



# Método para medir la credibilidad de los agentes en videojuegos empleando Descripción Lingüística de Fenómenos Complejos y su aplicación como herramienta de apoyo en el proceso de enseñanza-aprendizaje en un curso de Inteligencia Artificial \*†‡

Clemente Rubio-Manzano, Tomás Lermenda, Alejandra Segura, Claudia Martínez-Araneda y Christian Vidal  
SOMOS Research Group - Universidad del Bío-Bío & MCIS Research Group - Universidad de Cádiz

## Resumen

El objetivo de este trabajo es proponer un nuevo método para medir la credibilidad de los agentes artificiales que actúan en un mundo virtual empleando la Descripción Lingüística de Fenómenos Complejos y las Funciones de Equivalencia Restringidas. La combinación de ambas técnicas computacionales permite generar y comparar informes de comportamiento en lenguaje natural generados automáticamente. Con el fin de mostrar y explorar las posibilidades de este nuevo método, se desarrolló una aplicación web que puede emplearse como herramienta de apoyo a la docencia de Inteligencia Artificial y donde cada alumno puede evaluar por sí mismo la calidad de su implementación en los respectivos proyectos.

**Palabras Clave:** Inteligencia Artificial, Videojuegos, Descripción Lingüística de Datos, Funciones de Equivalencia Restringidas, Lógica Difusa.

## I. Introducción

La Inteligencia Artificial (IA) tiene como objetivo lograr que una máquina tenga una inteligencia similar a la humana [7]. Sin embargo, históricamente, han coexistido dos visiones (IA fuerte e IA débil) que se corresponden con las siguientes definiciones dadas por Marvin Minsky y John McCarthy, respectivamente: “IA es la ciencia de

*crear máquinas capaces de realizar tareas que requerirían inteligencia si las tuviera que hacer un humano*”; “IA es la ciencia y la ingeniería de crear máquinas inteligentes”. Como se indica en [7]: “Todos los avances logrados hasta ahora en el campo de la IA son manifestaciones de la IA débil”, es decir, dotar a las máquinas de la capacidad de realizar tareas específicas mejor que las personas se puede realizar con éxito, incluso, una IA específica puede superar a la pericia humana.

Por otro lado, un aspecto fundamental en el campo de la IA es determinar si las máquinas son realmente inteligentes, entonces dicha evaluación dependerá de la visión de IA que se tenga. En el caso de la IA fuerte, Alan Turing propuso su famoso test para determinar si un computador exhibía inteligencia humana. En el caso de la IA débil se optó por chequear si las máquinas realizan correctamente las tareas para las cuales fueron diseñadas y programadas.

La IA débil se ha aplicado con éxito en numerosas disciplinas. En el ámbito de los videojuegos, la IA se centra en crear entidades software (creadas mediante algoritmos/programas de computador que actúan sobre un entorno virtual) que produzcan en el espectador la ilusión de inteligencia. Por otro lado, se ha comprobado que los agentes (también llamados NPCs, por sus siglas en inglés Non-player Characters, o Bots) más divertidos y “jugables” son aquellos que se comportan como los jugadores humanos [13]. Surge así el desafío de tratar de desarrollar NPCs que posean un buen grado de imitación y, al mismo tiempo, diseñar e implementar instrumentos que permitan medirlo y evaluarlo [6]. No solo se trata en evaluar que un NPC realice las tareas específicas como un jugador humano, además, queremos que la IA se parezca a la IA del jugador humano lo más posible, debe ser creíble, cuanto más, mejor.

Esta característica se conoce como “credibilidad” (“believability”) y habitualmente se evalúa empleando una extensión el mencionado Test de Turing [5]: “Supongamos que estamos jugando e interactuamos con algún agente, ¿se podría distinguir observándolo al agente si es un

\* Clemente Rubio-Manzano. Departamento de Matemáticas, Universidad de Cádiz, España

† Clemente Rubio-Manzano, Tomás Lermenda, Christian Vidal y Alejandra Segura, Departamento de Sistemas de Información, Universidad del Bío-Bío, Chile

‡ Claudia Martínez-Araneda. Departamento de Ingeniería Informática, Universidad Católica Santísima Concepción, Chile

*jugador humano o un bot? Si no se puede distinguir, entonces se considera que el bot ha pasado el test* ". Esta prueba se diseñó para evaluar las habilidades de los bots para suplantar a un jugador humano.

El test anterior se puede considerar una evaluación basada en el chequeo visual de una determinada funcionalidad. Este hecho tiene como principal limitación a la subjetividad del proceso porque la observación visual es una tarea compleja al requerir procesar una gran cantidad de información generada durante las partidas. Incluso, cabe la posibilidad, de pérdida información debido a la rapidez de ejecución de los algoritmos de IA.

Nuestro trabajo consiste en enriquecer estos test alternativos para reducir la subjetividad mediante la captura de todos los datos posibles generados durante las partidas y entregar automáticamente un conjunto de descripciones lingüísticas en lenguaje natural que proporcionen información sobre el comportamiento de los agentes. Para ello, emplearemos la Descripción Lingüística de Fenómenos Complejos (DLFC)[14] y las Funciones de Equivalencia Restringidas (REFs, de sus siglas en inglés) [1]. En este sentido, el "comportamiento de un agente" está relacionado directamente con el comportamiento del algoritmo IA elegido para su implementación, en este caso, algoritmos de búsqueda en un espacio de estados.

La estructura del trabajo es como sigue, en la Sección II se realiza una introducción desde el punto de vista del proceso de enseñanza-aprendizaje al ser éste el fin último de su aplicación. La Sección III detalla un método para medir la credibilidad de los agentes implementados y comparar el comportamiento de los jugadores humanos con los agentes. Posteriormente, en la Sección IV se explica la experimentación y la evaluación implementadas sobre los proyectos de los estudiantes. Finalmente, la Sección de conclusiones y trabajo futuro proporciona algunos comentarios finales sobre el presente trabajo y el futuro del mismo.

## II. Motivación

Desde el año 2011 se ha implementado una nueva metodología de enseñanza-aprendizaje basada en videojuegos en el curso de IA de la Universidad del Bío-Bío (Chile). Su objetivo es proporcionar a los estudiantes una herramienta de ayuda en el proceso de comprensión de los diferentes algoritmos heurísticos que se enseñan en las clases de teoría.

En particular, en el año 2017, el proyecto consistió en desarrollar un videojuego cuyo agente protagonista tenía que capturar un conjunto de recompensas (que proporcionan energía) distribuidas por el escenario. El agente, que pierde energía cada cinco segundos, debía mantenerse en el escenario durante el mayor tiempo posible sin agotar

energía. Tres agentes adversarios (también programados por los estudiantes) tratarán de robarle la energía, en el caso de que ésta se agote la partida termina.

En este contexto, un agente (y su algoritmo de comportamiento asociado) está correctamente implementado cuando es capaz de imitar a un jugador experto humano (ver sección III) . El proceso de evaluación consiste en chequear la calidad de la imitación desde dos puntos de vista: visualmente, observando el comportamiento del agente artificial en una partida; formalmente, analizando el código del proyecto del alumno. Este proceso tiene algunas limitaciones:

- 1) Tiempo excesivo que se requiere para chequear de forma visual la funcionalidad de los proyectos. Se podría convertir en un problema cuando el número de estudiantes es alto y hay un único profesor.
- 2) Tarea compleja debido a la dificultad de evaluar un gran número parámetros importantes en el proyecto: cantidad (memoria utilizada, iteraciones realizadas, estructuras de datos usadas) y calidad (como es la IA, es decir, que habilidad tiene a la hora de realizar ciertas acciones)
- 3) Se pueden perder numerosos detalles en la inspección visual al transcurrir las partidas muy rápidamente: el depurador genera mucha información que es difícil de entender; gran cantidad de datos que se generan en la ejecución.
- 4) Imposibilidad (dificultad) de realizar una evaluación individual

Para superar estas limitaciones se propone incorporar la DLFC al proceso de enseñanza-aprendizaje de la asignatura IA y, en particular, a su evaluación basada en proyectos. Utilizando DLFC los estudiantes pueden obtener retroalimentación inmediata, técnica y personalizada sobre los fallos cometidos durante el desarrollo de sus proyectos y conseguir más información sobre todos los detalles de ejecución de los algoritmos heurísticos empleados para programar los agentes inteligentes. En la literatura, existen trabajos que han estudiado la aplicación de la DLFC (y en la algunos casos de la Descripción Lingüística de Datos-LDD) al proceso de enseñanza-aprendizaje [9], [3], [2] y [12].

Por otro lado, esta propuesta es muy beneficiosa para los profesores ya que les permitirá:

- Ahorrar tiempo al permitirles centrarse en otros aspectos a evaluar lo que implicará un mejor entendimiento de la implementación de los algoritmos.
- Mejorar el proceso clásico de evaluación proporcionando retroalimentación personalizada a los estudiantes.
- Apoyar el desarrollo individual de proyectos al permitir un seguimiento más cercano y la oportunidad



de centrarse en los aspectos más debilidades de los estudiantes.

Con el fin de mostrar y explorar las posibilidades de esta herramienta se ha desarrollado una plataforma web y se ha incorporado al proceso de enseñanza aprendizaje de la asignatura. Ahora cada estudiante puede obtener, en tiempo real, retroalimentación sobre su proyecto, además establecer su propio plan de aprendizaje. Adicionalmente, los perfiles de comportamiento para tales agentes y jugadores humanos se pueden utilizar para medir la calidad de las IA mediante una adaptación del test Turing al contexto de los videojuegos.

### III. Un método para medir la credibilidad de los agentes y compararlo con jugadores humanos

En esta sección se presenta un método para medir la credibilidad de los agentes basado en cuatro pasos.

#### Paso 1. Generación de la traza de ejecución.

Se ha empleado una traza de ejecución para capturar y almacenar los datos generados a partir de los movimientos de los humanos, agentes y oponentes. Una traza de ejecución es una técnica empleada habitualmente para capturar y almacenar los valores de las estructuras de datos empleadas en una aplicación. Se emplea, también, como herramienta de depuración y análisis del rendimiento.

Concretamente, la traza de ejecución implica la detección y almacenamiento de eventos relevantes durante la ejecución de un programa para posteriormente realizar un análisis más detallado. En particular, empleamos las trazas de ejecución para almacenar los valores de las métricas a partir de los cuales se pueden analizar el comportamiento de las entidades. Las métricas son las siguientes:

- Distancia entre jugador ( $P$ ) y adversarios ( $O$ ) calculada a partir de sus posiciones.
- Distancia entre jugador, adversarios y la recompensa más cercana ( $R^*$ ) calculada a partir de sus posiciones.
- Tiempo en llegar a la recompensa.
- Energía del jugador en cada momento.
- Memoria ocupada e iteraciones realizadas.

#### Paso 2. Definición de la red de percepciones computacionales

Ampliamos la red de percepciones computacionales presentada en [10]. En este caso, la energía, memoria ocupada e iteraciones realizadas se incorporan como nuevas variables. Se extiende la red, reglas y las plantillas de generación de lenguaje natural para soportar los nuevos requisitos.

Las variables lingüísticas se definen a partir de las métricas establecidas previamente. Mediante la agregación de los términos lingüísticos que componen cada una de las variables se definen las reglas de comportamiento. Por ejemplo, la percepción computacional (CP, de sus siglas en inglés) de situación ( $CP_{Situation}$ ) quedaría definida de la siguiente forma:

$$CP_{Situation} = CP_{Protection}(P, O), CP_{Distance}(P, O), CP_{Energy}(P).$$

Esta ecuación establece que la CP de situación se calcula en función de la protección, la distancia entre el jugador-oponente y la energía del jugador. Los valores correspondientes para esta CP se calculan usando las siguientes reglas:

Consecuente (Situación)	Antecedente (Protección, Distancia, Energía)
Risky	Intermediate, Close, Normal
Dangerous	Low, Close, Normal
Safe	Intermediate, Normal, Normal
Easy	Low, Normal, Normal
Dangerous	Low, Normal, Low
Dangerous	Normal, Close, Low
Dangerous	Normal, Normal, Low

El resto de CPs se definen de forma análoga:

- $CP_{Attitude} = CP_{Distance}(P, R^*), CP_{Distance}(O, R^*)$
- $CP_{Movement} = CP_{Distance}(P, R^*), CP_{Distance}(P, O), CP_{Energy}(P)$
- $CP_{Ability} = CP_{Attitude}, CP_{Movement}, CP_{Time}$
- $CP_{Skill} = CP_{Attitude}, CP_{Movement}, CP_{Situation}$
- $CP_{Resources} = CP_{Iterations}, CP_{Memory}$

#### Paso 3. Generación automática de los perfiles de comportamiento.

En cada periodo de tiempo (habitualmente cada segundo) se calculan los valores para cada una de las CPs instantáneas. Con el fin de agruparlas y resumirlas se emplea el concepto de CP Resumen, denotada por  $\Sigma CP$  y definida como  $\Sigma CP = ((a_1, w_1), \dots, (a_n, w_n))$ . El objetivo de cada  $\Sigma CP$  es sintetizar en un vector lingüístico difuso el resultado global obtenido tras la partida para una CP. En nuestro caso, este tipo de CP nos permite obtener el número de veces que un valor  $(a_1, \dots, a_n)$  ocurrió durante la partida.

Estos tipos de CP nos proporcionan un conjunto de variables, su valor asociado y el grado  $\alpha$ , que indica la media difusa para un valor determinado. Por ejemplo, un valor para la CP de Situación podría ser *Safe* con un grado de 0.8 en un instante  $i$ ,  $X = Safe$  con 0.7 en un instante  $i + 1$ , y etc. Por tanto, al final de la ejecución, tendremos que  $a_i$  (en el ejemplo “safe”) se ha dado  $N$  veces con  $M$  grados diferentes  $\beta_1, \dots, \beta_m$  (por supuesto, algunos de

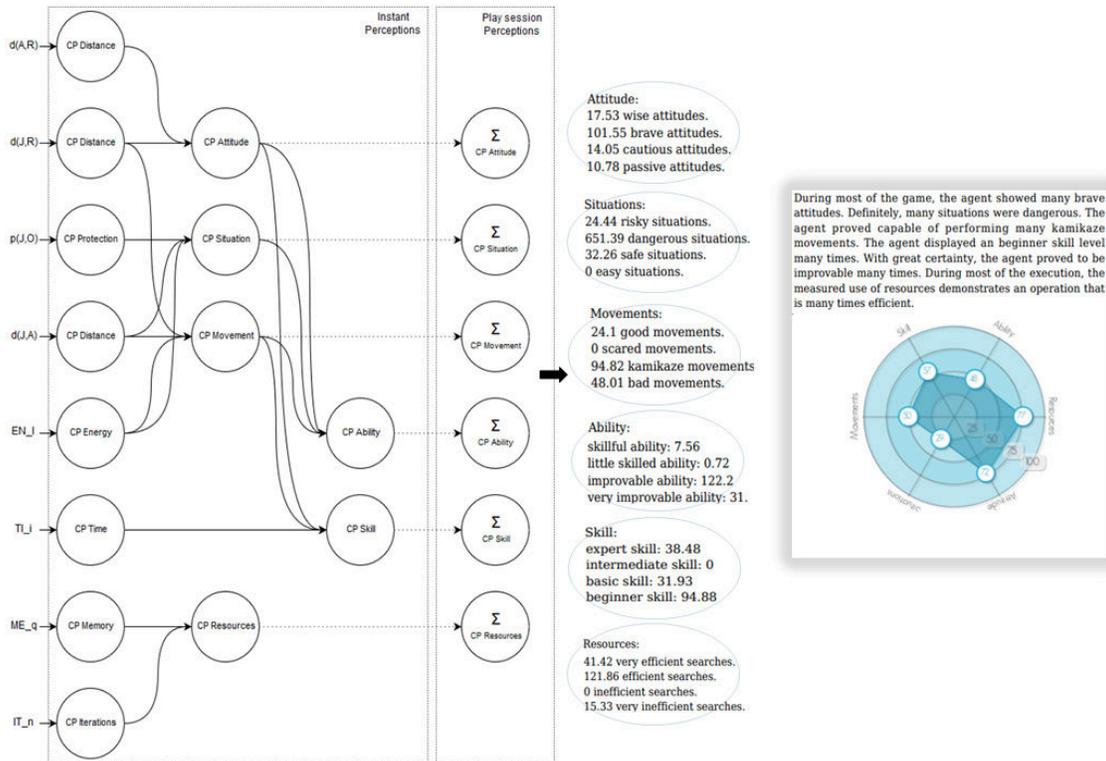


Fig. 1. Generación automática de informe de comportamiento a partir de los valores capturados en la traza de ejecución

estos grados podrían ser iguales). El grado final se calcula como sigue:  $\alpha_i = ((\beta_1 + \dots + \beta_n)/N)$ .

Por ejemplo, los siguientes resúmenes pueden obtenerse a partir de diferentes  $\Sigma CP$  (ver Figura 1). La generación del perfil de comportamiento se realiza empleando el conjunto de  $\Sigma CP$ . Para cada CP se crea una descripción lingüística en función del par  $(a_i, w_i) \in \Sigma CP$ . Los porcentajes se calculan para cada  $\Sigma CP$ . El porcentaje  $p_i$  se transforma en un término lingüístico de cantidad como sigue: "few" cuando  $p_i \in [0, 1/3]$ ; "several" cuando  $p_i \in [1/3, 2/3]$  o "many" cuando  $p_i \in [2/3, 1]$ . Entonces, vamos a considerar cuatro casos:

- 1) Existe un par  $(a_i, p_i) \in \Sigma CP$  cuyo  $p_i$  es mayor o igual a 66 por ciento.
- 2) Existe un par  $(a_i, p_i) \in \Sigma CP$  cuyo  $p_i$  es mayor o igual a 33 por ciento.
- 3) Existe dos pares  $(a_1, p_1), (a_2, p_2) \in \Sigma CP$  cuyos  $p_i$  son mayor o igual a 33 por ciento.
- 4) No existe ningún par  $(a_i, p_i) \in \Sigma CP$  cuyos  $p_i$  son mayor a 33 por ciento.

Y generar las siguientes descripciones en función de ellos. Mostramos aquí la tabla de CP de situación. El informe completo se puede ver en la Figura 2.

Casos	Descripción Lingüística
1	Definitely, <i>degree</i> situations were <i>value</i>
2	<i>degree</i> situations were <i>value</i>
3	<i>degree</i> situations were <i>value</i> <sub>1</sub> , although <i>degree</i> situation also were <i>value</i> <sub>2</sub>
4	Diverse situations were detected during the most part of the play session

#### Paso 4. Método para comparar perfiles de comportamiento entre jugadores humanos y agentes.

Como se ha mencionado en la introducción, uno de los objetivos más importantes en IA en el ámbito de los videojuegos es crear agentes que imiten la habilidad de un jugador humano. Por lo tanto, debemos definir la referencia de mejor jugador humano analizando las partidas y obteniendo un perfil de comportamiento referencia. Tras este proceso se obtuvo que:

- La actitud del jugador es valiente durante la mayor parte de la partida;
- Las situaciones por las que pasa son principalmente seguras;
- Los movimientos realizados son principalmente buenos;
- Se obtiene que es experto y hábil con un uso eficiente de tiempo y espacio.

Ahora, se puede proceder a la comparación de los mismos



mediante la definición de una medida de similitud basada en las funciones de equivalencia restringidas (REFs) entre las CPs Resumen obtenidas para cada caso.

**Definición 3.1:** Una REF,  $f$ , es una correspondencia  $[0, 1]^2 \rightarrow [0, 1]$  que satisface las siguientes condiciones:

- 1)  $f(x, y) = f(y, x)$  para todo  $x, y \in [0, 1]$
- 2)  $f(x, y) = 1$  si y sólo si  $x = y$
- 3)  $f(x, y) = 0$  si y sólo si  $x = 1$  y  $y = 0$  o  $x = 0$  y  $y = 1$
- 4)  $f(x, y) = f(c(x), c(y))$  para todo  $x, y \in [0, 1]$ ,  $c$  siendo una negación fuerte.
- 5) Para todo  $x, y, z \in [0, 1]$ , si  $x \leq y \leq z$ , entonces  $f(x, y) \geq f(x, z)$  and  $f(y, z) \geq f(x, z)$

Por ejemplo,  $g(x, y) = 1 - |x - y|$  satisface las condiciones (1)-(5) con  $c(x) = 1 - x$  para todo  $x \in [0, 1]$ . Una medida de similitud basada en REFs entre dos términos lingüísticos para ampliar el motor de inferencia del lenguaje Bousi Prolog se propuso en [11].

Un algoritmo de IA, usado para implementar un agente, tiene un comportamiento cercano al jugador humano cuando sus respectivos perfiles de comportamiento son similares. La puntuación final (de 1 a 7) se calcula mediante una medida de similitud para cada CP. La ecuación para calcular el grado de credibilidad es la siguiente:

$$FG = G_{Min} + S(Actitud) + S(Situacin) + S(Movimiento) + S(Habilidad) + S(Competencia) + S(Eficiencia)$$

$G_{Min} = 1$  punto (mínimo puntuación que un alumno en la Universidad del Bío-Bío puede obtener). Donde  $S(Attitude) = S_{REF}(\Sigma CP_{Attitude}^{Human}, \Sigma CP_{Attitude}^{Bot})$  es la similitud entre la actitud del jugador humano y la actitud del agente. El resto de relaciones de similitud se definen de forma análoga. La siguiente definición formaliza esta media:

**Definición 3.2:** Dadas dos percepciones computacionales  $\Sigma CP_i$ ,  $\Sigma CP_j$  cuyos vectores de porcentajes lingüísticos son  $\{(a_1, p_1) \dots, (a_n, p_n)\}$  y  $\{(b_1, q_1) \dots, (b_n, q_n)\}$  respectivamente. Una medida de similitud entre  $\Sigma CP_i$  y  $\Sigma CP_j$  se define como

$$S_{REF}(\Sigma CP_i, \Sigma CP_j) = \sum_{i=0}^n (REF(p_i, q_i)) / n$$

siendo  $REF(p_i, q_i) = 1 - |p_i - q_i|$

Por ejemplo, dadas dos CP Resumen  $CP_{Actitud}^{Humano}$ ,  $CP_{Actitud}^{Agente}$ , entonces  $\Sigma CP_{Actitud}^{Humano} = \{(wise, 122.35), (brave, 289), (cautious, 87.59), (passive, 8.75)\}$  y  $\Sigma CP_{Actitud}^{Agente} = \{(wise, 17.53), (brave, 101.55), (cautious, 14.05), (passive, 10.78)\}$ . Entonces el vector de porcentaje lingüístico se calcula para cada  $\Sigma CP$  empleando los totales:

$Total_{\Sigma CP_{Actitud}^{Humano}} (507.69)$   $Total_{\Sigma CP_{Actitud}^{Agente}} (143.61)$ , respectivamente:

- $\Sigma CP_{Actitud}^{Human} = \{(wise, 0.240), (brave, 0.569), (cau-$

$tious, 0.172), (passive, 0.017)\}$

- $\Sigma CP_{Actitud}^{Bot} = \{(wise, 0.122), (brave, 0.709), (cautious, 0.097), (passive, 0.075)\}$

Ahora, la similitud  $S_{REF}(\Sigma CP_{Attitude}^{Human}, \Sigma CP_{Attitude}^{Bot})$  se calcula con:  $REF(0.240, 0.122) = 1 - |0.240 - 0.122| = 0.882$ ,  $REF(0.569, 0.172) = 1 - |0.569 - 0.172| = 0.882$ ,  $REF(0.172, 0.097) = 1 - |0.172 - 0.097| = 0.925$  y  $REF(0.017, 0.075) = 1 - |0.017 - 0.075| = 0.942$  Por tanto,  $S_{REF}(\Sigma CP_{Attitude}^{Human}, \Sigma CP_{Attitude}^{Bot}) = \frac{3.402}{4} = 0.838$ . El resto de similitudes se calcula de forma similar. La puntuación final, junto al perfil de comportamiento obtenido para el jugador humano y el agente, proporciona al alumno una referencia sobre la calidad de los algoritmos implementados para los agentes actuando en el videojuego.

#### IV. Hacia una herramienta de apoyo a la docencia y el aprendizaje de algoritmos

Para llevar a cabo las ideas planteadas y definidas en las secciones anteriores se optó por el desarrollo una plataforma web que permite a cada estudiante obtener un informe de evaluación de forma automática que le proporcionará retroalimentación de forma inmediata sobre la calidad de implementación de sus agentes. La herramienta permite realizar una comparación entre el comportamiento de un jugador humano y los agentes desarrollados en los proyectos. De esta forma el alumno obtiene una nota de referencia de credibilidad de los mismos.

Esto puede verse como un test de turing permitiéndole testear a partir de las trazas de ejecución <sup>1</sup> que genera su proyecto un informe sobre el perfil de comportamiento de sus agentes inteligentes. A la herramienta se accede a través de la siguiente URL:

<http://www.youractionsdefineyou.com/assess>

La ventana principal muestra dos opciones: log in y register. El registro de un usuario consiste en introducir el correo electrónico, nombre de usuario, RUT (el equivalente al DNI español) y una clave. Para confirmar que el registro en la aplicación se ha realizado correctamente se enviará un correo electrónico al usuario. Una vez realizado el registro, el usuario puede ingresar introduciendo su usuario y clave. Una vez dentro de la aplicación, el informe de perfil de comportamiento se puede obtener seleccionando y cargando un archivo de traza de ejecución, cuando este archivo se carga el informe se obtiene automáticamente.

#### V. Agradecimientos

Al proyecto interno de la Universidad del Bío-Bío 170915

<sup>1</sup>[http://youractionsdefineyou.com/assess/web/examples\\_traces](http://youractionsdefineyou.com/assess/web/examples_traces)

## HUMAN BEHAVIOR PROFILE

The human player showed several brave attitudes. Definitely, many situations were safe. The human player proved to be capable of performing several good movements. The human player displayed an expert skill level several times. The human player proved to be skillful several times. During the most of the execution, the measured use of resources demonstrates an operation that is many times very efficient.

Grade: 7



## BOT BEHAVIOR PROFILE

During the most of the game, the bot showed many brave attitudes. Definitely, many situations were dangerous. The bot proved to be capable of performing several kamikaze movements. The bot displayed an beginner skill level several times. With great certainty, the bot proved to be improvable many times. During the most of the execution, the measured use of resources demonstrates an operation that is many times efficient.

Grade: 5.15



Fig. 2. Similitud entre los informes de comportamiento generados para el jugador humano y los agentes

2/R. Al grupo de investigación SOMOS (Software-MOdeling-Science) con referencia GI170315/EF

## VI. Conclusiones y trabajo futuro

En este trabajo se ha tratado el tema de la evaluación de los sistemas de Inteligencia Artificial (fuerte y débil) en el ámbito de los videojuegos. Se ha planteado la idea de evaluar la “inteligencia humana” de los agentes artificiales y se ha presentado un método para medir la “credibilidad” de los mismos analizando su comportamiento y comparándolo con el de los jugadores humanos. El método permite obtener de forma automática un informe sobre el comportamiento del jugador humano que puede compararse con la del agente. Para ello, se ha definido una medida de similitud entre las percepciones computacionales resumen encargadas de capturar y sintetizar los eventos ocurridos durante las partidas.

Esta es una primera aproximación hacia un marco más amplio y aún queda mucho trabajo por hacer. Nos gustaría incorporar en un futuro más detalles que enriquezcan la evaluación de los proyectos. Permitir que el sistema pueda entregar recomendaciones más ricas y detalladas sobre los fallos cometidos por los alumnos durante el diseño e implementación de los algoritmos de Inteligencia Artificial.

## References

- [1] Bustice, H., Barrenechea, E., Pagola, M. Restricted Equivalence Functions. *Fuzzy Sets and Systems* 157, pp. 2333-2346 (2006).
- [2] García-Honrado, I., Fortuny, J. M., Ferrer, M., Morera, L. (2016). Análisis del aprovechamiento de oportunidades de aprendizaje generadas en la discusión en gran grupo de un problema de transformaciones geométricas. *Investigación en Educación Matemática XX*, 253-264.
- [3] Gkatzia, D., Hastie, H., Janarthnam, S., & Lemon, O. Generating student feedback from time-series data using Reinforcement Learning. In *Proceedings of the 14th European Workshop on Natural Language Generation* (pp. 115-124).
- [4] Hernández-Orallo, J. (2017). Evaluation in artificial intelligence: from task-oriented to ability-oriented measurement. *Artificial Intelligence Review*, 48(3), 397-447.
- [5] Hingston, P. (2009). A turing test for computer game bots. *IEEE Transactions on Computational Intelligence and AI in Games*, 1(3), 169-186.
- [6] Livingstone, D. (2006). Turing’s test and believable AI in games. *Computers in Entertainment (CIE)*, 4(1), 6.
- [7] López de Mantaras Badia, R., Mesenguer González, P. (2017) ¿Qué sabemos de? *Inteligencia Artificial Editorial: CSIC y Catarata*
- [8] Phedes 2017 Phedes Lab: Linguistic Description of Complex Phenomena. <http://phedes.com/>
- [9] Ramos-Soto, A., Vazquez-Barreiros, B., Bugarín, A., Gewerc, A., & Barro, S. Evaluation of a Data To Text System for Verbalizing a Learning Analytics Dashboard. *International Journal of Intelligent Systems*, 32(2), 177-193.
- [10] Rubio-Manzano, C., & Trivino, G. Improving player experience in Computer Games by using players’ behavior analysis and linguistic descriptions. *International Journal of Human-Computer Studies*, 95, 27-38.
- [11] Rubio-Manzano, C. Similarity measure between linguistic terms by using restricted equivalence functions and its application to expert systems 9th European Symposium on Computational Intelligence and Mathematics. *Faro (Portugal)*, October 4th-7th, 2017.
- [12] Sánchez-Torrubia, M. G., Torres-Blanc, C., & Trivino, G. (2012). An approach to automatic learning assessment based on the computational theory of perceptions. *Expert Systems with Applications*, 39(15), 12177-12191.
- [13] Soni, P. & Hingston, P. (2008, June). Bots trained to play like a human are more fun. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence)*. IEEE International Joint Conference on (pp. 363-369). IEEE.
- [14] Trivino, G & Sugeno, M. Towards linguistic descriptions of phenomena. *International Journal of Approximate Reasoning*, 54(1), 22-34.