

hazLógica: o cómo aprender lógica haciendo lógica

Carmen López Rincón. Dpto. Lógica y Filosofía Teórica. Email: clrincon@ucm.es
ORCID: 0000-0002-1232-8199

Jaime Sarabia Álvarez-Ude. Dpto. Lógica y Filosofía Teórica. Email: sarabia@ucm.es
ORCID: 0000-0002-5504-6693

Resumen: Presentamos *hazLógica*, una aplicación de acceso libre en la red, originada en la experiencia de enseñanza de lógica a estudiantes de humanidades, y desarrollada por dos profesores de lógica en Filosofía. *hazLógica* es, en la doble acepción de *haz*, una invitación a hacer lógica y un haz de herramientas para simplificar los pasos iniciales de esta tarea. En sus versiones beta ha sido empleada durante este curso y los dos anteriores con resultados muy satisfactorios: ha mejorado la adquisición y comprensión de los contenidos propios de las asignaturas Lógica I y II (estudios de grado en Filosofía, dobles grados de Derecho-Filosofía y Políticas-Filosofía), con una mejora sustancial en las calificaciones. También se ha utilizado este curso por primera vez en la asignatura Lógica y Filosofía del Lenguaje (Máster de Investigación en Lengua Española) con resultados igualmente alentadores. Su dirección es <https://verbalogica.org/hazLogica/>

Palabras clave: lógica en humanidades, aprendizaje de lógica asistido por ordenador, ludificación en lógica, demostración automática

1. El problema

La asignatura de Lógica (I y II), perteneciente al grado de Filosofía y a los dobles grados de Derecho-Filosofía y CC. Políticas-Filosofía, con frecuencia resulta formal y metodológicamente extraña al perfil de un gran número de estudiantes.

Son dificultades habituales la necesidad de aprender lenguajes especializados, no naturales; la existencia de definiciones explícitas, tanto en términos sintácticos como semánticos; la práctica de la demostración rigurosa y la resolución de problemas. Estas dificultades, sin embargo, constituyen a la vez una de las aportaciones más valiosas de la asignatura a su formación, tanto desde un punto de vista filosófico: qué se puede decir, cómo decirlo con rigor; como práctico: son cada vez más los contextos en que es necesario conciliar perfiles de humanidades y ciencias sociales con una competencia suficiente en el uso de lenguajes especializados, particularmente de programación.

La existencia de herramientas de apoyo virtual al aprendizaje permite que los estudiantes, en función de sus necesidades, puedan reforzar los contenidos presentados en clase. En grupos numerosos (40-60) esto es particularmente interesante, dado que resulta imposible atender de forma personalizada las dificultades con las que muchos se encuentran. Esta fue la principal motivación que nos llevó a desarrollar *hazLógica*.

2. Descripción general

hazLógica es un sistema cliente-servidor: un servidor en la nube da servicio a clientes en un navegador. El servidor realiza las tareas de generación y análisis de objetos: lenguajes, oraciones, interpretaciones, etc. y en él residen, entre otros, los subsistemas de demostración automática y juegos. La interfaz de usuario efectúa la gestión y presentación

de los objetos. No es necesario instalar complemento o aplicación alguna para usarlo; basta una página del navegador.

El núcleo de *hazLógica* es un conjunto de sistemas de deducción automática, implementadas en SWI-Prolog, un lenguaje de programación lógica. Es destacable en *hazLógica* la capacidad de generar de forma automática oraciones e interpretaciones siempre nuevas, así como derivaciones y demostraciones tanto en sintaxis como en semántica lógica. Esto permite que el estudiante pueda enfrentarse con problemas inéditos y esté en condiciones de modular el esfuerzo exigido según sus preferencias y necesidades. Se propicia, por otra parte, el abandono de prácticas de aprendizaje basadas en la mera solución repetitiva de ejercicios-tipo.

En la actualidad consta de once espacios, siete de aprendizaje estándar y cuatro dedicados a juegos. Entre los siete primeros hay cuatro que tratan de, respectivamente, lógica de enunciados (*Ingrid LE*), lógica monádica de primer orden (*Ingrid LPO1*), lógica de primer orden (*Ingrid LPO*) y LPO con identidad (*Ingrid LPO=*). Cada uno de ellos es una extensión conservadora del anterior. En el quinto espacio (*OnGraph*) se estudian propiedades de las relaciones diádicas: reflexividad, simetría, etc; el sexto (*Semántica lógica*), se ocupa de nociones básicas como modelo, verdad lógica, consecuencia y equivalencia; y el séptimo (*Deducción*) presenta dos versiones del cálculo de tablas analíticas.

Además de estos siete módulos de aprendizaje estándar, hemos integrado en *hazLógica* un conjunto de juegos que hace posible una aproximación alternativa, para ciertos perfiles más estimulante, al aprendizaje de lógica. Éste adquiere así un cierto aspecto de desafío intelectual y de competición, que tiende a eliminar, al menos parcialmente, el rechazo del estudiante medio a una materia que, sobre todo al principio, le resulta extraña y contraintuitiva.

En lo que sigue nos centraremos en la presentación de dos aspectos de *hazLógica*: la demostración automática de propiedades sintácticas y semánticas; y los juegos.

3. Demostración automática

Como hemos apuntado antes, la práctica de la demostración rigurosa es una dificultad habitual entre quienes se disponen a aprender lógica. Esto se hace particularmente agudo para estudiantes de humanidades por dos razones fundamentales: su falta de experiencia previa, por un lado; y la ausencia de un ámbito de conocimiento familiar sobre el que hacer las demostraciones, por otro. Conviene considerar que son las matemáticas el campo por excelencia de aplicación de la demostración y que éste con demasiada frecuencia no es un ámbito precisamente popular entre estudiantes de humanidades.

En *hazLógica* hemos tratado de vencer esta dificultad usando la teoría lógica misma como ámbito de aprendizaje de la práctica de la demostración. Entendemos que la mejor forma de aprender lógica es haciendo lógica. Así, están incorporados en el programa subsistemas de demostración tanto en sintaxis como en semántica.

3.1. Derivación gramatical

El primer ámbito en el que *hazLógica* instruye en la práctica de la demostración es el de la derivación gramatical. Dada una gramática sintagmática, expresada alternativamente en lenguaje natural o Backus-Naur, *hazLógica* es capaz de justificar mediante una

derivación por qué una secuencia de signos del alfabeto considerado es gramaticalmente aceptable, es decir, está bien formada.

La Figura 1 es un ejemplo de la derivación gramatical de la oración $\neg \exists x (R(x,b) \rightarrow S(x,b))$ usando la notación Backus-Naur:

```
<oración> ::= <o. compuesta>
<oración> ::= <o. monádica>
<oración> ::= <conectiva monádica> <oración>
<oración> ::= ¬ <oración>
<oración> ::= ¬ <o. compuesta>
<oración> ::= ¬ <o. cuantificada>
<oración> ::= ¬ <cuantificador> <variable> <oración>
<oración> ::= ¬ ∃ <variable> <oración>
<oración> ::= ¬ ∃ x <oración>
<oración> ::= ¬ ∃ x <o. compuesta>
<oración> ::= ¬ ∃ x <o. diádica>
<oración> ::= ¬ ∃ x <p. izd.> <oración> <conectiva diádica> <oración> <p. dch.>
<oración> ::= ¬ ∃ x ( <oración> <conectiva diádica> <oración> <p. dch.>
<oración> ::= ¬ ∃ x ( <o. simple> <conectiva diádica> <oración> <p. dch.>
<oración> ::= ¬ ∃ x ( R(x,b) <conectiva diádica> <oración> <p. dch.>
<oración> ::= ¬ ∃ x ( R(x,b) → <oración> <p. dch.>
<oración> ::= ¬ ∃ x ( R(x,b) → <o. simple> <p. dch.>
<oración> ::= ¬ ∃ x ( R(x,b) → S(x,b) <p. dch.>
<oración> ::= ¬ ∃ x ( R(x,b) → S(x,b) )
```

Figura 1. Derivación gramatical de una oración. Fuente: verbalogica.org/hazLogica

3.2. Evaluación semántica

El segundo campo donde se realizan demostraciones es el de la determinación del valor de verdad de oraciones en una situación dada. El punto de partida es el valor semántico de los signos descriptivos de cada lenguaje: oraciones elementales en LE; predicados y nombres en LPO. Este valor viene dado por la interpretación. Oraciones e interpretaciones pueden ser generadas automáticamente por el programa o propuestas por el estudiante.

Es interesante destacar que *hazLógica* admite dos tipos de interpretación de los predicados: extensional e intensional. La primera constituye la aproximación estándar en lógica. La segunda facilita el aprendizaje de la primera presentando una versión más cercana a la forma de entender el significado en lenguaje natural¹. Expresiones como *cuadrado*, *violeta*, *estar a la izquierda* prestan su intensión para fijar la extensión, su interpretación estándar, respecto de una situación dada. Este préstamo es provisional, en la medida en que los signos descriptivos mantienen la característica de los lenguajes lógicos de no tener restricciones intensionales, más allá de las categoriales.

¹ Carnap (1946) es una referencia clásica en la historia de estas ideas.

El proceso deductivo que, como hemos dicho, parte del valor semántico de los signos elementales del lenguaje, consiste en establecer -de forma recursiva y composicional- el valor semántico de las subfórmulas hasta llegar al valor de la oración en su conjunto.

La Figura 2 presenta un ejemplo de deducción del valor de verdad de la oración $\forall z \neg (Q(z) \wedge S(z))$ en la interpretación que aparece en la Figura 3:

1	$Q(a)$	vr	def.i	$[a] \in [Q/1]$
2	$S(a)$	vr	def.i	$[a] \in [S/1]$
3	$(Q(a) \wedge S(a))$	vr	$\vee \wedge$	(1,2)
4	$\neg(Q(a) \wedge S(a))$	fl	$f \neg$	(3)
5	$\forall z \neg(Q(z) \wedge S(z))$	fl	$f \forall$	(4)

Figura 2. Evaluación semántica de una oración. Fuente: verbalogica.org/hazLogica

En esta, como en general en el resto de las demostraciones, se usan fórmulas signadas (columnas 2 y 3), siendo la columna 4 una alusión a la regla de deducción que se usa y la última columna, una indicación de la/s premisa/s sobre la/s que se aplica la regla.

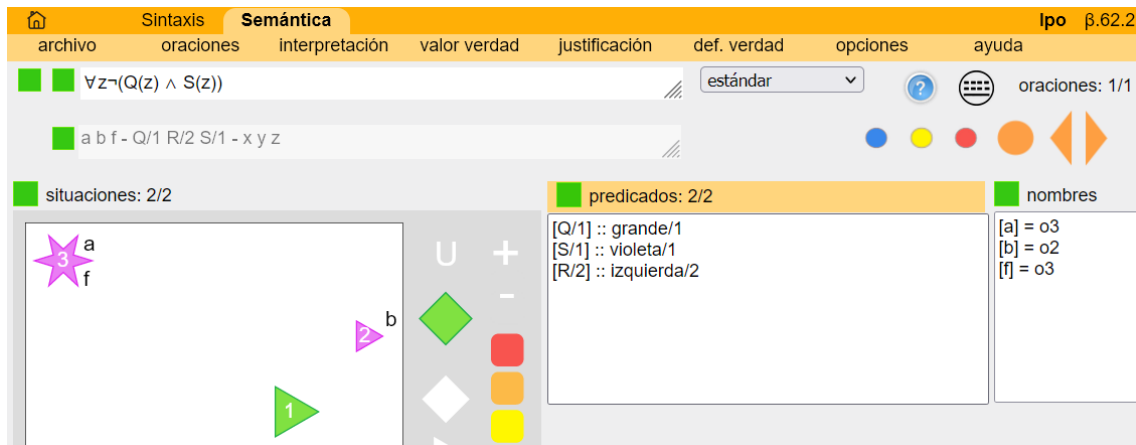


Figura 3. Interpretación de primer orden. Fuente: verbalogica.org/hazLogica

3.3. Semántica lógica

La justificación del tipo semántico de una oración (contingencia, verdad y falsedad lógicas) también exige demostraciones rigurosas. *hazLógica* sirve de instructor en este campo.

En el caso de las oraciones contingentes, el programa construye dos interpretaciones: una en la que la oración es verdadera; otra en la que es falsa. En ambos casos *hazLógica* justifica deductivamente el valor de verdad de la oración considerada.

Para las oraciones lógicamente verdaderas y falsas, *hazLógica* genera dos tipos de demostración: directa y por reducción al absurdo. En la primera se demuestra que la oración adopta en toda interpretación el mismo valor de verdad (verdadero / falso, respectivamente). En la segunda se justifica su tipo semántico deduciendo una contradicción del supuesto de la negación de lo que se quiere probar.

La construcción de contramodelos es también empleada en la demostración de la no-consecuencia de una oración respecto de un conjunto de premisas. Aquí la justificación consiste en evidenciar que en la interpretación propuesta las premisas son verdaderas y la

conclusión falsa. De forma similar, se prueba la inequivalencia de dos oraciones presentando una interpretación que proporciona distinto valor de verdad a cada una de las oraciones.

Los casos positivos de consecuencia y equivalencia se tratan de forma coherente: están también disponibles las dos formas antes aludidas: directa e indirecta.

La siguiente imagen presenta un ejemplo de demostración de no consecuencia:

Figura 4. Demostración en semántica lógica. Fuente: verbalogica.org/hazLogica

4. Juegos

Por el momento son cuatro los juegos públicamente accesibles: *hablamos LE*, *hablamos LPO*, *juegos sintácticos*, *juegos semánticos* (López Rincón, Sarabia et al. 2019). Además, en el marco de un proyecto complutense de innovación docente², están en desarrollo otros tres juegos más: uno sobre propiedades de las relaciones diádicas, otra versión de juego semántico, y un tercero sobre sorites y silogística.

Los dos primeros, *hablamos LE*, *hablamos LPO*, son juegos heurísticos en los que el estudiante ha de descubrir el significado de los signos (lógicos y descriptivos) a partir de ejemplos de su uso en oraciones que los contienen, y cuyo valor de verdad con respecto a una situación dada está determinado por el sistema. Hay dos modalidades de juego: *ingrid* y *user*. En la primera prevalece la interpretación generada por el programa; en la segunda se acepta la ideada por el usuario, siempre que ésta resulte consistente con las pistas. El proceso inferencial en este caso es abductivo.

² Nuevas herramientas de ludificación aplicadas al aprendizaje de Lógica (convocatoria 2020-21, UCM).

La siguiente figura ilustra un momento del juego en *hablamos LPO*:

Figura 5. *hablamos LPO*. Fuente: verbalogica.org/hazLogica

Los juegos sintácticos tienen un doble objetivo: identificar (i) la buena formación de expresiones en distintos lenguajes y notaciones, y (ii) el carácter de abierta/cerrada para las oraciones de LPO. En el primer caso, el sistema propone secuencias de signos sobre un alfabeto, y el estudiante debe decidir si la expresión es gramaticalmente correcta. En el segundo, la propuesta es de oraciones y la decisión ha de ser si está cerrada o no, esto es, si está en condiciones de ser interpretada en términos semánticos. En ambos casos, el programa justifica por qué la respuesta del estudiante es correcta o no lo es.

Los juegos semánticos implementan una variante de la semántica de juegos definida por Lorentzen y Lorenz (1978) y Hintikka y Sandu (2009), usados también por Barwise & Etchemendi (1993). En éstos el usuario juega contra el programa y gana si sabe evaluar adecuadamente el valor de verdad de una oración en la situación presentada.

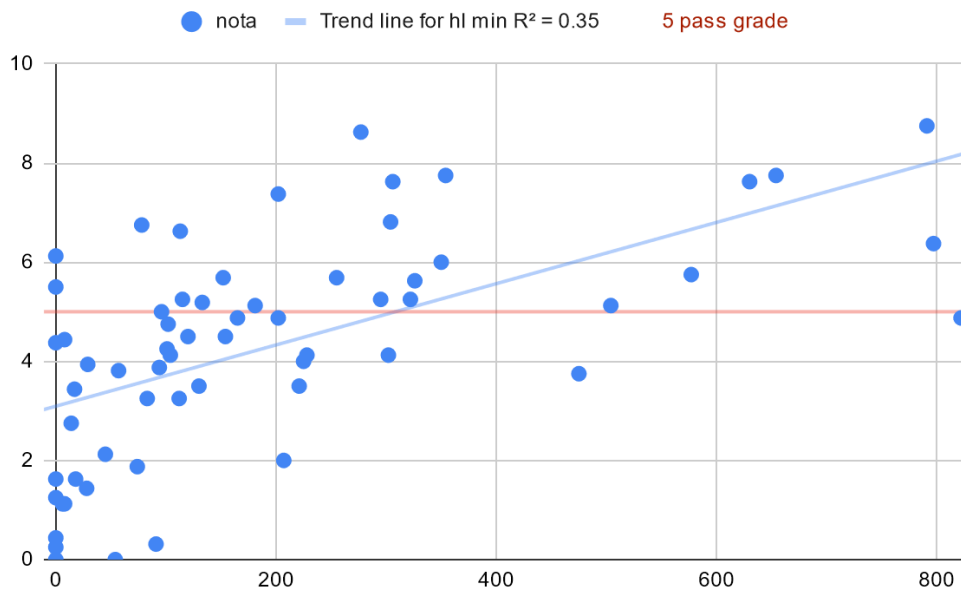
5. Resultados

Nuestra impresión subjetiva, influida fundamentalmente por el testimonio de los estudiantes, es que en general el programa les sirve de ayuda, particularmente para la adquisición de destrezas en el ámbito de la demostración rigurosa y la familiarización con el uso de lenguajes artificiales.

hazLógica ha sido usado durante los tres últimos cursos. Durante el primero, 2018-19, no recogimos datos de uso más allá de las impresiones informales y no sistemáticas aludidas antes. Tratando de corroborar si estas impresiones eran correctas, durante el segundo curso, 2019-20, ya recopilamos número de visitas a la página. Ha sido durante este curso, 2020-21, cuando hemos podido recoger de forma precisa datos sobre la utilización real del programa usando el registro de minutos de uso.

El siguiente gráfico recoge la correlación entre la calificación obtenida (en las preguntas del examen que no son tipo test) y los minutos de trabajo en *hazLógica*.

Gráfico 1. Uso de *hazLógica* – calificaciones



Fuente: elaboración propia

Es notoria la tendencia ascendente: la mayor inversión de tiempo en el programa redonda en términos generales en una calificación mejor.

Es también interesante considerar los siguientes datos:

Tabla 1. Resultados sobre aprobados

R ²	0.3504710268
aprobados que usaron hL	91.67%
aprobados que NO usaron hL	8.33%

Fuente: elaboración propia

Entendemos que éstos corroboran las impresiones que tuvimos desde el primer curso en que usamos *hazLógica* y nos animan a continuar su desarrollo.

Referencias bibliográficas

Barwise, Jon y Etchemendy, John. 1993. *Tarski's world*. Stanford. California: CSLI Publications.

Carnap, Rudolf. 1946. *Meaning and Necessity*. Chicago: The University of Chicago Press.

Hintikka, Jaakko y Sandu, Gabriel. 2009. "Game-Theoretical Semantics". En: Allan, K. (Ed.). *Concise Encyclopedia of Semantics*, pp. 341–343. Elsevier

López Rincón, Carmen y Sarabia Álvarez-Ude, Jaime y otros. 2019. "Desarrollo, integración y evaluación de técnicas de ludificación (*gamification*) en un entorno virtual de aprendizaje de lógica" [Proyecto de innovación docente]. Universidad Complutense de Madrid. <https://eprints.ucm.es/id/eprint/56072/> (Consultado el 15 de junio de 2021)

Lorenzen, Paul y Lorenz, Kuno. 1978: *Dialogische Logik*. Darmstadt: Wissenschaftliche Buchgesellschaft.