



Universidad Politécnica
de Madrid

**Escuela Técnica Superior de
Ingenieros Informáticos**



Grado en Grado en Ingeniería Informática

Memoria de Seguimiento

**Desarrollo de un Chatbot para Público
Infantil para Clasificar Sonidos del Cielo.
Versión 2**

Autor: Marcos Pino Gamazo
Tutor(a): Raquel Cedazo León

Madrid, «Noviembre 2020»

Resumen del Trabajo Realizado

Este Trabajo Final de Grado tiene como misión el diseño y desarrollo de una aplicación cuya principal funcionalidad es, la clasificación de meteoros mediante sonidos generados por ordenador a partir de detecciones de llevadas a cabo mediante radiofrecuencia.

El proyecto tiene como objetivo ser accesible a personas con discapacidad visual y para ello, se pretende desarrollar una herramienta mediante la cual se pueda seguir una conversación en *lenguaje natural*, estas herramientas se conocen por el término en inglés *chatbot*. Será necesario que la herramienta sea capaz de reconocer la voz y a su vez transmitir mediante voz la información generada por la aplicación. Esto va a facilitar el acceso a la herramienta no sólo a personas con problemas de visión sino también a niños que aún no sepan leer o escribir.

Por tanto, será necesario realizar un diseño estructural tanto de la base de datos para guardar los distintos parámetros de las radio detecciones, así como de la API que comunique la base de datos con la aplicación y del sitio Web, que deberá tener un diseño *responsive*, es decir, que se adapte a los diferentes dispositivos desde los que se pueda acceder.

Hasta ahora, se ha realizado el diseño y volcado de la base de datos antigua que contenía la información de las radio detecciones, y se ha comenzado la creación de un API Rest para conectar la base de datos con la aplicación.

Además, se han mantenido reuniones semanales entre todos los miembros del proyecto, así como la confección de los documentos intermedios y finales del trabajo.

Explicación y justificación de las modificaciones al Plan de Trabajo

El avance del proyecto se ha visto afectado debido a que depende de personas externas y equipamiento que no está bajo el control de los miembros del equipo. Para poder realizar el proyecto necesitamos generar sonidos a través de las detecciones de los meteoros. Hasta ahora los sonidos eran generados mediante varios programas en Python desarrollados por Samuel Lemes Perera para el Citizen Science Lab [1] de la Universidad Politécnica de Madrid, aunque el resultado es un sonido bastante artificial y metálico motivo por el cual estamos investigando nuevas formas de generarlos, algo que ha repercutido en la planificación inicial del proyecto.

Debido a estos motivos, la planificación se encuentra en el estado que se desarrolla en los siguientes apartados.

Revisión de la lista de objetivos del trabajo

En lo que llevamos de proyecto se ha diseñado y desarrollado tanto una nueva base de datos como una parte de la API y la documentación de esta. La lista de objetivos sigue siendo la siguiente:

- Mejora de la interfaz conversacional de un chatbot.
- Integración de los módulos de reconocimiento y sintetizador de la voz.
- Desarrollo de una interfaz web del experimento de clasificación de meteoros para público infantil.
- Módulo para analizar los resultados obtenidos.
- Desarrollo de una guía de usuario.

Revisión de la lista de tareas

Aunque existan ciertos contratiempos y retrasos en el desarrollo del proyecto, la lista de tareas continúa como estaba planificada inicialmente.

- T1: Estado del arte de los chatbots. (20 horas).
- T2: Análisis de módulos de reconocimiento y sintetizador de voz. (20 horas).
- T3: Ampliación del dataset para entrenar el chatbot. (20 horas).
- T4: Integración de los módulos de reconocimiento y sintetizador de voz. (50 horas)
- T5: Implementación de las mejoras del chatbot (70 horas).
- T6: Desarrollo de la interfaz web del experimento. (50 horas).
- T7: Despliegue y pruebas con grupos de interés. (44 horas).
- T8: Documentación. (50 horas).

Total: 324 horas. **12 ECTS**

Dado que se sigue una metodología *agile* se realiza una reunión un día a la semana de una hora de duración, en la que el equipo cuenta los avances que han realizado y se planifican las siguientes tareas a realizar. Debido a la situación actual, estas reuniones se realizan de forma telemática mediante Skype.

Revisión del diagrama de Gantt

De la planificación realizada al inicio del Trabajo de Fin de Grado se creó este diagrama de Gantt. Aunque existan cierto retraso, se mantiene la planificación inicial.

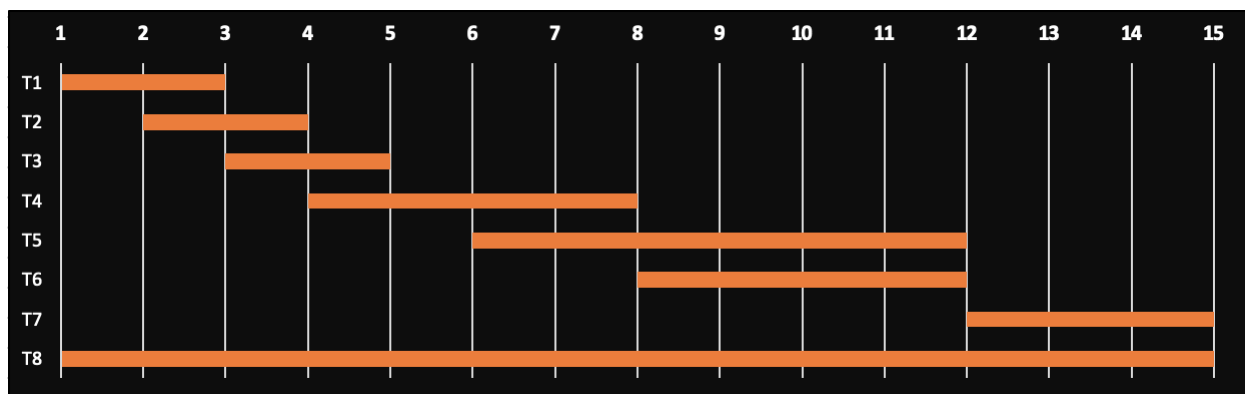


Figura 1: Diagrama de Gantt

Tabla de contenidos

Resumen del Trabajo Realizado	i
Explicación y justificación de las modificaciones al Plan de Trabajo	iii
1. Introducción	1
1.1. Motivación del proyecto	1
2. Estado del Arte	3
2.1. Historia de los chatbots	3
2.2. Frameworks y librerías para el desarrollo de chatbots	4
2.3. Frameworks	4
2.3.1. DialogFlow	4
2.3.2. Microsoft Bot Framework	5
2.3.3. IBM Watson	5
2.3.4. Amazon Lex	5
2.3.5. Rasa	6
2.4. Librerías	6
2.4.1. Chatterbot	6
2.4.2. Natural Language ToolKit	6
2.4.3. ChatbotAI	6
2.4.4. Tensorflow	6
Bibliografía	7

Capítulo 1

Introducción

1.1. Motivación del proyecto

La correlación entre hombre y máquina es cada día mayor, avanzando y evolucionando según las necesidades de la sociedad y de los usuarios. En los últimos años se puede observar como se ha ido evolucionando hacia el desarrollo de dispositivos, no necesariamente ordenadores, que estén conectados entre sí, esto se conoce como el Internet de las Cosas (IoT).

Durante la pandemia producida por la COVID-19, hemos podido comprobar las dificultades que existen en el entorno de la educación para poder desarrollar las tareas de una forma normal. Pero ¿y si pudiésemos tener un profesor en la palma de la mano accesible las 24 horas del día los 7 días a la semana?.

Según una encuesta realizada por el Instituto Nacional de Estadística (INE), entorno al 90 % de los niños entre 10 y 12 años utilizan ordenadores y navegan por internet de manera habitual [2]. Hoy en día están muy extendido el uso de asistentes virtuales para la simplificación de tareas. Nuestro objetivo es utilizar estas herramientas para hacer a los niños partícipes del proyecto y hacerlo de una manera entretenida, mediante técnicas de *gamificación*¹ nos ayuden a clasificar los distintos tipos de meteoros mediante su sonido. Esto hace que tengamos que pensar en que no todos los usuarios son iguales, la interfaz de la aplicación no puede ser igual para un niño de 5 años que apenas sabe leer y escribir que para un adulto. Además queremos que sea una aplicación accesible y sirva también para que personas con problemas de visión puedan participar en el proyecto. Una vez los usuarios hayan clasificado un meteoro un número determinado de veces; científicos analizarán y comprobarán la validez de las clasificaciones.

¹Gamificación: uso de técnicas, elementos y dinámicas propias de los juegos para potenciar la motivación y mejorar el aprendizaje.

Capítulo 2

Estado del Arte

Para el desarrollo de este Trabajo de Fin de Grado es necesario utilizar múltiples tecnologías y herramientas. En este capítulo se explicará brevemente el estado actual de los chatbots, las distintas herramientas para desarrollarlos, así como los lenguajes de programación utilizados, qué es el procesamiento natural del lenguaje.

2.1. Historia de los chatbots

Un chatbot es una aplicación de software capaz de mantener una conversación con un usuario dando una serie de respuestas automáticas, establecidas con anterioridad a diferentes entradas que pueda dar el usuario. Existen distintas teorías sobre el origen de los chatbots.

La primera de ellas, defiende que en la década de 1950, el matemático inglés Alan Turing investigó si una máquina sería capaz de imitar las respuestas de un humano mediante el análisis de una conversación de texto entre un humano y una máquina.

La segunda, más extendida, sitúa el origen en el año 1966 en el Instituto Tecnológico de Massachusetts (MIT), allí el profesor de informática Joseph Weizenbaum desarrolló en el laboratorio de inteligencia artificial el programa *Eliza*, el concepto es que actuara como si se tratase de un terapeuta. Funcionaba de la siguiente manera: examinaba palabras clave que tenía el enunciado del emisor para poder responder con una serie de oraciones que tenía previamente registradas.

En 1972, surgió el chatbot *Parry*, que simulaba ser una persona con esquizofrenia paranoide. A diferencia de Eliza disponía de una estrategia de comunicación cimentada en premisas y respuestas emocionales.^{en} base a las interacciones con los usuarios.

Como detalle curioso, Eliza y Parry fueron puestos a conversar entre sí mediante la red ARPANET ¹

Posteriormente fue desarrollado *Jabberwacky* por el programador inglés Rollo Carpenter, capaz de mantener una conversación mediante la voz. Aunque fue terminado en 1981 no fue hasta 1997 cuando fue publicado online.

¹ARPANET: red de ordenadores creada por el Departamento de Defensa de los Estados Unidos para conectar varias instituciones académicas y estatales.

2.2. Frameworks y librerías para el desarrollo de chatbots

A partir del año 2006 han surgido una gran cantidad de chatbots entre los que cabe destacar:

IBM Watson, nombrado así por el primer director ejecutivo de IBM. En un principio fue desarrollado para responder preguntas y respuestas ideadas por humanos para el concurso Jeopardy. El concurso es el típico de preguntas y respuestas solo que las preguntas son formuladas mediante juegos de palabras y giros lingüísticos. Se presentó al concurso en 2011 por primera vez y fue capaz de ganar a dos especialistas. A partir de ese instante, ha pasado por varias adaptaciones utilizando procesamiento de lenguaje natural y machine learning² para procesar una gran cantidad de datos. En 2013, IBM anunció que Watson podía ser utilizado para la toma de decisiones en el tratamiento del cáncer de pulmón.

Tal vez el chatbot más conocido mundialmente sea el asistente virtual desarrollado por Apple, *Siri*. Utiliza consultas dadas mediante comandos de voz para ayudar al usuario de diversas formas, realizar tareas, recordatorios, búsquedas y modificar configuraciones del sistema.

De este mismo estilo tenemos los chatbot *Alexa* creado por Amazon, *Cortana* desarrollado por Microsoft y *Google Now* desarrollado por Google

2.2. Frameworks y librerías para el desarrollo de chatbots

Lo primero que debemos conocer es la diferencia entre un *Framework* y una librería. Un framework es un tipo de estructura con una serie de archivos y pautas que se utiliza para desarrollar proyectos con una estructura y metodología, es decir, algo así como una plantilla que simplifica la creación de una solución. Por otro lado, una librería es uno o varios archivos escritos en algún tipo de lenguaje de programación, que proporcionan varias funcionalidades. Al contrario que un framework, una librería no establece la estructura sobre cómo debe realizarse el desarrollo, sino que da funcionalidades genéricas que han sido programadas con anterioridad y evitan que haya que escribir código de más, aumentando la calidad del código y reduciendo el tiempo de desarrollo.

2.3. Frameworks

2.3.1. DialogFlow

Framework de desarrollo de chatbots creado en 2010 y mantenido por Google[3]. Es capaz de comprender el lenguaje natural y nos brinda herramientas para la fabricación de diálogos y la recreación de conversaciones. Destaca por la gran cantidad de interfaces de conversación en los que se puede desplegar (Google home, google assistant, wearables, teléfonos, coches). Tiene soporte para más de 14 idiomas y es capaz de resolver abreviaturas y funcionar con faltas de ortografía. Posee una interfaz muy intuitiva y permite crear chatbots en una cantidad pequeña de tiempo.

²Machine Learning: disciplina dentro de la Inteligencia Artificial que crea sistemas y software capaces de aprender automáticamente

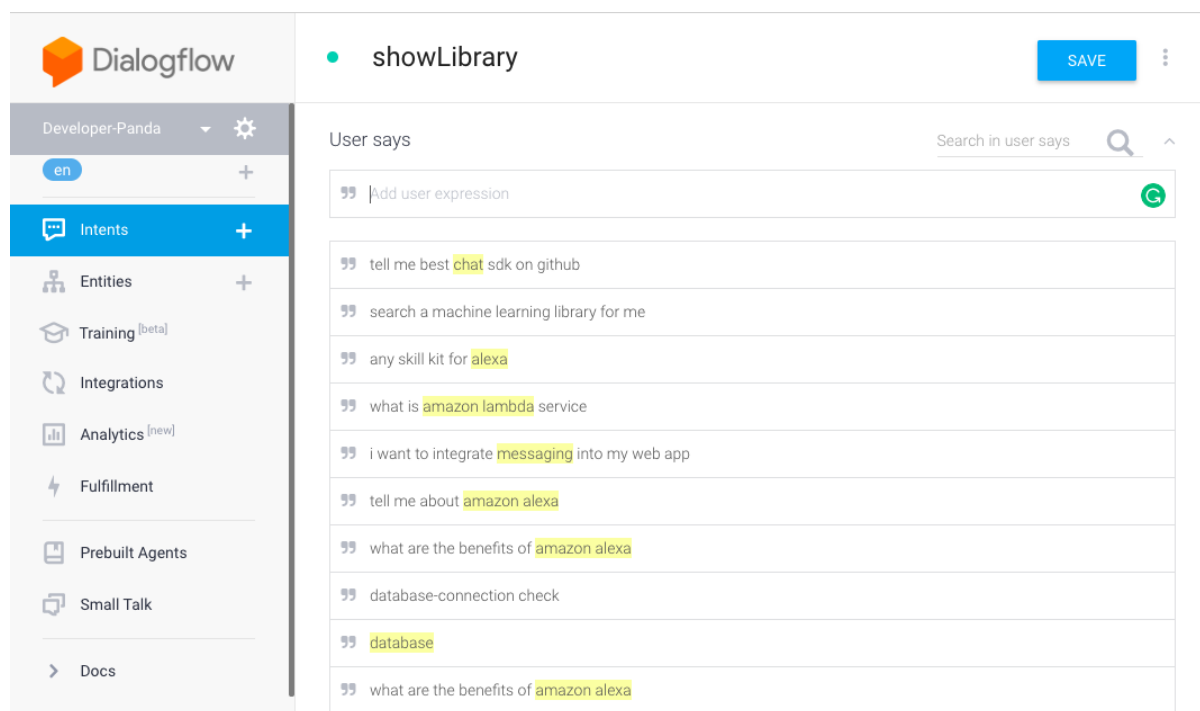


Figura 2.1: Interfaz de dialogflow

2.3.2. Microsoft Bot Framework

Desarrollado por Microsoft, crea chatbots rapidamente a través de la herramienta Microsoft Bot Builder y los conecta con Azure Bot Service, lo que nos permite una rápida creación del bot ya que nos proporciona diversas plantillas para seleccionar cuando se está creando el bot y nos brinda todas las mejoras de la nube creada por Microsoft. Puede editarse con directamente desde la página web usando el editor de Azure o algún IDE de desarrollo como Visual Studio o Visual Studio Code [4]. Posee su propio sistema de procesamiento natural del lenguaje llamado LUIS (*Language Understanding Intelligent Service*).

2.3.3. IBM Watson

Creado por IBM, es capaz de comprender y responder a las preguntas de los usuarios mediante lenguaje natural [4]. Watson está compuesto actualmente por un clúster de al menos 750 servidores *IBM Power 750*, con unos 16 TB de memoria RAM, lo que proporciona una potencia de cálculo bruto de unos **80 petaflops**, convirtiéndolo así en uno de los supercomputadores más potentes del mundo.

2.3.4. Amazon Lex

Creado y gestionado por Amazon, permite establecer comunicaciones con todos sus productos Eco y con su asistente virtual Alexa. Es una de las mejores herramientas en cuanto a conversión de voz a texto. Con esta herramienta se pueden crear bots con un lenguaje natural sofisticado. A diferencia de los anteriores, tiene una interfaz más intuitiva para principiantes aunque por el contrario, dispone de menos herramientas, aunque posee todas las necesarias en un chatbot.

2.3.5. Rasa

Por último cabe destacar Rasa, una framework *Open Source* de machine learning que nos permite crear conexiones entre las máquinas y el usuario. Posee herramientas para entender al usuario mediante el componente Rasa NLU (Natural Language Understanding), generar el diálogo con Rasa NLG (Natural Language Generation) y un motor (Rasa Core) capaz de definir cuál será la siguiente acción a tomar en función del mensaje transmitido por el usuario.

2.4. Librerías

Vamos a analizar las librerías existentes para el lenguaje de programación Python, debido a que se trata de un lenguaje simple, elegante ordenado y portable.

2.4.1. Chatterbot

Se trata de una librería de machine learning basada en las conversaciones y diálogos tradicionales. Lo más destacable de esta librería es que está diseñada de tal manera que permite crear chatbots que soporten varios idiomas.

Es también compatible con librerías para aportar más funcionalidades como puede ser la conversión de texto a voz y así poder interactuar con el usuario sin necesidad de escribir.

2.4.2. Natural Language Toolkit

NLTK por sus siglas en inglés, se trata de una plataforma para crear chatbots con lenguaje humano. Dispone de funcionalidades muy interesantes desde el punto de vista del reconocimiento del lenguaje como la tokenización , derivación, etiquetado, análisis y razonamiento semántico.

2.4.3. ChatbotAI

Nos permite crear un chatbot con muy pocas líneas de código. Genera un controlador del chat y bots con inteligencia artificial que permiten una integración muy sencilla con API Rest. Esta inteligencia artificial nos genera múltiples características como aprender, memorizar, manejar conversaciones según el tema [6].

2.4.4. Tensorflow

Es una plataforma de código abierto gestionada por Google. Se trata de una biblioteca de aprendizaje automático con la que es posible construir y entrenar redes neuronales para detectar patrones y correlaciones en el aprendizaje y razonamiento de los humanos.

Bibliografía

- [1] Citizen Science Lab. *Contadores de Estrellas - Server Side - Echoes generator*.
<https://github.com/cslab-upm/Echoes-stream-generator>
- [2] Insntituto Nacional de Estadística. *Encuesta sobre Equipamiento y Uso de Tecnologías de Información y Comunicación en los Hogares* https://www.ine.es/prensa/tich_2019.pdf
- [3] Carlos Denis *Dialogflow: la herramienta de Google para la creación de Chatbots* <https://www.makingscience.com/blog/dialogflow-la-herramienta-de-google-para-la-creacion-de-chatbots/>
- [4] Planeta Chatbot *Información sobre desarrollo de bots de Microsoft* <https://planetachatbot.com/informacion-para-articulo-de-bots-de-microsoft-943a25eddd5>
- [5] Rodolfo de Juana *IBM Watson: casi todo lo que tienes que saber* <https://www.muycomputerpro.com/2019/09/24/ibm-watson-casi-todo-lo-que-tienes-que-saber>.
- [6] Ahmad Faizal B H *Chatbot* <https://github.com/ahmadfaizalbh/Chatbot>