



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**
título del TFG



Presentado por Pablo Santidrián Tudanca
en Universidad de Burgos — 16 de marzo
de 2020

Tutor: José Francisco Díez Pastor y Álgvar
Arnaiz Gonzalez



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. nombre tutor, profesor del departamento de nombre departamento, área de nombre área.

Expone:

Que el alumno D. Nombre del alumno, con DNI 71362353T, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado título de TFG.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 16 de marzo de 2020

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. José Francisco Díez Pastor

D. Álar Arnaiz Gonzalez

Resumen

En este primer apartado se hace una **breve** presentación del tema que se aborda en el proyecto.

Descriptores

Palabras separadas por comas que identifiquen el contenido del proyecto Ej: servidor web, buscador de vuelos, android ...

Abstract

A **brief** presentation of the topic addressed in the project.

Keywords

keywords separated by commas.

Índice general

Índice general	III
Índice de figuras	V
Índice de tablas	VI
Introducción	1
1.1. Estructura de la memoria	2
Objetivos del proyecto	3
Conceptos teóricos	5
3.1. Minería de datos	5
Técnicas y herramientas	7
4.1. Gestión del proyecto	7
4.2. Entorno de desarrollo	8
4.3. Documentación	9
4.4. Bibliotecas	10
Aspectos relevantes del desarrollo del proyecto	11
5.1. Inicio del proyecto	11
5.2. Metodologías	11
5.3. Formación	12
5.4. Recogida de datos	12
Trabajos relacionados	13

Conclusiones y Líneas de trabajo futuras	15
Bibliografía	17

Índice de figuras

3.1. Fases del proceso de minería de datos.[18]	6
---	---

Índice de tablas

Introducción

El cambio climático está provocando que los fenómenos naturales extremos sean cada vez más habituales afectando a la poblaciones locales. Una de estas poblaciones locales son la medusas.

Las medusas tienen periodos de aparición estacionales y se alimentan de plancton por lo que su densidad es mayor en zonas donde este abunda. Estas zonas suelen ser lugares cercanos al talud continental donde además se reproducen [4]. Las medusas son organismos asexuales, un mismo individuo puede generar descendencia y esta se lleva a cabo mediante la fecundación de gametos que se convertirán en larvas, más adelante estas se adherirán a alguna superficie donde se transformarán en pólipos para finalmente separará la medusa adulta.[13]

La aparición de medusas cerca de las costas, es un fenómeno que se da cada vez con mayor frecuencia. Estas floraciones tiene efectos perjudiciales en ámbitos como el turismo o la pesca, así como los daños que pueden provocar a la salud de las personas llegando en algunos casos a causar enfermedades graves [19, 20].

Alguno de los factores que están provocando el aumento de estos acercamientos son[4, 17]:

- En la **climatología** influye principalmente el cambio climático con el descenso del nivel de lluvias y el aumento de la temperatura, que favorece el aumento de la salinidad y de la temperatura del agua.
- La **contaminación** provocada por la modificación de las zonas costeras o los vertidos cercanos a los costas provocan la proliferación de bacterias o plancton que sirve de alimento para las medusas.

- La **sobrepesca** causa un descenso de depredadores así como de otras especies con las que las medusas compiten por el alimento.

Con este proyecto se pretende predecir el comportamiento de las poblaciones de medusas en las costas de Chile en función de datos meteorológicos y marítimos obtenidos mediante el programa europeo *Copernicus*. Estos datos se tratarán para eliminar la información que no es útil y delimitar la zona geográfica de estudio. A partir de ahí se entrenará a una serie de modelos para predecir el comportamiento de las medusas.

1.1. Estructura de la memoria

Objetivos del proyecto

Este apartado explica de forma precisa y concisa cuales son los objetivos que se persiguen con la realización del proyecto. Se puede distinguir entre los objetivos marcados por los requisitos del software a construir y los objetivos de carácter técnico que plantea a la hora de llevar a la práctica el proyecto.

Conceptos teóricos

3.1. Minería de datos

Actualmente se recopila una gran cantidad de información de todos los ámbitos y es necesario darla un uso práctico. La **minería de datos** es un campo de la ciencia de la computación por el cual se tratan de descubrir nuevos patrones o relaciones en conjuntos de datos y así, conseguir un conocimiento obtenido de manera automática (*Machine Learning*). Con estas nuevas relaciones se trata de explicar comportamientos actuales o predecir resultados futuros. [18]

Sin embargo, la minería de datos es solo una fase del proceso de descubrimiento del conocimiento (KDD) pues es necesario tratar los datos antes de analizarlos así como validarlos posteriormente. Este proceso podemos obtener las siguientes fases:

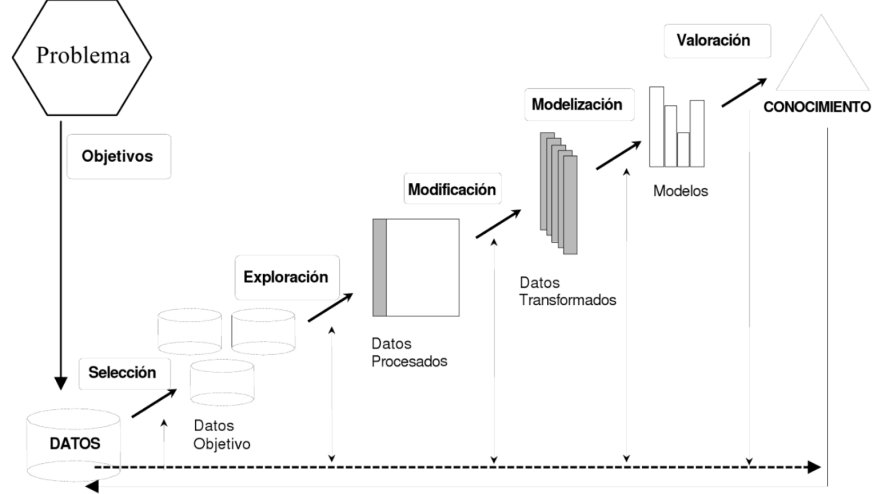


Figura 3.1: Fases del proceso de minería de datos.[18]

Preprocesamiento de los datos

Los datos obtenidos inicialmente no pueden ser utilizados directamente para la minería de datos. Estos datos deben ser preprocesados para eliminar posibles variables que no son necesarias o datos incorrectos y así conseguir transformar nuestro conjunto de datos iniciales en un conjunto más útil y menos pesado por lo que el algoritmo de análisis posterior, tendrá una menor carga de trabajo.

Machine learning

Técnicas y herramientas

Este apartado se plasman las diferentes técnicas metodológicas y herramientas de desarrollo que se han utilizado en la realización del proyecto, así como las posibles alternativas que se han tenido en cuenta y el motivo de haberlas desechado.

4.1. Gestión del proyecto

Scrum

Scrum se trata de un marco de trabajo ágil destinado al manejo de proyectos de desarrollo *software*. Está destinado para equipos pequeños dividiendo el trabajo en objetivos que se van logrando de manera incremental a través de iteraciones denominadas *sprints*. [21]

Git

Git se trata de un sistema de control de versiones gratuito y de código abierto para el manejo de proyecto. Se trata de un software de control de versiones de manera que se puede llevar un registro de cambios en los archivos y posibilita la coordinación de trabajos entre diferentes personas. [3, 15]

GitHub

GitHub es un plataforma destinada a alojar proyecto basándose en el software de control de versiones Git. Esta plataforma utiliza un interfaz web desde la que se nos permite realizar control de código, documentación,

gestión de tareas y otros muchas funcionalidades además de integración con otros servicios. GitHub es gratuito para proyectos *open source*. [16, 9]

Alternativas

Otras alternativas a GitHub fueron, Gitlab y Bitbucket. Ambos servicios son bastante similares a GitHub en funcionalidades y basados en Git.

Bitbucket fue rechazado rápidamente por la falta de familiaridad en su uso ya que no lo había usado nunca, únicamente había visto repositorios en la web.

Gitlab es un entorno más conocido ya que es el software que he utilizado en las prácticas de empresa para el control de código dentro del equipo de trabajo del que formo parte.

Finalmente se decidió usar GitHub por haber sido utilizado en clase de gestión de proyectos por lo que se tenía un mayor conocimiento de su funcionamiento, así como por su integración con ZenHub del que se hablará a continuación.

ZenHub

ZenHub es una herramienta para gestión de proyectos que se integra con GitHub. Este complemento añade a GitHub un tablero canvas en el cual se representan las *issues*. Es posible estimar tareas, así como darlas prioridades y visualizar gráficos como el gráfico *burndown*. [11]

Alternativas

Se plantearon otras alternativas como Trello o GitHub projects pero finalmente se eligió ZenHub por su facilidad de uso y el haber sido usado anteriormente, requisito que las otras alternativas no cumplían.

4.2. Entorno de desarrollo

Python

Para el desarrollo en Python se eligió Visual Studio Code. Esta es una aplicación totalmente gratuita basada en el framework Electron y posee gran cantidad de extensiones para facilitar la tarea a la hora de la programación como auto completado con IntelliSense. Además tiene integración con git para el control de versiones [8, 7].

Alternativas

Se estudiaron otras alternativas como PyCharm o Jupyter Notebook pero finalmente se eligió Visual Studio Code por la familiaridad con la misma y gran compatibilidad con diferentes lenguajes.

4.3. Documentación

LaTeX

LateX es un sistema de composición de texto plano destinado a la composición de textos con una calidad tipográfica alta. Incluye características diseñadas para la elaboración de documentación técnica y científica siendo un estándar en la publicación de documentos de investigación. Latex es totalmente gratuito [5].

Alternativas

Otras opciones valoradas fueron Microsoft Word y OpenOffice. La universidad proporciona una plantilla para LaTeX y OpenOffice por lo que Microsoft Word fue la primera descartada. Finalmente debido al enfoque más técnico que proporciona LaTeX frente a OpenOffice se decidió utilizarlo.

TeXstudio

TeXstudio se trata de un editor de textos gratuito, que ofrece diversas herramientas para elaborar documentación con LaTeX haciendo la escritura más confortable e intuitiva [6].

Alternativas

Las alternativas a este editor que se estudiaron fueron Texmaker y Overleaf. Se terminó eligiendo TeXstudio por ser un editor instado en local así como por ser más intuitivo y fácil de utilizar que Texmaker.

Zotero

La herramienta Zotero es un software gratuito para la gestión de referencias pudiendo recoger , organizar y citar creando referencias bibliográficas para cualquier editor. También cuenta con integración en le navegador. Una vez recopilados todas las citas, se puede exportar a un fichero BibTex para la utilizarse con LaTeX [12].

4.4. Bibliotecas

Flask

Flask es un framework ligero para el desarrollo de aplicaciones web en Python bajo el modelo Modelo-Vista-Controlador (MVC). Está diseñado para desarrollar aplicaciones de manera rápida y sencilla y con capacidad de escalar a aplicaciones más complejas [2].

Xarray

Xarray se trata de un proyecto de código abierto desarrollado para Python que facilita el trabajo con matrices multidimensionales etiquetadas utilizando la librería NumPy. Consta de una gran variedad de funciones para el análisis y la visualización de estructuras de datos. Está inspirado en el funcionamiento de la librería pandas y diseñado para funcionar con archivos de tipo netCDF [10].

Pandas

Esta librería se trata de una extensión de NumPy y está destinada a la manipulación y análisis de datos en lenguaje Python. Permite trabajar con estructuras de datos y operaciones para su transformación pudiendo estas ser tablas temporales o series numéricas [14].

FtpLib

Librería destinada a la implementación de la parte del cliente en el protocolo FTP. Desarrollada para el lenguaje Python, nos permite automatizar accesos a servidores FTP [1].

tqdm

Pequeña librería utilizada para mostrar una barra de progreso a la hora de realizar la descarga de los datos del FTP.

Aspectos relevantes del desarrollo del proyecto

5.1. Inicio del proyecto

5.2. Metodologías

En el desarrollo del proyecto se ha seguido una metodología de trabajo ágil, en concreto Scrum. Esta metodología se ha seguido con ciertas limitaciones al tratarse de un proyecto desarrollado por una sola persona.

- Se ha seguido una estrategia de desarrollo incremental mediante iteraciones o *Sprints*.
- La duración de estos *Sprints* fue de una o dos semanas.
- Al final de cada *Sprint* se realiza una reunión con los tutores para ver el avance y marcar nuevos objetivos que se estiman a través de un tablero canvas (*ZenHub*).
- El seguimiento de cada *Sprint* se realiza a través de los gráficos burn-down.

5.3. Formación

5.4. Establecimiento de requisitos

5.5. Recogida de datos

La obtención de los datos necesarios para el posterior análisis, se obtuvieron a través de la organización europea *Copernicus*. Esta cuenta con una serie de datos recopilados por satélites de todo el mundo.

Para la descarga de estos datos, se encontraban disponibles dos alternativas. Por un lado, podían ser descargados a través de un servidor FTP y por el otro, había disponible una API de reciente lanzamiento.

En un principio se priorizó la opción de la descarga a través de la API ya que podían descargarse los datos ya tratados reduciendo el trabajo previo al análisis. Finalmente se descartó esta idea por la lentitud de respuesta del servicio así como diferentes errores que se producían haciendo que la descarga de la totalidad de los datos no estuviese asegurada por lo que se descargaron los datos a través del servidor FTP y posteriormente eran tratados eliminando las variables y las zonas geográficas que no eran necesarias.

Trabajos relacionados

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Bibliografía

- [1] 20.8. ftplib — FTP protocol client — Python 2.7.17 documentation.
- [2] Flask. Library Catalog: palletsprojects.com.
- [3] Git.
- [4] Las proliferaciones de medusas. Library Catalog: www.miteco.gob.es.
- [5] LaTeX - A document preparation system.
- [6] TeXstudio.
- [7] Visual Studio Code - Code Editing. Redefined. Library Catalog: code.visualstudio.com.
- [8] Visual Studio Code - Wikipedia, la enciclopedia libre.
- [9] The world's leading software development platform · GitHub.
- [10] xarray: N-D labeled arrays and datasets in Python — xarray 0.14.1 documentation.
- [11] ZenHub - Agile Project Management for GitHub. Library Catalog: www.zenhub.com.
- [12] Zotero | Your personal research assistant.
- [13] Reproducción de las medusas, sus etapas, pólipos y larvas plánulas., October 2016. Library Catalog: medusas.wiki Section: Información general.
- [14] Pandas, November 2019. Page Version ID: 121498623.

- [15] Git, February 2020. Page Version ID: 123779424.
- [16] GitHub, March 2020. Page Version ID: 124035823.
- [17] Lisandro Benedetti-Cecchi, Antonio Canepa, Veronica Fuentes, Laura Tamburello, Jennifer E. Purcell, Stefano Piraino, Jason Roberts, Ferdinando Boero, and Patrick Halpin. Deterministic Factors Overwhelm Stochastic Environmental Fluctuations as Drivers of Jellyfish Outbreaks. *PLOS ONE*, 10(10):e0141060, October 2015. Publisher: Public Library of Science.
- [18] Daniel Santín González César Pérez López. Minería de datos: técnicas y herramientas - César Pérez López - Google Libros.
- [19] Lisa-ann Gershwin, Anthony J. Richardson, Kenneth D. Winkel, Peter J. Fenner, John Lippmann, Russell Hore, Griselda Avila-Soria, David Brewer, Rudy J. Kloser, Andy Steven, and Scott Condie. Biology and ecology of Irukandji jellyfish (Cnidaria: Cubozoa). *Adv. Mar. Biol.*, 66:1–85, 2013.
- [20] James Tibballs, Ran Li, Heath A. Tibballs, Lisa-Ann Gershwin, and Ken D. Winkel. Australian carybdeid jellyfish causing "Irukandji syndrome". *Toxicon*, 59(6):617–625, May 2012.
- [21] Wikipedia contributors. Scrum (software development) — Wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Scrum_\(software_development\)&oldid=943108748](https://en.wikipedia.org/w/index.php?title=Scrum_(software_development)&oldid=943108748), 2020. [Online; accessed 10-March-2020].