

Análisis de Consistencia de la Legislación de Defensa del Consumidor mediante métodos formales

Sebastián Sznur¹, María Belén Ruffa¹, Máximo Martínez¹, Juan Martín Brun²,
Fernando Schapachnik³

¹ Facultad de Ingeniería, Universidad FASTA, Argentina
{sz.sebas, belenruffa, maximomrtz}
}@gmail.com

² Facultad de Ciencias Jurídicas y Sociales, Universidad FASTA, Argentina
juanbrun.jmb@gmail.com

³ Departamento de Computación, FCEyN, Universidad de Buenos Aires, Argentina
fschapachnik@dc.uba.ar

Resumen. El problema de las lagunas legales o inconsistencias en la legislación es añejo y continúa existiendo. El grupo de investigación FormaLex del Departamento de Computación de la Facultad de Ciencias Exactas y Naturales de la UBA viene trabajando hace varios años en adaptar métodos de detección de inconsistencia basados en lógica al ámbito legal. Este trabajo relata la experiencia de la utilización de esos métodos formales para analizar un texto legal real, la Ley de Defensa al Consumidor de la República Argentina. Durante el proyecto se realizaron varios hallazgos interesantes. Por un lado, se detectaron limitaciones en la versión actual del lenguaje FL que permitieron delinear mejoras y correcciones. Por otro lado, se detectó que durante la codificación se generaba conocimiento que valía la pena profundizar y sistematizar: la redacción de ciertas partes de la norma analizada dificultaba su comprensión con vistas a la codificación y el posterior análisis. Ese conocimiento amerita ser sistematizado y a futuro se buscará elaborar un conjunto de buenas prácticas que ayude a escribir normas más propensas para el análisis.

1 Introducción

El problema de las lagunas legales o inconsistencias en la legislación es añejo, y continúa existiendo. La solución tradicional al mismo consiste en dejar en manos de los jueces, cotidianos usuarios de la legislación vigente, la interpretación de las normas para subsanar estos problemas. Sin embargo, abundan casos en donde, a pesar de la jurisprudencia y otros elementos con los que cuenta el Poder Judicial para lograr ecuanimidad, situaciones muy similares reciben tratamientos distintos, ambos igualmente razonables a priori.

Por otra parte, varios autores defienden cierto grado de ambigüedad en las normas, justamente para dejarle a los jueces margen de acción. Esta idea es debatible desde el aspecto teórico y filosófico, dado que la división en poderes de los sistemas

democráticos occidentales presupone como principio fundacional la periodicidad de los cargos y la elección lo más directa posible de los representantes. En la mayor parte de los casos los jueces tienen puestos prácticamente permanentes y son elegidos de manera indirecta. Entonces, ¿hasta qué punto es razonable que sean ellos los encargados de legislar? Aún si no se comparte esta visión, creemos que el grado de ambigüedad de una norma debería ser decisión de los legisladores, y no mero accidente. Así, una herramienta que ayude a detectar inconsistencias y omisiones sería de gran valor, aportando a la calidad democrática.

El grupo de investigación FormaLex del Departamento de Computación de la Facultad de Ciencias Exactas y Naturales de la UBA viene trabajando hace varios años en adaptar métodos de detección de inconsistencia basados en lógica, provenientes de la Ingeniería del Software, al ámbito legal [GMS2010, GMS2011, GMS2012]. Su trabajo comienza por la determinación de las similitudes entre los textos normativos y las especificaciones de software, esos documentos que describen el comportamiento que una pieza de software debería seguir: ambos son documentos deónticos, susceptibles de inconsistencias, y la Ingeniería del Software lleva varias décadas desarrollando técnicas de creciente efectividad para tratar con ese tipo de problemas.

Este trabajo relata la experiencia de la utilización de métodos formales para analizar un texto legal real, la Ley de Defensa al Consumidor de la República Argentina.

El objetivo de la experiencia consistió en la utilización del lenguaje formal para intentar codificar por completo la mencionada Ley, como una forma de explorar las capacidades y limitaciones del lenguaje, de manera tal de que éste pudiera expandirse y mejorarse para solucionarlas.

A continuación se presenta el lenguaje FL, para posteriormente detallar la metodología utilizada durante el desarrollo de este trabajo, explicando cada paso del proceso de traducción mediante un ejemplo concreto de la Ley. Finalmente se plantearán los resultados obtenidos y las conclusiones a las que se han llegado.

2 Estado del arte

El lenguaje está dividido en dos partes: una *background theory*, o *teoría marco*, y un conjunto de reglas. Mientras que las reglas son fórmulas LTL con operadores deónticos para expresar permisos, prohibiciones y obligaciones, la teoría marco provee mecanismos sencillos para describir la clase de modelos sobre los que predicen las reglas. Allí se expresan cosas como precedencia de eventos (e.g., el día ocurre antes de la noche), unicidad (e.g., las personas nacen una sola vez), etc.

Si algo es obligatorio, entonces debe valer en todo modelo legalmente válido y por ende $O(\phi)$ se interpreta como la fórmula $\Box \phi$, es decir, la fórmula que dice que en todo estado del sistema ϕ es válida. Si bien el lenguaje se basa en una traducción a LTL la sintaxis original de las reglas se preserva, y eso permite realizar algunos análisis a un nivel meta-lógico. Los detalles de dicho análisis se presentan en [GMS2011].

En FL la prohibición de algo es la obligación de su contrario ($F(\phi) \equiv O(\neg\phi)$). Para el lector avanzado en el tema, vale la pena remarcar que nuestra forma de codificar el lenguaje evita la mayoría de las llamadas "paradojas deónticas" (ver [GMS2011] para más detalles). El permiso se piensa como ausencia de prohibición, pero se trata no como un operador que modifica el conjunto de comportamientos legalmente válidos, sino como un predicado que el resto de los modelos legalmente válidos deben cumplir.

El principal componente de la teoría marco es la *acción*. Una acción puede estar sucediendo o no en cada instante de tiempo. En FL las acciones se interpretan como si fuesen *señales digitales*, que pueden estar prendidas o apagadas por una cantidad arbitraria de estados consecutivos. Las acciones pueden representar acciones del agente implícito propiamente dichas (por ejemplo, **action ConducirAuto**) o eventos externos, no controlables (e.g., **action Choque**).

Algunos requerimientos parecen a veces fáciles de formalizar, como por ejemplo el de tener registro para poder conducir un auto. Parecería que alcanza con prohibir la acción **ConducirAuto** si no hay una acción previa **ObtenerRegistro**. Esta facilidad es sólo aparente, ya que los individuos no sólo consiguen autorizaciones para conducir vehículos, también pueden perderlas, de manera que si también consideramos como posible a la acción **RevocarRegistro**, escribir una fórmula que establezca si una persona puede o no manejar tiene la misma complejidad que el problema de contar paréntesis para ver si están o no balanceados. Esto puede ser muy difícil o directamente imposible dependiendo de la lógica utilizada. En FL existe la noción de *intervalo*, similar a los *fluentes* de [GM2003]. Un intervalo está delimitado por sus acciones de comienzo y fin, de manera tal que no hay anidamiento ni se puede cerrar un intervalo ya cerrado. Durante la vigencia de un intervalo se hace verdadera una variable proposicional, de manera tal que el caso del ejemplo se puede escribir como el intervalo

interval autorizado_a_conducir delimited by actions ObtenerRegistro-RevocarRegistro

seguido de la fórmula $F(\neg is_autorizado_a_conducir \wedge ConducirAuto)$, donde el prefijo "is_" seguido del nombre del intervalo es una variable proposicional que se hace verdadera dentro del mismo.

Los intervalos también se pueden utilizar para acotar la ocurrencia de otras acciones:

interval periodo_escolar delimited by actions ComienzoDeCursada-FinDeCursada

action `RendirExamen` occurs only in scope `periodo_escolar`

Si se desea considerar cotas temporales (e.g., "los libros de la biblioteca deben devolverse dentro del periodo escolar"), los intervalos pueden usarse dentro de la obligación estándar: $O(\diamond_{\text{periodo_escolar}}(\text{DevolverLibro}))$. En LTL, la lógica en la que se basa FL, el diamante (\diamond) se interpreta como "en algún momento en el futuro". Los diamantes de FL, que incluyen el intervalo, como por ejemplo $\diamond_{\text{periodo_escolar}}$ se interpretan como "en algún momento dentro del periodo escolar".

FL también provee *contadores*, permitiendo utilizar expresiones como $O(\text{clp} > 0 \rightarrow \diamond(\text{clp} = 0))$, donde la variable entera **clp**, *cantidad de libros prestados*, es incrementada ante cada **RetirarLibro** y decrementada con cada **DevolverLibro**. La fórmula se lee "es obligatorio que cada vez que **clp** se haga mayor que cero, en algún momento posterior vuelva a ser cero", y es interesante porque es la manera correcta de expresar que cada libro prestado debe ser devuelto, ya que la fórmula intuitiva -e incorrecta- $O(\text{RetirarLibro} \rightarrow \diamond \text{DevolverLibro})$ es satisfecha al retirar varios libros y devolver sólo uno, que claramente no es lo deseado.

3 Metodología

Durante el desarrollo del presente proyecto, se trabajó en forma conjunta e interdisciplinaria en la traducción al lenguaje FL de cada uno de los artículos de la ley de Defensa del Consumidor.

FL hasta entonces se había aplicado a casos más pequeños, como reglamentos, o disposiciones de menor cantidad de artículos, razón por la que uno de los objetivos principales de este proyecto consistió, específicamente, en testear sus características y aptitudes para el uso en una ley específica, amplia y compleja. Por tal motivo es que se decidió traducir capítulos de la Ley de Defensa al Consumidor al lenguaje FL, para probarlo en un caso real y concreto.

Siendo que el lenguaje FL tiene como objetivo reflejar el mundo real mediante la abstracción, se estableció que el proceso de traducción de la ley sería mediante la división y análisis de cada uno de los capítulos que la componen. A su vez, se tomó cada uno de los artículos y se los desglosó de forma tal que se pudiera definir, para cada uno de ellos, una o varias fórmulas lógicas mediante los elementos que componen el lenguaje, es decir roles, acciones, que junto con operadores lógicos forman reglas que abstraen y formalizan cada artículo de la ley.

3.1 Ejemplo de formalización

Para ejemplificar el trabajo que se realizó durante el desarrollo del presente proyecto, se propone el siguiente artículo de la Ley de Defensa al Consumidor y el análisis realizado para el mismo.

Capítulo II. Artículo 5

Protección al consumidor. Las cosas y servicios deben ser suministrados o prestados en forma tal que, utilizados en condiciones previsibles o normales de uso, no presenten peligro alguno para la salud o integridad física de los consumidores o usuarios.

Se define el **Marco**, compuesto por **Roles** y **Acciones**. Un rol está conformado por un actor que participa de alguna “actividad” definida en el artículo y una acción es justamente esa “actividad” la cual el actor desempeña. Para este ejemplo, el Marco se define de la siguiente manera:

Marco
role Consumidor
role Proveedor
action ProveerByS performable by {Proveedor} output values {Correctamente, Incorrectamente}
action UtilizarByS performable by {Consumidor} output values {Gratuitamente, Onerosamente, Correctamente, Incorrectamente}
action RiesgoalConsumidor

Una vez definidos los roles y acciones, se define la regla que formaliza el artículo en lenguaje FL:

$O(\text{ProveerByS.Correctamente} \wedge \text{UtilizarByS.Correctamente} \rightarrow \neg \text{RiesgoalConsumidor})$

Esta regla se podría leer de la siguiente manera: Es obligatorio (“O” operador “obligación” de la lógica deóntica) que si se provee y utiliza el bien y servicio de manera correcta, es decir en condiciones previsibles y normales de uso, entonces no debe existir un riesgo para el consumidor.

El proceso de traducción antes ejemplificado, se realizó con cada uno de los artículos que componen los capítulos del I al XII de la ley. Al avanzar con dicho análisis, se fueron presentando situaciones que resultaban difíciles de ser formalizadas con el lenguaje FL. A partir de las limitaciones encontradas en el lenguaje, al momento de intentar modelar sus artículos, desde el Departamento de Computación de la UBA comenzaron a trabajar en la ampliación del lenguaje para considerar las problemáticas detectadas.

Cada uno de estos agregados implica un replanteo de la solución, así como un análisis integral de las capacidades del lenguaje y cambios en la sintaxis para modelar estas nuevas capacidades.

4 Resultados

Se presenta a continuación las limitaciones encontradas como resultado de este trabajo:

4.1 Necesidad de subroles y herencia

El art. 2 define al proveedor, indica su naturaleza y describe sus posibles actividades en un detalle bastante pormenorizado. También se indica quiénes no tienen dicho rol excluyéndolos especialmente ya que indica que se rigen por sus instituciones propias, aunque si integra la publicidad que estos pudieran realizar.

La necesidad identificada es la de poder tener en el lenguaje la posibilidad de decir que cierto rol se particiona en otros (en el caso identificado se trataría de tener varios tipos de "proveedor"), a la vez que potenciar el modelo de análisis en la presencia de roles. Puntualmente, la herramienta no consideraba todas las combinaciones posibles de roles, capacidad que se incorporó materializándola en el concepto de agente, que cumple con uno o más roles.

4.2 Necesidad de auto-referencia

El art 2 indica que “Todo proveedor está obligado al cumplimiento de esta ley”.

La autorreferencia es uno de los "talones de Aquiles" de los lenguajes formales, comenzando por los resultados famosos de Gödel y Turing. Sin embargo, en este caso se trata de incorporar la posibilidad de una autorreferencia acotada. Si bien se tenía conciencia de esta limitación, se la consideraba menos prioritaria, y se vio que algún tipo de solución es necesario proveer para poder modelar normativas reales. Para resolver esto se investigará el uso de "unfoldings parciales", una técnica ya conocida en la Teoría de Modelos.

4.3 Necesidad de referencia a otro artículo de la misma ley

Es el caso de los art. 38 y 39, Capítulo 9 donde se invoca elementos que cumplan con una característica mencionada en otro art: Por ejemplo: “Cuando los contratos a los que se refiere el artículo anterior requieran la aprobación de otra autoridad nacional o provincial, ...”

Un caso similar se da en el artículo 7°, donde tuvimos que determinar qué "CondicionesOferta" incluía: fecha de comienzo y finalización de la misma, modalidades, condiciones o limitaciones. Si bien se pudo modelar que era necesario difundir la revocación, no se encontró la forma de establecer que debía ser por los mismos "medios" que la revocación. El artículo remite a las sanciones del artículo 47, situación que se resumió como "sancionar", pero quizás habría que especificar las sanciones que explicita el artículo.

Por último, se puede mencionar como ejemplo el art. 32.– ... *El contrato debe ser instrumentado por escrito y con las precisiones establecidas en los arts. 10 y 34 de la presente ley.*

4.4 Necesidad de establecer “atributos”

Es el caso del Artículo 8°, donde se tuvo que subsumir algunas cuestiones como "compras telefónicas, catálogos o correos" en "ofertar por medios", expresión que

incluyera todas las anteriores. Lo mismo con "Datos Oferentes", que incluye el nombre, domicilio y el CUIT del oferente como obligatorios.

Estos artículos, principalmente enumerativos, se agruparon en una única expresión que engloba estos puntos. El problema puede llegar a presentarse mayormente cuando quizás al omitir esta información se pierda alguna idea o cuestión que debería tenerse en cuenta.

Por ejemplo, en el caso de la garantía de un contrato de prestación de servicios, deberá constar cierta información que se enumera en el artículo pero no se detalló en la expresión, sino que se definió como *EntregarGarantiaPorEscrito.Completa*

Art. 24.– Garantía. La garantía sobre un contrato de prestación de servicios deberá

documentarse por escrito haciendo constar:

- a) La correcta individualización del trabajo realizado;*
- b) El tiempo de vigencia de la garantía, la fecha de iniciación de dicho período y las condiciones de validez de la misma;*
- c) La correcta individualización de la persona, empresa o entidad que la hará efectiva.*

4.5 Necesidad de referenciar art. de otras leyes

Esta necesidad surge en virtud de invocar desde la misma ley art. del Código Civil o Código Penal. Incluso si estos estuvieran modelados sería interesante poder invocarlos al estilo de un "include" de los lenguajes de programación. Ej: Art 8 bis entre otros.

4.6. Necesidad de interactuar con otras leyes completas

Poder referirse a un conjunto de reglas como un todo.

Ej: *A instancia del consumidor se aplicará de pleno derecho el art. 2176 del Código Civil; (art. 18)*

4.7 Necesidad de mejorar los contadores numéricos y los intervalos

Sería deseable que los contadores numéricos se pudieran pausar o interrumpir por un evento ajeno. Ej. Artículo 11 indica *"El tiempo durante el cual el consumidor está privado del uso de la cosa en garantía, por cualquier causa relacionada con su reparación, debe computarse como prolongación del plazo de garantía legal.*

Es necesario contar con cierta flexibilidad al momento de tener que definir intervalos con condiciones especiales de inicio y de finalización, donde, entre otras cosas, puedan definirse uno o más casos de inicio y uno o más casos de fin. Esto permitiría modelar, por ej. el caso siguiente:-

Art. 34.– (Texto según ley 26361, art. 14) Revocación de aceptación. En los casos previstos en los arts. 32 y 33 de la presente ley, el consumidor tiene derecho a revocar la aceptación durante el plazo de diez (10) días corridos contados a partir de la fecha en que se entregue el bien o se celebre el contrato, lo último que ocurra, sin responsabilidad alguna. ...

Producto de estos hallazgos se identificó la necesidad de enriquecer el lenguaje FL dotándolo de contadores que no sólo aumenten sus valores sino también que los disminuyan, que lo hagan a raíz de múltiples acciones, que comiencen en valores arbitrarios y que tengan acciones de reset que también los lleven a valores arbitrarios.

A la vez, se detectó que la definición actual de los intervalos era demasiado restrictiva: es necesario dotarlos de la capacidad de comenzar y terminar no por una única acción, sino por un conjunto de ellas. En realidad, algunas condiciones de inicio o de fin que necesitan ser especificadas son en realidad predicados lógicos: condiciones, eventualmente complejas, que pueden incluir la ocurrencia de múltiples acciones, combinadas de forma arbitraria con valores de contadores y encajes en otros intervalos. Aparentemente estos predicados podrían ser proposicionales, sin necesidad de llegar a ser modales.

4.8 Necesidad de poder excluir reglas en una situación dada

Poder armar reglas de la forma: “Si ocurre tal cosa no tener en cuenta tal conjunto de reglas”. Actualmente tal proceso es engorroso, una línea como la siguiente llevaría a reescribir todo un artículo (y encima de otra ley):

El art. 2170 del Código Civil no podrá ser opuesto al consumidor. (art. 18).

Art 25 “Los servicios públicos domiciliarios con legislación específica y cuya actuación sea controlada por los organismos que ella contempla serán regidos por esas normas y por la presente ley. En caso de duda sobre la normativa aplicable, resultará la más favorable para el consumidor”.

Se trata nuevamente de un caso de autorreferencia desde el punto de vista lógica: la norma habla de sí misma, aunque en este caso las implicancias son distintas. Este tipo de artículos evidencian la necesidad de un estudio para determinar qué tipo de inconsistencias podrían producirse (al menos de forma más común) a partir de este tipo de cláusulas, como para encontrar el balance entre conveniencia y complejidad lógica de las posibles soluciones.

4.9 Necesidad de incluir ciertas operaciones aritméticas

En el Capítulo 8 se prevé el cálculo de tasas de interés anual, gastos extras, seguros adicionales, entre otros. Art 31. “Cuando una empresa de servicio público domiciliario, con variaciones regulares estacionales, facture en un período consumos que exceden en un setenta y cinco por ciento (75%) el promedio de los consumos correspondientes al mismo período de los dos años anteriores se presume que existe error en la facturación.”

Si bien es sabido desde el punto de vista lógico que la inclusión de aritmética entera lleva rápidamente hacia la indecidibilidad, debería estudiarse la posibilidad de brindar algún tipo de alternativa simbólica que ayude en estos casos. Por ejemplo, la herramienta podría alertar que la misma variable está siendo influida por dos fórmulas, aunque no pueda determinar si son distintas o no.

4.10 Necesidad de agentes que interpreten roles

En conexión con lo dicho anteriormente, algunas inconsistencias podrían detectarse solamente mediante la acción conjunta de más de un agente en cada rol. Por ejemplo, con la existencia de más de un comprador o más de un vendedor. Surge entonces la necesidad de parametrizar los análisis para ciertas cantidades de agentes en cada rol. Este punto debe manejarse con cuidado ya que rápidamente podría llevar a una explosión en el número de estados.

4.11 Necesidad de que el rol no sea sólo sujeto sino también objeto directo.

Ej: art 45 ...” El recurso deberá interponerse ante la misma autoridad que dictó la resolución,”

Algunas acciones tienen, además de un sujeto que las ejecuta, otro que es destinatario o participe de las mismas, y eso hoy en día no está contemplado. Este punto se conecta con la sincronización entre acciones: si Juan le vende a Pedro el producto X, Pedro a su vez está realizando la acción recíproca, que es la de comprarle a Juan. Este tipo de situaciones deberían poder especificarse con mayor claridad y facilidad en FL.

4.12 Necesidad de incluir cálculos con fechas

La ley establece en ciertas ocasiones la necesidad de que ciertas situaciones solo pueden ocurrir, dentro de un plazo a partir de una fecha determinada, ej: Fecha de Vencimiento. *Art. 23.– Deficiencias en la prestación del servicio. Salvo previsión expresa y por escrito en contrario, si dentro de los treinta (30) días siguientes a la fecha en que concluyó el servicio se evidenciar en deficiencias o defectos en el trabajo realizado....* Art. 29 “Las facturas deberán ser entregadas al usuario con no menos de diez(10) días de anticipación a la fecha de su vencimiento.”

Si bien hoy en día FL brinda soporte para acciones temporales que de alguna manera permiten hablar de fechas, la interacción entre éstas y los contadores es muy limitada y este tipo de especificaciones se hace tediosa. Por lo tanto, es necesario proveer algún tipo de infraestructura que vaya indicando el paso del tiempo en unidades fijas, permita almacenar su valor en alguna variable cuando una acción suceda, y usar esa misma variable tanto como condición de otras como también como parte de un predicado deóntico, tal como sucede en el artículo citado.

4.13 Necesidad de expresar fórmulas que utilicen cuantificadores

Algunas expresiones encontradas requieren poder hablar de todos los agentes que interpretan determinado rol o de algunos de ellos, pudiendo determinar con claridad

cuando se trata de un caso y cuándo del otro. Para eso se modificaron las fórmulas deónticas de FL para incorporar cuantificadores como se muestra en los ejemplos:

Si un conductor cruza un semáforo en rojo, algún policía debe multarlo.

$O(\exists i: \text{conductor}, i. \text{cruzaEnRojo} \Rightarrow \diamond \exists j: \text{policia}, j. \text{haceMulta})$

Si un consumidor solicita un servicio y todos los proveedores lo niegan, la autoridad debe multarlos (obligaciones colectivas).

$O(\exists i: \text{consumidor}, i. \text{solicitaServicio} \wedge$
 $\forall j: \text{proveedor}, j. \text{niegaServicio} \Rightarrow$
 $\exists h: \text{autoridad}, \forall j: \text{proveedor} h. \text{aplicaMulta}(j))$

5 Conclusiones

Se tradujo la ley de Defensa del consumidor al lenguaje deóntico FL, con el objetivo de probar por primera vez la versatilidad del conjunto de esas herramientas en un caso amplio y real.

Durante el proyecto se realizaron varios hallazgos interesantes. Por un lado, se detectaron limitaciones en la versión actual del lenguaje FL que permitieron delinear mejoras y correcciones, algunas de las cuales ya se llevaron a cabo, otras están en curso y otras esperan un análisis más profundo desde el punto de vista de su viabilidad técnica.

Por otro lado, se detectó que durante la codificación se generaba conocimiento que valía la pena profundizar y sistematizar: la redacción de ciertas partes de la norma analizada dificultaba su comprensión con vistas a la codificación y el posterior análisis. Una vez superada la dificultad muchas veces surgían propuestas alternativas de redacción que hubieran facilitado la tarea. También sucedió que ciertas partes del texto eran plausibles a ser codificadas de más una manera, siendo algunas de ellas más ventajosas que otras.

Todo ese conocimiento amerita ser sistematizado, como muestra también la experiencia del mundo de la Ingeniería del Software. A futuro se buscará elaborar un conjunto de buenas prácticas que ayude a escribir normas más propensas para el análisis. En simultáneo, se buscará generar recomendaciones para estos futuros expertos en análisis de normas, de manera tal que poder ir generando conocimiento que simplifique la traducción de los textos normativos en lenguaje natural, a un lenguaje formal como FL.

Por último, se buscará capacitar recursos humanos para que conozcan la herramienta FL de manera tal que puedan mejorarla, ampliarla y corregirla, de manera de generar un círculo virtuoso que progresivamente vaya reduciendo la brecha de esfuerzo necesario para analizar un texto legal.

Agradecimientos. Los autores de este artículo desean agradecer al resto del equipo de trabajo, conformado por la Ing. Ana Haydée Di Iorio, la Dra. Bibiana Luz Clara, la Dra. Fernanda Giaccaglia, la Ing. Verónica Uriarte y Cristian Marcos, como así

también al Ing. Roberto Giordano Lerena, Decano de la Facultad de Ingeniería de la Universidad Fasta, y a la Lic. Mónica Pascual, Secretaria de Investigación y Posgrado.

Bibliografía

[GM2003] Giannakopoulou, D., Magee, J.: Fluent model checking for event-based systems. ACM SIGSOFT Software Engineering Notes 28 (2003) 266

[GMS2010] Gorín, D., Mera, S., Schapachnik, F.: Model Checking Legal Documents. In: Proceedings of the 2010 conference on Legal Knowledge and Information Systems: JURIX 2010. (2010) 111--115

[GMS2011] Gorín, D., Mera, S., Schapachnik, F.: A Software Tool for Legal Drafting. In: FLACOS 2011: Fifth Workshop on Formal Languages and Analysis of Contract-Oriented Software, Elsevier (2011) 1--15

[GMS2012] Giménez, M., Mera, S., Schapachnik, F.: Detección automática de defectos normativos, una invitación a la colaboración. In: CIDDI 2012, Congreso Iberoamericano de Docentes e Investigadores en Derecho e Informática. (2012)