Revista Innovación y Software
Vol. 5, No. 2, Mes Septiembre - Febrero, 2024
ISSN: 2708-0935
Pág. 129-141
https://revistas.ulasalle.edu.pe/innosoft

Tipo de artículo: Artículos originales
Temática: Desarrollo de aplicaciones informáticas
Recibido: 20/07/2024 | Aceptado: 12/09/2024 | Publicado: 30/09/2024

Identificadores persistentes:
DOI: 10.48168/innosoft.s16.a180
ARK: ark:/42411/s16/a180
PURL: 42411/s16/a180

Sistema de reconocimiento de voz y texto: Una herramienta para la autenticación basada en lectura aleatoria

Speech and text recognition system: A tool for authentication based on random reading

Campos Gamarra Alejandro Román[1][0009-0009-8492-1133]*, **Ugaz Julian Edson Alexis**[2][0009-0006-1872-5685], **Avila Rebaza Sergio Fernando**[3][0009-0005-3022-4798], **Mendoza de los Santos Alberto Carlos**[4][0000-0002-0469-915X]

[1]Afiliación institucional completa. Dirección postal. t1053301021@unitru.edu.pe
[2]Afiliación institucional completa. Dirección postal. t1513300221@unitru.edu.pe
[3]Afiliación institucional completa. Dirección postal. t1053301021@unitru.edu.pe
[4]Afiliación institucional completa. Dirección postal. amendoza@unitru.edu.pe

*Autor para correspondencia: t1513300121@unitru.edu.pe

Resumen

El presente artículo tiene como objetivo principal el desarrollo de un sistema de reconocimiento de voz y texto para mejorar la seguridad en la identificación de usuarios. Para el desarrollo del sistema se implementaron metodologías de aprendizaje profundo y diversas librerías de Python, incluyendo Speech_recognition, Pyttsx3, y Librosa, entre otras. El sistema fue evaluado en un entorno controlado utilizando 50 muestras de voz, obteniendo una precisión del 74 %. Los resultados indicaron que el 61.53 % de los errores se debieron a fallos en la identificación de la voz y el 30.76 % a discrepancias en la coincidencia del texto generado. Estos hallazgos subrayan la efectividad general del sistema, aunque también señalan la necesidad de ajustar los umbrales de similaridad y mejorar los algoritmos de reconocimiento para incrementar su precisión y robustez. Se concluye que el sistema presenta una solución prometedora para la autenticación biométrica de voz, mostrando un balance entre precisión y áreas de mejora que refuerzan su utilidad en aplicaciones de seguridad informática.

Palabras claves: autenticación, biometría, reconocimiento de voz, seguridad informática.

Abstract

The main objective of this paper is the development of a speech and text recognition system to improve security in user identification. For the development of the system, deep learning methodologies and several Python libraries were implemented, including Speech_recognition, Pyttsx3, and Librosa, among others. The system was evaluated in a controlled environment using 50 speech samples, obtaining an accuracy of 74 %. The results indicated that 61.53 % of the errors were due to failures in voice identification and 30.76 % were due to discrepancies in matching the generated text. These findings underscore the overall effectiveness of the system, although they also point to the need to adjust the similarity thresholds and improve the recognition algorithms to increase their accuracy and robustness. It is concluded that the system presents a promising solution for biometric voice

Revista Innovación y Software
Vol. 5, No. 2, Mes Septiembre - Febrero, 2024
ISSN: 2708-0935
Pág. 129-141
https://revistas.ulasalle.edu.pe/innosoft

authentication, showing a balance between accuracy and areas for improvement that reinforce its usefulness in computer security applications.

Keywords: *authentication, biometrics, voice recognition, computer security.*

Introducción

En la actualidad, la necesidad de implementar modelos de información de seguridad informática ha crecido exponencialmente, en este contexto, la biometría, que comprende el análisis de datos biológicos, ha ganado mucha importancia [1]. Estos sistemas están basados en el aprendizaje profundo (deep learning) y ya han demostrado éxito en conseguir resultados en muchas tareas como visión por computadora, reconocimiento de voz y procesamiento del lenguaje natural [2].

La biometría es una tecnología que estudia las medidas y análisis de patrones físicos o de comportamiento empleados para identificar a una persona [1]. Esta tecnología está creciendo rápidamente debido a la gran necesidad de asegurar las pertenencias de las personas y organizaciones, desde bienes hasta información, en el abrumador crecimiento de las tecnologías digitales en los aspectos de la sociedad [3]. Específicamente, hablando de crecimientos acelerados, el rápido desarrollo de tecnología de síntesis de voz ha mejorado notoriamente en aspectos como la naturalidad y similitud al habla natural de los humanos. Sin embargo, a medida que las barreras para la síntesis de voz están disminuyendo rápidamente, el número de actividades ilegales como el fraude y la extorsión está incrementando, lo que propone una amenaza directa a los sistemas de autenticación basados en reconocimiento de voz [4].

En [1] mencionan que el uso de la voz en la biometría para la autenticación de usuarios es muy importante. Además, nos deja en claro la problemática que estos sistemas enfrentan actualmente, la suplantación de identidad, que supone grandes desafíos para la implementación de estos. Los ciberdelincuentes pueden obtener acceso a muestras pregrabadas o sintetizadas de voz con herramientas de inteligencia artificial de usuarios legítimos para suplantar el modo de hablar del usuario objetivo para saltarse la seguridad del sistema de reconocimiento de voz [5]. Ante estos hechos han surgido muchos intentos de suprimir estas vulnerabilidades, por ejemplo, en [4] se propuso un modelo de detección de audios sintetizados que apunta a mejorar la precisión de los sistemas de reconocimiento de voz automáticos (automatic speech recognition, ASR) que buscaba diferenciar entre audios generados sintéticamente.

Otro método que se intenta implementar para hacer más robusto el proceso de autenticación es el reconocimiento de sonidos que aún no son reproducibles mediante una sintetización digital, o que son muy difíciles de imitar, se trata de los ruidos *pop*, estos ruidos son aquellos que se producen por el flujo de aire del usuario cuando

habla muy cerca al micrófono [5].

En [6] se menciona el concepto de modo de detección de vida (liveness detecction mode, LDM), que indica una forma de detectar si un intento de uso del sistema se trata de una sinterización, conversipn o reproducción de un audio pregrabado y proponen una solución en base a este método.

Los servicios de reconocimiento de voz también son conocidos como servicios de autenticación de huella de voz (Voiceprint Authentication as a Service, VAaS), en [7] se menciona los beneficios de implementar estos servicios como su ubicuidad, generalidad y usabilidad. Sin embargo, a pesar de su atractividad, sufre de vulnerabilidades como la fuga de las huellas de voz de los usuarios por el aire o la nube, lo que retrasa su amplia adopción.

En este trabajo intentamos aportar información valiosa para impulsar la implementación de estos modelos basados en inteligencia artificial para la autenticación de usuarios mediante reconocimiento de voz, que se captará por un micrófono y evaluará en ambientes controlados para reducir la bulla y obtener mejores resultados.

Materiales y métodos o Metodología computacional

Estado del arte

Deep Learning Approaches for Speech Recognition: A Review

El presente artículo [8] revisa exhaustivamente los enfoques de aprendizaje profundo aplicados al reconocimiento de voz, comenzando con una introducción a los conceptos fundamentales y el papel del aprendizaje profundo en su evolución. Se detallan los modelos y arquitecturas de redes neuronales como CNNs, RNNs y transformadores, así como las técnicas de preprocesamiento y características acústicas para mejorar el rendimiento. Además, se exploran avances en la reducción de ruido y robustez frente a diferentes condiciones ambientales, incluyendo estudios de caso y aplicaciones prácticas.

End-to-End Automatic Speech Recognition Systems: An Overview

El presente artículo [9] ofrece una visión general de los sistemas de reconocimiento automático de voz de extremo a extremo, describiendo su evolución y el impacto del aprendizaje profundo en su desarrollo. Se analizan las arquitecturas clave, como las redes neuronales recurrentes (RNNs), las redes neuronales convolucionales (CNNs) y los transformadores, destacando cómo eliminan la necesidad de componentes intermedios tradicionales. Además, se discuten técnicas de preprocesamiento y características acústicas que mejoran el rendimiento de estos sistemas. El artículo también presenta estudios de caso y aplicaciones prácticas, demostrando implementaciones

en diversos entornos y dispositivos.

Fundamentación teórica

- **Reconocimiento de Voz**
  El reconocimiento de voz implica la conversión del habla en texto mediante algoritmos de procesamiento de señales y modelos estadísticos de lenguaje. Utiliza técnicas de aprendizaje profundo para mejorar la precisión [10].

- **Autenticación Biométrica**
  La autenticación biométrica utiliza características físicas o comportamentales únicas de una persona para verificar su identidad. Las características de la voz son una modalidad común debido a su unicidad y facilidad de captura [11].

- **Técnicas de Preprocesamiento de Audio**
  El preprocesamiento de audio incluye técnicas como la eliminación de ruido, la normalización de volumen y la extracción de características acústicas (e.g., MFCCs) para mejorar el rendimiento del sistema de reconocimiento de voz [12].

Materiales

- **Micrófono**, se utilizó el micrófono integrado de una laptop Dell G15 ryzen edition, aunque con cualquier otro micrófono funcional hubiese sido suficiente.

- **Visual Studio Code**, editor de código abierto multiplataforma creado por Microsoft en el año 2015. Este editor provee una interfaz gráfica que se integra con los componentes esenciales comúnmente utilizados por programadores, por ejemplo, autocompletar código, mini editores incorporados, y menús contextuales, entre otras [13].

- **Python**, es un lenguaje abierto de alto nivel que cuenta con facilidades para la programación orientada a objetos, imperativa y funcional [14]. Además, cuenta con una amplia gama de librerías, tiene una sintaxis simple, clara y sencilla [15].

Librerías utilizadas de este lenguaje en el proyecto:

1. **Speech_recognition**, es una librería que se utiliza para realizar reconocimiento de voz en tiempo real o en archivos de audio pregrabados [16].

Revista Innovación y Software
Vol. 5, No. 2, Mes Septiembre - Febrero, 2024
ISSN: 2708-0935
Pág. 129-141
https://revistas.ulasalle.edu.pe/innosoft

2. **Pyttsx3**, es una librería que se utiliza para convertir texto a voz.
3. **Librosa**, es una librería que se utiliza para el procesamiento y análisis de audio y música. Proporciona los componentes básicos necesarios para crear sistemas de recuperación de información musical.
4. **Numpy**, es una librería que se utiliza principalmente para realizar cálculos matemáticos y científicos. Ofrece muchas funciones y herramientas que pueden ser útiles para proyectos de Data Science.
5. **Sounddevice**, es una librería que se utiliza para grabar audio mientras que se ejecuta código en segundo plano.
6. **Wavio**, es una librería que se utiliza para facilitar la lectura y escritura de archivos WAV, permitiendo manipular estos archivos de forma sencilla y eficiente.
7. **Random**, es una librería que se utiliza para generar números pseudoaleatorios y realizar operaciones relacionadas con la aleatoriedad.

Creación de sistema

Creación de los audios

Para comenzar tenemos que hacer la creación de los audios. Para eso usaremos el script que veremos en la Figura 1 para grabar la voz y posteriormente guárdalo como un audio de formato .wav. En la función estamos usando una duración predeterminada de 5 segundos, aunque se puede cambiar a conveniencia al usarlo, además usamos el parámetro ”file_path” para poner el archivo al que se quiere guardar el audio.

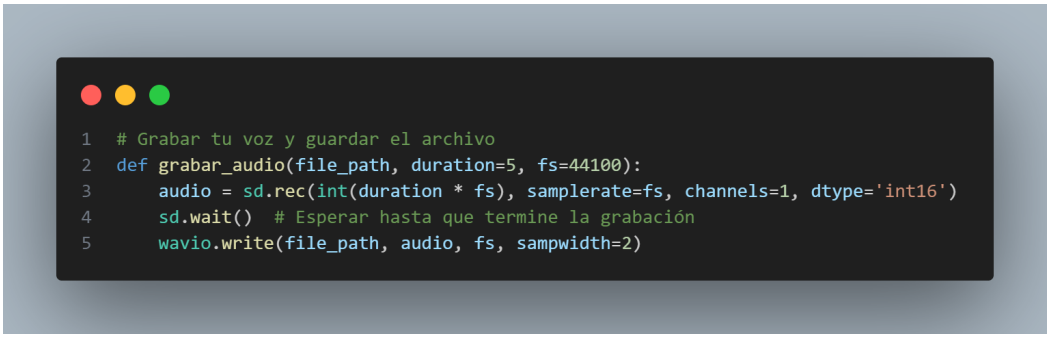


Figura 1. Script para hacer la grabación de voz

Teniendo el script para grabar audio. Necesitaremos dos audios, en el primero se guardará nuestra voz y en

Revista Innovación y Software
Vol. 5, No. 2, Mes Septiembre - Febrero, 2024
ISSN: 2708-0935
Pág. 129-141
https://revistas.ulasalle.edu.pe/innosoft

el segundo será la grabación en el momento para posteriormente compararlo con el primero. En la Figura 2 observamos el script para calibrar nuestra voz y guardar la grabación en el archivo ”mi_voz.wa’

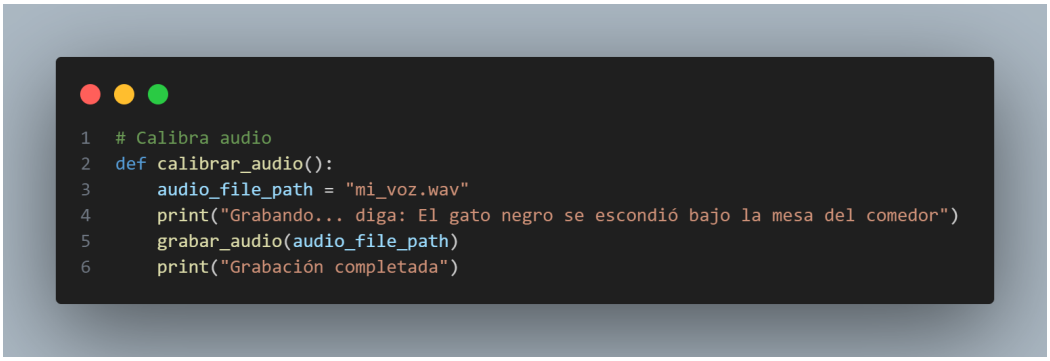


Figura 2. Script para calibrar nuestra voz

En la Figura 3 observamos el script para generar el nuevo audio que se comparara posteriormente a nuestra voz.

Figura 3. Script para generar el nuevo audio

Revista Innovación y Software
Vol. 5, No. 2, Mes Septiembre - Febrero, 2024
ISSN: 2708-0935
Pág. 129-141
https://revistas.ulasalle.edu.pe/innosoft

Creación y transcripción de los textos aleatorios

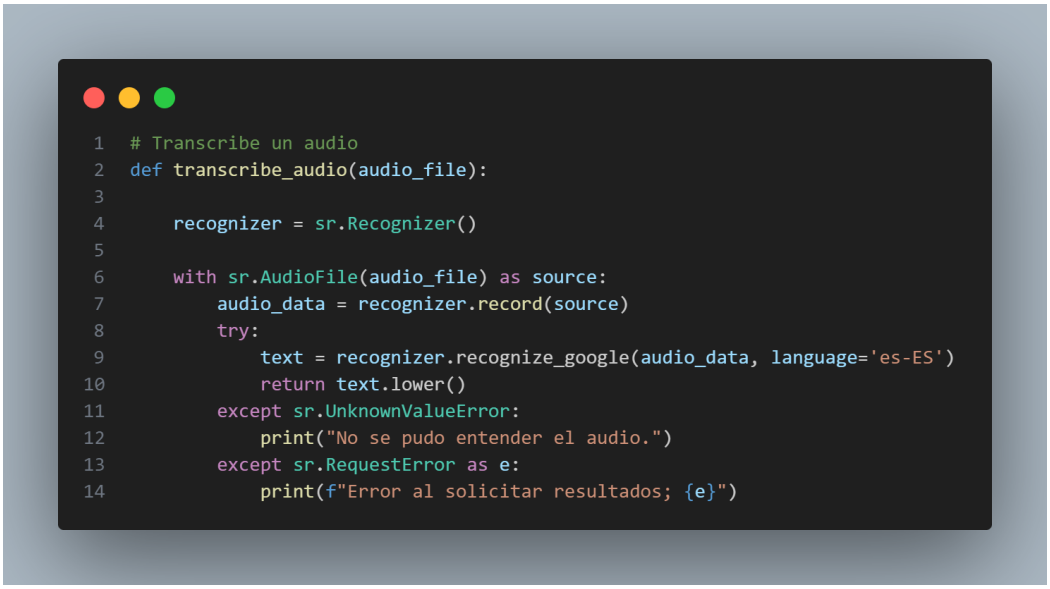
Ahora veamos la parte de la creación de textos aleatorios para posteriormente ser reconocidos por el nuevo audio. Usaremos 10 sujetos con 10 adjetivos. Cada sujeto tendrá concordancia vocálica con cada adjetivo. Y así no formar palabras extrañas. En la Figura 4 podemos observar el método en la que se generar los textos aleatorios.

```python
# Listas de palabras para los sujetos y verbos/sustantivos
sujetos = ['aventura', 'historia', 'misterio', 'viaje', 'experiencia', 'narrativa', 'personaje', 'paisaje', 'conocimiento', 'descubrimiento']
adjetivos = ['emocionante', 'fascinante', 'intrigante', 'inolvidable', 'enriquecedora', 'captivante', 'complejo', 'deslumbrante', 'valioso', 'sorprendente']

def generar_texto():
    sujeto = random.choice(sujetos)
    verbo_sustantivo = random.choice(adjetivos)
    return f"{sujeto} {verbo_sustantivo}"
```

Figura 4. Script para generar texto aleatorio

Ahora en la Figura 5 podemos ver el Script para transcribir lo dicho en un audio para devolverlo como un texto.



```python
# Transcribe un audio
def transcribe_audio(audio_file):

    recognizer = sr.Recognizer()

    with sr.AudioFile(audio_file) as source:
        audio_data = recognizer.record(source)
        try:
            text = recognizer.recognize_google(audio_data, language='es-ES')
            return text.lower()
        except sr.UnknownValueError:
            print("No se pudo entender el audio.")
        except sr.RequestError as e:
            print(f"Error al solicitar resultados; {e}")
```

Figura 5. Script para transcribir audio

Revista Innovación y Software
Vol. 5, No. 2, Mes Septiembre - Febrero, 2024
ISSN: 2708-0935
Pág. 129-141
https://revistas.ulasalle.edu.pe/innosoft

Ya teniendo el script para transcribir audio. Basta con crear un script para transcribir lo que dice en el nuevo audio que podemos ver en la Figura 6.

Figura 6. Script para transcribir el nuevo audio

Validación

Ya teniendo la función para grabar los audios y la función para transcribir los audios. Ahora pasaremos a la verificación. Primero usaremos el script de la Figura 7 para extraer las características de un audio.

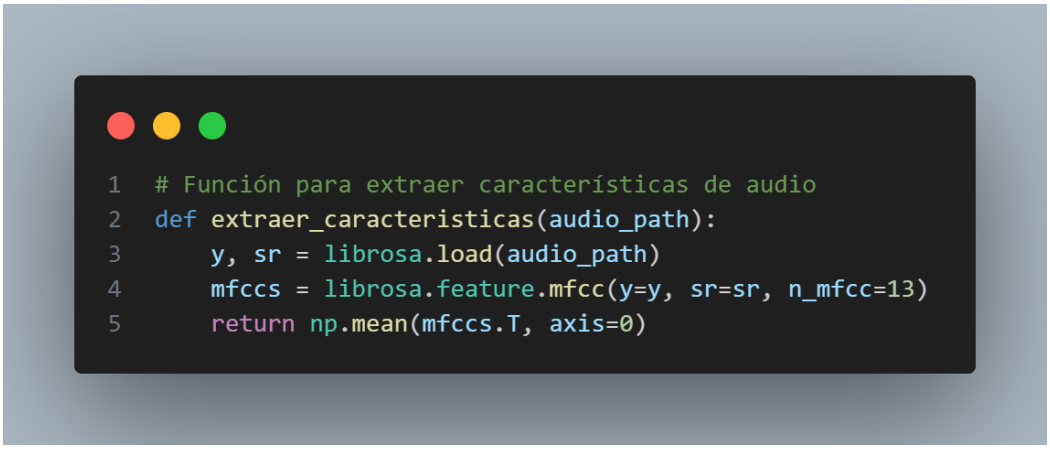
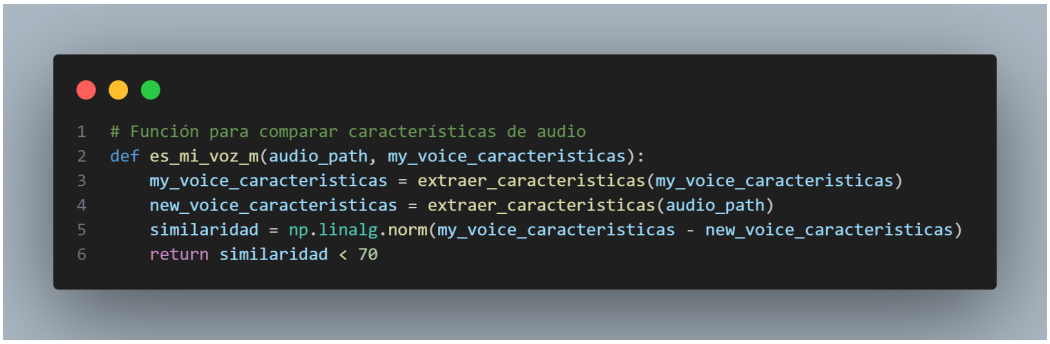


Figura 7. Script para extraer características de un audio

Teniendo el script para extraer características de un audio. Solo nos queda comparar las características del

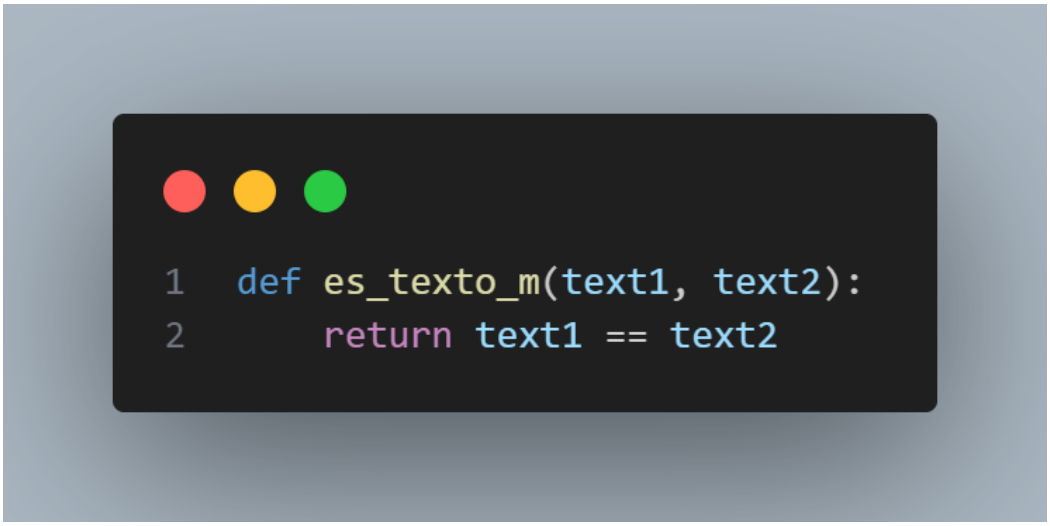
nuevo audio y mi voz por lo que usaremos el script de la Figura 8.



```python
# Función para comparar características de audio
def es_mi_voz_m(audio_path, my_voice_caracteristicas):
    my_voice_caracteristicas = extraer_caracteristicas(my_voice_caracteristicas)
    new_voice_caracteristicas = extraer_caracteristicas(audio_path)
    similaridad = np.linalg.norm(my_voice_caracteristicas - new_voice_caracteristicas)
    return similaridad < 70
```

Figura 8. Script para extraer características de mi voz y en nuevo audio

Como vemos en la Figura 8, usamos una similaridad de 70. Esto se puede ajustar a conveniencia. Sin embargo, hemos usado 70 como parámetro. Con ello podemos saber si ambos audios pertenecen a la misma voz. Ahora veremos cómo saber si lo dicho en el audio es la frase correcta o no. Para eso comenzaremos con un script simple que verifica que ambos textos sean iguales el cual veremos en la Figura 9.

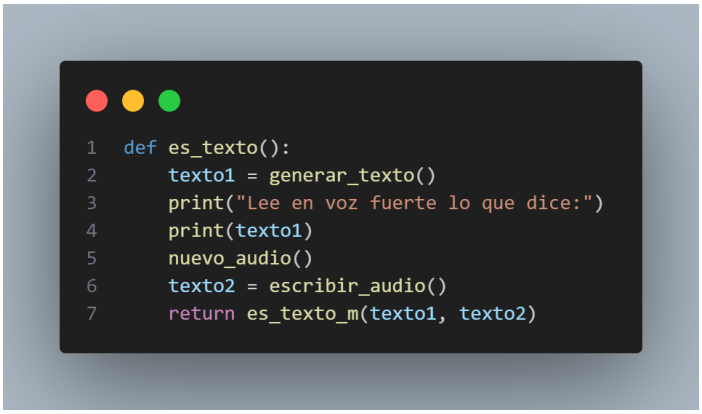


```python
def es_texto_m(text1, text2):
    return text1 == text2
```

Figura 9. Script para comparar 2 textos

Ahora usaremos el script de la Figura 10 para que genere un texto aleatorio y luego grabar el nuevo audio diciendo esa frase aleatoria para posteriormente compararlo.

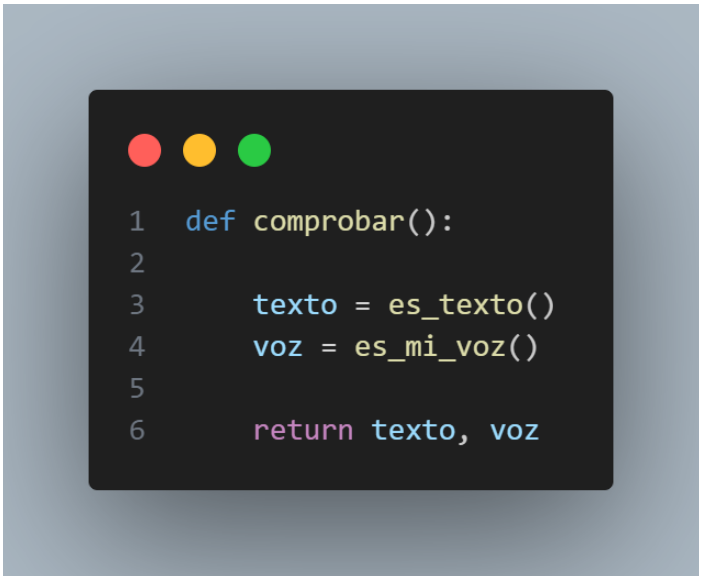
Revista Innovación y Software
Vol. 5, No. 2, Mes Septiembre - Febrero, 2024
ISSN: 2708-0935
Pág. 129-141
https://revistas.ulasalle.edu.pe/innosoft



```
def es_texto():
    texto1 = generar_texto()
    print("Lee en voz fuerte lo que dice:")
    print(texto1)
    nuevo_audio()
    texto2 = escribir_audio()
    return es_texto_m(texto1, texto2)
```

Figura 10. Script para comparar el texto del nuevo audio con el texto generado

Ya con ellos tenemos los dos métodos de verificación el cual serian, validar la voz y validar el texto. Simplemente usaremos el script de la Figura 11. Para hacer esas 2 verificaciones y nos vote *true o false* dependiendo de si la voz o el audio están bien o mal.



```
def comprobar():

    texto = es_texto()
    voz = es_mi_voz()

    return texto, voz
```

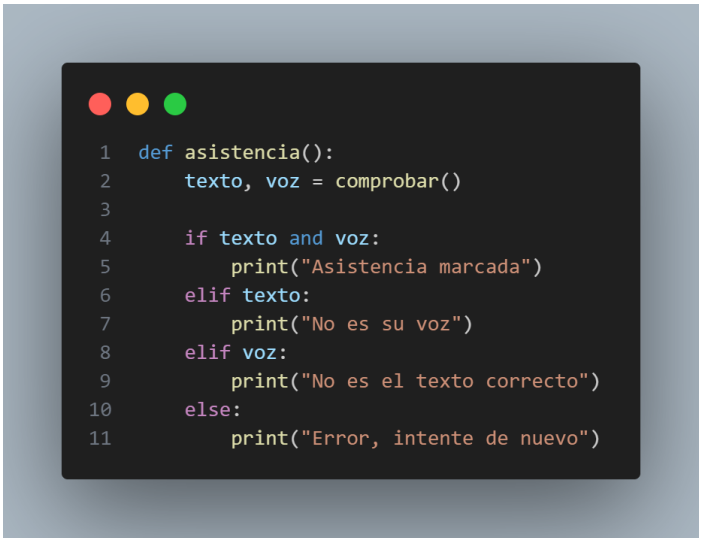
Figura 11. Script que agrupa las 2 verificaciones

Ya con esto podemos integrar el script al sistema que queramos, sin embargo, esta vez lo usaremos para una simulación de asistencia.

Revista Innovación y Software
Vol. 5, No. 2, Mes Septiembre - Febrero, 2024
ISSN: 2708-0935
Pág. 129-141
https://revistas.ulasalle.edu.pe/innosoft

Aplicación

Para ello usaremos el script de la Figura 12 el cual validará ambas verificaciones.



```python
def asistencia():
    texto, voz = comprobar()

    if texto and voz:
        print("Asistencia marcada")
    elif texto:
        print("No es su voz")
    elif voz:
        print("No es el texto correcto")
    else:
        print("Error, intente de nuevo")
```

Figura 12. Script para simular una asistencia

En el script usaremos el script comprobar para validar ambas verificaciones. Luego se armará para en caso la voz y el texto sean correctos, se mandará un mensaje de ”Asistencia marcada” en caso solo el texto sea correcto, significa que la voz no es la correcta por lo que marcara ”No es su voz”. En caso la voz sea la correcta, significa que el texto no es el correcto por lo que marcara ”No es el texto correcto” y ya si de plano ni uno de los dos es correcto, se marcara ”Error, intente de nuevo”.

Resultados y discusión

En el análisis de 50 pruebas realizadas, se obtuvieron 37 resultados de ”Asistencia marcada”, 8 de ”No es su voz”, 4 de ”No es el texto correcto” y 1 de ”Error, intente de nuevo”. Esto se traduce en una precisión del sistema del 74 %, calculada como el porcentaje de asistencias marcadas sobre el total de pruebas. Dentro de las respuestas no marcadas como asistencia, el 61.53 % se debió a errores en la identificación de la voz incorrecta, mientras que el 30.76 % correspondió a discrepancias en la coincidencia exacta del texto generado.

Estos resultados resaltan la eficacia general del sistema en la identificación de la voz y la coincidencia del texto generado, aunque también indican áreas de mejora específicas para fortalecer la precisión en la identificación de la voz correcta y la correspondencia exacta del texto generado.

Conclusiones

El sistema ha logrado una precisión del 74 % en el reconocimiento de voz y texto generados aleatoriamente. Dentro de las discrepancias encontradas, el 61.53 % se debieron a errores en la identificación de la voz correcta, mientras que el 30.76 % correspondió a discrepancias en la coincidencia exacta del texto generado.

El umbral puede ser ajustado según sea conveniente. Sin embargo, con un umbral de 70 hemos conseguido muy buenos resultados.

Aunque la voz fuese correcta, aun así, nos genera un texto aleatorio para asegurarnos de que la persona este en el lugar designado. Y con la precisión del 74 % podemos indicar que el sistema es bastante útil.

Contribución de Autoría

Alejandro Román Campos Gamarra: Conceptualización, Investigación, Metodología, Software, Validación, Redacción - borrador original.
Sergio Fernando Avila Rebaza: Conceptualización, Investigación, Metodología, Análisis formal, Visualización, Supervisión, Recursos, Supervisión, Escritura, revisión y edición.
Edson Alexis Ugaz Julián: Conceptualización, Investigación, Metodología, Análisis formal, Visualización, Supervisión, Escritura, revisión y edición.
Alberto Carlos Mendoza de los Santos: Supervisión, Validación.

Referencias

[1] U. Sumalatha, K. K. Prakasha, S. Prabhu, and V. C. Nayak, "A comprehensive review of unimodal and multimodal fingerprint biometric authentication systems: Fusion, attacks, and template protection," *IEEE Access*, vol. 12, pp. 64 300–64 334, 2024.

[2] S. Minaee, A. Abdolrashidi, H. Su, M. Bennamoun, and D. Zhang, "Biometrics recognition using deep learning: a survey," *Artificial Intelligence Review*, vol. 56, no. 8, pp. 8647–8695, 2023.

[3] W. H. Abdulla, F. Marattukalam, and V. Krivokuča Hahn, "Exploring human biometrics: A focus on security concerns and deep neural networks," *APSIPA Transactions on Signal and Information Processing*, vol. 12, no. 1, 2023.

[4] J. Zhang, G. Tu, S. Liu, and Z. Cai, "Audio anti-spoofing based on audio feature fusion," *Algorithms*, vol. 16, no. 7, p. 317, 2023.

[5] P. Jiang *et al.*, "Securing liveness detection for voice authentication via pop noises," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 2, pp. 1702–1718, 2023.

[6] R. Zhang, Z. Yan, X. Wang, and R. H. Deng, "Livoauth: Liveness detection in voiceprint authentication with random challenges and detection modes," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 6, pp. 7676–7688, 2023.

[7] ——, "Volere: Leakage resilient user authentication based on personal voice challenges," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 2, pp. 1002–1016, 2023.

[8] T. Zeng, "Deep learning in automatic speech recognition (asr): A review," in *Proceedings of the 2022 7th International Conference on Modern Management and Education Technology (MMET 2022)*. Paris: Atlantis Press SARL, 2023, pp. 173–179.

[9] D. Wang, X. Wang, and S. Lv, "An overview of end-to-end automatic speech recognition," *Symmetry*, vol. 11, no. 8, p. 1018, 2019.

[10] J. Li, "Recent advances in end-to-end automatic speech recognition," https://arxiv.org/abs/2111.01690, 2022, accedido el 22 de octubre de 2023.

[11] O. Adenuga and A. Oduroye, "Biometric authentication: A review," 08 2023.

[12] M. Labied, A. Belangour, M. Banane, and A. Erraissi, "An overview of automatic speech recognition preprocessing techniques," in *2022 International Conference on Decision Aid Sciences and Applications (DASA)*, 2022.

[13] P. Masci and C. A. Muñoz, "An integrated development environment for the prototype verification system," *Electronic Proceedings in Theoretical Computer Science*, vol. 310, pp. 35–49, 2019.

[14] C. P. Ivet, D. R. Yanet, and B. G. Roberto, "El lenguaje de programación python," *Ciencias Holguín*, 2014.

[15] G. D. Raúl, *Python para todos*. Creative Commons Reconocimiento, 2011.

[16] S. M. M. Soldara, D. E. Campos Camacho, C. A. M. Guzmán, J. L. Torres Ramírez, J. Luís, and H. Chávez, "Selección de una herramienta de reconocimiento de voz analizando sus características," *Tecnológico Nacional de México en Celaya Pistas Educativas*, vol. 2023, no. 146, 2023.