Generación y adaptación de explicaciones "Human-in-the-Loop" en la "Smart Home"

Antoni Mestre
VRAIN Institute
Universitat Politècnica de València
Valencia, Spain
anmesgas@inf.upv.es

Miriam Gil
Departament d'Informàtica
Universitat de València
Burjassot, Spain
miriam.gil@uv.es

Manoli Albert
VRAIN Institute
Universitat Politècnica de València
Valencia, Spain
malbert@pros.upv.es

Vicente Pelechano
VRAIN Institute
Universitat Politècnica de València
Valencia, Spain
pele@pros.upv.es

Ignacio Panach
Departament d'Informàtica
Universitat de València
Burjassot, Spain
ignacio.panach@uv.es

ABSTRACT

Conseguir una colaboración persona-sistema efectiva y eficiente en sistemas autónomos con participación del humano ("human-in-the-loop"), requiere de sistemas transparentes, comprensibles, y que generen confianza en las personas. Las explicaciones constituyen la herramienta esencial para conseguir esto. Sin embargo, el camino hacia conseguir unas explicaciones comprensibles y no intrusivas presenta muchos desafíos. Es necesario identificar cuándo un usuario en un contexto particular requiere o no de una explicación, de esta forma se evita abrumar a los usuarios con explicaciones innecesarias, y generar una explicación comprensible ante esta situación, todo esto ayudando a la mejora de la experiencia de usuario. En este trabajo presentamos una propuesta para generar de forma automática explicaciones *Human-in-the-Loop*. Estas explicaciones se generan solo cuando se detecta la necesidad de explicación ante una acción de adaptación del sistema. Esto se consigue utilizando técnicas de Aprendizaje Automático que son capaces de inferir, a partir de la interacción que está llevando a cabo el humano con el sistema, si el usuario requiere explicaciones. Además, estas explicaciones se generarán en tiempo de ejecución de forma automática. Para ello, se utilizan técnicas de Generación de Lenguaje Natural para, a partir de la información sobre el sistema, generar una explicación.

CCS CONCEPTS

• Human-centered design. Ubiquitous and mobile computing. Ubiquitous and mobile computing systems and tools

• Human-centered design. Interaction design. Interaction design process and methods

KEYWORDS

Human in the Loop, Explicabilidad, Sistemas Autónomos, Explicaciones Auto-adaptables, Aprendizaje Automático, Experiencia de Usuario

1 Introducción

La introducción de los sistemas autónomos colaborando con las personas para ayudarse mutuamente en la realización de sus tareas (concepto acuñado como "mixed-initiative interaction" [12]), permitirá combinar lo mejor del conocimiento y las capacidades humanas junto con las máquinas (o sistemas). Por ejemplo, en el dominio de la automatización de vehículos, Miller y Ju [21] identificaron las capacidades superiores de las personas sobre las máquinas y viceversa. Combinar las capacidades de los sistemas autónomos junto con las capacidades de las personas es lo que permitirá lograr realmente construir sistemas con la deseada flexibilidad, capacidad de acción y autonomía. A esta colaboración de las personas con los sistemas es a lo que llamamos soluciones "*Human-in-the Loop*" (HiL). La fluidez de la interacción, la transparencia y la comprensibilidad del sistema desempeñan un papel clave en la aceptación de estos sistemas y la experiencia de usuario.

Diseñar de forma adecuada la coordinación entre la persona y el sistema es uno de los retos más importantes a los que se enfrenta la comunidad científica hoy en día. El problema que aparece en estos sistemas es que las personas no comprendan completamente el estado del sistema, los objetivos a alcanzar, o no tengan conocimiento suficiente para llevar a cabo sus acciones [29]. Más aún si tenemos en cuenta que el comportamiento de los sistemas autónomos puede ser emergente, lo que implica que su comportamiento puede sorprender a los usuarios por inesperado. Para hacer frente a este problema, hay un considerable interés reciente en dotar a los sistemas de capacidad para que puedan generar "explicaciones" ante decisiones que el sistema toma en tiempo de ejecución. De esta forma es posible mejorar la comprensión humana y su capacidad para cooperar con el sistema [1]. Las explicaciones son esenciales para mejorar la confianza y la comprensión entre la persona usuaria y el sistema [22], mejorar la colaboración, la

transparencia y aumentar la confianza [16]. Si bien se ha demostrado que las explicaciones son efectivas en muchas circunstancias, el camino hacia conseguir unas explicaciones comprensibles y adaptadas a cada situación o usuario presenta muchos desafíos. No todos los usuarios ni todas las situaciones requieren de las mismas explicaciones, por lo que para evitar abrumar a los usuarios con explicaciones innecesarias, es necesario identificar cuándo un usuario en un contexto particular requiere o no de una explicación.

Abordar adecuadamente estos desafíos requiere de un nuevo enfoque a la hora de diseñar las funciones de un sistema autónomo y su colaboración con las personas. Necesitamos soluciones que permitan mejorar la transparencia de los sistemas autónomos y conseguir sistemas auto-explicables. Estos sistemas auto-explicables deben poder, en tiempo de ejecución, 1) identificar cuándo es necesario ofrecer una explicación sobre lo que está haciendo el sistema, y 2) una vez detectada esta necesidad, generar una explicación adaptada al usuario y al contexto específico del sistema.

La principal contribución de este trabajo es la definición de una propuesta para construir sistemas HiL auto-explicables que generen y adapten las explicaciones en tiempo de ejecución. Estas explicaciones estarán enfocadas en informar al usuario acerca de las acciones de adaptación que ha llevado a cabo el sistema (causa de la adaptación y acción realizada) de forma que se mejore la confianza, el entendimiento y la colaboración entre el usuario y el sistema ante el comportamiento emergente del mismo. Un aspecto clave de la propuesta es la utilización de un diseño centrado en el usuario (DCU) [25] para la detección de la necesidad de explicación para cada situación específica (contexto del usuario, sistema y entorno), ya que dependiendo de lo que está ocurriendo en el sistema puede ser conveniente o no ofrecer una explicación. Con esto lo que se pretende es adaptar las explicaciones a cada situación concreta, consiguiendo que el usuario entienda el sistema pero sin necesidad de abrumarlo en exceso. Estas explicaciones no se definirán en tiempo de diseño (puesto que no se conoce si se requiere la explicación), sino que en tiempo de ejecución, una vez detectada la necesidad de explicación, el sistema generará la explicación de forma automática. Para ello, se utilizan técnicas de Inteligencia Artificial (IA) con dos objetivos:

- Detección de la necesidad de ofrecer una explicación. En concreto se utilizan técnicas de aprendizaje automático que son capaces de inferir, a partir de la interacción que está llevando a cabo el humano con el sistema, si se requieren explicaciones sobre la acción de adaptación que el sistema ha realizado.
- Generación de las explicaciones en lenguaje natural. En concreto se utilizan técnicas de Generación de Lenguaje Natural (GLN) para a partir de la acción de adaptación realizada por el sistema generar una explicación.

El resto del trabajo se estructura de la siguiente forma. La Sección 2 analiza los trabajos relacionados que abordan las explicaciones en sistemas autónomos. En la Sección 3 se detalla el problema que se aborda en el trabajo y se propone una solución para el mismo. La Sección 4 describe los elementos que componen la solución propuesta. En la Sección 5 se presenta un caso de estudio en el dominio de las 'Smart Homes'. Finalmente, la Sección 6 presenta el trabajo futuro y las conclusiones.

2 Estado del Arte

Recientemente, el papel de las explicaciones ha resurgido en el campo de la Inteligencia Artificial con la noción de Inteligencia Artificial Explicable (XAI) y en agentes y robots autónomos como una capacidad importante de éstos [1]. Hellström y Bensch [13] describieron cómo se captura el estado del sistema en la mente de un ser humano. Cuando no se explica el comportamiento del sistema, es posible que el estado en la mente no sea coherente con el estado real, lo que podría conducir a situaciones peligrosas. Además, la falta de un modelo mental para el ser humano para que pueda estimar las acciones del sistema puede desencadenar riesgos para la seguridad [3]. Por lo tanto, las explicaciones son necesarias en sistemas autónomos, como también ratifica la legislación Europea sobre "explicabilidad". En el contexto de las leyes de protección de datos recientes, las leyes europeas exigen ahora que las decisiones que tomen los sistemas autónomos para personas les sean explicadas [7]. Los sistemas autónomos toman muchas decisiones, pero las razones de estas decisiones pueden ser opacas para la persona usuaria [4]. En este trabajo nos centramos en detectar la necesidad de ofrecer una explicación y en generarla adaptándola a cada contexto específico, por tanto, analizaremos trabajos centrados en estos dos aspectos.

La necesidad de explicaciones no siempre es fácil de detectar, un claro ejemplo de ello ocurre en el dominio de los vehículos autónomos, ya que los vehículos toman muchas decisiones a lo largo del tiempo y explicar todas estas decisiones al usuario podría ser contraproducente. Por ejemplo, en un momento dado el vehículo toma una decisión (explícita o implícita) de acelerar o desacelerar, y de realizar un ajuste de dirección (lo que se conoce como el nivel de control de Michon [20]): ¿Deberían los vehículos autónomos explicar estas decisiones continuamente? ¿O solo deberían explicarse las decisiones que generen cambios relevantes en el funcionamiento normal del vehículo como podría ser la selección de cambios de rutas? ¿O solo se necesita una explicación retrospectiva en torno a posibles accidentes? Aunque idealmente un sistema debería poder dar múltiples explicaciones, si lo hacen puede afectar a la atención del usuario y también podría tener un impacto en el rendimiento del sistema (al dedicar capacidad al almacenamiento de decisiones). Desde la perspectiva de las soluciones HiL, la explicabilidad debe estar presente al menos para evitar la 'confusión de modos de automatización' [15] y para evitar la 'fatiga de estar en alerta' [27].

Wüest et al. [30] utilizaron un enfoque de ciclo de retroalimentación para identificar situaciones en las que es valioso pedirle al usuario retroalimentación sobre el comportamiento del sistema. Los autores compararon el comportamiento del usuario con un modelo de objetivos y solicitaban comentarios cuando las personas lograban

subobjetivos o cuando se desviaban de un subobjetivo esperado. Li et al. [17] investigaron cuándo se debían proporcionar explicaciones en sistemas auto-adaptativos dependiendo del coste generado por los retrasos en las adaptaciones y el beneficio de la explicación con respecto a la función de utilidad del sistema. Sin embargo, los autores tratan la explicación como una acción preparatoria. En este trabajo, se trata la explicación como una acción para que el humano comprenda el sistema y se utilizan técnicas de inteligencia artificial que toman como entrada datos del contexto del usuario y su interacción con el sistema para determinar si se necesita una explicación. A diferencia de otros trabajos en nuestra propuesta un aspecto clave es evitar abrumar al usuario con explicaciones innecesarias. Para ello se adaptarán las explicaciones al contexto del usuario.

En cuanto a la generación de la explicación, muchos trabajos se han centrado en este aspecto. Neerincx et al. [23] distinguieron tres fases para proporcionar una explicación: generación de la explicación, comunicación de la explicación y recepción de la explicación. La generación de explicaciones tiene como objetivo generar dos categorías [28]: 1) 'explicación del qué' (*what-explanation*), una descripción de la solución de un problema de planificación; y 2) 'explicación del por qué' (*why-explanation*), una justificación de por qué se selecciona esa solución. Se pueden adoptar modelos de requisitos basados en objetivos para ofrecer una explicación de cómo un sistema cumple sus requisitos [29]. Lim et al. [18] propusieron explicaciones automáticas para los diferentes tipos de explicación. Sukkerd et al. [28] presentaron un método para generar un argumento de cómo se prefiere una solución a otras alternativas. Estos trabajos existentes se centran en el contenido de la explicación en la etapa de generación de la explicación o en los efectos de la explicación en el humano en las etapas de comunicación y recepción. Sin embargo, ninguno de ellos adapta las explicaciones en base al contexto.

Drechsler et al. [5] propusieron los primeros pasos hacia un marco conceptual para un sistema ciberfísico auto-explicativo. En su aproximación proponen agregar una capa de auto-explicación que incluye un modelo abstracto del sistema, y proponen ajustar la granularidad de las explicaciones para diferentes grupos de usuarios. Proponen construir cadenas de causa-efecto para acciones observables utilizando el modelo abstracto. Los usuarios pueden acceder a estas cadenas para comprender la causa de las acciones. Blumreiter et al. [2] propusieron un framework de referencia para construir sistemas ciberfísicos auto-explicativos introduciendo capacidades de auto-reflexión en el bucle de control MAPE-K. Sin embargo, en ninguno de estos trabajos se tienen en cuenta las necesidades de las personas ni el contexto para adaptar las explicaciones.

3 Planteamiento del Problema y Visión General de la Solución

Los sistemas autónomos son capaces de ajustar su comportamiento de acuerdo con cambios en su entorno. Para ello, el sistema realiza adaptaciones de su comportamiento de forma autónoma. Esto puede ocasionar que los usuarios no entiendan lo que está sucediendo en el sistema y se comporten y cooperen de forma errónea con él. Para solucionar esta falta de comprensión y mejorar la cooperación de los humanos con los sistemas, se deben utilizar explicaciones que traten de argumentar a los usuarios los cambios en el comportamiento del sistema. Estas explicaciones no pueden ser vistas como algo estático, sino que deben ser dinámicas, adaptándose a factores internos y externos tales como la capacidad, experiencia, género, o estado de atención de las personas, aparición de fallos o conflictos en el sistema, condiciones del entorno, etc. Además, no siempre se deberían explicar los cambios en el comportamiento del sistema, puesto que podemos abrumar a los usuarios con explicaciones que no sean necesarias. Por lo tanto, se deberían proporcionar las explicaciones solo cuando se detecte que es necesario. Por ejemplo, supongamos una tarea de una 'Smart Home' que se encarga de realizar el pedido de la compra a un supermercado de forma autónoma. Puede ocurrir que cuando el sistema envía el pedido al supermercado, este servicio no esté disponible. En ese caso, el sistema se adapta a este suceso enviando el pedido a otro supermercado. Este cambio en la forma habitual del comportamiento del sistema podría causar que un usuario con poca experiencia en el sistema, pierda la noción de lo que está ocurriendo. En caso de que se detectara que efectivamente el usuario no está entendiendo lo que sucede, por ejemplo, si detectamos que está intentando hacer la compra de forma manual, se debería ofrecer una explicación al usuario

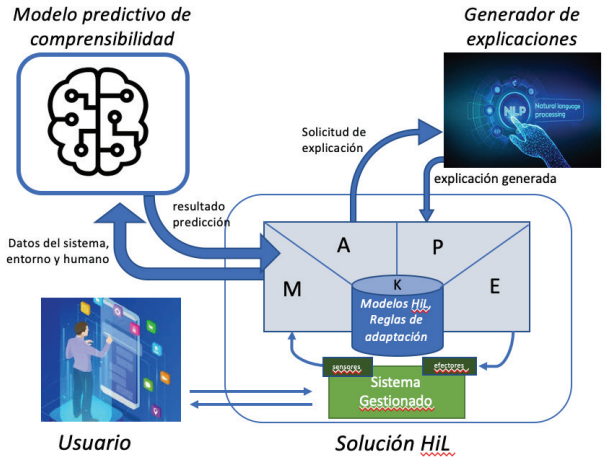

Figura 1: Arquitectura de la solución propuesta

En este trabajo se propone una propuesta para el diseño de soluciones HiL auto-explicables que utiliza un modelo predictivo de comprensibilidad, para inferir cuando es necesario ofrecer una explicación al usuario ante una acción de adaptación del sistema, y un generador de explicaciones para construir la explicación necesaria en tiempo de ejecución. La arquitectura de la solución propuesta se muestra en la Figura 1. El componente base de la arquitectura es el sistema gestionado más un bucle MAPE-K para sistemas auto-adaptativos [14]. El sistema gestionado se obtiene

aplicando el marco de desarrollo propuesto en [9] para el diseño e implementación de tareas colaborativas humano-sistema (estas tareas colaborativos conforman la solución HiL). Para adaptar las tareas al contexto, el bucle MAPE-K es responsable de monitorizar el entorno del usuario y el sistema, además de adaptar la funcionalidad y la comunicación humano-sistema en función de unas reglas de adaptación y de unos modelos que describen las tareas colaborativas.

Partiendo de la arquitectura presentada en la Figura 1, la operacionalización del funcionamiento del sistema gestionado con los componentes de la solución propuesta se puede abstraer en los siguientes pasos:

- **Identificación de la necesidad de explicación**. El módulo de monitorización del bucle MAPE-K, cuando detecta que se ha ejecutado una acción de adaptación, manda una solicitud al modelo predictivo de comprensibilidad. Este modelo es capaz de inferir a partir de los datos de la interacción que está llevando a cabo el humano con el sistema si se requieren explicaciones sobre la acción de adaptación que el sistema ha realizado. La necesidad de explicación puede inferirse, ya sea porque ante una acción de adaptación, el humano no está colaborando de forma correcta con el sistema o porque ni siquiera colabora con él.
- **Generación de la explicación**. Si se detecta la necesidad de dar una explicación ante una acción de adaptación del sistema, el módulo de análisis del bucle MAPE-K manda una solicitud al generador de explicaciones. Este módulo se basa en técnicas de PLN para generar una explicación del comportamiento del sistema utilizando como base de conocimiento las reglas de adaptación.
- **Ejecución de la explicación**. Una vez obtenida la explicación, el módulo de planificación define la acción para ofrecer la explicación. Esta acción la llevará a cabo el módulo de ejecución a través del sistema gestionado.

4 Diseño de los componentes

Una vez presentada la arquitectura de la solución propuesta, en esta sección detallamos el diseño de los componentes involucrados en cada paso de la operacionalización del funcionamiento del sistema.

4.1 Identificación de la necesidad de explicaciones

Para la identificación de la necesidad de explicaciones se propone diseñar y desarrollar un modelo predictivo que permita inferir si es necesario mostrar una explicación al usuario ante una acción de adaptación realizada por el sistema. Esta predicción inferirá si es necesario o no dar una explicación. Con esta información, el componente de PLN (ver Sección 4.2) generará la explicación en caso necesario.

El modelo predictivo se crea utilizando un algoritmo de aprendizaje supervisado que aprenda del comportamiento del usuario, del contexto de ejecución del sistema y de la información del entorno. Los datos que debe utilizar el modelo como entrada para inferir si es necesaria una explicación o no, estarán relacionados con:

- **El Sistema**: datos sobre las acciones que está realizando el sistema. Estas acciones quedan registradas en el log del sistema.
- **El Entorno**: datos sobre el entorno proporcionados por los sensores que monitorizan las variables del entorno del sistema (i.e. temperatura, hora, etc.).
- **El Humano**: datos sobre el perfil del humano, como su experiencia y su capacidad (el perfil se especifica utilizando el modelo OWC el cual categoriza las características de los humanos en tres factores: *opportunity*, *willingness* y *capability* [6]), la ocupación actual del humano (a partir de las interacciones del humano con el sistema se puede obtener esta información) y su interacción con el sistema (información obtenida a través del log del sistema).

Tabla 1: Variables de entrada para el modelo predictivo de la 'Smart Home'.

Id	Variable	Descripción
X_1	Acción de adaptación	Descriptor de la adaptación que se está llevando a cabo en el sistema
X_2	Localización de la adaptación	Identificador de la estancia de la casa a la que afecta la adaptación
X_3	Localización del usuario	Identificador de la estancia de la casa donde se encuentra localizado el usuario
X_4	Perfil del usuario	Descriptor de perfil de usuario que está interactuando con el sistema
X_5	Ocupado	Booleano, si el usuario está ocupado o no lo está
X_6	Acción t_0	Descriptor de la acción que está sucediendo en el sistema en el instante actual
X_7	Dispositivo t_0	Identificador de dispositivo con el que el usuario está interactuando en la acción t_0
X_8	Acción t_{-1}	Descriptor de la acción que está sucediendo en el sistema en el instante previo al actual
X_9	Dispositivo t_{-1}	Identificador de dispositivo con el que el usuario está interactuando en la acción t_{-1}
X_{10}	Acción t_{-2}	Descriptor de acción que está sucediendo en el sistema en el instante previo al anterior
X_{11}	Dispositivo t_{-2}	Identificador de dispositivo con el que el usuario está interactuando en la acción t_{-2}
X_{12}	Temperatura	Grados centígrados de la casa
X_{13}	Hora	Hora actual

Por ejemplo, en el caso de la 'Smart Home', un vector de entrada para el modelo predictivo contendría datos para las variables que se muestran en la Tabla 1. La salida del modelo predictivo es un valor booleano que indica la necesidad o no de proporcionar explicación cuando el sistema ha llevado a cabo una acción de adaptación. Como ejemplo del modelo predictivo que se propone, hemos creado un modelo para el caso de la 'Smart Home'. Para su creación, se ha implementado un algoritmo de aprendizaje supervisado. Estos algoritmos aprenden a partir de muestras de entrenamiento etiquetadas. En nuestro caso, los datos de entrenamiento los hemos obtenido de datos recogidos de un prototipo de una 'Smart Home' en la que disponíamos de los siguientes dispositivos: 2 teléfonos inteligentes, 2 relojes inteligentes, altavoces, 1 simulador de frigorífico inteligente con registro de existencias, 4 bombillas inteligentes, 2 simuladores de aires acondicionados, 2 simuladores de ventanas de apertura y cierre automático, 2 sensores de temperatura, y 2 sensores de presencia.

Los datos de entrenamiento se han generado a través de 20 sesiones con 20 usuarios para los que ejecutábamos 3 escenarios de adaptación en la 'Smart Home': regulación de temperatura, compra automática y regulación de la iluminación. Estos datos de entrenamiento han sido etiquetados por los propios usuarios, por lo que la construcción del modelo predictivo sigue un enfoque de diseño centrado en el usuario. En la Sección 5 se detallan los dos primeros escenarios. La Tabla 2 presenta 3 muestras de los datos de entrenamiento recogidos de acuerdo con las variables identificadas en la Tabla 1.

Tabla 2: Datos recogidos en el ejemplo de la 'Smart Home'

Id	Muestra 1	Muestra 2	Muestra 3
X_1	[actualización sistema]: bloqueo temperatura	[fallo bombilla principal]: activación bombilla auxiliar	[aumento temperatura]: activación aire acondicionado
X_2	Smart Home	Cocina	Smart Home
X_3	Salón	Dormitorio	Salón
X_4	Usuario1	Usuario3	Usuario no identificado
X_5	Ocupado	Ocupado	No ocupado
X_6	Aumentar la temperatura 1 grado	Aumentar la temperatura 1 grado	Ø
X_7	Teléfono móvil del usuario	Teléfono móvil del usuario	Ø
X_8	Aumentar la temperatura 1 grado	Aumentar la temperatura 1 grado	Ø
X_9	Teléfono móvil del usuario	Teléfono móvil del usuario	Ø
X_{10}	Disminuir la temperatura 1 grado	Disminuir la temperatura 1 grado	Ø
X_{11}	Teléfono móvil del usuario	Teléfono móvil del usuario	Ø
X_{12}	19 grados	19 grados	16 grados
X_{13}	08:55:10	08:55:10	22:35:26
Etiqueta	Explicación	No explicación	No explicación

Como se puede ver en la tabla, cada registro se ha etiquetado según el observador veía que el usuario necesitaba o no una explicación de la acción de adaptación que se estaba llevando a cabo. Se ha aumentado el conjunto de muestras mediante la generación de datos sintéticos [8]. Para ello, se ha utilizado el método *Synthetic Minority Oversampling Technique* (SMOTE). Con este método, los datos sintéticos creados se obtienen mediante interpolación de varias muestras de la misma clase que se encuentran cercanas entre sí en el espacio de características. De manera intuitiva, los datos sintéticos son creados emulando las características de las muestras reales.

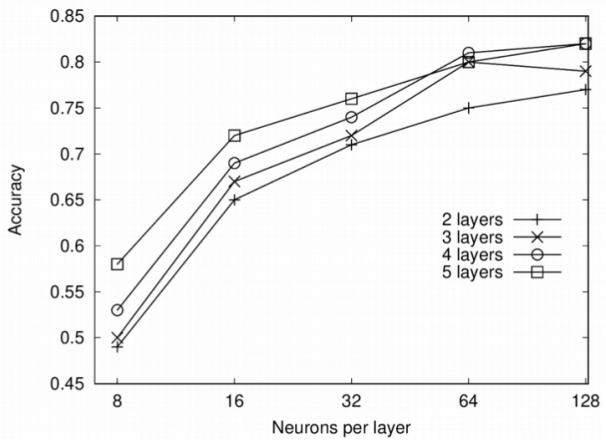

Figura 2: Precisión entre el número de capas ocultas y el número de neuronas por capa

El algoritmo implementado hace uso de redes neuronales densamente conectadas entrenadas mediante aprendizaje supervisado. La función de activación empleada es la *Relu - Rectified Linear Units*, utilizada por su facilidad de entrenamiento en contraposición a otras funciones [11]. Para la estimación de los parámetros se han realizado diferentes pruebas variando el número de capas ocultas y el número de neuronas en cada capa (ver Figura 2).

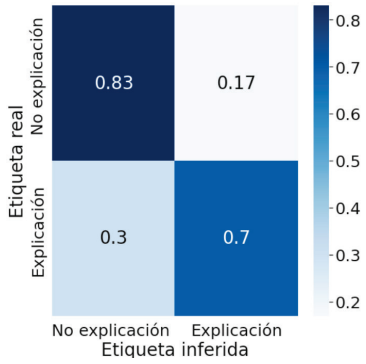

Figura 3: Matriz de confusión de la red neuronal

La configuración óptima es de 4 capas ocultas con 64 neuronas en cada una de ellas. La arquitectura de la red neuronal consta de 6 capas; la capa de input, 4 capas ocultas y la capa de output, compuesta por dos neuronas correspondientes al número de clases: explicación y no explicación. La partición de los datos de aprendizaje es de 75% para entrenamiento, 15% para validación y 10% para test. Se ha obtenido una precisión de 80.6% (ver Figura 3).

4.2 Generación de explicaciones mediante PLN

Para la generación automática de explicaciones en lenguaje natural se propone el diseño y desarrollo de un modelo de PLN que sea capaz de componer explicaciones para explicar al usuario las acciones de adaptación que lleva a cabo el sistema. Las técnicas de PLN, en concreto el área de Generación de Lenguaje Natural (GLN) [19], permiten dotar a los sistemas de capacidad para crear texto en lenguaje humano de manera autónoma. La generación de texto en lenguaje natural requiere, en primer lugar, decidir la información a reproducir, y posteriormente, determinar cómo organizarla y cómo producir el texto en lenguaje natural.

El modelo de GLN utiliza como entrada para generar las explicaciones una regla de adaptación del bucle MAPE-K. Esta regla es la que ha disparado la acción de adaptación que necesita explicación. En concreto, una regla de adaptación especifica las acciones que se han de llevar a cabo ante un evento si se satisface una condición. Para ello, una regla de adaptación se define de la siguiente manera: *Evento [Condición] Acción.* Por ejemplo, supongamos que en la 'Smart Home' queremos que cuando se detecte una subida de temperatura que conlleve que el valor de la temperatura aumente más de 5 grados, se active el aire acondicionado. La regla de adaptación, que llamamos "*AumentoTemperaturaRelevante*", se definiría como:

AumentoTemperatura [Incremento > 5] Activar AC

Se ha construido un modelo de GLN para la 'Smart Home' que utiliza como entrada la última regla de adaptación que se ha ejecutado en el sistema y con esta información genera un texto con la explicación a ofrecer al usuario. El modelo utiliza la siguiente estructura sintáctica genérica para construir las explicaciones:

Acción por *Condición [*y por *Condición,] [*debido a *Evento [*y a *Evento]].*

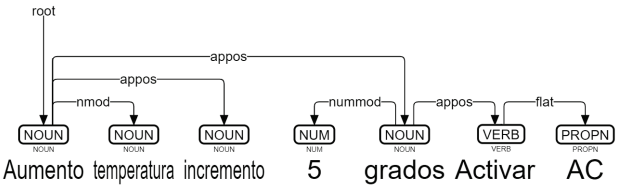

Figura 4: Análisis morfológico de regla de adaptación

En primer lugar, el modelo elimina los caracteres que no son alfanuméricos y tokeniza la regla (es decir, desglosa la regla en palabras). En segundo lugar, realiza un análisis morfológico [26] para identificar la categoría gramatical de las palabras, en el caso de los números, no se les asigna categoría gramatical. La Figura 4 muestra el análisis morfológico de la regla "AumentoTemperaturaRelevante". En este paso, también se añaden preposiciones o artículos para aumentar la comprensibilidad de la sentencia. Por ejemplo, dados dos sustantivos consecutivos, se inserta la preposición "de".

Por último, es necesario que se expliquen las acciones que se han llevado a cabo utilizando el tiempo verbal correcto. En toda regla, la parte Acción contiene un verbo en forma infinitiva que indica la acción realizada en la adaptación. Para indicar que la acción se ha realizado con la forma verbal correcta, el módulo de GLN aplica stemming al verbo (extrae la raíz sin el sufijo infinitivo), y le concatena la terminación de participio. Siguiendo con el ejemplo anterior, dada la acción de la regla: *ActivarAC*, el módulo de GLN transforma este texto en: Activado AC.

Con todo esto, para la regla "AumentoTemperatura [Incremento > 5 grados] ActivarAC", la explicación generada por el módulo de GLN sería:

Activado AC por incremento de 5 grados debido a aumento de temperatura.

Una vez generada la explicación, el módulo de planificación seleccionaría los mecanismos de interacción más adecuados para ofrecer la explicación. A partir de esta selección, el módulo de ejecución ofrecería la explicación mediante estos mecanismos de interacción. Para ello, se propone usar AdaptIO [10], una infraestructura software para adaptar la interacción de notificaciones en base al contexto del usuario. AdaptIO monitoriza el contexto y adapta los mecanismos de interacción de las notificaciones (en nuestro caso las explicaciones) en términos de nivel de molestia. Esta parte queda fuera del alcance del presente trabajo y se plantea como trabajo futuro en la Sección 6.

5 Ejemplo de aplicación de la propuesta

Como ejemplo de aplicación de la propuesta utilizamos el escenario de regulación de temperatura y compra autónoma propuesto por Nilsson et al. [24] en el ámbito de la 'Smart Home'.

5.1 Escenario de regulación de la temperatura

Este escenario implica la regulación de la temperatura dentro de la casa de forma automática monitorizando variables del entorno, tales como la ocupación de la casa. Sin embargo, algunos trabajos [24] reflejan como este comportamiento autónomo muchas veces causa confusión en los usuarios originando una pérdida de confianza de estos.

Supongamos que, debido a una actualización del sistema, se ha ejecutado una acción de adaptación que ha bloqueado

temporalmente la tarea regular temperatura. La regla de adaptación que se ha ejecutado es la siguiente:

ActualizaciónSistema [] BloquearReguladorTemperatura.

El modelo predictivo a partir de la información de la que dispone del sistema, del usuario y del entorno, infiere si es necesario proporcionar una explicación al usuario. Un registro de los datos de entrada del algoritmo se muestra en la Tabla 3.

Los datos de la Tabla 3 reflejan que lo que está sucediendo en el sistema es que el usuario está intentando regular la temperatura (x6, x8, x10) mediante su teléfono móvil (x7, x9, x11), el perfil del usuario es conocido (x4) y este se encuentra en el salón (x3). La acción de adaptación que ha llevado a cabo el sistema es un bloqueo de la regulación de la temperatura, lo que afecta a toda la casa (x2), ante una actualización del sistema (x1). La red neuronal infiere que es necesario proporcionar una explicación con un 84% de probabilidad. Al inferirse necesidad de proporcionar explicación, el componente de GLN genera la sentencia. La explicación sería la siguiente:

Bloqueado regulador de temperatura debido a actualización de sistema.

Tabla 3: Vector de datos que recibe el modelo predictivo de la 'Smart Home' en el escenario de regulación de temperatura

Id	Dato
X_1	ActualizaciónSistema []: BloquearReguladorTemperatura
X_2	Smart Home
X_3	Salón
X_4	Usuario1
X_5	Ocupado
X_6	Aumentar la temperatura 1 grado
X_7	Teléfono móvil del usuario
X_8	Aumentar la temperatura 1 grado
X_9	Teléfono móvil del usuario
X_{10}	Disminuir la temperatura 1 grado
X_{11}	Teléfono móvil del usuario
X_{12}	19 grados
X_{13}	08:55:10

5.2 Escenario de compra autónoma

Este escenario implica la compra automática de alimentos cuando la nevera se está quedando sin existencias. Las acciones de adaptación que se llevan a cabo en este escenario pueden causar confusión en los usuarios. Por ejemplo, al solicitar la compra al supermercado habitual, el servicio de compra no responde. Ante esto, el sistema de compra autónoma decide hacer la compra en otro supermercado puesto que hay un alimento marcado como indispensable. La regla de adaptación que se ha ejecutado es la siguiente:

SolicitudCompra [ServicioCompraNoResponde && CantidadArticuloPrioritario = 0] SolicitarCompraSupermercadoAuxiliar.

El modelo predictivo a partir de la información de la que dispone del sistema, del usuario y del entorno, infiere si es necesario proporcionar una explicación al usuario. Un registro de los datos de entrada del algoritmo se muestra en la Tabla 4.

Tabla 4: Vector de datos que recibe el modelo predictivo de la 'Smart Home' en el escenario de compra autónoma

Id	Dato
X_1	SolicitudCompra [ServicioCompraNoResponde && CantidadArticuloPrioritario = 0] SolicitarCompraSupermercadoAuxiliar
X_2	Smart Home
X_3	Cocina
X_4	Usuario1
X_5	Ocupado
X_6	Buscar mail compra supermercado
X_7	Teléfono móvil del usuario
X_8	Leer mail
X_9	Teléfono móvil del usuario
X_{10}	Leer mail
X_{11}	Teléfono móvil del usuario
X_{12}	19 grados
X_{13}	14:05:15

Los datos de la Tabla 4, reflejan que lo que está sucediendo en el sistema es que el usuario ha consultado su mail y ha buscado un mail del supermercado (x6, x8, x10) mediante su teléfono móvil (x7, x9, x11), el perfil del usuario es conocido (x4) y éste se encuentra en la cocina (x3). La acción de adaptación que ha llevado a cabo el sistema es una solicitud de cambio de supermercado, lo que afecta a toda la casa (x2), ante una actualización del sistema (x1). Ante estos datos, la red neuronal infiere que es necesario proporcionar una explicación con un 82% de probabilidad. Al inferirse necesidad de proporcionar explicación, el modelo de GLN genera una explicación:

Solicitado compra supermercado auxiliar por servicio de compra no responde y cantidad artículo prioritario 0 debido a solicitud compra

6 Conclusiones y trabajo futuro

La comunicación humano-máquina es clave para el éxito de los sistemas autónomos con participación del humano. El trabajo colaborativo requiere el entendimiento del sistema por parte del humano. La necesidad de garantizar o maximizar este entendimiento hace que se planteen nuevas formas de abordar el diseño de la interacción humano-máquina en sistemas autónomos con participación del humano. El diseño de esta interacción no puede hacerse de forma personalizada y adaptable

si se aborda en tiempo de diseño, cuando no se conoce el comportamiento de los usuarios ni se sabe qué posibles situaciones se va a encontrar el sistema. Ante este reto, en este trabajo se propone el uso de técnicas de IA y PLN para la generación automática de explicaciones ante acciones de adaptación que lleva a cabo el sistema. Estas acciones de adaptación implican un cambio en el comportamiento habitual del sistema, lo que puede confundir al humano llevándole a cometer errores en el trabajo colaborativo. Utilizando técnicas de IA se identifican las situaciones en las que los humanos requieren explicaciones, y se generan explicaciones de forma automática en tiempo de ejecución utilizando técnicas de PLN. Este trabajo abre el camino hacia futuros trabajos donde se pretende aprender de la reacción del usuario ante las explicaciones del sistema para continuar aprendiendo y retroalimentando el modelo predictivo.

Como trabajo futuro se pretende validar la propuesta con usuarios. Para ello planteamos reclutar a sujetos con distintos perfiles (expertos y no expertos en el dominio de la 'Smart Home'), y ubicarlos en contextos diferentes (dedicados en exclusividad a la tarea o haciendo otras acciones en paralelo). Para cada contexto, definiremos un conjunto de acciones de adaptación experimentales donde el usuario debe indicar dada la ejecución de una acción de adaptación, si desearía recibir una explicación y en caso afirmativo, la explicación a recibir. Compararemos si la preferencia indicada para ese contexto y ese perfil de usuario coincide o no con el resultado proporcionado con el conjunto de datos de entrenamiento y los algoritmos propuestos en el presente artículo. Las variables respuesta a analizar serán: la correctitud del sistema para inferir la necesidad de explicación y su generación (mide el grado de coincidencia entre la solución ofrecida y las preferencias de los usuarios); la intención de uso de este sistema por los usuarios finales, la utilidad percibida del sistema de recomendación y la facilidad de uso percibida. Además, también se pretende integrar la solución con AdaptIO [10] para adaptar dinámicamente los mecanismos de interacción a utilizar para ofrecer las explicaciones generadas.

ACKNOWLEDGMENTS

Trabajo financiado por la Generalitat Valenciana bajo el proyecto GV/2021/072 y cofinanciado por el MINECO bajo el proyecto AVANTIA PID2020-114480RB-I00.

REFERENCES

[1] Biran, O., Cotton, C. "Explanation and justification in machine learning: A survey," in IJCAI-17 workshop on explainable AI (XAI), vol. 8, 2017, p. 1.

[2] Blumreiter, M. et al., "Towards Self-Explainable Cyber-Physical Systems," 2019 ACM/IEEE 22nd International Conference on MODELS-C, 2019, pp. 543-548.

[3] Broekens, J., Harbers, M., Hindriks, K., Van Den Bosch, K., Jonker, C. and Meyer, J.-J. "Do you get it? user-evaluated explainable bdi agents," in German Conference on Multiagent System Technologies, 2010, pp. 28–39.

[4] Burrell, J., 2016. How the machine 'thinks': understanding opacity in machine learning algorithms. Big Data Soc. 3 (1), 205395171562251 2.

[5] Drechsler, R., Lüth, C., Fey, G., and Güneysu, T. "Towards self-explaining digital systems: A design methodology for the next generation," in 2018 IEEE IVSW, 2018, pp. 1–6.

[6] Eskins, D. And Sanders, W. H.: The Multiple-Asymmetric-Utility System Model: A Framework for Modeling Cyber-Human Systems. QEST '11, 233-242 (2011).

[7] European Union (2018) Corrigendum to Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data Directive 95/46/EC. http://data. euro-pa.eu/eli/reg/2016/679/corrigendum/2018-05-23/oj.

[8] Fernandez, A., Garcia, S., Herrera, F., Chawla, N. V., "SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary". 2018.

[9] Gil, M., Albert, M., Fons, J. et al. Modeling and "smart" prototyping human-in-the-loop interactions for AmI environments. Personal and Ubiquitous Computing (2021).

[10] Gil, M., Pelechano, V. Self-adaptive unobtrusive interactions of mobile computing systems. J. Ambient Intelligence and Smart Environments. 9(6): 659-688 (2017).

[11] Goodfellow, I., Bengio, Y. and Courville, A. (2016). Deep Learning. MIT Press. http://www.deeplearningbook.org.

[12] Hearst, M. A. Mixed-initiative interaction. IEEE Intelligent Systems. pp 14-24, 1999.

[13] Hellström, T. and Bensch, S. "Understandable robots-what, why, and how," Paladyn, Journal of Behavioral Robotics, vol. 9, no. 1, pp. 110– 123, 2018.

[14] IBM, An architectural blueprint for autonomic computing, White Paper 2005

[15] Janssen, C. P., Donker, S. F., Brumby, D. P., Kun, A. L. History and future of human-automation interaction. International Journal of Human-Computer Studies 131 (2019) 99-107.

[16] Le Bras, P., Robb, D. A., Methven, T. S., Padilla, S., and Chantler, M. J. "Improving user confidence in concept maps: Exploring data driven explanations," in CHI Conference on Human Factors in Computing Systems. ACM, 2018, pp. 1–13.

[17] Li, N. Cámara, J. Garlan, D. & Schmerl, B. (2020) "Reasoning about When to Provide Explanation for Human-involved Self-Adaptive Systems," 2020 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS), 2020, pp. 195-204.

[18] Lim, B.Y., Dey, A.K, Avrahami, D. "Why and why not explanations improve the intelligibility of context-aware intelligent systems," in CHI 2009, 2009, pp. 2119–2128.

[19] Martínez, E., Nogales, A., Morales, J., Garcia-Tejedor, A. J. "A light method for data generation: a combination of Markov Chains and Word Embeddings". Procesamiento del Lenguaje Natural. 2020, 64: 85-92.

[20] Michon, J.A., 1985. A critical view of driver behavior models: what do we know, what should we do? Human Behavior and Traffic Safety. Springer, pp. 485–524.

[21] Miller, D., Ju, W., 2015. Joint cognition in automated driving: combining human and machine intelligence to address novel problems.

[22] Nahavandi, S., 2017. Trusted autonomy between humans and robots: toward human-on-the-loop in robotics and autonomous systems. IEEE Syst. Man Cybern. Mag. 3 (1), 10–17.

[23] Neerincx, M. A., vander Waa, J., Kaptein, F., and van Diggelen," J. Using perceptual and cognitive explanations for enhanced human-agent team performance," in Engineering Psychology and Cognitive Ergonomics - EPCE 2018, 2018, pp. 204–214.

[24] Nilsson, T., Crabtree, A., Fischer, J. et al. (2019) Breaching the future: understanding human challenges of autonomous systems for the home. Pers Ubiquit Comput (2019) 23: 287.

[25] Norman, D. A. & Draper, S. W. (Editors) (1986) "User-Centered System Design: New Perspectives on Human-Computer Interaction". Lawrence Earlbaum Associates, Hillsdale, NJ.

[26] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton and Christopher D. Manning. 2020. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In Association for Computational Linguistics (ACL) System Demonstrations. 2020.

[27] Sorkin, R.D., 1989. Why are people turning off our alarms. Hum. Factors Bull. 32 (4), 3–4.

[28] Sukkerd, R. "Improving transparency and understandability of multi-objective probabilistic planning," Thesis Proposal – School of Computer Science Institute for Software Research Software Engineering, pp. 1–41, 2018.

[29] Welsh, K., Bencomo, N., Sawyer, P., Whittle, J. "Self-explanation in adaptive systems based on runtime goal-based models," Trans. Comput. Collective Intell., v. 16, pp. 122–145, 2014.

[30] Wüest, D., Fotrousi, F. and Fricker, S. "Combining monitoring and autonomous feedback requests to elicit actionable knowledge of system use," in RE, 2019, pp. 209–225.