Framework para el preprocesamiento de texto extraído de la red social Twitter

Autores
Manuel Alejandro Alvarado Cobo
Manuel Alejandro Hurtado Sarria

Universidad del Valle
Escuela de Ingeniería de Sistemas y Computación
Ingeniería de Sistemas
Tuluá - Valle del Cauca
2013

Framework para el preprocesamiento de texto extraído de la red social Twitter

Autores

Manuel Alejandro Alvarado Cobo*

Manuel Alejandro Hurtado Sarria**

manalco2@gmail.com*, lejohurtado@gmail.com**

Código. 200767082*, Código. 200767243**

Documento presentado como requisito parcial para la obtención de grado de Ingeniero de Sistemas

Directora

Ing. Yuri Mercedes Bermúdez Mazuera

Universidad del Valle

Escuela de Ingeniería de Sistemas y Computación

Ingeniería de Sistemas

Tuluá - Valle del Cauca

2013

Trabajo de grado presentado por:
Manuel Alejandro Alvarado Cobo
Manuel Alejandro Hurtado Sarria
Como requisito parcial para la obtención del título de Ingeniero de Sistemas.

Ing. Yuri Mercedes Bermúdez Mazuera
Directora

Jurado 1

Jurado 2

Agradecimientos

Manuel Alejandro Alvarado Cobo
Doy infinitas gracias a mi madre María Benilda Cobo Narváez, mi principal inspiración, mi modelo a seguir, mi guía, mi apoyo y mi motivación.

A nuestra directora Yuri Mercedes Bermúdez Mazuera por guiarnos a través de nuestro proceso de formación profesional y por supuesto durante el desarrollo de este proyecto.

A mis compañeros y amigos con los que compartí tanto durante estos años de formación profesional.

Manuel Alejandro Hurtado Sarria
Agradezco principalmente a mis padres y al resto de mi familia que me brindaron su apoyo en toda la carrera.

A la profesora Yuri Mercedes Bermúdez que brindó su incondicional apoyo en el proceso de desarrollo del trabajo de grado, a los diferentes profesores me brindaron su conocimiento.

Índice general

Índice de figuras

Índice de tablas

Resumen

El preprocesamiento de información tiene como propósito fundamental la “manipulación y trasnformación de los datos en bruto de manera que permita exponer o al menos facilitar la exposición de la información contenida en el arreglo de datos. [1]” Existen diversas herramientas que ofrecen la posibilidad de ejecutar subtareas de la extracción de información y procesamiento de lenguaje natural con este propósito, sin embargo son escasas aquellas que brindan la posibilidad de ejecutar más de una de estas tareas, esto sin mencionar que del mismo modo son escasas aquellas que permiten el tratamiento de datos en idiomas diferentes al inglés.

Tomando como fuente de datos a procesar a Internet, debe considerarse que es una fuente inagotable de información, basta y no muy bien ordenada; no obstante dicha información debe tener cierto grado de orden para poder ser tratada por las subtareas de extracción de información y generar resultados útiles a partir de los datos procesados, como por ejemplo en el área del marketing para mejorar ventas o productos, analizar tendencias, entre otros.

En el presente documento se expone el proceso de desarrollo de una aplicación que integra diferentes herramientas de preprocesamiento de información adaptadas para el idioma español que toma como datos de entrada texto recuperado de la red social Twitter. Inicialmente se implementa un módulo que permite recuperar los documentos usando palabras clave como criterio de búsqueda. Así mismo se detalla la implementación de módulos de corrección de palabras mal escritas, eliminación de palabras vacías, anotación gramatical, reconocimiento de nombres de entidades y el proceso realizado en su integración.

Abstract

The main purpose of information preprocessing is “to manipulate and transform raw data so the information content enfolded in the data set can be exposed. [1]” There are plenty of information extraction subtasks tools to achieve this purpose, however, most of them offer the possibility to execute only one of this tasks. In addition, the majority of the developed tools for this type of tasks have only been implemented for english.

Internet can be considered as a resource to obtain a large amount of data, it must be said that the global network is a vast and a not very organized source of information. Nevertheless this information must be at some point of organization to bring out a useful outcome from its processed data, for example, in the marketing area, this results could be used to increase sales, improve products, analyse tendencies, among others.

In this document, it is exposed the followed process in the development of a software application that integrates different information preprocessing tools that were adapted to work with the spanish language, taking as the input data the text recovered from the Twitter social network. The process started with the development of an information retrieval module that allows to retrieve documents using keywords as the search criteria. Likewise, it is explained the implementation of a spelling correction module, stop words module, part of speech tagging module, name entity recognition module and the process of its integration.

Capítulo 1

Introducción

1.1. Descripción del Problema

Internet contiene una extensa cantidad de información, según estudios de Intel [12], a principios de 2012, se transferían alrededor de 639.800 GB (Gigabytes) por minuto a través de la red, lo que equivaldría a un poco más de 921 PB (Petabytes) al día. Sin embargo, toda esta información está ordenada por secciones siguiendo diferentes parámetros, algunas veces ninguno; por lo que al final, en general se puede decir que en realidad no está ordenada. El poco orden de la información en Internet es algo difícil de percibir dado que según define la actual Web, la Web 2.0 [2], el contenido es generado por los usuarios para los usuarios, es decir, se construye de manera colaborativa y bidireccional, existe la retroalimentación. Es por esto que hoy en día existe la preocupación por darle un orden a la información en la web, algunas personas la denominan la Web 3.0 [3], la Web del futuro, una red de información con sentido, en la cual las búsquedas sean más eficientes y los resultados más concretos, tal y como expresó Tim Berners-Lee durante la 18ª Conferencia Mundial de la WWW, "la Web del futuro pondrá orden al caos actual de información en Internet [13]."

En ciencias de la computación, la Extracción de Información es un tipo de Recuperación de Información que pretende extraer datos estructurados desde documentos, sin embargo se requiere que los datos almacenados en dichos documentos presenten un orden para poder ser procesados por las tareas típicas de extracción de información. Las redes sociales son un ejemplo claro de cómo existen porciones de información ordenada en cierta forma, cada red social define sus propios estándares. La red social Twitter por ejemplo, ordena la información generada por sus usuarios en documentos de hasta 140 caracteres, normalmente accesibles de manera pública, incluso para personas que no pertenecen a la red social.

La información contenida en Twitter es tan diversa, que diferentes personas alrededor del mundo han encontrado múltiples usos para dicha información, donde los proyectos resultantes emplean técnicas de extracción de información para centrarse en temas específicos. No obstante, la gran mayoría de tecnologías usadas para estos proyectos están diseñadas para trabajar en el idioma inglés o idiomas diferentes al español. Diariamente se publican mensajes de personas de diferentes edades, y al ser un recurso de internet atrae en primera instancia a usuarios jóvenes, quienes muchas veces no prestan atención a las reglas gramaticales al momento de publicar sus mensajes, es decir, mucha información publicada en Twitter contiene errores ortográficos o palabras mal escritas; esto aumenta el grado de dificultad en el tratamiento de estos datos. Para resolver esta dificultad se requiere la implementación de diccionarios que permitan eliminar palabras innecesarias del conjunto de información, como los artículos gramaticales (el, la, los, las) y otras palabras que no agregan ningún valor y de esta manera llevar a cabo técnicas de preprocesamiento de información, tales como el reconocimiento de nombres de entidades, la resolución de correferencias, la extracción de terminologías, la extracción de relaciones, entre otras. Sin embargo, la mayor parte de las herramientas disponibles no implementan varias técnicas y se centran en alguna en específico. Esto dificulta el preproesamiento de la información, siendo necesario utilizar varias herramientas que ayuden a cumpliar el objetivo deseado.

1.2. Formulación del Problema

¿Cómo integrar diferentes técnicas de preprocesamiento de textos para aplicarlas a un corpus de tweets obtenidos de la red social twitter?

1.3. Objetivos

1.3.1. Objetivo General

Desarrollar un framework para el preprocesamiento de textos extraídos de la red social Twitter.

1.3.2. Objetivos Específicos

1. Implementar un mecanismo para la recuperación dinámica de tweets.
2. Proveer un conjunto de diccionarios que complementen el buen desempeño de las herramientas de preprocesamiento del texto.
3. Implementar un mecanismo para la corrección automática de palabras mal escritas.
4. Integrar técnicas de preprocesamiento de texto.

1.4. Estructura del Documento

El presente documento se divide en cuatro capítulos, hasta este punto se concluye el capítulo 1 que constituye un concepto del proyecto donde se presentan la descripción del problema, la formulación del problema, el objetivo general y los objetivos específicos. En el capítulo 2 se definen conceptos y/o teorías referenciadas en este documento bien sea para describir los procesos del sistema o para contextualizar otras ideas en el Marco de Referencia con el Marco Conceptual, Marco Teórico y Antecedentes. En el capítulo 3 se encuentra la descripción de las actividades realizadas en el proceso de desarrollo. Finalmente en el capítulo 4, se presentan las conclusiones y trabajo futuro. Adicionalmente se anexan los diferentes diagramas generados, historias de usuario y pruebas realizadas.

Capítulo 2

Marco de Referencia

2.1. Marco Teórico

2.1.1. Procesamiento de Lenguaje Natural (PLN)

Se refiere al campo de las ciencias de la computación, la inteligencia artificial y la lingüística cuyo principal interés son las interacciones entre el lenguaje humano y las computadoras, donde recientemente se introduce el concepto (anteriormente ligado en sí mismo al PLN) de Entendimiento del Lenguaje Natural que pretende permitirle a las computadoras entender el significado de entradas de lenguaje en forma natural. El PLN se explica más específicamente como la "habilidad de los sistemas para procesar oraciones en lenguajes naturales como el inglés en lugar de un lenguaje artificial especializado de computadoras como lo es C++ [14]." El procesamiento del lenguaje natural se subdivide en ciertos procesos para encapsular la interpretación de la entrada del lenguaje como tal, estas subdivisiones son:

- **Procesamiento de Señales:** Toma como entrada palabras habladas y las convierte a texto.
- **Análisis semántico:** Se encarga del sentido que toman las palabras y oraciones usadas, así como su relación con las entidades mencionadas.
- **Análisis sintáctico:** Se encarga de la estructura y gramática de las oraciones.
- **Pragmática:** Estudia el lenguaje en su relación con los usuarios y las circunstancias de la comunicación.

En general, se supone que un sistema de procesamiento de lenguaje natural en toda su capacidad, debería poder no sólo interpretar entradas de lenguaje natural, ya fuese en texto o no, sino

generarlo e interactuar en una conversación con otra entidad de procesamiento de lenguaje natural independientemente de si se trata de un ser humano u otra entidad artificial [14].

2.1.2. Part of Speech Tagging o Anotación Gramatical

También conocida como POS-Tagging o sólo POST, es una técnica de software que obtiene como entrada texto en lenguaje natural y automáticamente le asigna una categoría gramatical a cada palabra, como por ejemplo: sustantivo, verbo, adjetivo, nombre propio, etc. Cabe resaltar que las aplicaciones de anotación gramatical pueden tener una granularidad mucho más completa en cuanto a las categorías gramaticales que se le asignan a cada palabra, por ejemplo, puede dividirse el grupo de sustantivos en sustantivos singulares y sustantivos plurales, o más complejo aún, puede dividirse el grupo de verbos en verbos en pasado, verbos en futuro, verbos en infinitivo, etc. [16].

Entre los métodos más comunes para abordar esta técnica del procesamiento de lenguaje natural, se encuentran los basados en reglas, los estocásticos y los basados en transformación. Los métodos basados en reglas tienen definidos su conjunto de restricciones dependiendo del lenguaje a tratar, lo que teóricamente los obliga a ser capaces de procesar un único lenguaje. Los POST estocásticos (probabilísticos) y los basados en transformación pueden ser un poco más flexibles en cuanto al idioma a tratar, sin embargo, ambos requieren de entrenamiento para generar un modelo que permita interpretar las entradas de texto.

2.1.3. Named Entity Recognition o Reconocimiento de Nombres de Entidades

El Reconocimiento de Nombres de Entidades (NER) es una subtarea de la extracción de información que consiste en la búsqueda, localización y etiquetado de partes atómicas de un texto para categorizarlos ya sean nombres de cosas, personas, empresas, lugares, fechas, cantidades, etc. [17]. Sin embargo solo tres de estas son globalmente aceptadas como Nombres de Entidades: Persona, Lugar y Organización.

2.2. Marco Conceptual

2.2.1. Corpus

"Conjunto lo más extenso y ordenado posible de datos o textos científicos, literarios, etc., que pueden servir de base a una investigación. [26]"

2.2.2. Recuperación de Información

Es un área de estudio que se concentra en la búsqueda información en documentos, en metadatos de los documentos, en bases de datos relacionales, en la World Wide Web y especialmente se concentra en la recuperación de textos, imágenes, sonidos, etc. [6].

2.2.3. Stop Words o Palabras Vacías

Las Palabras Vacías son palabras que se encuentran con mucha frecuencia a lo largo de un texto, algunos autores, como Singhal llegan a considerarlas como "ruido" en la información dado que no agregan un valor significativo a la misma. [7]

2.2.4. Software Libre

El software que respeta la libertad de los usuarios y la comunidad. En términos generales, los usuarios tienen la libertad de copiar, distribuir, estudiar, modificar y mejorar el software. Con estas libertades, los usuarios (tanto individual como colectivamente) controlan el programa y lo que puede hacer. Donde "las cuatro libertades esenciales son:

- La libertad de ejecutar el programa para cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y cambiarlo para que haga lo que usted quiera (libertad 1). El acceso al código fuente es una condición necesaria para ello.
- La libertad de redistribuir copias para ayudar a su prójimo (libertad 2).
- La libertad de distribuir copias de sus versiones modificadas a terceros (libertad 3). Esto le permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello [18]."

Lo anterior no implica que el software libre no esté protegido bajo alguna licencia, existen diferentes tipos de licencias libres que además de otorgar las cuatro libertades anteriores, otorgan algunos deberes.

2.3. Antecedentes

La perspectiva de red social se usa para modelar la estructura social de un grupo y cómo esta estructura cambia con el paso del tiempo. [8] En los últimos años y con el creciente uso de Internet, el uso de las redes sociales ha presentado un aumento considerable ubicándose como centros de opinión.

La variada cantidad de información contenida en sitios como Twitter, que es de acceso público, ha permitido que diferentes grupos de personas encuentren múltiples usos para esta. A continuación se mencionan algunos proyectos que han implementado varias subtareas de la extracción de información en ambientes o con condiciones afines, sin embargo, es necesario aclarar que la gran mayoría de los trabajos realizados en el campo de la extracción de información y minería de datos sociales para propósitos puntuales emplean tecnología de procesamiento del lenguaje natural para el idioma inglés.

2.3.1. Social-based traffic information extraction and classification

Extracción y clasificación de información de tráfico basada en redes sociales, fue un proyecto del Centro Nacional de Tecnologías de Cómputo y Electrónicas (NECTEC por sus siglas en inglés) bajo la supervisión de la Agencia Nacional de Desarrollo de Ciencia y Tecnología de Tailandia (NSTDA por sus siglas en inglés). El sistema generado extrae y analiza información en tiempo real de la red social Twitter que se relacione con el tráfico urbano, clasificando lugares y rutas mientras determina el estado del tráfico en dichas rutas. El proyecto concluye que el sistema tenía un 76.85 % de precisión al clasificar lugares y un 98.23 % de precisión al clasificar rutas. [9].

2.3.2. EquatorNLP: Pattern-based Information Extraction for Disaster Response

Desarrollado por Lars Döhling y Ulf Leser para la Universidad Humboldt de Berlín, es un sistema de extracción de información de redes sociales enfocado a la información relacionada con desastres, el sistema clasifica y muestra información actualizada y relevante para la toma de decisiones en el accionar post-desastre, como lo son la planeación de rescates o monitoreo de zonas por ejemplo. En éste sistema se utilizó procesamiento del lenguaje natural en conjunto con técnicas de síntesis basadas en grafos, donde los grafos son representaciones semánticas de relaciones entre entidades. La extracción de la información se realizó con relaciones entre cinco entidades: objeto,

cantidad, modificador, indicador y negación. Se empleó una máquina basada en aprendizaje para la cual se necesitó un corpus de información previamente anotada y ordenada para su entrenamiento. [10].

2.3.3. Using contextual spelling correction to improve retrieval effectiveness in degraded text collection

Desarrollado por Patrick Ruch para el Instituto Federal de Tecnología de Suiza, en Lausanne, propone un sistema de recuperación de información aplicado a documentos, que le hace frente a los problemas gramaticales (ortografía) donde las palabras son evaluadas por medio de diccionarios correctamente formados teniendo en consideración como primera instancia la distancia entre cadenas (el valor de separación que define qué tan similares son dos cadenas de texto), palabra por palabra y guardando una lista de posibles candidatos, como segunda instancia se realiza un tratamiento de POS-tagging en el que se tienen en cuenta los dos tokens anteriores y posteriores a cada palabra y por último se clasifican los candidatos con base en las probabilidades de transición. [11].

2.3.4. Métodos para la corrección ortográfica automática del español

Tesis de maestría presentada por el Lic. Daniel Castro Castro para la Universidad de Oriente en Santiago de Cuba donde propone un corrector ortográfico basado en reglas y recursos estadísticos, en el que se clasifican los errores encontrados en los textos en dos tipos: ortográficos (Non-Word) y gramaticales (Real-Word), tomando en cuenta 4 casos para diferenciar dicho error, los cuales son: errores por sustitución de caracteres, errores por inserción de caracteres, errores por eliminación de caracteres y errores por transposición de caracteres. El algoritmo implementado se divide en dos etapas, la primera donde se buscan sugerencias para la corrección de los errores ortográficos y la segunda etapa donde se buscan sugerencias para errores gramaticales. En cada etapa se usan recursos como diccionarios de palabras, listado de los errores más comunes, ficheros de reglas, entre otros. [5].

Capítulo 3

Desarrollo del Proyecto

Para la organización en el proceso de desarrollo se utilizó la metodología Scrum [4], que se clasifica como una metodología ágil con base en el control de procesos. Scrum consta de dos eventos, *Spring* y *Scrum*, tres artefactos *Product Backlog*, *Carta burndown* y *Sprint Backlog*, y tres roles *Product owner*, *Development Team* y *Scrum Master* (Figura 3.1).

Las tareas asignadas en el spring son seleccionadas de las actividades para el cumplimiento de los objetivos, estas tareas deben cumplir con sus respectivas pruebas y documentación.

Para los eventos de *Scrum Day* se estableció una periodicidad de una semana en la que toman parte todos miembros del equipo, para los eventos del *Spring* por su parte, se estableció un ciclo de 30 días. Los roles de la metodología se dividieron en: dueño del producto (*Product owner*) Director de Trabajo de grado, el equipo de desarrollo (*Development team*) conformado por los estudiantes que presentan este documento y finalmente el rol de *Scrum Master* se le asignó a uno de ellos (Manuel Alvarado).

En el desarrollo de ciertas actividades se presentan problemas que llevan un poco más de tiempo de lo planeado y/o se convierten en situaciones más complejas, que ameritan ver dicha actividad desde otro enfoque, de esta forma se puede encontrar una nueva manera de afrontar el problema encontrado y dicha actividad puede ser modificada para el cumplimiento de un objetivo. Scrum permite realizar este tipo de cambios sin provocar un gran impacto en el desarrollo de las demás actividades proponiendo la reestructuración de la actividad afectada generando un nuevo plazo de tiempo para terminarla. Uno de los artefactos de Scrum es el DevBurndown que permite visualizar el estado actual del proyecto y establecer estrategias para mantener o mejorar el ritmo de trabajo,

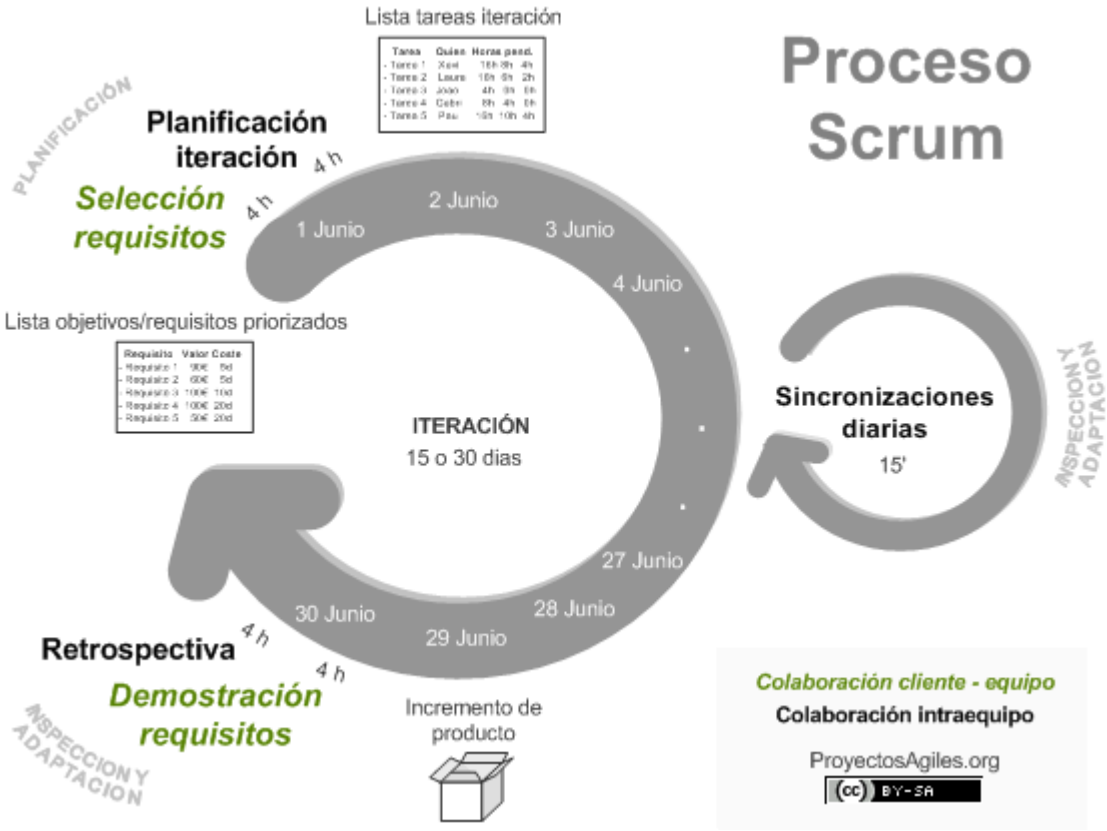

Figura 3.1: Principios clave de Scrum [24].

dicho gráfico puede observarse en la figura 3.2. La línea azul representa el trabajo planeado en horas (restantes) y la línea roja representa el trabajo realizado en horas (restantes). Dicho esto, se puede observar en el gráfico una inestabilidad que incurre en un pico de horas trabajadas al iniciar el mes de noviembre a medida que intenta estabilizarse se intersectan las curvas en el mes de enero para finalmente alcanzar un ritmo estable desde el mes de marzo.

Por otra parte, como lo define IMB lo menciona en una de sus Wiki, “una historia de usuario es una representación de un requerimiento de software escrito en una o dos frases utilizando el lenguaje común del usuario. Las historias de usuario son utilizadas en las metodologías de desarrollo ágiles para la especificación de requerimientos (acompañadas de las discusiones con los usuarios y las pruebas de validación). Cada historia de usuario debe ser limitada, esta debería poderse escribir sobre una nota adhesiva pequeña.

Las historias de usuario son una forma rápida de administrar los requerimientos de los usuarios

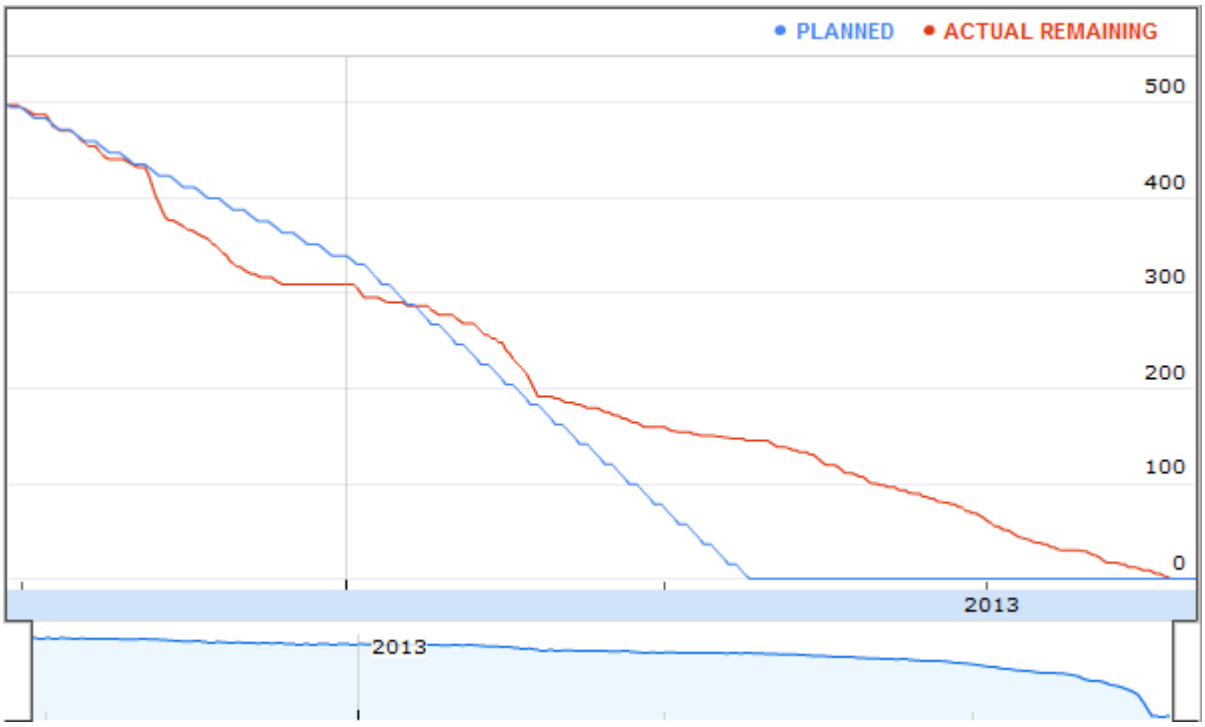

Figura 3.2: Artefacto SCRUM: DevBurndown.

sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos. Las historias de usuario permiten responder rápidamente a los requerimientos cambiantes. [25] ”

3.1. Diseño

3.1.1. Diagrama de Clases

3.1.1.1. Sistema

En la figura B.2 se presenta el diagrama de clases general del sistema.

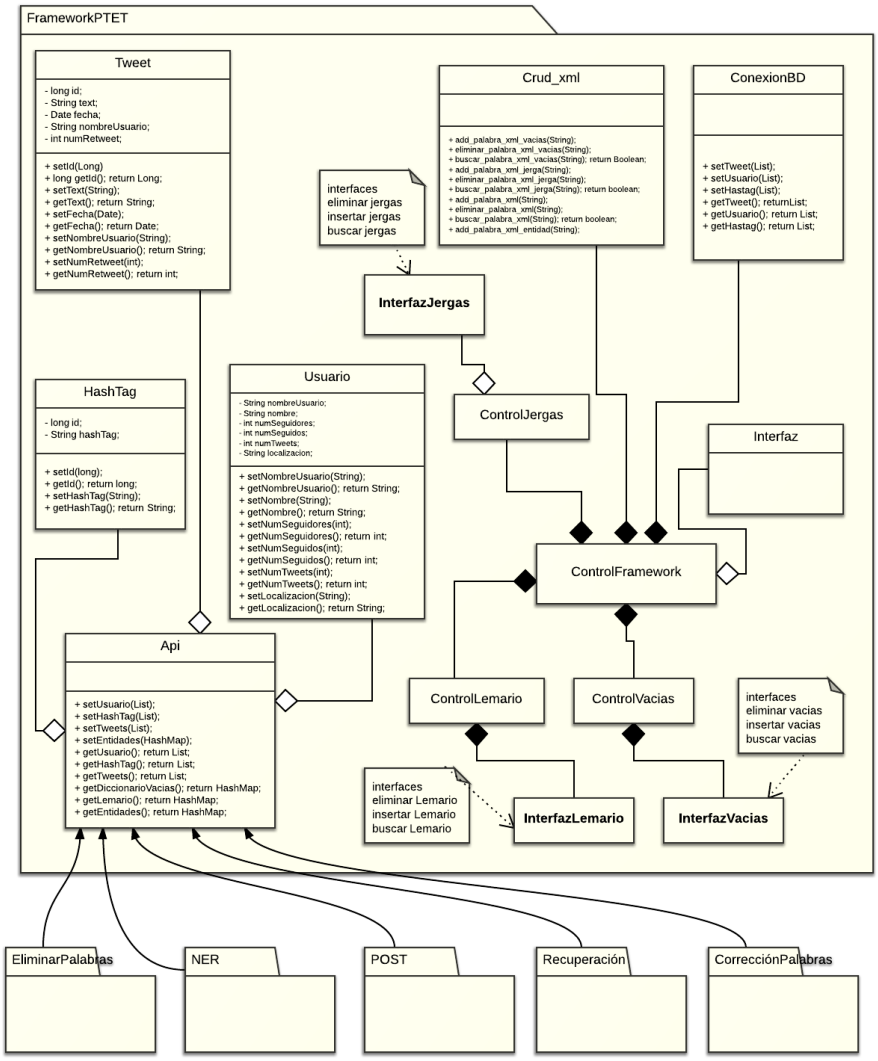

Figura 3.3: Diagrama de Clases: Sistema

3.1.1.2. Eliminación de Palabras Vacías

En la figura B.6 se presenta el diagrama de clases del módulo de Eliminación de Plabras Vacías.

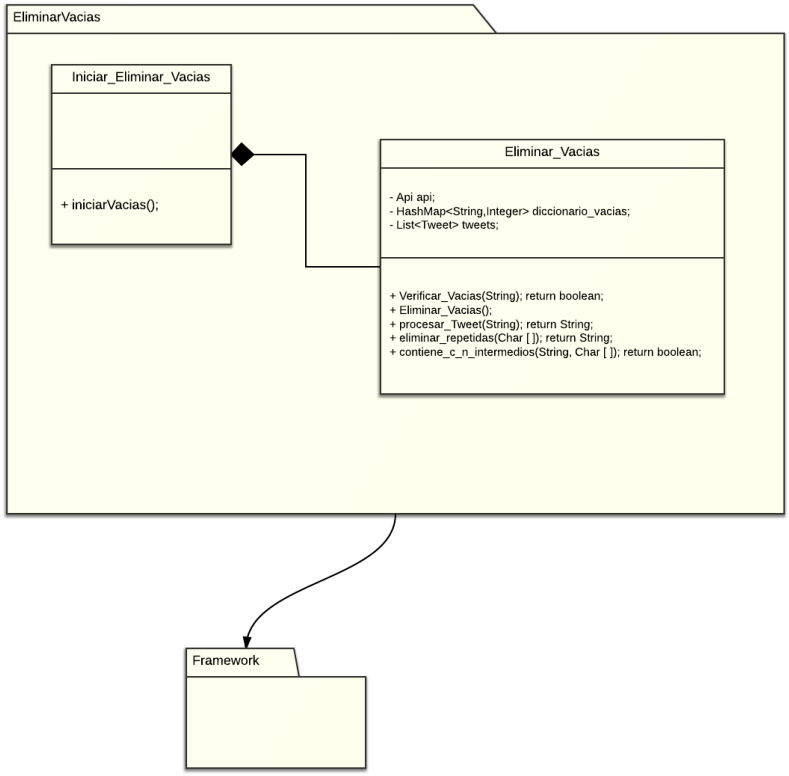

Figura 3.4: Diagrama de Clases: Eliminación de Palabras Vacías

3.1.1.3. Corrección de Palabras Mal Escritas

En la figura B.5 se presenta el diagrama de clases del módulo de Corrección Automática de Palabras Mal Escritas.

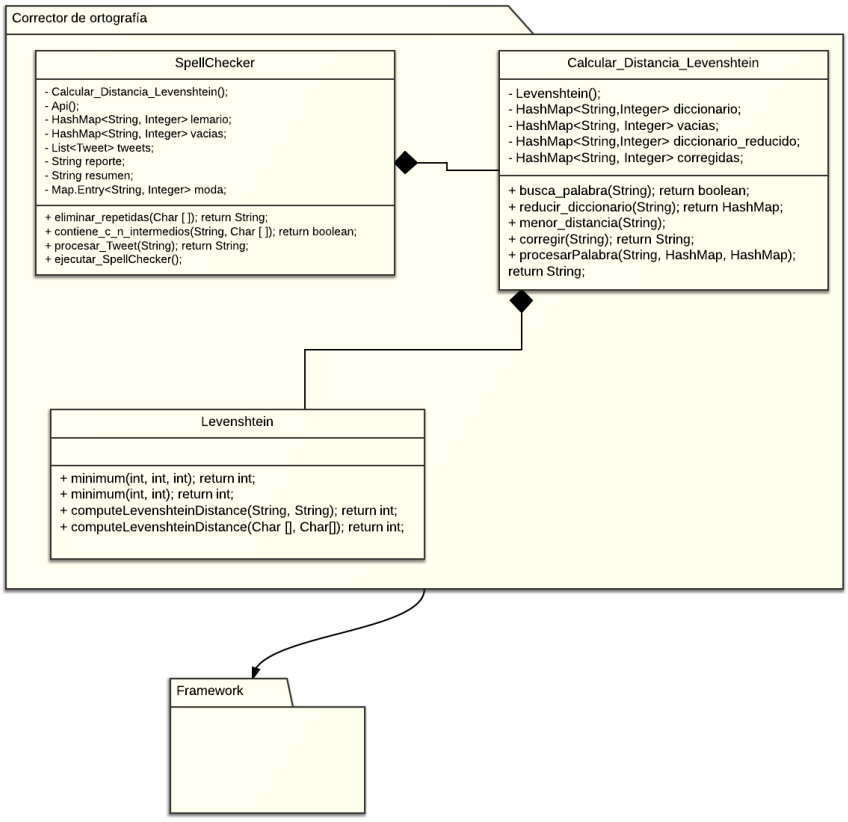

Figura 3.5: Diagrama de Clases: Corrección de Palabras Mal Escritas

Los diagramas de clases del sistema y sus componentes pueden referenciarse en el documento de anexos. Apéndice B.

3.1.2. Diagrama de Componentes

En la figura 3.6 se presenta el diagrama de componentes general del sistema.

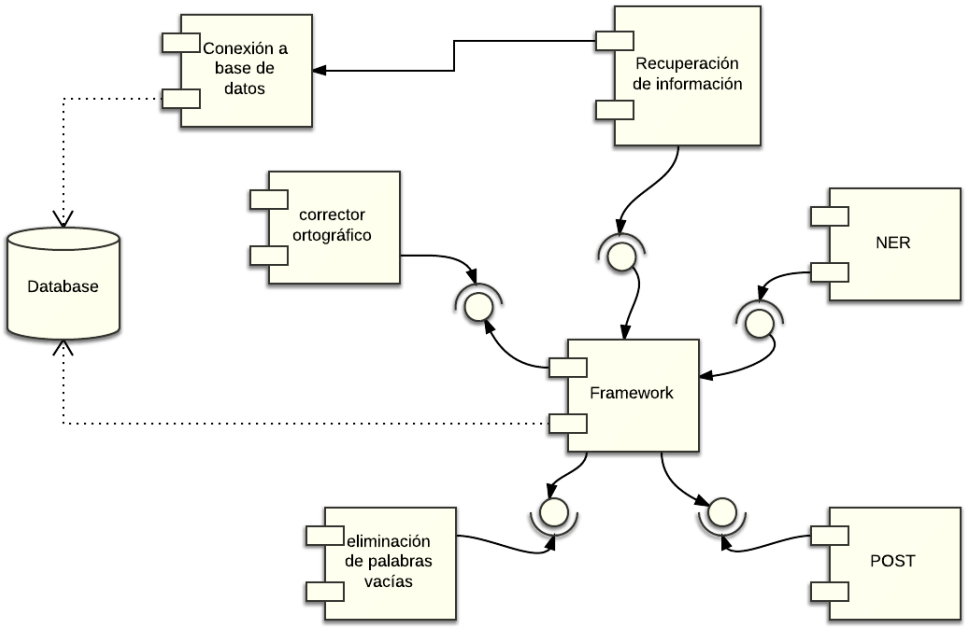

Figura 3.6: Diagrama de Componentes.

3.1.3. Historias de Usuario

3.1.3.1. Sistema

En la siguiente figura se muestra una historia de usuario correspondiente al uso de la interfaz principal del sistema.

Historias de Usuario					
Título	Framework				
ID	7	Sistema			
Descripción					
Como usuario quiero que el sistema me permita ejecutar módulos de preprocesamiento de texto para obtener resultados estructurados.					
Estimación	1	**Prioridad**	5	**Dependencia**	Módulos
Prueba de			Integración - Rendimiento		
Comportamiento individual de los módulos con el framework					

Tabla 3.1: Historia de Usuario: Sistema.

3.1.3.2. Recuperación de Tweets

En la siguiente figura se muestra una historia de usuario correspondiente al uso del módulo de recuperación de tweets.

Historias de Usuario					
Título	Recuperación de tweets con palabras clave.				
ID	1	**Módulo**	Recuperación		
Descripción					
Como usuario del sistema quiero realizar búsquedas de tweets para recuperarlos y ejecutar otras tareas sobre ellos.					
Estimación	2	**Prioridad**	5	**Dependencia**	API twitter
Prueba de			Sistema		
Conexión con la red social twitter.					
Busqueda de tweets por palabras dadas.					
Recuperación de tweets.					

Tabla 3.2: Historia de Usuario: Módulo de recuperación de tweets.

3.1.3.3. Reconocimiento de Nombres de Entidades

En la siguiente figura se muestra una historia de usuario correspondiente al módulo de reconocimiento de nombres de entidades (NER).

Historias de Usuario					
Título	Reconocimiento de nombres de entidades				
ID	6	**Módulo**	NER		
Descripción					
Como usuario quiero que el sistema etiquete y clasifique las entidades presentes en el texto para identificar qué entidades tienen relación con las palabras clave de búsqueda y recuperación.					
Estimación	1	**Prioridad**	4	**Dependencia**	Ninguna
Prueba de			Rendimiento		
Porcentajes de aciertos en identificación de entidades miselaneas y personas en el idioma español					

Tabla 3.3: Historia de Usuario: Módulo de reconocimiento de nombres de entidades (NER).

Las historias de usuario pueden encontrarse en el documento de anexos. Apéndice A.

3.1.4. Diagrama de Secuencia

3.1.4.1. Sistema

El siguiente es el diagrama de secuencia de la interfaz principal del sistema ejecutando la cola de procesos.

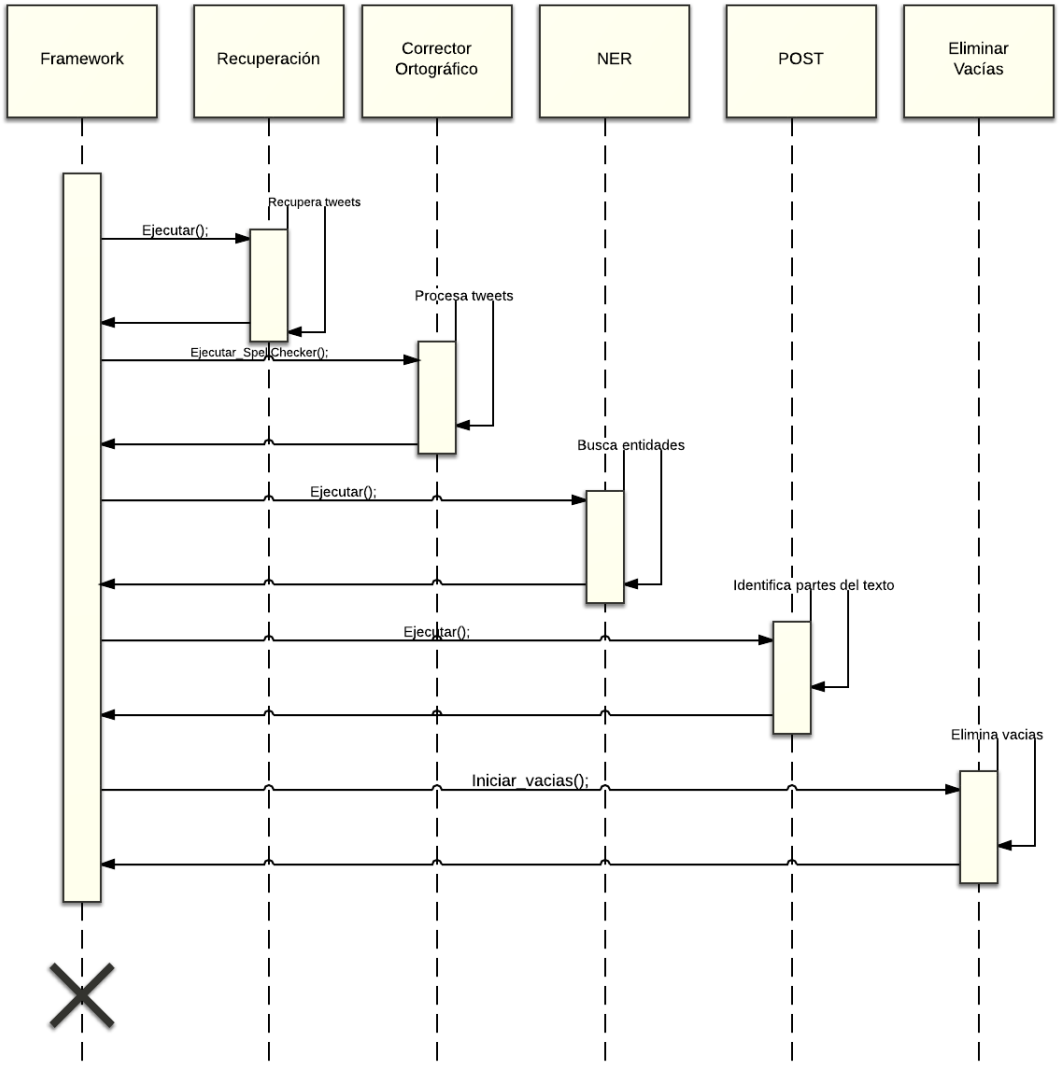

Figura 3.7: Diagrama de Secuencia: interfaz principal del sistema ejecutando la cola de procesos.

3.1.4.2. Recuperación de Tweets

El siguiente es el diagrama de secuencia del del módulo de recuperación al recuperar los tweets.

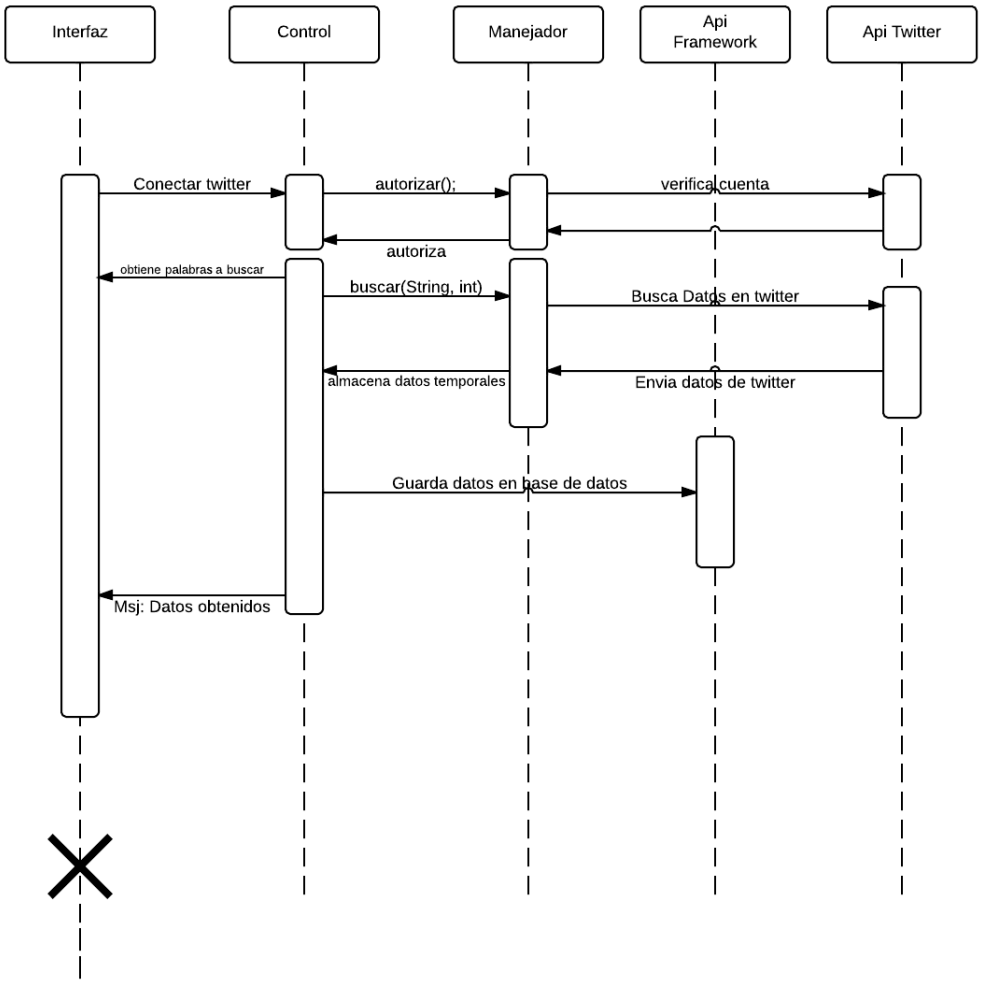

Figura 3.8: Diagrama de Secuencia: módulo de recuperación al recuperar los tweets.

3.1.4.3. Reconocimiento de Nombres de Entidades

El siguiente es el diagrama de secuencia del del módulo de Reconocimiento de Nombres de Entidades.

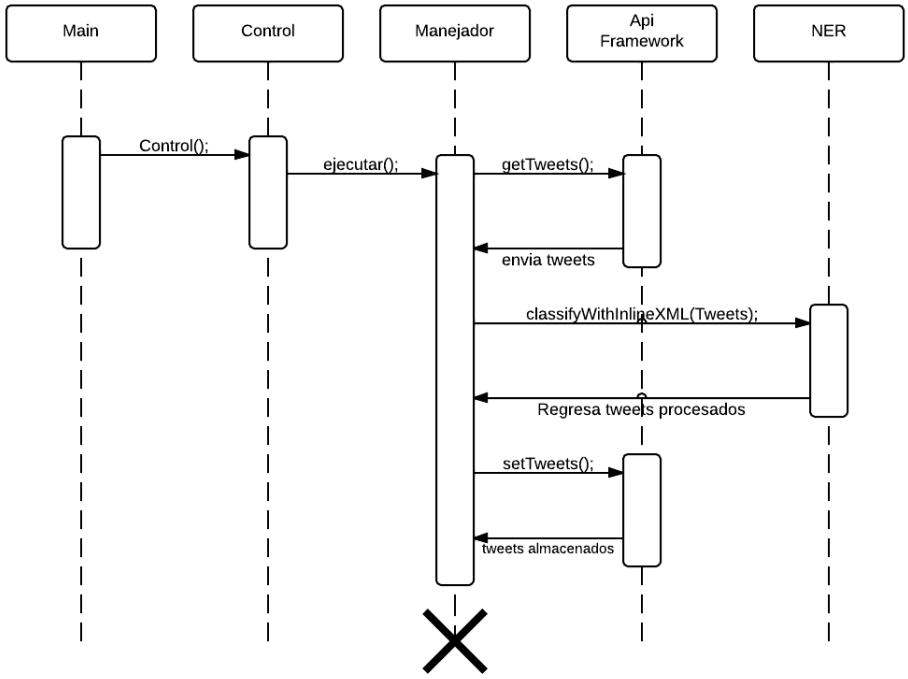

Figura 3.9: Diagrama de Secuencia: módulo de Reconocimiento de Nombres de Entidades.

Los diagramas de secuencia pueden ser encontrados en el documento de anexos. Apéndice C.

3.2. Descripción General del Sistema

El sistema desarrollado consta de cinco módulos, una interfaz gráfica de usuario y una interfaz de programación de aplicaciones (API). En los cinco módulos integrados se encuentran:

3.2.1. Interfaz Gráfica de Usuario

La interfaz gráfica de usuario ofrece la ejecución de los métodos de los módulos del núcleo así mismo como incluir dichos métodos en una cola de procesos que ejecutará uno a uno los procesos incluidos en la misma. Cabe resaltar que no es estrictamente necesario el uso de la interfaz gráfica para llevar a cabo estas tareas.

3.2.2. Módulo de Recuperación de Documentos (tweets) de la Red Social Twitter

La aplicación encargada de poblar la base de datos del sistema es el módulo de recuperación de tweets, por medio del uso de palabras clave éste módulo recupera desde la red social tweets que contienen dichas palabras.

3.2.3. Módulo de Eliminación de Palabras Vacías

El módulo de eliminación de palabras vacías retira palabras del texto que se encuentren clasificadas como *Stop Words*, es decir, que no le agregan un valor significativo a la información. Éste proceso se lleva a cabo usando un diccionario (lista de palabras) que contiene las palabras que se consideran vacías.

3.2.4. Módulo de Corrección de Palabras

La corrección de palabras en el sistema se realiza por un módulo que implementa el algoritmo de distancia entre cadenas Damerau-Levenshtein que permite corregir errores de sustitución, eliminación, inserción y sustitución, comparando las palabras de cada tweet con las que se encuentran en el diccionario de palabras en español.

3.2.5. Módulos de Reconocimiento de Nombres de Entidades y Anotación Gramatical

De los módulos de reconocimiento de nombres de entidades y anotación gramatical, incluyen aplicaciones producto del Grupo de Investigación de Procesamiento de Lenguaje Natural de la Universidad de Stanford [15], los cuales son el Name Entity Recognizer [17] y el Part-of-Speech Tagger [16].

3.3. Arquitectura

La figura 3.10 enseña el esquema de relación que tienen los componentes del sistema. Los componentes de la aplicación interactúan entre sí por medio de la interfaz de programación de aplicaciones, de manera que esta actúa como mediador de cada proceso que se ejecuta y la acción que lo inicia.

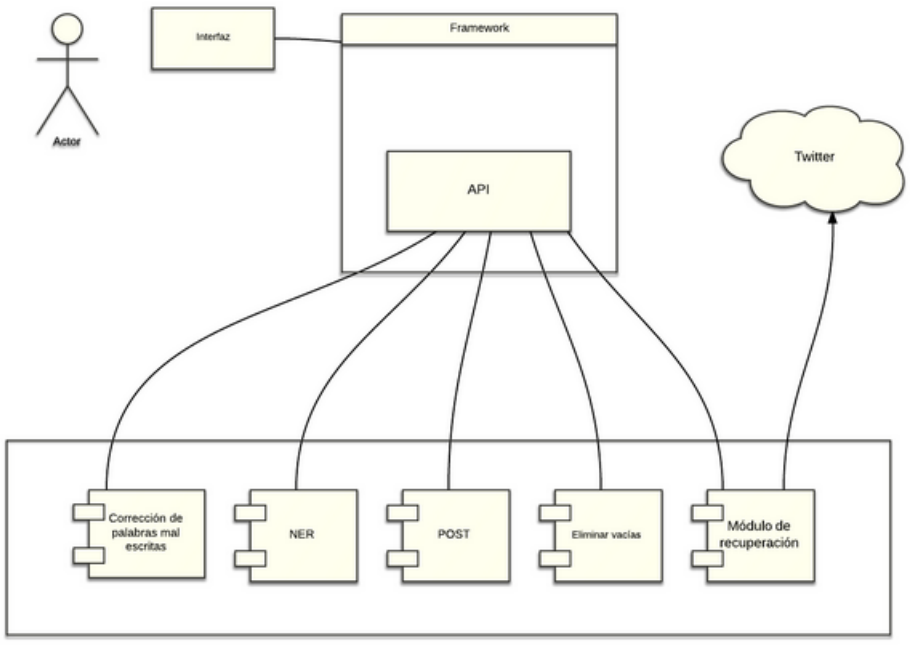

Figura 3.10: Arquitectura del sistema.

Para la ejecución de cualquier subtarea de extracción de información, es necesario que las entradas de las mismas existan en el sistema, es decir, que en la base de datos del sistema se encuentren documentos sin ser procesados (tweets), de forma que el proceso debe iniciar con la ejecución del

módulo de recuperación de tweets. Este módulo almacena la información en la base de datos local, manteniendo copias originales de cada documento recuperado.

La ejecución de los demás módulos obtiene como entrada el conjunto de documentos almacenados en la base de datos del sistema; el módulo de reconocimiento de entidades y el módulo de anotación gramatical evalúan uno a uno el texto de los documentos (tweets) y anotan las palabras con la etiqueta correspondiente. Durante este proceso estos módulos tratan los tweets como una instancia de un objeto, una vez finalizado el proceso de anotación el texto es almacenado nuevamente en la base de datos del sistema (siempre conservando una copia del documento original).

El módulo de palabras vacías obtiene sus datos de entrada de la base de datos del sistema por medio del API. De los datos de entrada se procesa palabra por palabra, donde las que se encuentren registradas en el diccionario de palabras vacías son eliminadas. Al final del proceso los datos de salida son enviados nuevamente a la base de datos del sistema.

Finalmente la ejecución del módulo de corrección de palabras mal escritas procesa una a una las palabras de cada tweet comparándolas con algunas dentro del diccionario de palabras en español. Se elige qué palabras son comparadas teniendo en cuenta ciertos criterios, como el número de caracteres de la palabra que se desea corregir. El algoritmo usado para la comparación de las palabras es Dermau-Levenshtein, que calcula la distancia (grado de separación, le asigna un valor dependiendo de qué tan parecidas son las palabras) entre palabras y si se cumplen algunos criterios se elige la palabra cuya distancia es menor (la más similar) y se reemplaza por la palabra de entrada.

3.4. Detalles de la Implementación

Antes de entrar en detalle con la implementación de la aplicación de software, es necesario entender la composición de la unidad de información a la cual se le denomina *tweet* en este documento. Un tweet es un documento de texto limitado hasta un máximo de 140 caracteres publicado por un usuario de la red social Twitter. (Figura 3.11)

Figura 3.11: Tweet publicado desde la cuenta en Twitter de la Real Academia Española.

En el texto de un tweet, se pueden encontrar objetos como *Hashtags*, *Menciones* o *Enlaces* que no deben ser procesados por las tareas definidas. (El trato para estos componentes del tweet se expone más adelante en esta misma sección). Un *Hashtag* es una palabra que se caracteriza por comenzar con el símbolo numeral (#); hace las veces de etiqueta para clasificar el tweet en el marco de un tema específico. (Figura 3.12)

Figura 3.12: Ejemplo de un tweet con el uso de Hashtags.

De manera similar a un Hashtag, una *Mención* se puede explicar como una invocación a un usuario de la red social, de manera que se le notifica al usuario mencionado del tweet y de su contenido. Las menciones son una de las formas de interacción entre usuarios que se tienen en Twitter. Otras formas de interacción incluyen *Mensajes Directos* entre usuarios, *Retwits*, *Favoritos* y *Seguimiento*, aunque ninguna de estas últimas se ve reflejada directamente sobre el texto contenido en un tweet y por lo tanto no se procesa en el framework. (Figura 3.13)

Figura 3.13: Ejemplo de tweet con el uso de Menciones.

Para el desarrollo del **primer objetivo** (*Implementar un mecanismo para la recuperación dinámica de tweets*), se inició con la implementación de un pequeño aplicativo de recuperación de información usando palabras clave como criterio para obtener los tweets desde la red social. Este aplicativo, con las modificaciones necesarias durante el proceso de desarrollo general del proyecto, se convertiría finalmente en el módulo de recuperación de tweets. En conjunto con este módulo se inició el primer modelo de base de datos del sistema. (Figura 3.14)

El sistema tiene a disposición cuatro diccionarios o listas de palabras (Diccionario de Palabras Vacías, Diccionario de Palabras en Español, Diccionario de Expresiones Populares y Diccionario de Nombres de Entidades) en español que pueden ser usados por medio del API, algunos de estos diccionarios son necesarios para la ejecución de otros módulos que fueron implementados posteriormente; la compilación de estos diccionarios corresponde con el desarrollo del **objetivo número dos** (*Proveer un conjunto de diccionarios que complementen el buen desempeño de las herramientas de preprocesamiento del texto*). El proceso de desarrollo del framework continuó con la compilación de dos de estos diccionarios: diccionario de palabras vacías (diccionario número 1, con un poco más de 400 palabras) y diccionario de expresiones populares (jergas. Diccionario número 2, con un poco más de 2400 palabras). (Figura 3.15)

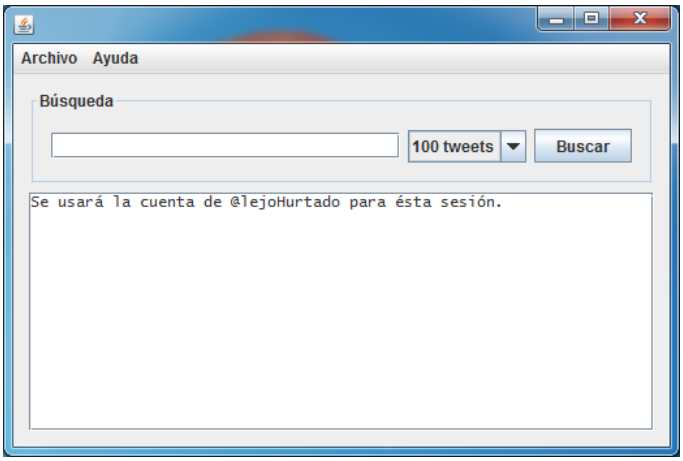

Figura 3.14: Interfaz gráfica del Módulo de Recuperación de Tweets.

Producto de la investigación sobre mecanismos para la corrección automática de palabras mal escritas y el método elegido para el desarrollo del módulo de corrección de palabras, se generó un diccionario adicional (diccionario número 3, con un poco más de 87300 palabras) en el que se compilaron palabras del español como referencia para el funcionamiento de dicho módulo, por lo que el proceso de desarrollo del sistema continuó con la compilación de este nuevo diccionario y después con la implementación del módulo de corrección de palabras mal escritas, que corresponde al cumplimiento del **objetivo número tres** (*Implementar un mecanismo para la corrección automática de palabras mal escritas*). Para el desarrollo de este módulo se inició un proceso de búsqueda de algoritmos de corrección de palabras, en este proceso, se encontraron dos tipos de ellos: semánticos y tipográficos. La implementación de un corrector semántico es extensa debido a que debe contar con reglas para cada dialecto del español, por lo que se decidió tomar el enfoque de los algoritmos tipográficos que se basan en corpus o diccionarios de palabras, en este caso palabras en español. En conjunto con un algoritmo de distancia entre cadenas se logra determinar una posible corrección a una palabra que no se encuentre en el diccionario. El algoritmo de distancia implementado en el módulo es el algoritmo de Damerau-Levenshtein que permite cuatro operaciones requeridas para transformar una cadena, bien sea inserción, eliminación, sustitución o transposición de dos caracteres. Por la cantidad de palabras en este diccionario, el compararlas una a una no es muy eficiente, por lo que se buscó una forma de reducir la cantidad de comparaciones que se deberían hacer. Esto se logró comparando únicamente palabras con una cantidad igual de caracteres con un margen de error de un carácter.

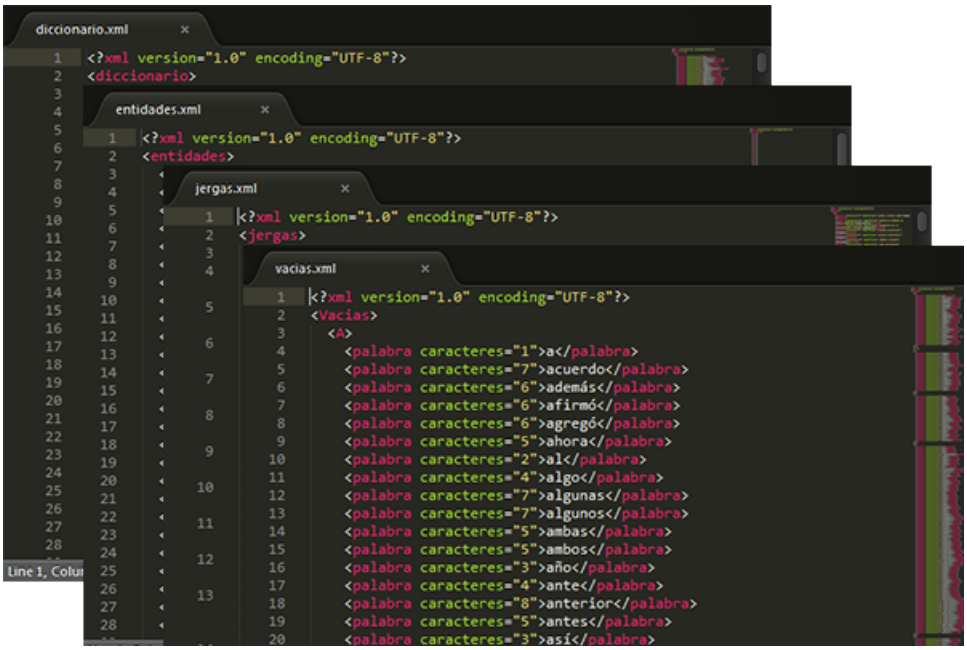

Figura 3.15: Diccionarios del sistema.

La integración de técnicas de preprocesamiento de texto en español, para el cumplimiento del **objetivo específico número cuatro** (*Integrar técnicas de preprocesamiento de texto*), se hizo por medio del desarrollo e integración de los módulos de eliminación de palabras vacías, anotación gramatical (POST) y reconocimiento de nombres de entidades (NER). La generación del módulo de Eliminación de Palabras Vacías se realizó con la ayuda del Diccionario de Palabras Vacías, de modo que se consulta si la palabra procesada está incluida en el diccionario, de ser así, es eliminada del texto.

Con ambos módulos, corrección de palabras y eliminación de palabras vacías se encontraron dificultades con la forma en que los usuarios de la red social escriben, encontrando casos donde repiten caracteres, hay caracteres especiales o hay números en una misma palabra, entre otros casos, por lo cual se generaron ciertas reglas para poder obtener un mejor resultado a la hora de procesar las palabras y reducir inconvenientes con objetos propios de twitter como lo son los hashtag y las menciones de usuario. Para ello, las reglas generadas fueron:

- No se procesarán palabras con caracteres numéricos intermedios.

  En el ejemplo de la Figura 3.16, no se procesan las cadenas *F3L1Zzzz N4V1DAD*.

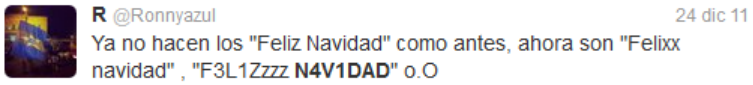

Figura 3.16: Ejemplo de tweet con caracteres numéricos intermedios.

- No se procesarán palabras con caracteres especiales intermedios.

  En el ejemplo de la Figura 3.17, no se procesa la cadena *USD$1.000*.

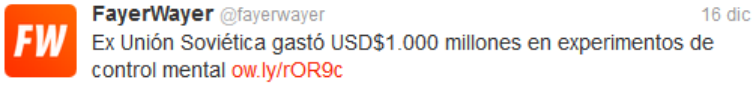

Figura 3.17: Ejemplo de tweet con caracteres especiales intermedios.

- No se procesarán palabras iniciadas con @.

  En el ejemplo de la Figura 3.18, no se procesan las cadenas *@RTVE @microsiervos*.

Figura 3.18: Ejemplo de tweet con palabras iniciadas en @.

- No se procesarán palabras iniciadas con #.

  En el ejemplo de la Figura 3.19, no se procesa la cadena *#Google*.

- Se eliminarán caracteres repetidos en las palabras.

  En el ejemplo de la Figura 3.20, La cadena *Doooooooooooooormir* se reduce a *Dormir*.

  La eliminación de caracteres repetidos, eliminaría de una palabra con “cc” y/o “rr” únicamente cuando las palabras que contengan estos caracteres estén mal escritas debido a que primero se buscan en el diccionario y si no se encuentran entonces los caracteres repetidos son eliminados.

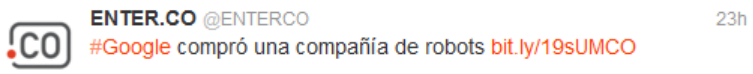

Figura 3.19: Ejemplo de tweet con palabras iniciadas en #.

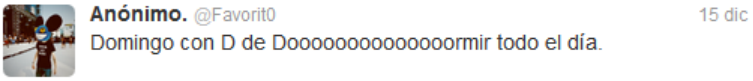

Figura 3.20: Ejemplo de tweet con palabras con caracteres repetidos.

- Los caracteres al inicio y/o al final de una palabra deberán volver a incluirse para no perjudicar procesos posteriores.

  En el ejemplo de la Figura 3.21, las cadenas *¿Cuántos* y *5s?* se procesan retirando los caracteres especiales, determinando si la subcadena generada debe ser corregida y finalmente se agregan de nuevo los signos.

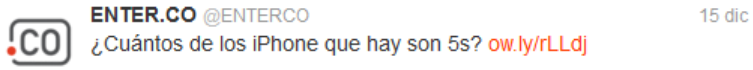

Figura 3.21: Ejemplo de tweet con palabras iniciadas o terminadas con un carácter especial.

En el desarrollo del módulo de reconocimiento de nombres de entidades, se integró por medio de una interfaz una herramienta de software desarrollada por el Grupo de Procesamiento de Lenguaje Natural de la Universidad de Stanford [15], liberada bajo licencias libres. Para la integración del NER de Stanford fue necesario realizar un entrenamiento con al rededor de cinco mil palabras. Dado que los datos de entrenamiento deben ser correctos, se determinó entrenar el modelo para identificar dos tipos de entidades para enrutar el desarrollo de este módulo dentro de las limitaciones de tiempo del proyecto. Las entidades definidas para el entrenamiento son *PERSONA* y *MISC* (miscelaneo). Con esta integración, el módulo no sólo etiqueta las entidades mencionadas en los textos sino también genera un diccionario de las entidades encontradas (diccionario número 4).

Cabe resaltar que ambos entrenamientos, para el módulo de Reconocimiento de Nombres de Entidades y Anotación Gramatical, se realizaron con el propósito de generar el modelo que le permitiría a su respectivo módulo realizar el procesamiento del texto que se le envíe por medio del framework.

De manera similar, para el desarrollo del módulo de anotación gramatical, se integró el Anotador Gramatical del Grupo de Procesamiento de Lenguaje Natural de la Universidad de Stanford [15]. Esta integración igualmente requirió de un entrenamiento, para el cual era necesario elegir uno de los modelos de entrenamiento de la aplicación. Con el objetivo de entrenar la herramienta para trabajar con el idioma español, se realizó el entrenamiento con el modelo para el idioma francés, dado que al igual que el español se clasifica como una lengua romance [19]. El entrenamiento se llevó a cabo con más de cinco mil palabras. Debido a esto, el resultado final logra resultados aceptables para entradas en idioma español.

Para integrar los aplicativos mencionados como módulos del sistema, se desarrolló una interfaz que hiciera las veces de mediador entre las entradas del sistema y los métodos de procesamiento ofrecidos por las aplicaciones. Mientras se llevaba a cabo este proceso, se actualizaban los módulos desarrollados para establecer las conexiones requeridas y permitir la interacción entre los mismos. (Figura 3.22)

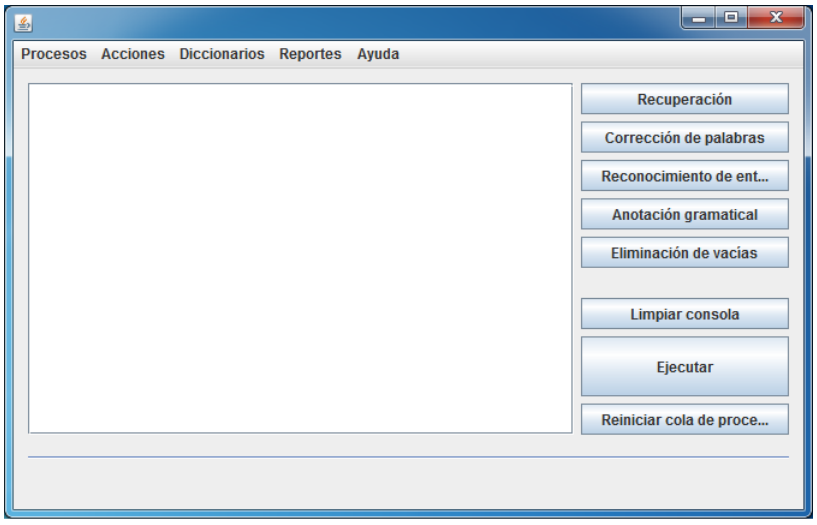

Figura 3.22: Interfaz gráfica de usuario.

Finalmente, para concretar el objetivo general del proyecto, se integraron los módulos anteriormente descritos por medio de una interfaz y una capa de programación de aplicaciones, de manera que se permite preprocesar y almacenar de forma local la información recuperada de la red social

Twitter, permitiéndole a los usuarios del sistema acceder a dichos datos en búsqueda de información implícita gracias a la estructuración post-recuperación que se obtiene por medio de la ejecución de los módulos implementados.

Las tecnologías usadas en el desarrollo del sistema son:

- **JDom:** JDom [20] es una librería de Java que permite interactuar con archivos XML, usada para generar y editar los diferentes diccionarios que ofrece el sistema.
- **Twitter4j:** Twitter4j [21] es una librería para Java que permite integrar aplicaciones con el API de Twitter, completamente construida en Java y funcional en plataformas móviles Android. Twitter4j se usa en el módulo de recuperación de tweets para realizar las búsquedas por palabra clave y recuperación de los textos.
- **XML:** XML [22] es un lenguaje especializado para permitir el intercambio de información entre diferentes aplicaciones. XML es el formato elegido para la generación de los diccionarios del sistema.
- **Stanford Name Entity Recognizer:** El NER de Stanford [17] es una implementación de reconocimiento de nombres de entidades que clasifica y etiqueta entidades en un texto. Esta aplicación requiere de entrenamiento si se emplea en alguno de los idiomas no soportados o si se pretenden modificar los tipos de entidades a reconocer. Se emplean sus métodos principales en el módulo de reconocimiento de nombres de entidades para reconocer y clasificar las entidades mencionadas en los tweets.
- **Stanford Part-of-Speech Tagger:** El POST de Stanford [16] es una pieza de software que lee un texto y le asigna etiquetas a cada palabra dependiendo de su clasificación gramatical, ya sea sustantivo, verbo, adjetivo, etc. Esta herramienta ofrece algunos modelos de reconocimiento para algunos idiomas; sin embargo, para poder procesar el idioma español la herramienta debe ser entrenada. Se emplea en el módulo de anotación gramatical para etiquetar gramaticalmente las palabras de los tweets.
- **Java:** Java [23] es un lenguaje de programación interpretado multiplataforma, soportado por Oracle.

3.5. Pruebas

Las pruebas realizadas al sistema para verificar el cumplimiento de los objetivos fueron pruebas de Sistema, de Integración y de Rendimiento.

Las pruebas de Sistema son similares a las pruebas de caja negra buscan probar que el sistema realice las acciones deseadas por el usuario.

3.5.1. Prueba de Sistema

A continuación se presenta alguna prueba de sistema correspondiente al módulo de recuperación de tweets, tabla 3.4.

3.5.1.1. Recuperación de Tweets

Prueba de Sistema			
Título		Búsqueda para obtener tweets	
Descripción de la prueba		Se realizará la ejecución de una libreria para el api de twitter, se parametrizara una búsqueda por medio de palabras y se obtendran tweets relacionados.	
ID			1
Fecha de realizaciòn			
Realizada por		Manuel Alejandro Alvarado	
Pre-requisitos		Api Twitter	
Detalles casos de prueba			
Accion Usuario		**Resultado esperado**	**Resultado obtenido**
Se ejecutará una llamada al api de Twitter		Conectado	Conectado
Parametrizar la búsqueda y obtener tweets.		Los resultados cumplen con los parámetros especificados.	Los resultados cumplen con los parámetros especificados.
Observaciones adicionales			
Aprovado	x	**No aprovado**	

Tabla 3.4: Prueba de Sistema: Búsqueda de tweets.

Las pruebas de Integración se realizan para probar que al realizar la combinación de las diferentes partes de la aplicación, en nuestro caso los módulos al ir siendo agregados al framework el

sistema debe tener un funcionamiento acertado.

3.5.2. Prueba de Integración

La siguiente es la prueba de integración correspondiente al sistema, tabla 3.5.

3.5.2.1. Sistema

Prueba de Integración		
Título	Ejecución de módulos desde el framework	
Descripción de la prueba	Demostrará el comportamiento individual de los modulos obteniendo y enviando datos al framework	
ID	7 - 2	
Fecha de realizaciòn		
Realizada por		
Pre-requisitos	Módulo eliminar palabras vacias, NER, POST, corrección de palabras, Recuperación	
Detalles casos de prueba		
Accion Usuario	**Resultado esperado**	**Resultado obtenido**
Ejecución completa módulo de recuperación	Tweets almacenados en la base de datos. Con la ejecución del modulo desde el framework	Tweets almacenados en la base de datos
Ejecución completa módulo de eliminar palabras vacías	Tweets actualizados en la base de datos sin las palabras vacias. Con la ejecución del módulo desde el framework	Tweets actualizados en la base de datos sin palabras vacías
Ejecución completa módulo de NER	Tweets actualizados en la base de datos con las palabras etiquetadas de las entidades personas y miselaneas. Con la ejecución del módulo desde el framework	Tweets actualizados en la base de datos con entidades Personas y Miselaneas etiquetadas
Ejecución completa módulo de POST	Tweets actualizados en la base de datos con las partes del texto etiquetadas. Con la ejecución del módulo desde el framework.	Tweets actualizados en la base de datos con etiquetas de partes de texto.
Ejecución completa módulo de corrección de palabras	Tweets actualizados en la base de datos sin las palabras vacias. Con la ejecución del módulo desde el framework.	Tweets actualizados en la base de datos con palabras corregidas
Observaciones adicionales		
Aprobado x	**No Aprobado**	

Tabla 3.5: Prueba de Sistema: Sistema.

Las pruebas de Rendimiento se realizan desde una perspectiva para verificar los atributos de la calidad del sistema.

3.5.3. Prueba de Rendimiento

A continuación se presenta la prueba de rendimiento realizada para el módulo de Reconocimiento de Nombres de Entidades. En la primera prueba presentada en la tabla 3.6, se muestran los resultados obtenidos de la anotación de 100 tweets con un modelo entrenado para reconocer 5 entidades. Como se puede notar, se obtuvo un 57.51 % de efectividad reconociendo correctamente las entidades.

3.5.3.1. Reconocimiento de Nombres de Entidades

NER			
Total			
LUGAR	29		
PERSONA	40		
USUARIO	71		
ORGANIZACIÓN	15		
MISC	38		
Total	**193**		
Reconocidas		**Error**	**Aciertos**
LUGAR	53	82,76%	17,24%
PERSONA	18	55,00%	45,00%
USUARIO	40	43,66%	56,34%
ORGANIZACIÓN	16	6,67%	93,33%
MISC	45	18,42%	81,58%
Total	**172**	10,88%	89,12%
Reconocidas correctamente			
LUGAR	26	10,34%	89,66%
PERSONA	18	55,00%	45,00%
USUARIO	39	45,07%	54,93%
ORGANIZACIÓN	5	66,67%	33,33%
MISC	23	39,47%	60,53%
Total	**111**	42,49%	57,51%
Reconocidas incorrectamente			
LUGAR	27		
PERSONA	23		
USUARIO	31		
ORGANIZACIÓN	11		
MISC	20		
Total	**112**		

Tabla 3.6: Prueba de Rendimiento uno Reconocimiento de Entidades

Para la segunda prueba, referenciada en la tabla 3.7, se resolvió reducir el modelo para reconocer dos entidades, con esta modificación, se logró conseguir una efectividad del 65.38 % en la correcta clasificación de las entidades.

NER			
Total			
PERSONA	70		
MISC	85		
Total	**155**		
Reconocidas		**Error**	**Aciertos**
PERSONA	30	57.14%	42.86%
MISC	100	17.65%	82.35%
Total	**130**	16.13%	83.87%
Reconocidas correctamente			
PERSONA	30	0.00%	100.00%
MISC	55	45.00%	55.00%
Total	**85**	34.62%	65.38%
Reconocidas incorrectamente			
PERSONA	0		
MISC	30		
Total	**30**		

Tabla 3.7: Prueba de Rendimiento dos Reconocimiento de Entidades

Las demás pruebas se encuentran en el documento de anexos. Apéndice D.

Capítulo 4

Conclusiones y Trabajo Futuro

4.1. Conclusiones

- Es posible integrar diferentes herramientas que ejecutan subtareas de la extracción de información y el procesamiento de lenguaje natural para realizar procesos básicos con estas. Sin embargo, para obtener información implicita de los datos preprocesados se requiere de un análisis adicional que permita relacionar dichos datos y concluir nueva información.

- La documentación de las aplicaciones de software, especialmente las que se encuentran bajo licencias libres, son muy importantes para permitirle a otros usuarios el correcto uso de las mismas o incluso su crecimiento, ya sea para mejorar sus resultados o abarcar nuevos problemas.

- En el campo del procesamiento de lenguaje natural son de mucha utilidad los diccionarios de palabras o entidades ya que funcionan como punto de referencia y se establece que su contenido es una verdad absoluta en el sistema.

- Uno de los problemas más difíciles de tratar con datos recuperados de redes sociales es la modificación del idioma escrito, donde se encuentran por ejemplo, textos con tantas abreviaciones que es imposible encontrar un sentido sintáctico debido a que se pretende procesar lo que dice el texto de manera literal.

- El entrenamiento de un modelo para el etiquetado automático de palabras requiere de grandes cantidades de información correctamente anotada para lograr resultados satisfactorios.

4.2. Trabajo Futuro

- Dadas las propiedades de los módulos de reconocimiento de nombres de entidades y anotación gramatical, es posible establecer relaciones entre entidades e incluso generar un grafo de relaciones de las entidades encontradas.
- Implementar nuevas reglas para la corrección de palabras mal escritas con el objetivo de mejorar el porcentaje de error presentado con las actuales restricciones del sistema.
- Implementar un módulo de reportes estadísticos en tiempo real de las entidades encontradas por el módulo de recuperación.
- Mejorar los modelos de entrenamiento para los módulos de NER y POST de modo que se etiquete correctamente un porcentaje más alto de palabras y entidades en su categoría correspondiente.
- Generar nuevos modelos para el reconocimiento y clasificación de entidades con el objetivo de reconocer más tipos de estas y enriquecer la estructura de los datos procesados.
- Integrar módulos de procesamiento de lenguaje natural que permitan trabajar con el aspecto semántico del idioma, como bien lo puede ser un corrector semántico, que mejoraría la calidad de los textos procesados.

Referencias

[1] Pyle D. Data Preparation for Data Mining. Morgan Kaufmann Publishers; 1999.

[2] Fumero A. y Roca G. Web 2.0. Fundación Orange; 2007.

[3] Watson, Mark. Scripting Intelligence, Web 3.0 Information Gathering and Processing. USA: Apress; 2009.

[4] Schwaber K. y Sutherland J. La guía de Scrum. USA: Scrum.org; 2011.

[5] Castro D. Métodos para la corrección ortográfica automática del español [Tesis de maestría] Santiago de Cuba: Universidad de Oriente. Facultad de Matemática y Computación. 2012.

[6] Singhal A. Modern Information Retrieval: A Brief Overview. IEEE Computer Society Technical Committee on Data Engineering; 2001. p. 35–43.

[7] Falahati M. THE CONCEPT OF STOPWORDS IN PERSIAN CHEMISTRY ARTICLES: A DISCUSSION IN AUTOMATIC INDEXING. University of Mysore. p. 2.

[8] Wasserman S. y Faust K. Social Network Analysis in the Social and Behavioral Sciences. Cambridge University Press: 1994. p. 1–27.

[9] National Electronics and Computer Technology Center (NECTEC). Social-based traffic information extraction and classification. 2011.

[10] Do L. y Leser U. EquatorNLP: Pattern-based Information Extraction for Disaster Response. Berlin: Universidad de Humblodt. 2011.

[11] Ruch P. Using contextual spelling correction to improve retrieval effectiveness in degraded text collection. Suiza: Swiss Federal Institute of Technology.

[12] Temple, Krystal. What Happens in an Internet Minute? [Artículo en línea]. Intel: Inside Scoop. URL: http://scoop.intel.com/what-happens-in-an-internet-minute/ [Consultado: 2013-06-02].

[13] Jáuregui, Pablo. La web del futuro pondrá orden al caos actual de la información en Internet. [Artículo en línea]. El Mundo. URL: http://www.elmundo.es/elmundo/2009/04/22/navegante/1240412890.html [Consultado: 2013-06-02].

[14] Consortium on Cognitive Science Instruction. Introduction to Natural Language Processing [Artículo en línea] URL: http://www.mind.ilstu.edu/curriculum/protothinker/natural_language_processing.php [Consultado: 2013-09-29].

[15] «Stanford Natural Language Processing Group» [En línea] URL: http://nlp.stanford.edu [Consultado: 2012-06-04]

[16] «Stanford Natural Language Processing Group» Stanford Log-linear Part-Of-Speech Tagger [En línea] URL: http://nlp.stanford.edu/software/tagger.shtml [Consultado: 2012-06-04]

[17] «Stanford Natural Language Processing Group» Stanford Named Entity Recognizer (NER) [En línea] URL: http://nlp.stanford.edu/software/CRF-NER.shtml [Consultado: 2012-06-04]

[18] «¿Qué es Software Libre? - Proyecto GNU - Free Software Foundation» [En línea] URL: http://www.gnu.org/philosophy/free-sw.html [Consultado: 2012-06-04]

[19] «Romance languages - Wikipedia, the free encyclopedia» [En línea] URL: http://en.wikipedia.org/wiki/Romance_languages [Consultado: 2013-10-10]

[20] «JDOM» [En línea] URL: http://www.jdom.org/ [Consultado: 2013-10-27]

[21] «Twitter4J - A Java library for the Twitter API» [En línea] URL: http://twitter4j.org/en/index.html [Consultado: 2013-10-27]

[22] «Extensible Markup Language (XML)» [En línea] URL: http://www.w3.org/XML/ [Consultado: 2013-10-27]

[23] «Java.com: Java y Tú» [En línea] URL: http://www.java.com/es/ [Consultado: 2013-10-27]

[24] «Qué es SCRUM — Proyectos Ágiles» [En línea] URL: http://www.proyectosagiles.org/que-es-scrum [Consultado: 2013-11-06]

[25] «Rational Team Concert for Scrum Projects - SCRUM como metodología» [En línea] URL: https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/Rational+Team+Concert+for+Scrum+Projects/page/SCRUM+como+metodolog%C3%ADa [Consultado: 2013-11-06]

[26] «Diccionario de la Lengua Española — Real Academia Española — Corpus» [En Línea] URL: http://lema.rae.es/drae/?val=corpus [Consultado: 2013-12-09]

ANEXOS

Apéndice A

Historias de Usuario

En este Anexo, se describen las historias de usuario definidas para el desarrollo de la aplicación. Se define la estructura COMO, QUIERO, PARA; donde la palabra COMO identifica al usuario del sistema, QUIERO define la acción o resultado esperado por el usuario y finalmente PARA indica el objetivo de dicha acción.

Historias de Usuario					
Titulo	Recuperación de tweets con palabras clave.				
ID	1	**Módulo**	Recuperación		
Descripción					
Como usuario del sistema quiero realizar búsquedas de tweets para recuperarlos y ejecutar otras tareas sobre ellos.					
Estimación	2	**Prioridad**	5	**Dependencia**	API twitter
Prueba de			Sistema		
Conexión con la red social twitter.					
Búsqueda de tweets por palabras dadas.					
Recuperación de tweets.					

Tabla A.1: HU1: Recuperación de tweets con palabras clave.

Historias de Usuario					
Titulo	Inserción de palabras en el diccionario de palabras vacías.				
ID	2.1.1	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero agregar una palabra en el diccionarios de palabras vacías para influir en los resultados obtenidos por el módulo de eliminación de palabras vacías					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Inserta palabras en el diccionario de palabras vacías.					

Tabla A.2: HU2: Inserción de palabras en el diccionario de palabras vacías.

Historias de Usuario					
Titulo	Consulta de palabras en el diccionario de palabras vacías.				
ID	2.1.2	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero consultar una palabra en el diccionarios de palabras vacías para conocer el estado del diccionario					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Consulta de palabras en el diccionario de palabras vacías					

Tabla A.3: HU3: Consulta de palabras en el diccionario de palabras vacías.

Historias de Usuario					
Titulo	Eliminación de palabras en diccionario de palabras vacías.				
ID	2.1.3	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero eliminar una palabra en el diccionarios de palabras vacías para influir en los resultados obtenidos por el módulo de eliminación de palabras vacías					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Eliminación de palabras en el diccionario de palabras vacías					

Tabla A.4: HU4: Eliminación de palabras en diccionario de palabras vacías.

Historias de Usuario					
Titulo	Inserción de palabras en el diccionario de palabras en español.				
ID	2.2.1	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero agregar una palabra en el diccionarios de palabras en español para influir en los resultados obtenidos por el módulo de corrección de palabras					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Inserta palabras en el diccionario de palabras en español.					

Tabla A.5: HU5: Inserción de palabras en el diccionario de palabras en español.

Historias de Usuario					
Titulo	Consulta de palabras en el diccionario de palabras en español.				
ID	2.2.2	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero consultar una palabra en el diccionarios de palabras en español para conocer el estado del diccionario					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Consulta de palabras en los diccionarios.					

Tabla A.6: HU6: Consulta de palabras en el diccionario de palabras en español.

Historias de Usuario					
Titulo	Eliminación de palabras en diccionario de palabras en español.				
ID	2.2.3	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero eliminar una palabra en el diccionarios de palabras vacías para influir en los resultados obtenidos por el módulo de corrección de palabras					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Eliminación de palabras en el diccionario de palabras en español					

Tabla A.7: HU7: Eliminación de palabras en diccionario de palabras en español.

Historias de Usuario					
Titulo	Inserción de palabras en el diccionario de expresiones populares.				
ID	2.3.1	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero agregar una expresión popular en el diccionarios de expresiones populares para influir en los resultados obtenidos por el módulo de corrección de palabras					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Inserción de palabras en el diccionario de expresiones populares					

Tabla A.8: HU8: Inserción de palabras en el diccionario de expresiones populares.

Historias de Usuario					
Titulo	Consulta de palabras en el diccionario de expresiones populares.				
ID	2.3.2	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero consultar una expresiones populares en el diccionarios de palabras en español para conocer el estado del diccionario					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Consulta de palabras en el diccionario de expresiones populares.					

Tabla A.9: HU9: Consulta de palabras en el diccionario de expresiones populares.

Historias de Usuario					
Titulo	Eliminación de palabras en diccionario de expresiones populares.				
ID	2.3.3	**Módulo**	Sistema		
Descripción					
Como usuario del sistema quiero eliminar una expresión popular en el diccionarios de expresiones populares para influir en los resultados obtenidos por el módulo corrección de palabras					
Estimación	1	**Prioridad**	3	**Dependencia**	Ninguna
Prueba de			Sistema		
Eliminación de palabras en el diccionario de expresiones populares					

Tabla A.10: HU10: Eliminación de palabras en diccionario de expresiones populares.

Historias de Usuario					
Titulo	Corrección de palabras mal escritas				
ID	3	**Módulo**	Corrección de palabras mal escritas.		
Descripción					
Como usuario quiero que el sistema realice la corrección automática de palabras mal escritas en los textos recuperados para mejorar la comprensión y estructura gramatical de los mismos.					
Estimación	3	**Prioridad**	4	**Dependencia**	Diccionario de palabras en español
Prueba de			Rendimiento		
Porcentaje de errores de palabras corregidas.					

Tabla A.11: HU11: Corrección de palabras mal escritas.

Historias de Usuario					
Titulo	Anotación gramatical.				
ID	4	**Módulo**	POS-Tagging		
Descripción					
Como usuario quiero que el sistema etiquete las palabras con respecto a su clasificación gramatical para estructurar el texto.					
Estimación	2	**Prioridad**	4	**Dependencia**	Ninguna
Prueba de			Rendimiento		
Porcentajes de aciertos en la identificación de partes de texto en español					

Tabla A.12: HU12: Anotación gramatical.

Historias de Usuario					
Titulo	Eliminación de palabras vacías				
ID	5	**Módulo**	Eliminación de palabras vacías.		
Descripción					
Como usuario quiero que el sistema elimine del texto las palabras consideradas vacías para identificar información importante contenida en él.					
Estimación	2	**Prioridad**	3	**Dependencia**	Diccionario de palabras vacías
Prueba de			Rendimiento		
Porcentaje de aciertos en la identificación de palabras vacías					

Tabla A.13: HU13: Eliminación de palabras vacías.

Historias de Usuario					
Titulo	Reconocimiento de nombres de entidades.				
ID	6	**Módulo**	NER		
Descripción					
Como usuario quiero que el sistema etiquete y clasifique las entidades presentes en el texto para identificar qué entidades tienen relación con las palabras clave de búsqueda y recuperación.					
Estimación	1	**Prioridad**	4	**Dependencia**	Ninguna
Prueba de			Rendimiento		
Porcentajes de aciertos en identificación de entidades misceláneas y personas en el idioma español					

Tabla A.14: HU14: Reconocimiento de nombres de entidades.

Historias de Usuario					
Titulo	Framework				
ID	7	Sistema			
Descripción					
Como usuario quiero que el sistema me permita ejecutar módulos de preprocesamiento de texto para obtener resultados estructurados.					
Estimación	1	**Prioridad**	5	**Dependencia**	Módulos
Prueba de			Integración - Rendimiento		
Comportamiento individual de los módulos con el framework					

Tabla A.15: HU15: Framework.

Apéndice B

Diagramas de Clases

A continuación se presentan los diagrams de clases del sistema y sus módulos.

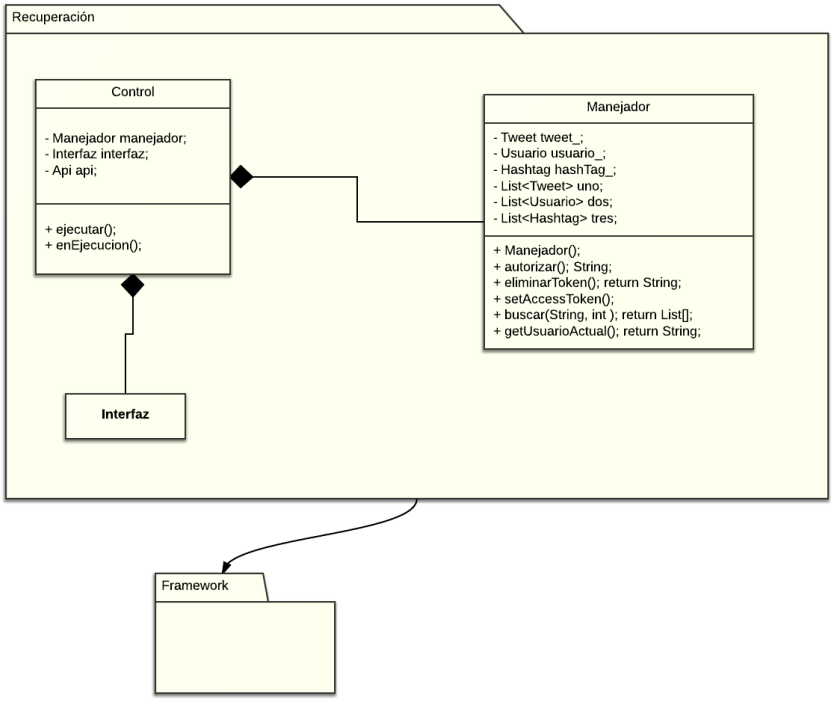

Figura B.1: DC1: Diagrama de clases, módulo de recuperación de tweets.

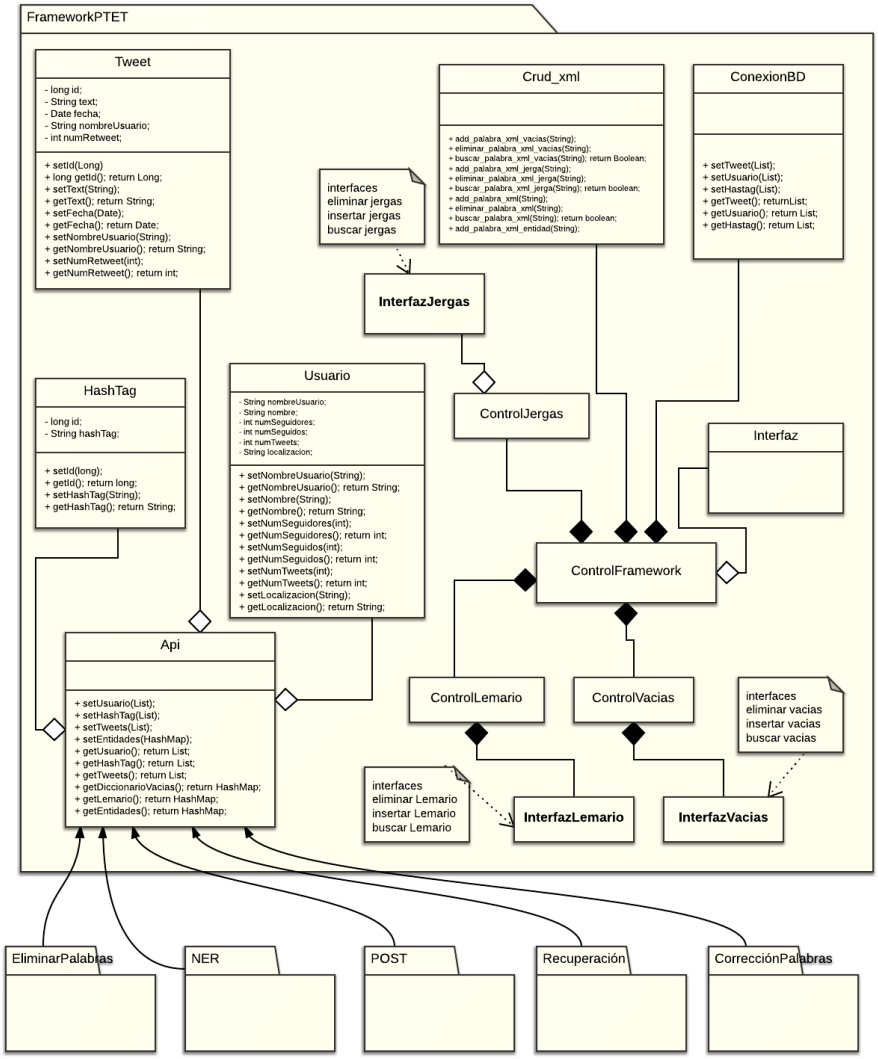

Figura B.2: DC2: Diagrama de clases, sistema central.

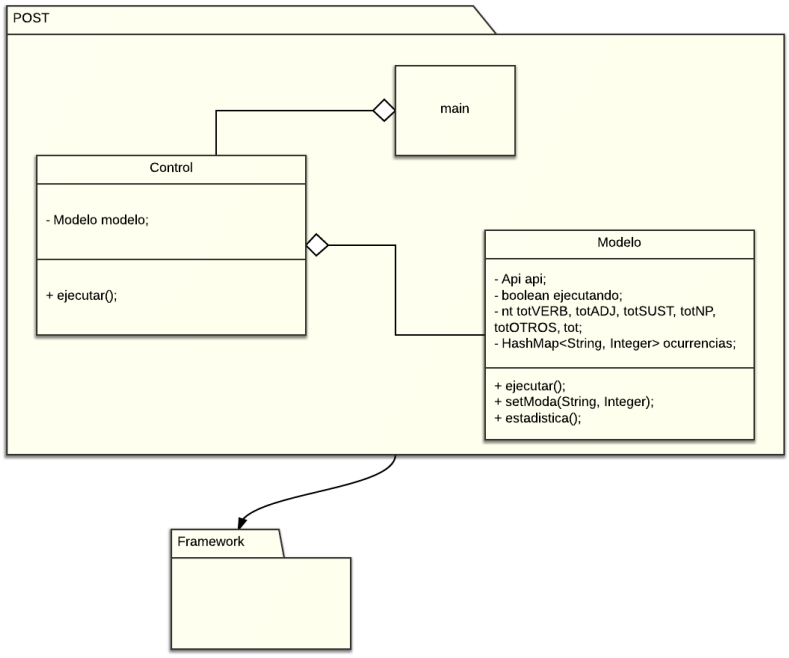

Figura B.3: DC3: Diagrama de clases, módulo anotación gramatical.

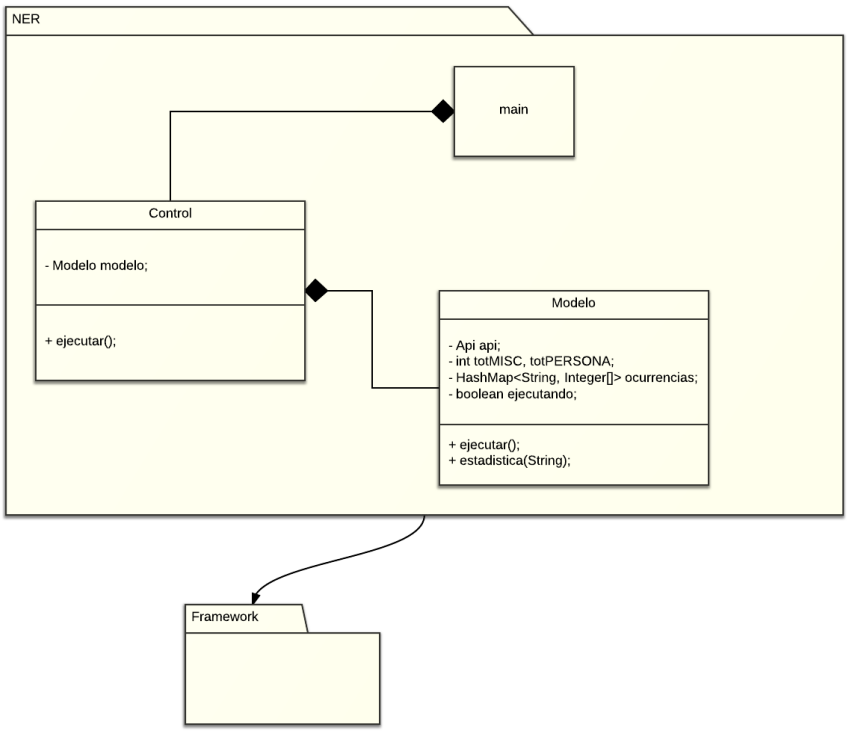

Figura B.4: DC4: Diagrama de clases, módulo de reconocimiento de nombres de entidades.

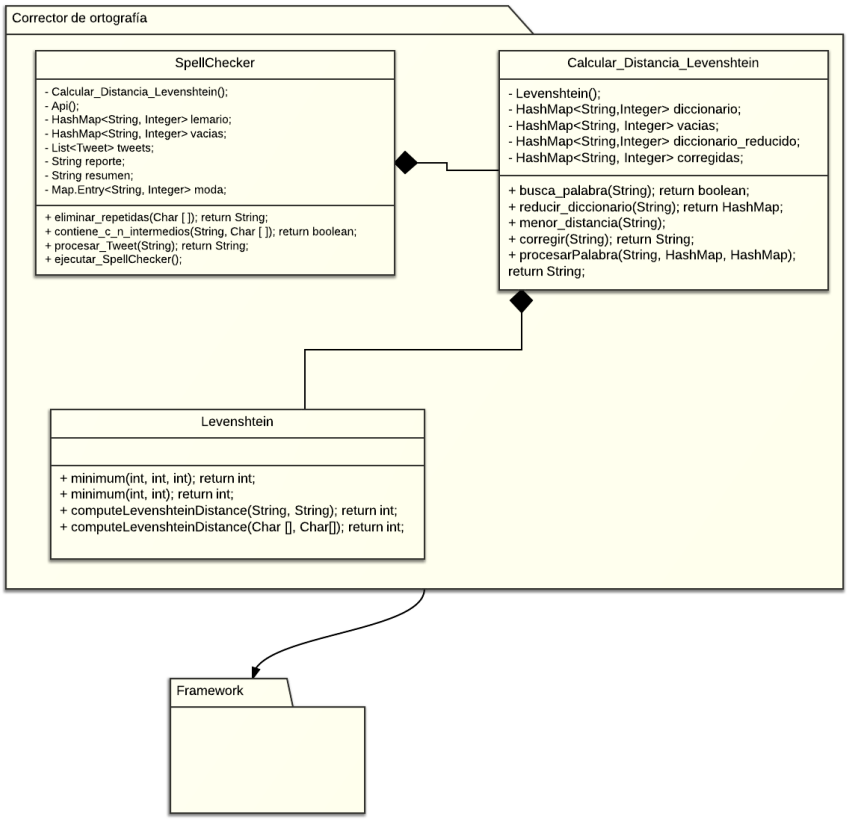

Figura B.5: DC5: Diagrama de clases, módulo de corrección automática de palabras mal escritas.

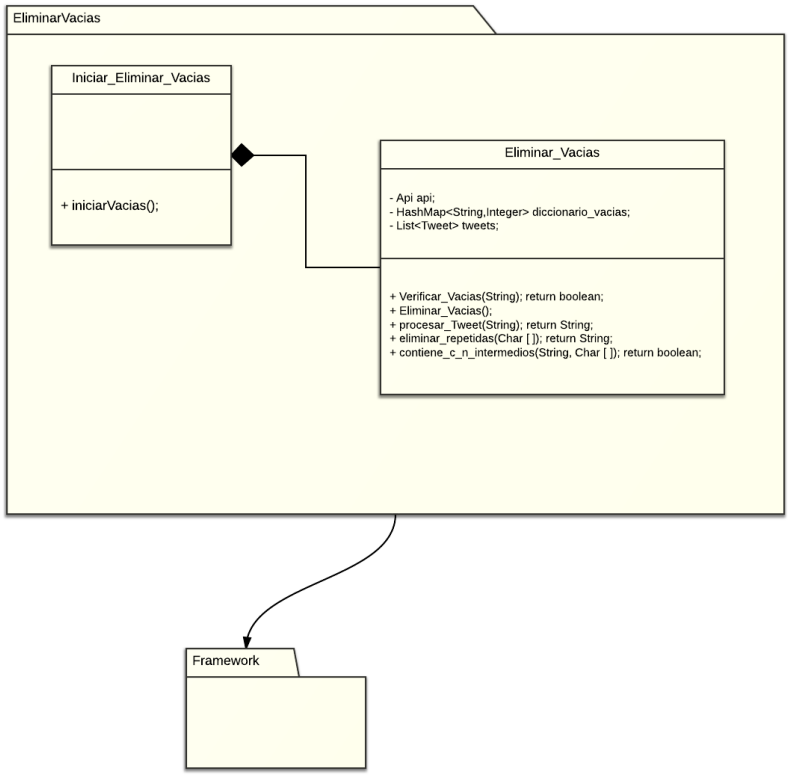

Figura B.6: DC6: Diagrama de clases, módulo de eliminación de palabras vacías.

Apéndice C

Diagramas de Secuencia

A continuación se presentan los diagramas de secuencia de los diferentes módulos del sistema.

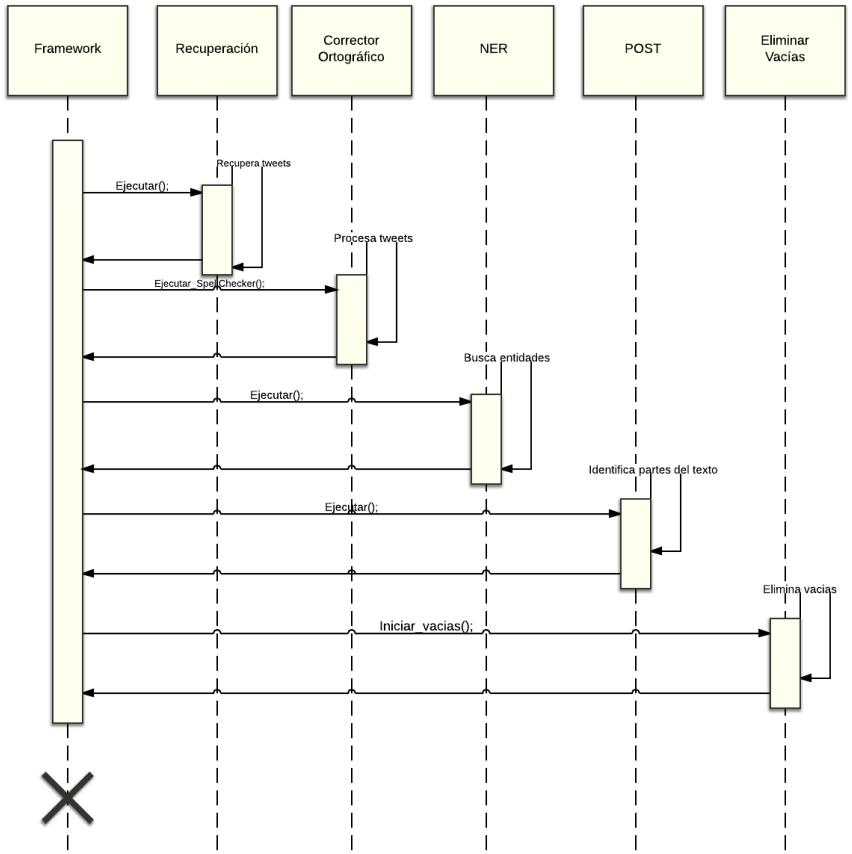

Figura C.1: DS1: Diagrama general de secuencia.

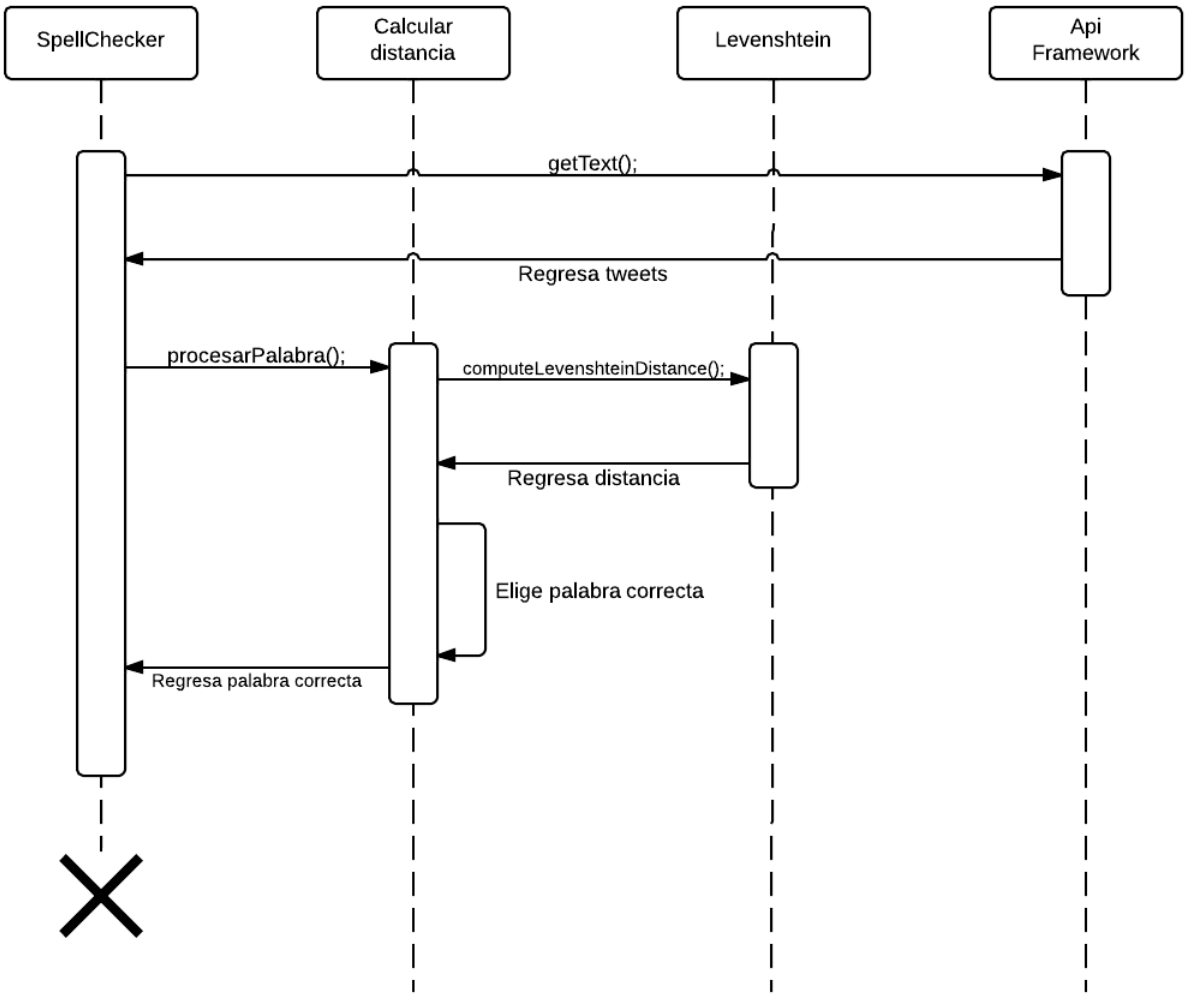

Figura C.2: DS1: Diagrama de secuencua: Módulo de corrección automática de palabras mal escritas.

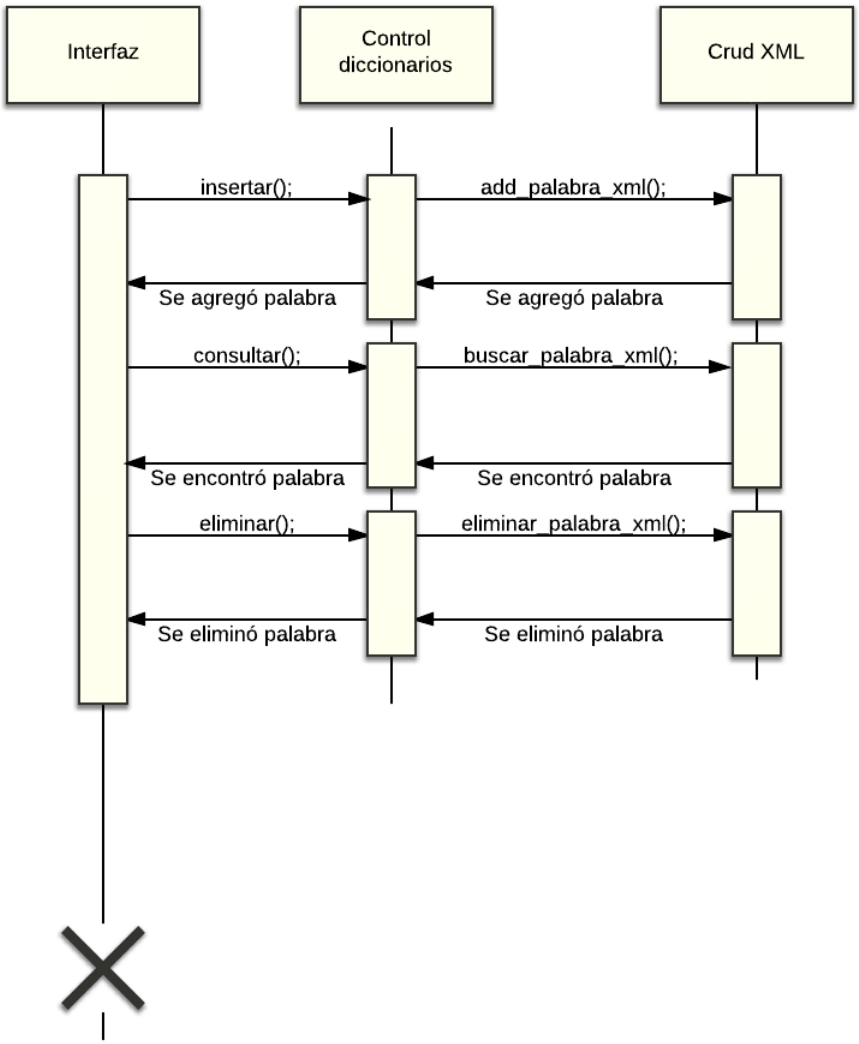

Figura C.3: DS1: Diagrama de secuencua: Interacción con los Diccionarios del Sistema.

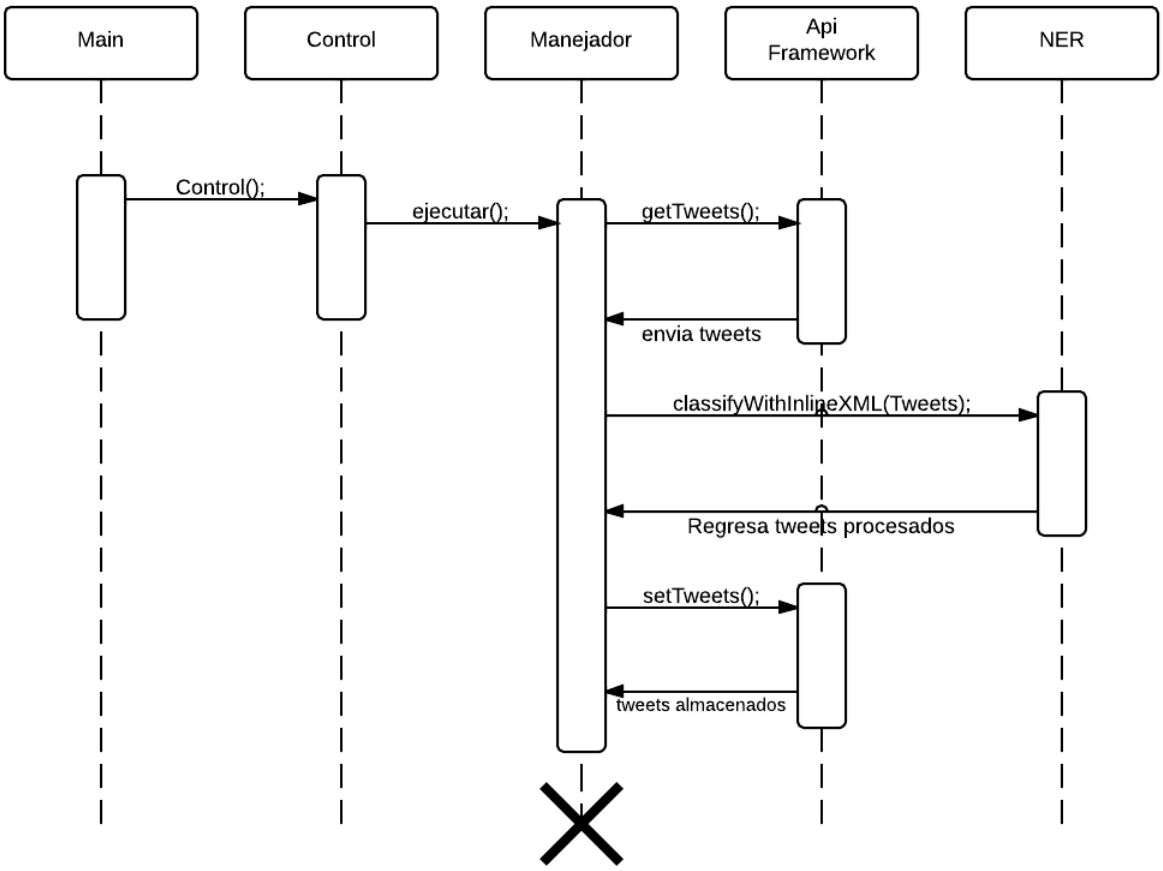

Figura C.4: DS1: Diagrama de secuencua: Módulo de Reconocimiento de Nombres de Entidades.

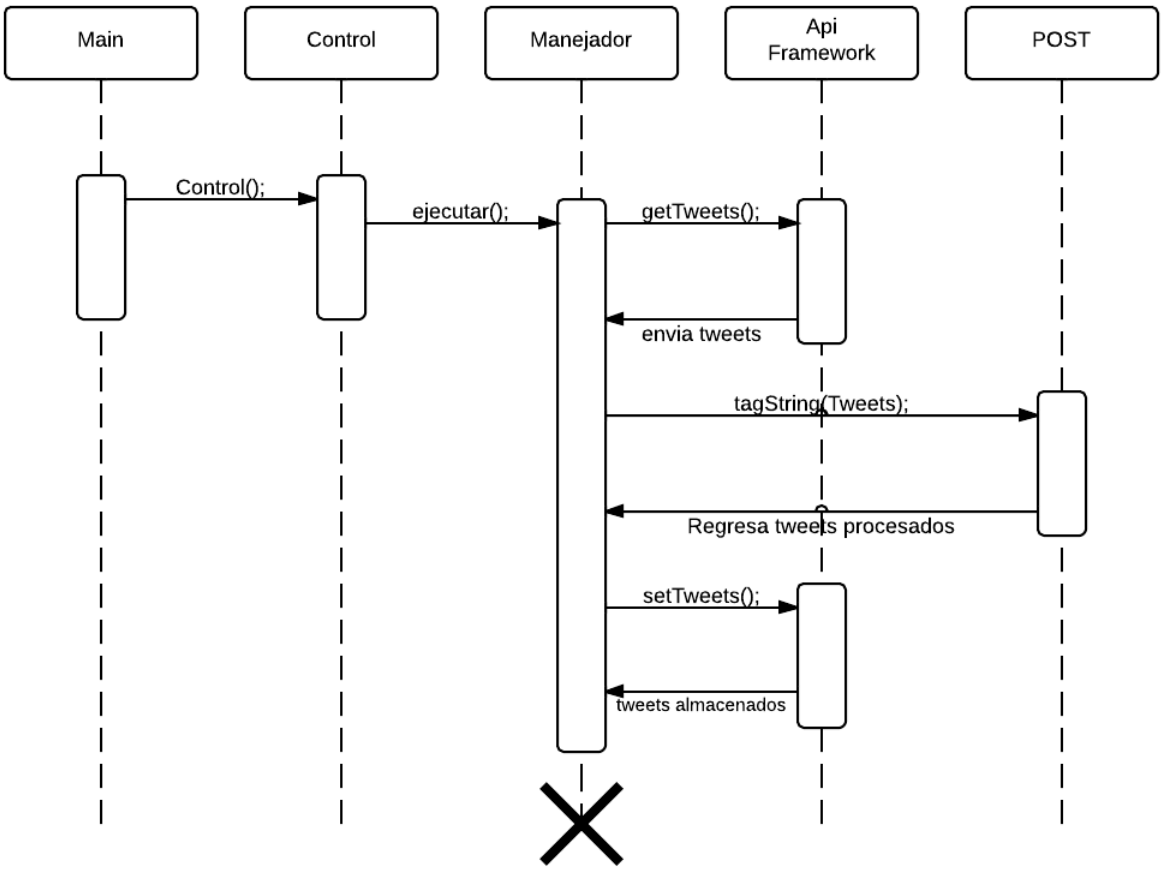

Figura C.5: DS1: Diagrama de secuencua: Módulo de Anotación Gramatical.

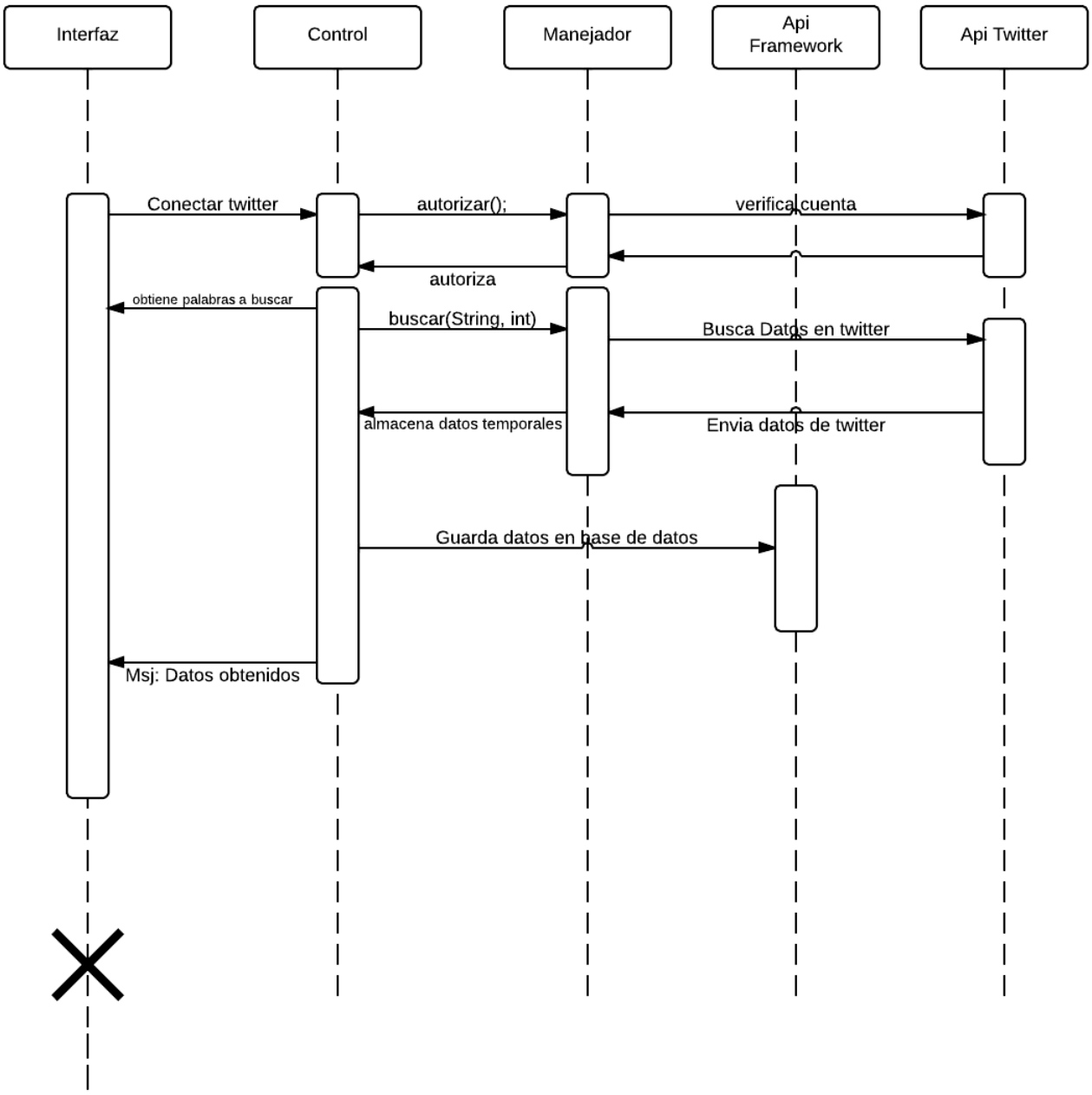

Figura C.6: DS1: Diagrama de secuencua: Módulo de Recuperación de Tweets.

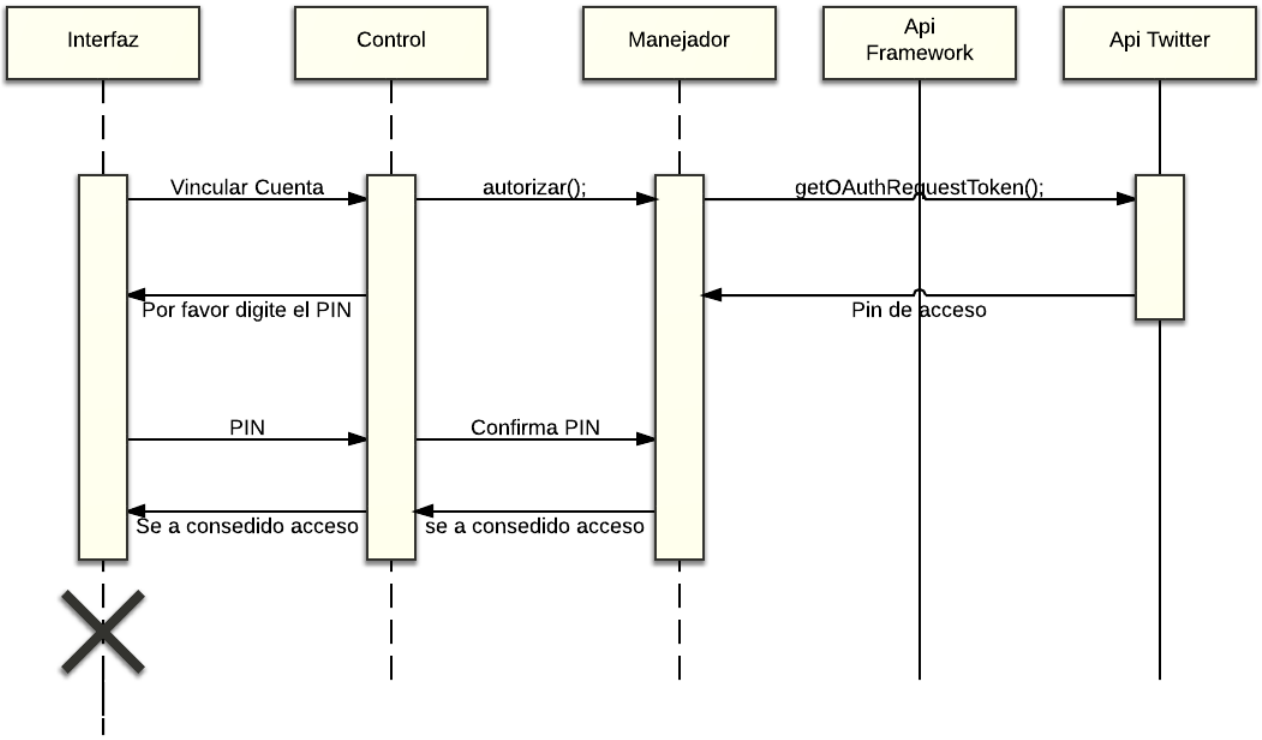

Figura C.7: DS1: Diagrama de secuencua: Módulo de Recuperación de Tweets, vincular cuenta.

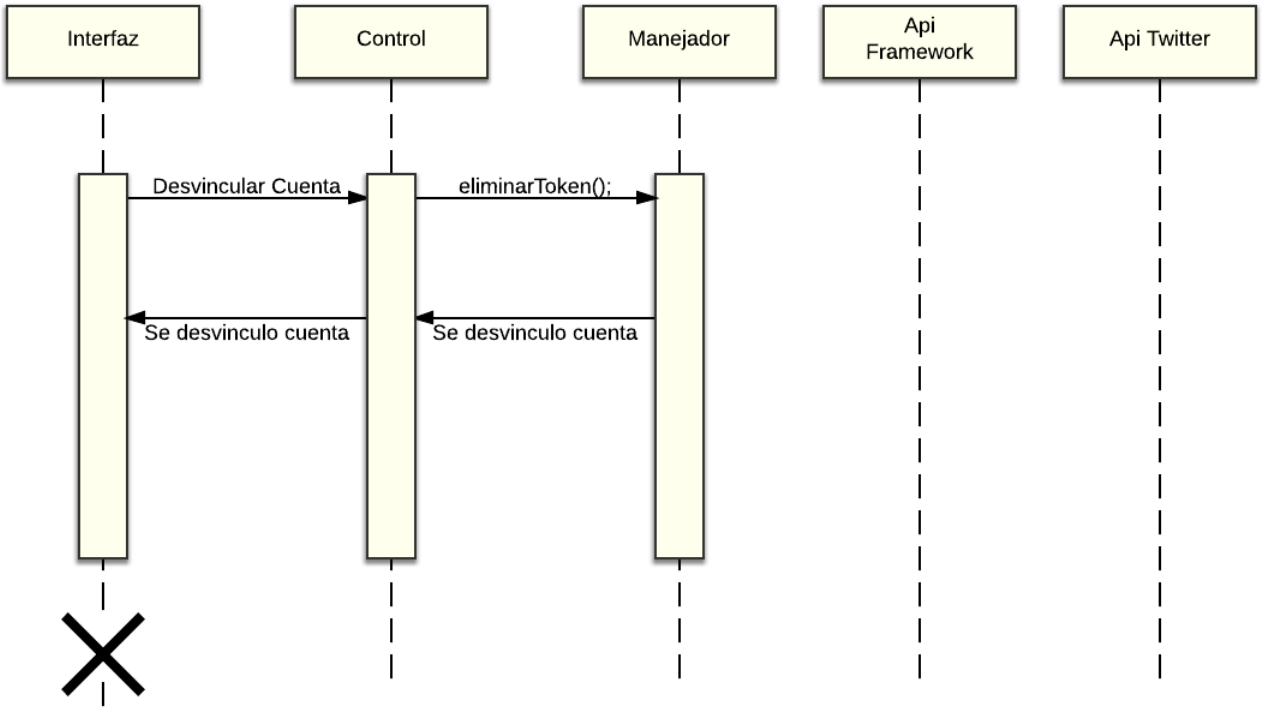

Figura C.8: DS1: Diagrama de secuencua: Módulo de Recuperación de Tweets, desvincular cuenta.

Apéndice D

Pruebas

Se realizaron pruebas de Sistema, Integración y Rendimiento.

Prueba de Sistema			
Título		Búsqueda para obtener tweets	
Descripción de la prueba		Se realizará la ejecución de una libreria para el api de twitter, se parametrizara una búsqueda por medio de palabras y se obtendran tweets relacionados.	
ID			1
Fecha de realizaciòn			
Realizada por		Manuel Alejandro Alvarado	
Pre-requisitos		Api Twitter	
Detalles casos de prueba			
Accion Usuario		**Resultado esperado**	**Resultado obtenido**
Se ejecutará una llamada al api de Twitter		Conectado	Conectado
Parametrizar la búsqueda y obtener tweets.		Los resultados cumplen con los parámetros especificados.	Los resultados cumplen con los parámetros especificados.
Observaciones adicionales			
Aprovado	x	**No aprovado**	

Tabla D.1: P1: Búsqueda para obtener tweets.

Prueba de Sistema			
Título		Inserción, consulta y eliminación de palabras en diccionarios.	
Descripción de la prueba		Se realizará la inserción, eliminación y consulta de palabras a los diccionario	
ID		2	
Fecha de realización		2012/12/10	
Realizada por		Manuel Alejandro Hurtado Sarria	
Pre-requisitos		Ninguno	
Detalles casos de prueba			
Acción Usuario		**Resultado esperado**	**Resultado obtenido**
Insertar, consultar y eliminar palabra al diccionario de palabras vacías		inserción realizada, consulta encontrada, eliminar palabra	Proceso realizado
Insertar, consultar y eliminar palabra al diccionario de palabras en español		inserción realizada, consulta encontrada, eliminar palabra	Proceso realizado
Insertar, consultar y eliminar palabra al diccionario de jergas		inserción realizada, consulta encontrada, eliminar palabra	Proceso realizado
Observaciones adicionales			
Aprobado	X	**No aprobado**	

Tabla D.2: P2 : Inserción, consulta y eliminación de palabras en diccionarios.

Pruebas de rendimientos		
Título	Eliminacón de palabras vacías	
Descripción de la prueba	Se determinará la cantidad de palabras correctamente eliminadas	
ID	5	
Fecha de realizaciòn	12/20/2012	
Realizada por	Manuel Alejandro Hurtado Saria	
Pre-requisitos	Diccionario de Palabras Vacías - Recuperación tweets	
Detalles casos de prueba		
Accion Usuario	**Resultado esperado**	**Resultado obtenido**
Se realizarán la ejecución del módulo con 100 tweets	Esperado un 60 % de corregidas	Se obtuvo un porcentaje de acierto del 75 %
Observaciones adicionales		
El porcentaje de acierto esta relacionado a la cantidad de palabras indexadas en el diccionario de palabras vacías		

Tabla D.3: P3 : Eliminación de Palabras Vacías.

Prueba de Rendimiento		
Título	Corrección de palabras mal escritas	
Descripción de la prueba	Se determinará la cantidad de palabras corregidas correctamente	
ID	3	
Fecha de realización	2013/2/15	
Realizada por	Manuel Alejandro Hurtado Sarria	
Pre-requisitos	Diccionario de Palabras en español - Recuperación de tweets	
Detalles casos de prueba		
Acción Usuario	**Resultado esperado**	**Resultado obtenido**
Se realizarán la ejecución del módulo con 100 tweets	Se espera un rendimiento del 60 % de corregidas	Se obtuvo un porcentaje de acierto del 64%
Observaciones adicionales		
El porcentaje de acierto esta relacionado a la cantidad de palabras indexadas en el diccionario de palabras en español		

Tabla D.4: P4 : Corrección de palabras mal escritas.

Prueba de Rendimiento		
Título	Anotación gramatical	
Descripción de la prueba	Se determinará la cantidad de palabras etiquetadas correctamente	
ID	4	
Fecha de realización	2013/08/3	
Realizada por	Manuel Alejandro Alvarado Cobo	
Pre-requisitos	Recuperación de tweets	
Detalles casos de prueba		
Acción del Sistema	**Resultado esperado**	**Resultado obtenido**
Se realizará la ejecución del modulo con 100 tweets	Se espera un acierto superior al 60 % de palabras correctamente etiquetadas	Se obtuvo un porcentaje de acierto del 78%
Observaciones adicionales		

Tabla D.5: P5 : Anotación gramatical.

Prueba de Rendimiento		
Título	Reconocimiento de nombres de entidades	
Descripción de la prueba	Se determinará la cantidad de entidades etiquetadas correctamente.	
ID	5	
Fecha de realización	2013/05/15	
Realizada por	Manuel Alejandro Alvarado Cobo	
Pre-requisitos	Recuperación de tweets	
Detalles casos de prueba		
Acción del Sistema	**Resultado esperado**	**Resultado obtenido**
Se realizará la ejecución del modulo con 100 tweets	Se espera un acierto superior al 60 % de entidades correctamente etiquetadas	Se obtuvo un porcentaje de acierto del 57.51%
Observaciones adicionales		
Se realizaron pruebas adicionales con el objetivo de mejorar el reconocimiento de entidades. Para ver los resultados refiérase a las tablas 3.2 y 3.3 del documento principal.		

Tabla D.6: P6 : Reconocimiento de Nombres de Entidades.