



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



TFG del Grado en Ingeniería
Informática



Presentado por Jorge Martínez Martín
en Universidad de Burgos — 5 de junio de 2024
Tutor: Álar Arnaz González



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. Álvar Arnaiz González, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Jorge Martínez Martín, con DNI 71482657Z, 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, 5 de junio de 2024

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. nombre tutor

D. nombre co-tutor

Resumen

La enfermedad del Parkinson afecta cada vez a más personas. Pese a ser una enfermedad incurable poder detectarla y ver su progreso es crucial para adecuar los tratamientos a cada persona.

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	iv
Índice de tablas	v
1. Introducción	1
2. Objetivos del proyecto	3
3. Conceptos teóricos	5
3.1. Enfermedad del Parkinson	5
3.2. Aprendizaje automático	6
4. Técnicas y herramientas	9
4.1. Técnicas	9
4.2. Herramientas	9
5. Aspectos relevantes del desarrollo del proyecto	13
6. Trabajos relacionados	15
6.1. A computer vision framework for finger-tapping evaluation .	15
6.2. The discerning eye of computer vision	16
6.3. Supervised classification of bradykinesia	17
7. Conclusiones y Líneas de trabajo futuras	19
Bibliografía	21

Índice de figuras

Índice de tablas

1. Introducción

El parkinson es una enfermedad crónica e irreversible que afecta a más de 150.000 personas en toda España, siendo esta la segunda enfermedad neurodegenerativa más común después del Alzheimer [13]. Esta enfermedad se caracteriza por la pérdida de neuronas dopaminérgicas lo que conlleva un desequilibrio en los circuitos neuronales que controlan el movimiento esto puede provocar temblores, rigidez muscular, lentitud de movimiento (bradicinesia) y problemas de equilibrio y coordinación [11]. Pese a que no exista una cura, saber la evolución del paciente y su respuesta a los medicamentos y tratamientos puede ayudar enormemente tanto a pacientes como a médicos a adecuarse mejor a las circunstancias del paciente y con ello mejorar su calidad de vida.

La bradicinesia es uno de los síntomas más importantes a la hora de clasificar el grado de la enfermedad. Este síntoma se hace muy visible en un ejercicio conocido como test de golpeteo rápido de los dedos (o «rapid finger tapping test» en inglés). Este ejercicio consiste en separar y juntar repetidas veces los dedos índice y pulgar de forma rápida y constante.

En las últimas décadas, el campo de la inteligencia artificial ha experimentado un gran avance, con aplicaciones prometedoras en el ámbito de la salud. Específicamente, las técnicas de aprendizaje automático han demostrado su potencial para el diagnóstico y clasificación de enfermedades neurodegenerativas a partir de datos clínicos, genéticos y de neuroimagen [12].

En este proyecto se ha hecho uso de la inteligencia artificial para la creación de modelos capaces de clasificar el grado de bradicinesia en vídeos del «rapid finger tapping test». Ver de manera individual la clasificación de cada vídeo puede no ser del todo útil para poder comprobar la evolución del

paciente. Es por ello que se ha decido crear, usando vídeos de distintos días, una gráfica en la que se pueden ver las dos características de la bradicenesia: la amplitud y la velocidad.

2. 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.

3. Conceptos teóricos

En este capítulo se definirán algunos conceptos teóricos para facilitar la comprensión de este proyecto.

3.1. Enfermedad del Parkinson

La EP es una enfermedad neurodegenerativa multisistémica progresiva que cada afecta principalmente a a gente de avanzada edad [16]. Entre los síntomas principales de esta enfermedad podemos encontrar: Pérdida significativa de parte de las células productoras de dopamina lo que produce que aparezcan mucho antes los síntomas relacionados con el movimiento que presenta la EP. Otro de los síntomas que podemos encontrar en la EP es la bradicinesia es cual se caracteriza por la lentitud al realizar movimientos voluntarios así como la ralentización y decremento de amplitud a la hora de realizar movimientos repetitivos. El diagnóstico de la EP se basa en criterios específicos. Los síntomas iniciales incluyen lentitud de movimientos, junto con rigidez muscular, temblores o problemas de equilibrio. Los controles posteriores descartan otras posibles causas y confirman la presencia de factores que sugieren claramente la EP. La EP suele empezar en un lado del cuerpo y extenderse a lo largo de unos años. Los síntomas incluyen postura encorvada, rigidez, letra más pequeña y marcha arrastrando los pies. Los temblores son frecuentes. Las alteraciones de la marcha (como pasos indecisos o congelación repentina) se vuelven habituales. La pérdida de equilibrio es un problema importante, que aumenta el riesgo de caídas y lesiones.

Rapid finger tapping test

El test de golpeteo rápido de los dedos, también conocido como «rapid finger tapping test» o RFTT es un procedimiento por el cual el paciente realiza golpeteos de manera repetitiva durante un periodo de entre 10 y 15 segundos en los cuales ha de intentar generar la mayor amplitud posible entre el dedo índice y el pulgar sin bajar la frecuencia a la que lo realiza. Este test es usado comúnmente para el diagnóstico de personas con la EP ya que como se explicó anteriormente uno de los síntomas que presenta esta enfermedad es la bradicinesia el cual produce que una persona con EP al realizar este test muestre deterioro ya sea en la amplitud o en la frecuencia según avanza la prueba. Este test será el que se utilizará en este proyecto para determinar el grado de EP en el que se encuentra el paciente.

Unified parkinson disease rating scale

La escala de evaluación unificada de la EP o UPDRS por sus siglas en inglés es una herramienta creada por la Movement Disorder Society que permite, mediante la evaluación de diversos parámetros de la EP, medir la gravedad de la enfermedad. En ella se miden diversos aspectos de las experiencias tanto motoras como no motoras de la vida diaria. Para el proyecto utilizaremos solo algunas de las medidas que se utilizan para la clasificación del estado de la persona ya que son los datos que se extraerán de los vídeos proporcionados por los pacientes.

3.2. Aprendizaje automático

El aprendizaje automático es una rama de la inteligencia artificial centrada en el desarrollo de métodos y algoritmos para que el computador de manera autónoma sea capaz de, mediante la experiencia y el procesamiento de datos, mejorar en esa tarea. De esta manera los modelos realizarán predicciones cada vez más precisas.

Dentro de un «dataset» podemos encontrar lo que se conocen como instancias. Una instancia es cada fila del «dataset» caracterizadas por atributos que pueden ser tanto categóricos (nombres, colores, categorías, etc) como valores numéricos (números).

Dentro del aprendizaje automático podemos distinguir tres tipos principales:

- Aprendizaje supervisado: a este aprendizaje se le proporciona un conjunto de datos completo con ejemplos etiquetados, y es mediante estos ejemplos con los que aprende a clasificar nuevos datos o realizar predicciones. Se puede dividir en regresión (predicción de datos numéricos) o clasificación (predicción de datos categóricos)
- Aprendizaje no supervisado: este aprendizaje a diferencia del supervisado se le entregan los datos sin clasificar de manera que tiene que ser él el que encuentre las relaciones. Se puede dividir en: clustering (agrupación de datos por su similitud) y reducción dimensional (disminuye el número de variables en un conjunto de datos sin perder información importante).
- Aprendizaje semi-supervisado: este aprendizaje es una combinación de los dos aprendizajes anteriores ya que se le proporcionan dos conjuntos de datos. En el primer conjunto los datos están etiquetados, este conjunto es el conocido como entrenamiento (training) que es con el cual el modelo se entrena para encontrar la relación entre los datos. En el segundo conjunto los datos no están etiquetados, este conjunto es conocido como test que es el que permite al modelo comparar como de precisas son las predicciones que se realizan ya que, pese a que al modelo se le introduzcan los datos sin etiquetar, el «dataset» si contiene las etiquetas de estas instancias. Al igual que el aprendizaje supervisado su objetivo principal es la creación de clasificadores o regresores.

Un clasificador se compone principalmente de dos fases: la fase de entrenamiento y la fase de predicción. Durante la primera fase se le introducen datos al clasificador con una etiqueta y el objetivo de este es encontrar una hipótesis que se cumpla para cualquier instancia incluso si no se encuentra dentro de los datos de entrenamiento. Para ello suponemos que la hipótesis que mejor se adapte a las instancias que no se encuentran en la fase de entrenamiento, es la que mejor se adapte al conjunto de entrenamiento. Es decir, el clasificador «aprende» del conjunto de entrenamiento.

4. Técnicas y herramientas

En este capítulo se muestran las diferentes técnicas y herramientas que se han utilizado para el desarrollo del proyecto.

4.1. Técnicas

En esta sección se muestran las técnicas principales empleadas para el desarrollo del proyecto.

Scrum

Para el desarrollo de este proyecto ha sido utilizada la metodología ágil conocida como Scrum.

La metodología Scrum trata de fraccionar la duración de un proyecto en lo que se conoce como «sprints», cuya duración varía de una a dos semanas. El objetivo de los «sprints» es decidir que parte del proyecto se va a desarrollar durante ese periodo de tiempo realizando revisiones diarias para ver como avanza así como una revisión de «sprint» en el que hace una valoración general del «sprint» y se decide en que va a consistir el siguiente.

4.2. Herramientas

En esta sección se describen las herramientas que se han utilizado durante la realización del proyecto.

Portales

Usados principalmente para el seguimiento de la metodología Scrum, creación de diagramas y bocetos y control de versiones:

- **Zube:** portal dedicado a la gestión de proyectos «software» [21]. Permite visualizar proyectos, sprints, gráficos propios de Scrum y crear tableros.
- **GitHub:** es una plataforma web que proporciona una interfaz gráfica para el control de versiones utilizando Git [7]. Además, GitHub permite a los usuarios revisar versiones anteriores de proyectos y realizar un seguimiento de los cambios a lo largo del tiempo.
- **DrawIO:** es una herramienta que permite la creación de diagramas entidad-relación, diagramas de flujo, casos de uso, etc [3]. Esta herramienta dispone además de los símbolos UML necesarios para todas las funcionalidades.

Librerías

Principalmente de Python:

- **Scikit-Learn:** esta librería ofrece una amplia variedad de algoritmos de aprendizaje supervisados, semisupervisados y no supervisados [14]. Además posee una gran compatibilidad con otras librerías como así como una extensa documentación que facilitan su implementación.
- **TSFresh:** es una biblioteca de Python, para la extracción y representación de características de series temporales [17]. Esta herramienta agiliza el proceso de extracción de características a partir de datos de series temporales para apoyar las tareas de aprendizaje automático. Ofrece una selección de características definidas que abarcan diferentes medidas estadísticas, como análisis de tendencias, patrones estacionales, métricas de correlación y evaluaciones de complejidad.

Entorno de desarrollo

Incluye el entorno integrado de desarrollo (IDE), lenguaje y otros programas usados.

- **Visual Studio Code:** editor y depurador de código, así como algunas de sus extensiones más importantes.

- **Python:** lenguaje de programación de alto nivel ampliamente usado. Ha sido escogido para realizar este proyecto debido a que posee una gran variedad de paquetes y librerías de Machine Learning [10].
- **Git:** es un software de control de versiones distribuido, utilizado para la control de versiones en proyectos software.
- **Github Copilot:** es un asistente de programación con inteligencia artificial que te permite realizar código de forma más rápida mediante la realización de consultas y las sugerencias que provee.

Desarrollo web

A continuación se exponen tanto librerías como recursos para el desarrollo del apartado web:

- **Bootstrap:** es un «framework» utilizado para la creación de páginas web [2]. En esencia es un librería de código que simplifica el proceso de desarrollo de una página web. Unas de sus principales características son:
 - **Diseño «responsive»:** Bootstrap permite de una forma muy sencilla implementar la posibilidad de que la página web con todos sus componentes se adapten automáticamente en tiempo real a cualquier pantalla incluso si se modifica el tamaño de la ventana.
 - **Documentación y comunidad:** dado que Bootstrap es un framework muy conocido, tanto la documentación del framework como la comunidad que ayuda a desarrollarlo facilitan mucho la posibilidad de aprenderlo y poder encontrar soluciones específicas a nuestro diseño en poco tiempo.
 - **Enfoque móvil:** Bootstrap está diseñado teniendo en cuenta también el diseño en dispositivos móviles, lo que ayuda al desarrollo de la web para multiplataformas.
- **Flask:** es un «framework» escrito en Python que simplifica y facilita la creación de aplicaciones web mediante el uso del patrón Modelo-Vista-Controlador [4]. Dado que es considerado un micro «framework» no viene con todas las funcionalidades por defecto; sin embargo, existen librerías compatibles con Flask que permiten añadir todas las funcionalidades que se necesiten. Entre ellas se han usado:

- **Werkzeug**: es un conjunto de librerías de Python usados para el desarrollo web [18]. Se ha utilizado Flask junto a Werkzeug ya que este contiene una librería de cifrado que se ha utilizado durante el desarrollo del proyecto.
 - **SQLAlchemy**: dado que Flask no puede hacer uso de forma directa de las bases de datos hace uso de la librería SQLAlchemy [15]. Esta librería permite la creación de bases de datos mediante la creación de clases en Python, definir el esquema y la relación entre las tablas y hacer consultas mediante métodos y objetos en vez de consultas SQL sin formato.
 - **Flask-Login**: es una extensión de Flask que simplifica añadir la funcionalidad de iniciar sesión, administrar sesión y cerrar sesión [6]. Permite además otorgar y restringir accesos de distintas secciones de la aplicación así como cargar datos relacionados con la sesión actual.
- **Jinja**: es un motor de plantillas rápido, seguro y fácil de usar para Python [8]. Permite la modificación de partes de la plantilla de la web mediante el uso de código como la agregación de contenido dependiendo del valor de variables mediante bucles o condicionales.
- **Font Awesome**: es una herramienta usada para el diseño y desarrollo de páginas web [1]. En vez de usar archivos de imágenes para iconos utiliza fuentes. Esto ofrece diversas ventajas:
- Escalabilidad: los iconos se escalan perfectamente a cualquier tamaño sin perder calidad.
 - Colores personalizables: permite modificar el color de los iconos usando solo CSS. Para poder utilizar Font Awesome con Flask se ha utilizado la librería Flask-FontAwesome que permite su implementación [5].

5. Aspectos relevantes del desarrollo del proyecto

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros³, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

6. Trabajos relacionados

Durante los últimos años se han utilizado diferentes métodos para evaluar la Enfermedad del Parkinson mediante la prueba del *Rapid Finger-tapping*. En este capítulo se recogen algunos de los trabajos de manera resumida.

6.1. A computer vision framework for finger-tapping evaluation

En este artículo [9] se documenta el uso de la visión por computador para la clasificación de individuos según el nivel de gravedad de la Enfermedad del Parkinson.

Se emplea un método en el que se emplea el uso de la cara para la calibración de la amplitud de la prueba. Para ello el sujeto ha de elevar las manos a la altura de la cara y apuntar con las puntas de los dedos hacia la misma.

El estudio se realizó con 387 vídeos de *rapid finger-tapping test* (RFT) de 13 pacientes diagnosticados con Enfermedad del Parkinson en estado avanzado y 84 vídeos de *rapid finger-tapping test* de 6 personas de control sanas. En total 471 vídeos.

Metodología

1. Se realiza un reconocimiento facial del individuo y se crea a partir de él 2 cuadrados en los costados de la cara para realizar la medición de la amplitud.

2. Se genera una serie temporal que representa la distancia desde el dedo índice al pulgar siendo esta la amplitud del movimiento.
3. Se extraen algunas características de la serie temporal, por ejemplo, la velocidad media de apertura y cierre de dedos, el número total de toques de los dedos, la amplitud máxima...
4. Se seleccionan las características no redundantes mediante el algoritmo chi-cuadrado
5. Se entrena una máquina de vectores de soporte (SVM) mediante las características obtenidas para realizar la clasificación.

Resultados

Se encontró una nueva característica representativa del ritmo de golpeteo llamada «correlación cruzada entre los picos normalizados», la cual mostró una fuerte correlación de Guttman con las valoraciones clínicas. Al utilizar el clasificador de máquina de vectores de soporte y una validación cruzada de 10 grupos, se logró categorizar las muestras de pacientes en los niveles UPDRS-FT con una precisión del 88 %. Este mismo esquema de clasificación también permitió discriminar entre las muestras **RFT** de los controles sanos y los pacientes con **EP**, logrando una precisión del 95 %.

6.2. The discerning eye of computer vision

Este artículo [20] utilizan 133 vídeos de manos realizando el **RFT** procedentes de 39 pacientes de Enfermedad del Parkinson y 30 pacientes de control.

A través de estos vídeos se han logrado extraer características y comprobar la relación que existe entre estas y la gravedad de la enfermedad en un paciente siendo esta valorada desde 0 (normal) hasta 4 (enfermedad muy grave).

Metodología

1. Se utiliza la librería de DeepCutLab la cual se trata de una librería de visión por computador para obtener la serie temporal de la amplitud.
2. Se normaliza la serie con la amplitud máxima siendo igual a 1 y escalando los valores conforme a esta medida.

3. Al igual que en el estudio anterior se extraen ciertas características de la serie siendo estas la velocidad, la amplitud y el ritmo.

Resultados

DeepLabCut rastreó y midió con fiabilidad el *finger-tap* en un vídeo estándar de smartphone. Las medidas del ordenador se relacionaron bien con las valoraciones clínicas de la bradicinesia.

6.3. Supervised classification of bradykinesia

Este artículo [19] utiliza 70 vídeos de evaluaciones de *finger-tap* en un entorno clínico (40 manos con **Parkinson**, 30 manos de control). Dos expertos clínicos en **Parkinson**, que desconocían los diagnósticos, evaluaron los vídeos para dar un grado de gravedad de la bradicinesia entre 0 y 4 utilizando la Escala Unificada de Calificación de la Enfermedad de Parkinson (UPDRS, por sus siglas en inglés)

Metodología

1. Extraer la frecuencia: La frecuencia de intervención se estimó como la frecuencia correspondiente al pico de amplitud máxima en el espectro de la transformada rápida de Fourier (FFT).
2. La densidad espectral de energía se calculó como la integral al cuadrado del espectro FFT, una medida que se espera que aumente con la amplitud del golpeteo.

Resultados

Una máquina de vectores de soporte con núcleos de función de base radial predijo la presencia de bradicinesia leve/moderada/grave con una precisión de prueba estimada del 0,8 %. Un modelo Naïve Bayes predijo la presencia de la enfermedad de Parkinson con una precisión de prueba estimada de 0,67.

7. 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] Font Awesome. Font awesome. <https://fontawesome.com/>, 2024.
- [2] Bootstrap. Bootstrap · the most popular html, css, and js library in the world. <https://getbootstrap.com/>, 2024.
- [3] DrawIO. Drawio. security-first diagramming for teams. <https://drawio.com/>, 2024.
- [4] Flask. Guía del usuario. <https://flask-es.readthedocs.io/>, 2024.
- [5] Flask-FontAwesome. Flask-fontawesome. <https://pypi.org/project/Flask-FontAwesome/>, 2024.
- [6] Flask-Login. Flask-login. <https://flask-login.readthedocs.io/en/latest/>, 2024.
- [7] Github. Github. let's build from here. <https://github.com/>, 2024.
- [8] Jinja. Jinja templating engine. <https://jinja.palletsprojects.com/en/3.1.x/>, 2024.
- [9] Taha Khan, Dag Nyholm, Jerker Westin, and Mark Dougherty. A computer vision framework for finger-tapping evaluation in parkinson's disease. *Artificial intelligence in medicine*, 60(1):27–40, 2014.
- [10] Phyton. 3.12.3 documentation. <https://docs.python.org/3/>, 2024.
- [11] Werner Poewe, Klaus Seppi, Caroline M. Tanner, Glenda M. Halliday, Patrik Brundin, Jens Volkman, Anette-Eleonore Schrag, and Anthony E. Lang. Parkinson disease. *Nature Reviews Disease Primers*, 3(1), March 2017.

- [12] R. Bharat Rao, Sriram Krishnan, and Radu Stefan Niculescu. Data mining for improved cardiac care. *SIGKDD Explor. Newsl.*, 8(1):3–10, jun 2006.
- [13] Diego Santos García, Marta Blázquez-Estrada, Matilde Calopa, Francisco Escamilla-Sevilla, Eric Freire, Pedro J. García Ruiz, Francisco Grandas, Jaime Kulisevsky, Lydia López-Manzanares, Juan Carlos Martínez Castrillo, Pablo Mir, Javier Pagonabarraga, Francisco Pérez-Errazquin, José María Salom, Beatriz Tijero, Francesc Valldeoriola, Rosa Yáñez, Arantxa Avilés, and María-Rosario Luquín. Present and future of parkinson’s disease in spain: Parkinson-2030 delphi project. *Brain Sciences*, 11(8), 2021.
- [14] Scikit-Learn. Scikit-learn. machine learning in python. <https://scikit-learn.org/stable/>, 2024.
- [15] SQLAlchemy. Sqlalchemy. the python sql toolkit and object relational mapper. <https://www.sqlalchemy.org/>, 2024.
- [16] Sigurlaug Sveinbjornsdottir. The clinical symptoms of parkinson’s disease. *Journal of Neurochemistry*, 139(S1):318–324, 2016.
- [17] TSfresh. Tsfresh.time series feature extraction based on scalable hypothesis tests. <https://tsfresh.readthedocs.io/en/latest/>, 2024.
- [18] Werkzeug. Werkzeug documentation. <https://werkzeug.palletsprojects.com/en/3.0.x/>, 2024.
- [19] Stefan Williams, Samuel D Relton, Hui Fang, Jane Alty, Rami Qahwaji, Christopher D Graham, and David C Wong. Supervised classification of bradykinesia in parkinson’s disease from smartphone videos. *Artificial Intelligence in Medicine*, 110:101966, 2020.
- [20] Stefan Williams, Zhibin Zhao, Awais Hafeez, David C Wong, Samuel D Relton, Hui Fang, and Jane E Alty. The discerning eye of computer vision: Can it measure parkinson’s finger tap bradykinesia? *Journal of the Neurological Sciences*, 416:117003, 2020.
- [21] Zube. Zube. developer collaboration, solved. <https://www.zube.io/>, 2024.