



**.XXXVII.**

# JORNADAS DE AUTOMÁTICA

---

FACULTAD DE INFORMÁTICA  
UNIVERSIDAD COMPLUTENSE DE MADRID

## LIBRO DE ACTAS

7, 8 y 9  
SEPTIEMBRE  
2016  
**MADRID**



# INTERFAZ INTELIGENTE Y ADAPTATIVA PARA PERSONAS CON DISCAPACIDAD

María Taboada, Juan Jesús Roldán y Antonio Barrientos

Centro de Automática y Robótica (UPM-CSIC)

C/ José Gutiérrez Abascal 2, 28006 Madrid

[maria.taboadap@alumnos.upm.es](mailto:maria.taboadap@alumnos.upm.es), [jj.roldan@upm.es](mailto:jj.roldan@upm.es), [antonio.barrientos@upm.es](mailto:antonio.barrientos@upm.es)

## Resumen

*La comunicación es una necesidad básica para los seres humanos, pero existen enfermedades y lesiones que la pueden restringir. Los Sistemas Aumentativos y Alternativos de Comunicación permiten a las personas con discapacidad comunicarse con las personas que las rodean y realizar algunas tareas cotidianas. Este trabajo propone ir más allá en el desarrollo de estas interfaces y proporcionarles un cierto nivel de inteligencia y adaptación. La interfaz desarrollada emplea técnicas de inteligencia artificial y minería de datos para aprender el comportamiento del operador y predecir sus siguientes peticiones. El objetivo de estos algoritmos es doble: por un lado, hacer una selección con las opciones más habituales y, por otro, asignar las posiciones más cómodas a las opciones más probables.*

**Palabras Clave:** Interfaz inteligente y adaptativa, Personas con discapacidad, Minería de procesos, Minería de datos, Inteligencia artificial.

## 1 INTRODUCCIÓN

La naturaleza social del ser humano hace de la comunicación una herramienta fundamental en el día a día de cualquier persona. A través de ella expresamos nuestras necesidades, pensamientos y emociones, lo que nos permite establecer relaciones personales, integrarnos y participar en la sociedad. Por esta razón es importante que ninguna persona se vea privada de esta capacidad, ya que es un factor fundamental para lograr una calidad de vida digna.

Sin embargo, existe un grupo de personas que sufren enfermedades o lesiones que merman o impiden totalmente su habilidad comunicativa. Algunos ejemplos pueden ser la Esclerosis Lateral Amiotrófica (ELA), la Parálisis Cerebral, los Trastornos del Espectro Autista o la Esclerosis Múltiple. Por este motivo, se han desarrollado Sistemas Aumentativos y Alternativos de Comunicación (SAAC) [1], que dotan a estas

personas de medios para expresarse de forma adecuada y facilitan su interacción con los individuos que les rodean. Estos sistemas se denominan aumentativos porque incrementan las capacidades de comunicación de las personas con disfunciones del habla y alternativos porque compensan las dificultades empleando formas de expresión distintas al lenguaje hablado.

En particular, una de cada quinientas personas en España sufre Parálisis Cerebral según la Confederación ASPACE. En todo el mundo, esta cifra alcanza 17 millones, de los cuales el 25% de ellos necesita utilizar algún tipo de SAAC. Por otro lado, en el mundo hay unas 400.000 personas con ELA y cada año se descubren unos 140.000 casos nuevos. En general, unas 7 personas de cada 1.000 que habitan el mundo necesitarán usar un SAAC en algún momento de su vida [2].

En los últimos años se han realizado numerosos estudios basados en la combinación de los distintos tipos de SAAC con las nuevas tecnologías, con el objetivo de responder a las distintas necesidades de las personas con graves disfunciones motoras y del habla. Por ejemplo, se han elaborado múltiples proyectos enfocados al desarrollo de tableros de comunicación basados en pictogramas en los que el usuario puede diseñar manualmente las plantillas. Sin embargo, esta personalización tiene una serie de límites, ya que se realiza en un momento concreto, no se adapta a los cambios de conducta, restringe el número de opciones y no depende del contexto.

Este trabajo propone la aplicación de disciplinas como la inteligencia artificial y la minería de datos al desarrollo de una interfaz para personas con discapacidad. El resultado es una interfaz inteligente y adaptativa controlada por un sistema de seguimiento de ojo. En concreto, la interfaz consiste en un tablero virtual pictográfico de comunicación aumentativa y alternativa gestionado por algoritmos de predicción del comportamiento. Los algoritmos implementados utilizan varias técnicas para adaptar automáticamente la interfaz a las necesidades y costumbres del usuario, de forma que la comunicación sea más rápida, eficaz y cómoda. Este



- Sección superior: esta sección contiene un cuadro de texto con la opción seleccionada y un botón que permite su confirmación.



Figura 3: Sección superior

- Sección central: esta zona no tiene elementos y permite al usuario situar el cursor cuando no quiera realizar ninguna selección.
- Sección inferior: esta sección contiene cuatro botones fijos: emergencia, que abre una ventana complementaria (ver figura 7); domótica, que muestra las opciones de domótica en la sección de categorías; actividades, que muestra las opciones fuera de la domótica en dicha sección; y cuidador, que llama al cuidador sin necesidad de confirmación.



Figura 4: Sección inferior

Por otra parte, los paneles predictivos (figuras 5 y 6) muestran las opciones seleccionadas y ordenadas por los algoritmos de predicción. Estos algoritmos trabajan con las selecciones del usuario en el pasado y tratan de predecir su comportamiento en el futuro. En concreto, la interfaz tiene los siguientes paneles predictivos:

- Sección izquierda: en esta sección se muestran las opciones seleccionadas y ordenadas por los algoritmos de predicción. Aquí sólo se muestran opciones que el usuario ha seleccionado y confirmado previamente.



Figura 5: Sección predictiva de selecciones confirmadas

- Sección derecha: en esta sección se encuentran todas las opciones que incorpora la aplicación organizadas por categorías. La idea es que el usuario utilice esta sección sólo cuando la opción que desea no haya sido seleccionada por los algoritmos y, por tanto, no esté entre las opciones de la sección izquierda. Para facilitar

la búsqueda, tiene integrado un algoritmo de predicción por árboles de decisión.

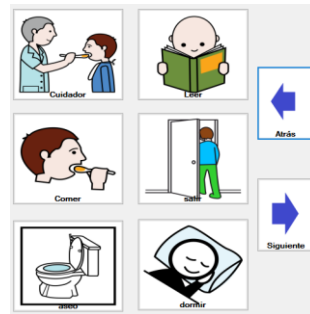


Figura 6: Sección predictiva por categorías

Cabe destacar que las predicciones se disponen en orden descendente de importancia (de acuerdo con los algoritmos) desde la zona central de la interfaz (zona de acceso más cómodo) hacia los extremos (zona de acceso menos cómodo). De esta forma, en la sección izquierda los botones de la columna de la derecha de esta sección tendrán las predicciones con más peso, mientras que los de la izquierda les corresponden las menos precisas.

### 2.2.2 Ventana de emergencia

La ventana de emergencia es complementaria a la ventana principal y accesible a través del botón de emergencia. Como se puede ver en la figura 7, esta ventana tiene opciones de dolor, picor, intensidad, partes del cuerpo y llamar al 112. También cuenta con un botón de confirmación de selección y un cuadro de texto donde aparece la selección confirmada.

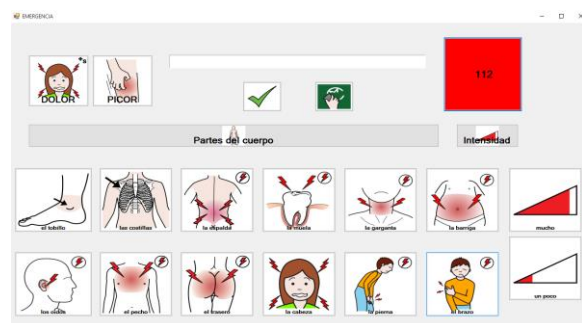


Figura 7: Ventana de emergencia

## 3 APRENDIZAJE

El proceso de aprendizaje consta de dos etapas: en la primera se elaboran registros de comportamiento, mientras que en la segunda se obtienen modelos de comportamiento.

### 3.1 REGISTROS DE COMPORTAMIENTO

En los registros de comportamiento se guardan los comandos del usuario, junto con la fecha y hora del momento en el que se han realizado. Las tablas 1 y 2 muestran dos ejemplos de estos registros y a continuación se describen sus características:

- Registros de categorías: a cada categoría le corresponde un registro y en ellos se recogen todas las selecciones realizadas en las secciones laterales. Por ejemplo, en el registro de “Comer” se guardan opciones como “Verduras” y “Pasta” (ver tabla 1).
- Registro de selecciones: en este registro sólo se almacenan aquellas opciones que el usuario ha seleccionado y confirmado. Así, si el usuario selecciona “Leer”, seguidamente “Periódico” y a continuación confirma la selección, solamente “Periódico” se almacenará en este registro, mientras que en los registro de categorías se guardarán ambas selecciones (ver tabla 2).

Tabla 1: Ejemplo de registro de la categoría “Comer”

BOTON	AÑO	MES	DIA	HORA	MINUTOS	SEGUNDO	DÍAS SEMANA
verdura	2016	6	14	16	36	5	Tuesday
habitacion	2016	6	14	16	37	11	Tuesday
verdura	2016	6	14	16	38	5	Tuesday
pasta	2016	6	14	16	39	17	Tuesday
pescado	2016	6	14	16	39	19	Tuesday
pasta	2016	6	14	17	6	36	Tuesday
pasta	2016	6	14	20	5	53	Tuesday
verdura	2016	6	14	20	6	0	Tuesday
legumbre	2016	6	14	20	22	49	Tuesday
legumbre	2016	6	14	20	37	55	Tuesday
pasta	2016	6	14	21	5	26	Tuesday
verdura	2016	6	14	21	5	26	Tuesday

Tabla 2: Ejemplo de registro de comportamiento de acciones seleccionadas

BOTON	AÑO	MES	DIA	HORA	MINUTOS	SEGUNDO	DÍAS SEMANA
despertar	2016	5	23	6	45	8	Moday
ducha	2016	5	23	6	50	4	Moday
peinar	2016	5	23	7	3	5	Moday
cafe	2016	5	23	7	8	51	Moday
cepillar lo	2016	5	23	7	17	48	Moday
darse crer	2016	5	23	7	21	48	Moday
luz off	2016	5	23	7	25	58	Moday
salir	2016	5	23	7	30	9	Moday
trabajar	2016	5	23	7	59	39	Moday
cafe	2016	5	23	11	0	29	Moday

### 3.2 MODELOS DE COMPORTAMIENTO

Los modelos de comportamiento permiten comprender y predecir el comportamiento del usuario, además de detectar posibles desviaciones en sus acciones.

Se generan a partir de registros de eventos sobre los que se aplican distintas técnicas de modelado. Entre

estas técnicas se encuentran la minería de datos y la minería de procesos. Por un lado, la minería de datos se centra en el análisis de datos para elaborar predicciones o clasificaciones [3]. Por otro lado, la minería de procesos elabora modelos a partir de los eventos, basándose generalmente en la sucesión de éstos [4].

Hay una gran variedad tanto de modelos como de algoritmos de modelado. En este trabajo se han utilizado árboles de decisión, obtenidos mediante minería de datos, y redes de Petri, generadas mediante minería de procesos [5].

#### 3.2.1 Árboles de decisión

Los árboles de decisión son una herramienta de clasificación y regresión común en minería de datos y aprendizaje automático [6]. Estos modelos establecen relaciones entre unas variables objetivo (en este caso, las opciones seleccionadas por el usuario) y unas variables observables (en este caso, la fecha y hora y las condiciones en las que se produjo esa selección). Las hojas de los árboles corresponden a las variables objetivo, mientras que los nodos representan las variables observables.

Hay varios algoritmos para obtener árboles de decisión [7]. En este trabajo se ha empleado un algoritmo de particionamiento recursivo implementado en *Rapidminer Studio 7.0* [8]. Este algoritmo parte de los registros de comportamiento, asignando la variable respuesta a las opciones seleccionadas y las variables predictivas a los demás atributos. A continuación realiza una iteración en la que va buscando las variables predictivas que separan las variables de respuesta obteniendo la entropía mínima. Esta iteración inaliza cuando las variables de respuesta están completamente separadas o cuando la entropía no se puede reducir. La primera condición se suele dar cuando hay más variables observables que objetivo, mientras que la segunda es común cuando hay más variables objetivo que observables.

En este caso, la interfaz maneja más opciones que condiciones, por lo que el árbol no proporciona en sus hojas una sola opción, sino un conjunto de opciones con sus probabilidades. Esta información permite a la interfaz seleccionar las opciones que debe mostrar en los paneles dinámicos y, además, ordenarlas según su probabilidad.

#### 3.2.2 Redes de Petri

Las redes de Petri son una herramienta matemática y gráfica para el modelado de sistemas [9]. Estos modelos tienen forma de grafos con lugares, transiciones y arcos que los conectan. Por lo tanto,

son modelos capaces de representar todas las posibles secuencias de estados de un sistema.

Hay varios procedimientos para generar redes de Petri. En este caso se ha optado por los algoritmos de descubrimiento de la minería de procesos [10]. La minería de procesos es una disciplina que estudia los procesos mediante registros de eventos y varias clases de modelos. Por su parte, los algoritmos de descubrimiento son capaces de generar automáticamente redes de Petri a partir de estos registros de eventos.

En este trabajo se ha empleado el programa *Fluxicon Disco* [11] para generar el registro de eventos a partir del registro de comportamiento de la interfaz. Los eventos contienen un caso (el día de la semana), una actividad (el botón seleccionado y confirmado), una fecha y hora y otros datos (por ejemplo, la temperatura). La elección del día de la semana como caso para los eventos no es trivial, sino que se basa en un análisis. Por un lado, esta selección consigue que el modelo represente un periodo acotado con un principio y un final definidos: por ejemplo, un día desde que el usuario se levanta hasta que se acuesta. Por otro lado, el modelo puede descubrir varios caminos diferentes según el caso: por ejemplo, una rutina laboral de lunes a viernes y una rutina de descanso los fines de semana.

A continuación, se ha utilizado el algoritmo de minado inductivo implementado en el programa *ProM 6.5.1 (Process Mining framework)* [12] para generar redes de Petri. Este algoritmo busca patrones comunes en los eventos y los divide mediante operadores lógicos y bucles.

La red de Petri permite determinar qué opciones pueden seguir a la actual y ordenarlas según su probabilidad. Para ello, las opciones a las que se puede evolucionar se enumeran de menor a mayor, correspondiendo el menor número a la opción más cercana y el mayor a la más alejada en el modelo.

## 4 PREDICCIÓN

El proceso de aprendizaje consta de una etapa de elaboración de registros de comportamiento y otra de obtención de modelos de comportamiento.

### 4.1 Predicción con árboles de decisión

Una vez obtenidos los registros de comportamiento, se obtienen los árboles mediante el programa *RapidMiner Studio 7.0*.

Al iniciar la aplicación, los árboles se cargan desde sus archivos a una lista en la memoria del programa. Estos árboles están compuestos desde la raíz hasta las

hojas por una serie de nodos conectados a un nodo padre y varios nodos hijos. Cada nodo contiene un número que identifica su posición en el árbol, un nombre, un número que identifica a su padre, una variable predictiva, un valor o un intervalo de valores y una lista de casos con probabilidades. En el caso de que un nodo sea una hoja del árbol, su nombre será el de la opción más probable y su lista de probabilidades se compondrá de las diferentes opciones y el número de veces que se escogieron.

Una vez cargados los árboles, el programa lee el árbol pertinente siguiendo las ramas que cumplen los valores de las variables predictivas de ese momento hasta llegar a la hoja adecuada. En ese momento, busca en la lista de casos de dicho nodo el máximo e introduce la opción correspondiente en el primer botón. Esta opción se elimina de la lista y se repite el proceso para el valor del segundo botón. Esto se produce sucesivamente hasta que se les hayan asignado valores a todos los botones. De esta manera, cuanto menor sea el número de botón, más probable es que el usuario lo seleccione.

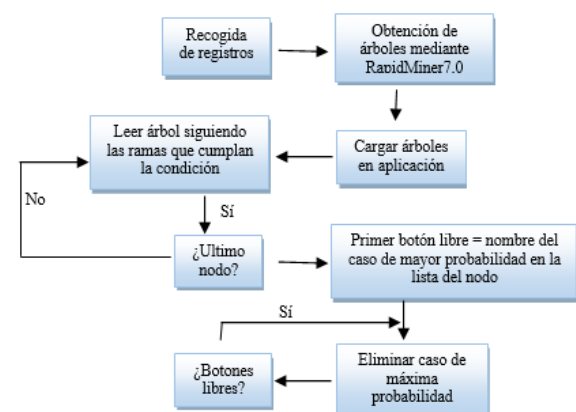


Figura 8: Predicción por árboles de decisión

Este método de predicción se utiliza en las dos secciones predictivas de la interfaz. En el caso de la zona de selecciones confirmadas, la lectura del árbol se realiza periódicamente, mientras que en la sección de categorías solo se lee el árbol de esa categoría cuando ésta ha sido seleccionada. En concreto, el árbol inicial que se lee en esta sección es el de las categorías generales.

### 4.2 Predicción con algoritmo de minado inductivo

En este caso, tras la obtención del modelo mediante el programa *ProM*, se deben realizar de forma manual las listas de evolución de cada una de las opciones. Estas listas se ordenan en función de la cercanía a la opción a tratar y la frecuencia de cada uno de los posibles caminos a seguir. Una vez elaboradas, se cargan en el programa en el momento

de su inicialización y se tratan de la misma manera que las listas de casos de los árboles de decisión.

## 5 EXPERIMENTOS

Los experimentos se han realizado registrando la actividad diaria de dos usuarios sanos de alrededor de 50 años durante varias semanas. A lo largo de este periodo, los modelos se han actualizado añadiendo cada semana los registros de actividad recogidos la semana anterior. Los datos almacenados para la predicción han sido las actividades desarrolladas por el usuario y la fecha y hora en la que tuvieron lugar.

Se han estudiado dos tipos de modelados para la predicción. Por un lado, árboles de decisión basados en los días de la semana, la hora y los minutos. Estos modelos clasifican los eventos de forma que, dadas unas determinadas condiciones, devuelven unas probabilidades para cada actividad. Por otro lado, se ha empleado el algoritmo de minado inductivo, que busca patrones en los eventos para predecir cuál será el siguiente a uno dado, para generar una red de Petri del proceso.

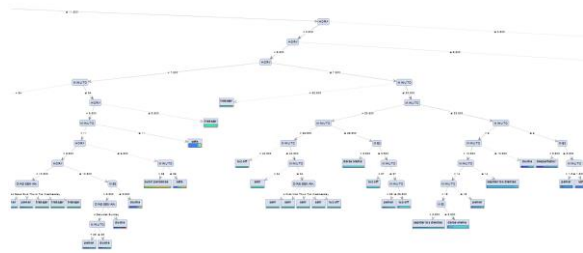


Figura 9: Ejemplo de árbol de decisión

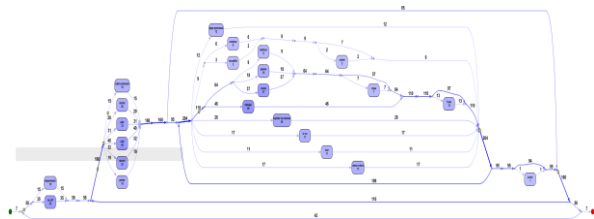


Figura 10: Ejemplo de red de Petri

Para el estudio de la eficiencia de los algoritmos se ha registrado el nivel de acierto en la predicción de la interfaz y se ha analizado su evolución a lo largo de cuatro semanas.

En las siguientes figuras se muestran los resultados obtenidos utilizando árboles de decisión para la predicción. En concreto, en la figura 11 se representa el número de predicciones acertadas, nulas y mayores de cero en las diferentes semanas. Las primeras corresponden a los casos en los que el usuario ha elegido la opción de mayor probabilidad según el

modelado. Las predicciones nulas son aquellas cuya probabilidad devuelta por el algoritmo es nula, pero el usuario las ha seleccionado. Por último, las predicciones no nulas son aquellas que tienen probabilidad mayor que cero y han sido seleccionadas por el usuario independientemente de que sean las de mayor valor.

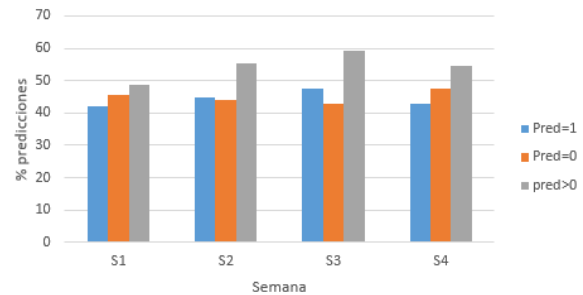


Figura 11: Evolución de las predicciones de los árboles de decisión

Por otra parte, los resultados obtenidos mediante las redes de Petri se muestran en la figura 12, donde se refleja la evolución de la eficiencia de predicción a lo largo de cuatro semanas. En ella se contabilizan las veces que el usuario selecciona una opción situada en la sección de predicción con minería de procesos.

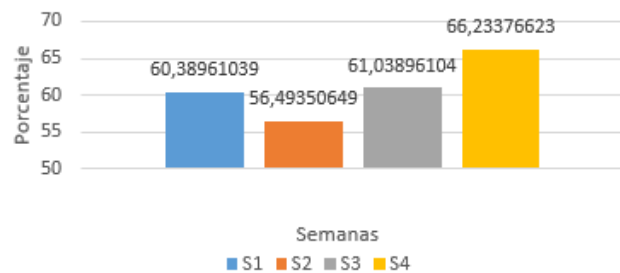


Figura 12: Evolución de la predicción mediante las redes de Petri.

## 6 CONCLUSIONES

En este trabajo se ha desarrollado una interfaz inteligente adaptativa de comunicación alternativa y aumentativa, orientada a personas con discapacidad motora y del habla. Para ello se ha dispuesto un diseño adecuado para su control mediante sistemas de seguimiento de ojos o cabeza. En concreto, se trata de un tablero virtual de comunicación mediante pictogramas con paneles grandes que se adapta al usuario para facilitar su comunicación con las personas que le rodean y, de este modo, mejorar su calidad de vida.

La interfaz contiene varias secciones con distintas funcionalidades y se adapta al usuario mediante algoritmos de predicción. En particular, se ha implementado un modelo de árbol de decisión y otro

de red de Petri y se han estudiado sus rendimientos por separado para la predicción del comportamiento humano.

Los resultados demuestran que el empleo de estos algoritmos mejora la comunicación, ya que prioriza la salida por pantalla de las opciones preferidas por el usuario. Además, las predicciones mejoran a lo largo del tiempo siempre y cuando el usuario mantenga una cierta rutina, puesto que los algoritmos presentan una cierta sensibilidad a las variaciones. En el algoritmo basado en árboles de decisión, la sensibilidad se encuentra en los parámetros de tiempo. Por su parte, en el algoritmo de minería de procesos, al basarse en la sucesión de eventos, los cambios de secuencia en estos pueden tener importantes connotaciones.

A pesar de que el comportamiento humano suele presentar determinadas rutinas, también puede experimentar variaciones aleatorias. Por ello es necesario desarrollar algoritmos robustos que permitan englobar las múltiples circunstancias que se pueden dar. En particular, una posible mejora en este caso sería un aumento del tiempo del estudio para que el registro de eventos contenga más ejemplos. También sería interesante estudiar la combinación de ambos algoritmos, estudiando el uso de los árboles de decisión en los puntos de decisión de las redes de Petri.

Por último, cabe resaltar que si bien este trabajo se ha desarrollado para ser aplicado a una interfaz para personas con limitaciones en sus capacidades de comunicación, los métodos de predicción y adaptación son plenamente aplicables a otros tipos de interfaces. En particular, su uso puede ser eficiente en aquellas interfaces en las que un operador debe manejar sistemas complejos, como la monitorización y control de procesos o flotas de robots. Estas aplicaciones son líneas de trabajo complementarias en las que también se están aplicando los desarrollos expuestos en este artículo.

### Agradecimientos

Esta investigación ha recibido fondos del proyecto RoboCity2030-III-CM (Robótica aplicada a la mejora de la calidad de vida de los ciudadanos, fase III; S2013/MIT-2748), financiado por los Programas de Actividades I+D en la Comunidad de Madrid y los Fondos Estructurales de la Unión Europea, y del proyecto DPI2014-56985-R (Protección robotizada de infraestructuras críticas), financiado por el Ministerio de Economía y Competitividad del Gobierno de España.

Este trabajo está enmarcado en el proyecto SAVIER (Situational Awareness Virtual Environment), que

está financiado y apoyado por Airbus Defence & Space.

### Referencias

- [1] K. Wiegand (2014), "Thesis Proposal: Disambiguation of Imprecise User Input Through Intelligent Assistive Communication". *NORTHEASTERN UNIVERSITY*.
- [2] Portal Aragonés de Comunicación Aumentativa y Alternativa. <http://arasaac.org/>
- [3] J. Han, J. Pei, M. Kamber (2011), "Data mining: concepts and techniques". *Elsevier*.
- [4] J. J. Roldán, J. del Cerro, A. Barrientos (2015), "Minería de procesos aplicada a la monitorización de las misiones robóticas". *XXXVI Jornadas de Automática, 2-4 septiembre 2015, Bilbao. ISBN 978-84-15914-12-9, Comité Español de Automática de la IFAC (CEA-IFAC)*.
- [5] W. Van Der Aalst (2011), "Process mining: discovery, conformance and enhancement of business processes". *Springer Science & Business Media*.
- [6] L. Rokach, O. Maimon (2014), "Data mining with decision trees: theory and applications". *World scientific*.
- [7] S. Lomax and S. Vadera (2013), "A survey of cost-sensitive decision tree induction algorithms", *ACM Computing Surveys (CSUR)*, 45(2), 16.
- [8] RapidMiner Studio 7.0 es un software que ofrece múltiples herramientas para hacer análisis predictivos en cualquier ámbito. <https://rapidminer.com/products/studio/>
- [9] T. Murata (1989), "Petri nets: Properties, analysis and applications". *Proceedings of the IEEE*, 77(4), 541-580.
- [10] B. F. van Dongen, A. K. A. De Medeiros, L. Wen (2009), "Process mining: Overview and outlook of petri net discovery algorithms", *Transactions on Petri Nets and Other Models of Concurrency II*, Springer, pp. 225-242.
- [11] Fluxicon Disco es un programa que genera información estructurada y visual a partir de datos brutos. <https://fluxicon.com/disco/>
- [12] ProM 6.5.1 es un programa desarrollado por la Universidad de Eindhoven que contiene



multitud de herramientas para realizar análisis  
de minería de procesos.  
<http://www.processmining.org/prom/start>