193002

SISTEMA EXPERTO BASADO EN ONTOLOGÍA PARA LA DETECCIÓN DE FRAUDE EN TARJETAS DE CRÉDITO

Juan Gabriel Ramírez Sosa

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
MAESTRÍA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ, D.C.
2019

193002

SISTEMA EXPERTO BASADO EN ONTOLOGÍA PARA LA DETECCIÓN DE FRAUDE EN TARJETAS DE CRÉDITO

Autor:

Juan Gabriel Ramírez Sosa

MEMORIA DEL TRABAJO DE GRADO REALIZADO PARA CUMPLIR UNO DE LOS REQUISITOS PARA OPTAR AL TITULO DE

MAGÍSTER EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

Director

Luis Manuel Vilches Blázquez

Comité de Evaluación del Trabajo de Grado

Regina Motz

Víctor Saquicela

Página web del Trabajo de Grado

https://livejaverianaedu.sharepoint.com/sites/Ingsis/TGMISC/193002

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
MAESTRÍA EN INGENIERIA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ, D.C.
Noviembre,2019

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
MAESTRÍA EN INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

Rector Magnífico

Jorge Humberto Peláez, S.J.

Decano Facultad de Ingeniería

Ingeniero Lope Hugo Barrero Solano

Director Maestría en Ingeniería de Sistemas y Computación

Ingeniera Ángela Carrillo Ramos

Director Departamento de Ingeniería de Sistemas

Ingeniero Efraín Ortiz Pabón

Artículo 23 de la Resolución No. 1 de Junio de 1946

"La Universidad no se hace responsable de los conceptos emitidos por sus alumnos en sus proyectos de grado. Sólo velará porque no se publique nada contrario al dogma y la moral católica y porque no contengan ataques o polémicas puramente personales. Antes bien, que se vean en ellos el anhelo de buscar la verdad y la Justicia"

AGRADECIMIENTOS

Agradezco primero que todo a Dios por permitirme iniciar y llevar a cabo esta enriquecedora experiencia estudiantil y personal.

A mi hija Hannah Stefanny porque siempre llena mi vida de felicidad y a quien dedico este proyecto, al igual que a mi adorada esposa Adriana que con su amor, comprensión y apoyo en todo momento me ha ayudado a culminar esta importante etapa de mi vida.

A mis padres y hermanas por el amor y el apoyo desde la distancia, que ha sido muy fortificante en esta etapa de mi vida.

A mi amigo Diego Parada por estar de manera incondicional y ser un apoyo en esta experiencia.

Al profesor Luis Manuel Vilches por su asesoría académica, por su apoyo moral y por ser el excelente ser humano que es.

CONTENIDO

LISTADO DE FIGURAS

LISTADO DE TABLAS

ABSTRACT

The increase in bank transactions using credit cards, through the Internet and mobile devices, is causing an increase in fraud on these products. To counter these frauds, financial institutions perform analysis and transactional fraud management tasks, supported by engines, rules for fraud detection that demand a large amount of infrastructure resources and non-approved information. This project develops an expert system based on ontologies and rules for the detection of credit card fraud, composed of a network of ontologies and SWRL rules, where information from transactional sources of credit card is also linked through a semantic integration perspective of information. The development of the project was guided by the methodology of science based on the design and supported by the NeOn and Linked Data methodologies.

RESUMEN

El incremento de las transacciones bancarias haciendo uso de tarjetas de crédito, a través de Internet y dispositivos móviles, está originando un incremento del fraude sobre estos productos. Para hacer contraparte a estos fraudes, las instituciones financieras realizan tareas de análisis y gestión de fraude transaccional, apoyados en motores, reglas para la detección de fraude que demandan gran cantidad de recursos de infraestructura e información no homologada. Este proyecto desarrolla un sistema experto basado en ontologías y reglas para la detección de fraude en tarjetas de crédito, compuesto por una red de ontologías y reglas SWRL, donde además se vincula información de fuentes transaccionales de tarjeta de crédito mediante una perspectiva de integración semántica de la información. El desarrollo del proyecto fue guiado por la metodología de la ciencia basada en el diseño y apoyada sobre las metodologías NeOn y Linked Data.

RESUMEN EJECUTIVO

En este trabajo se ha desarrollado un sistema experto basado en ontologías y reglas para la detección de fraude en tarjetas de crédito. Este sistema está compuesto por una red de ontologías, consultas SPARQL y reglas *Semantic Web Rule Language* (SWRL). Asimismo, esta propuesta recoge un proceso de integración semántica basada en los principios de Linked Data, donde se conecta información de fuentes transaccionales de tarjeta de crédito y de información de apoyo para la detección de fraude.

El desarrollo de este trabajo abordó el estudio y aplicación de las metodologías NeOn y Linked Data en cada una de sus fases y escenarios. Así, para lograr la creación de una red de ontologías se trataron diversos escenarios de la metodología NeOn que permitieron modelar el conocimiento relacionado con transacciones de tarjeta de crédito aplicada a la detección de fraude. Por otro lado, haciendo uso del lenguaje de reglas SWRL fue posible realizar la creación de reglas de alertamiento de fraude de tarjeta de crédito genéricas, las cuales fueron construidas con base en reglas de detección de fraude estándar del mercado.

En este trabajo se utilizaron diversas tecnologías semánticas, teniendo un rol fundamental el *framework* de Apache Jena, el cual se utilizó para la integración de compontes de adaptación, inferencia, almacenamiento y consulta, permitiendo materializar el sistema experto propuesto.

Con el objetivo de demostrar el funcionamiento del sistema experto desarrollado, se llevó a cabo una validación, donde se comparan los resultados obtenidos por el mencionado sistema con un *gold standard* proporcionado por una entidad bancaria colombiana. La validación se llevó a cabo sobre tres muestras de datos que presentan diferentes características y donde se utilizaron las métricas *precision*, *recall* y *F-measure* para el análisis de los resultados. Los resultados más prometedores del sistema experto se propician, principalmente, a nivel de montos de las transacciones y de velocidad de transacciones.

Con el desarrollo de este proyecto se logra poner a disposición de la comunidad educativa, una red de ontologías que modela la información transaccional de tarjeta de crédito, base para futuras integraciones con otras redes ontológicas y fuentes de información. Adicionalmente, se aporta la construcción de reglas semánticas de detección de fraude para transacciones de tarjetas de crédito con base en los lenguajes SPARQL y SWRL, de modo que puedan ser utilizadas en proyectos futuros que aborden estos temas de investigación.

INTRODUCCIÓN

El incremento de las necesidades de los ciudadanos para realizar transacciones bancarias, a través de Internet o de dispositivos móviles, ha llevado a que las organizaciones financieras ofrezcan, cada día más, productos que se puedan utilizar en estos canales. En este escenario, las tarjetas (crédito y débito) tienen un rol preponderante en torno a las cuales se articula el funcionamiento de muchos de estos productos financieros, al punto que la tarjeta de crédito se ha convertido en el segundo producto con mayor penetración entre los adultos colombianos, seguido por el crédito de consumo. Así, en marzo de 2019, en Colombia aparecían 9 millones de personas que tenían al menos una tarjeta de crédito vigente y 6.9 millones con algún tipo de crédito de consumo vigente. La totalidad de estas tarjetas de crédito y la mayoría de los créditos de consumo fueron ofrecidas por establecimientos de crédito [1]. Lo anterior ha dado lugar al incremento en los delitos vinculados a temas de fraude sobre los productos a disposición de los consumidores financieros colombianos, en gran medida, centrados en las tarjetas de crédito.

En la actualidad, el fraude [2] está presente en distintos frentes tratando de tomar ventaja de las debilidades que se pueden encontrar en los canales de interacción que las instituciones financieras ofrecen a sus clientes. En un gran porcentaje, el fraude es materializado a través de la utilización de tarjetas de crédito o débito tanto en operaciones presentes (realizadas por el cliente presencialmente en el comercio) como no presentes (compras por Internet, telefónicas, celular, entre otras).

Para paliar estas situaciones delictivas, las instituciones financieras incluyen en su operación tareas de análisis y gestión de fraude, las cuales se han realizado con motores especializados en creación de reglas para la detección de fraude. Estos motores para la gestión de fraude demandan gran cantidad de recursos de infraestructura tecnológica y diversos procesos para la homogenización de la información a analizar, que en la mayoría de las ocasiones se origina por múltiples fuentes de información que quedan desconectadas unas de otras, generando silos de información. Esta situación obliga a establecer métodos de traducción y carga de mensajería, tales como: adaptar mensajería base 24 [3] a ISO8583 [4], carga de archivos en diversos formatos (CSV, TXT y Microsoft Excel) para convertirlos en tablas de bases de datos, entre otros procesos.

Las reglas de detección de los motores de gestión de fraude, con frecuencia, han sido creadas y mantenidas por personal experto [5]. Las alertas generadas a partir de las reglas de detección de fraude se validan con los clientes bancarios por personal operativo del sistema de monitoreo, quienes proceden a gestionar el caso con base en la respuesta del cliente, procediendo a bloquear productos o a descartar la alerta.

Con el fin de contemplar un apoyo a los sistemas de detección de fraude, se desarrolló este proyecto el cual propone un sistema experto basado en ontologías y reglas para la detección

de fraude en tarjetas de crédito. Este sistema está compuesto por una red de ontologías, consultas SPARQL [6] y reglas SWRL (*Semantic Web Rule Language*) [7], que están basadas en reglas de detección de fraude estándar del mercado. A su vez, la red de ontologías desarrollada se utilizó para vincular la información de fuentes transaccionales de tarjeta de crédito y de información de apoyo para la detección de fraude, proponiendo una perspectiva de integración semántica de la información basada en los principios de Linked Data [8]. La utilización e integración de diversas tecnologías semánticas ha permitido obtener como producto el mencionado sistema experto. Además, se realiza una evaluación de los resultados obtenidos por dicho sistema contra los resultados de la calificación de fraude que produjo el sistema de monitoreo transaccional actual utilizado por una entidad bancaria. Para esta evaluación se utilizó un *gold standard*, proporcionado por la entidad bancaria, y se aplicaron las métricas *precision, recall* y *F-measure* para medir la efectividad de las reglas desarrolladas.

Este documento, se estructura de la siguiente manera: En el capítulo 1 se realiza la descripción general del proyecto, mostrando la oportunidad y problemática que motivó el mismo. En el capítulo 2 se hace la descripción del proyecto, tras ello se incorporan los objetivos, tanto generales como específicos, y la metodología con la cual se aborda el cumplimiento de estos en el proyecto. En el capítulo 3 se presenta el marco teórico, donde se describen los conceptos fundamentales considerados en el desarrollo del proyecto. El capítulo 4 recoge los trabajos relacionados con esta propuesta. En el capítulo 5 se presenta el desarrollo del proyecto, describiendo la solución a partir de su arquitectura y el desarrollo realizado. En el capítulo 6 se presenta la evaluación de los resultados. Finalmente, en el capítulo 7 se incluyen las conclusiones, aportes y trabajos futuros.

1. DESCRIPCIÓN GENERAL

En este capítulo se hace la presentación del trabajo de grado, describiendo la problemática objetivo, junto con la oportunidad de solución propuesta, para la construcción del sistema experto basado en ontologías y reglas para la detección de fraude en tarjetas de crédito.

1.1. Oportunidad y problemática

En la actualidad, se viene presentando un alto número de fraudes en el sistema financiero y, principalmente, estos fraudes se asocian al producto de tarjeta (crédito o débito). Este tipo de fraude, por definición, es descrito como una forma de robo de identidad, que implica la toma no autorizada de información de la tarjeta de crédito de otra persona con el fin de cargar compras en la cuenta o retirar fondos de esta [2].

A nivel mundial, cada año se causan miles de millones de dólares en pérdidas debido a transacciones fraudulentas con tarjetas de crédito [9], como se reflejó en el año 2016, donde se presentaron fraudes por un monto de $ 22.80 mil millones de dólares, creciendo un 4.4% con respecto al año 2015 [10]. Esta situación hace que la detección de fraude en los sistemas de compras en línea es el tema más candente hoy en día [11]. Los investigadores de fraude, los sistemas bancarios y los sistemas de pago electrónico, como por ejemplo PayPal, deben contar con un sistema de detección de fraudes eficiente y complejo para evitar actividades de fraude que cambian rápidamente. Según un informe de CyberSource [12], uno de los proveedores líderes mundiales en procesamiento de pagos en línea con tarjetas de crédito para empresas, en 2017, el porcentaje de pérdida de fraude en su tienda web fue del 74% y de un 49% en sus canales móviles.

En el caso de América Latina, los pagos en línea representan aún una pequeña, pero creciente, parte del total de ingresos por ventas en línea en el mundo con tarjetas de crédito. En este sentido, se estima que las cifras de *ecommerce* en la región ascenderán de los 4.800 millones de dólares reflejados en 2015 a 16.600 millones en 2020 [13]. En Colombia, en la actualidad, cerca del 80% del fraude que se realiza con tarjetas de crédito ocurre a través de canales no presenciales, es decir, Internet o dispositivos móviles [14].

Para tratar de minimizar las cifras anteriores, las instituciones financieras invierten anualmente buena parte de su presupuesto en temas de riesgo de fraude. Sin embargo, estos esfuerzos no se logran materializar en resultados contundentes debido al dinamismo de los ataques de los delincuentes. En este sentido, las instituciones financieras continúan optimizando la detección de fraude mediante la inclusión de sistemas de monitoreo, con el objetivo de reducir el riesgo de fraude y lograr trasmitir seguridad y satisfacción a sus clientes. No obstante, los esfuerzos por incorporar sistemas de monitoreo de fraude, a pesar de disminuir

en gran medida las incidencias, no ha logrado ser del todo efectivo, presentando persistentes problemas que afectan a los clientes, tales como:

- Transacciones legítimas que se etiquetan incorrectamente como fraude-falso positivo [12].
- Transacciones fraudulentas que se etiquetan como legítimas y falsas negativas [12].
- Problemas derivados de las actuales múltiples fuentes de información desconectadas unas de otras, generando silos de información que son de difícil incorporación en los sistemas de monitoreo de fraude.

Para atacar los problemas mencionados, se han adelantado estudios e implementaciones de soluciones automatizadas para mitigar los fraudes y sus consecuencias, y la comunidad está centrada en determinar las anomalías entre los patrones de comportamiento de fraude que han sufrido cambios en relación con el pasado [16]. Como ejemplo de estos sistemas y modelos de detección de fraude automatizados se encuentran diversas propuestas de sistemas expertos basados en ontologías [17], [18], [19], [20],[21], [22]. Igualmente, los sistemas basados en aprendizaje profundo (*Deep Learning*) están apareciendo para la detección de fraude en este contexto [16], [23], [24]. Sin embargo, en las propuestas de sistemas expertos para detección de fraude bancario no se ha contemplado un sistema experto basado en reglas SWRL que se integren en una red de ontologías. Además, resultan escasos los trabajos relacionados donde el sistema experto se alimente de una integración semántica de múltiples fuentes de información asociadas con transacciones de tarjeta de crédito.

Por tanto, la solución que se plantea en este proyecto pretende ahondar en la creación de sistemas expertos para el análisis y detección de fraude de tarjetas de crédito basado en redes de ontologías y reglas SWRL. Adicionalmente, se procederá a evaluar el sistema desarrollado mediante un comparativo de los resultados de análisis de fraude del sistema experto propuesto con los resultados de un sistema de monitoreo existente en una entidad bancaria.

2. DESCRIPCIÓN DEL PROYECTO

2. Objetivos

2.1. Objetivo general

Diseñar un sistema experto basado en ontologías y reglas que permita la detección de fraude en transacciones bancarias de tarjetas crédito.

2.2. Objetivos específicos

1. Construir una de red de ontologías que permitan modelar el conocimiento relacionado con transacciones bancarias de las tarjetas de crédito.

2. Desarrollar un conjunto de reglas SWRL (*Semantic Web Rule Language*) que ayuden al sistema experto en su proceso de análisis y detección de fraude en tarjetas de crédito.

3. Crear un sistema experto basado en ontologías y reglas para el análisis y detección de fraude de tarjeta de crédito.

4. Validar resultados del sistema experto propuesto comparado con los resultados del sistema de monitoreo existente.

2.3. Metodología

El proyecto utilizó como metodología general la metodología de ciencia basada en el diseño [25]. Esta metodología se completó con otras más específicas según se iban desarrollando las fases del proyecto. Así, utilizamos la metodología NeOn [26] para la generación de la red de ontologías y la metodología Linked Data [8] para la integración semántica de los conjuntos de datos tratados.

La metodología de la ciencia basada en el diseño contemplada abordó las siguientes fases:

2.3.1. Identificación del problema y motivación

Con el fin de realizar la definición del problema específico de investigación y justificar el valor de una solución, se planteó realizar las siguientes tareas:

- Conocer el dominio: Fraude bancario y, más específicamente, el fraude en tarjetas de crédito.
- Revisar el estado del arte sobre ontologías, Linked Data y reglas SWRL relacionadas con el fraude bancario en tarjetas crédito.
- Identificación del problema.
- Identificación de la motivación.

2.3.2. Objetivos de la solución

Buscando inferir los objetivos de una solución a partir de la definición del problema, se expuso realizar las tareas de:

- Identificación de posibles soluciones al problema identificado.
- Identificación de contribuciones de las soluciones posibles al problema.
- Establecer la solución al problema identificado.

2.3.3. Diseño y desarrollo

Con el fin de lograr el objetivo específico número 1: "*Construir una de red de ontologías que permitan modelar el conocimiento relacionado con transacciones bancarias de las tarjetas de crédito"*, se formularon las siguientes tareas:

- Aplicar escenarios de la metodología NeOn [26]
- Construir una red de ontologías de dominio en el ámbito del fraude en tarjetas de crédito.
- Ajustar la red de ontologías construida.
- Documentar la construcción de la red de ontologías construida.
- Desarrollar el proceso de integración semántica mediante la generación de RDF de las fuentes de información de fraude consideradas en el proyecto, lo que permitirá establecer la conexión entre las fuentes de información relevantes y la red de ontologías construida generando una base de conocimiento de fraude, para esto se aplicará la metodología Linked Data [8].

Para abordar el objetivo específico número 2 "*Desarrollar un conjunto de reglas SWRL (Semantic Web Rule Language) que ayuden al sistema experto en su proceso de análisis y detección de fraude en tarjetas de crédito"* se propuso diseñar e implementar las reglas utilizando la especificación de W3C para SWRL [7].

Por último, para cubrir el objetivo específico número 3 *"Crear un sistema experto basado en ontologías y reglas para el análisis y detección de fraude de tarjeta de crédito"*, se proponen las siguientes tareas:

- Desarrollar un proceso de carga del modelo de la red de ontologías por medio del API de Apache Jena.
- Implementar un proceso de almacenamiento de tripletas RDF con base en el modelo de la red de ontologías construida.
- Construir un proceso de consultas SPARQL insumo para la inferencia con las reglas SWRL diseñadas.
- Desarrollar un proceso de inferencia para la ejecución de las reglas SWRL.

2.3.4. Demostración, evaluación y comunicación

Por último, cubriendo las tres fases finales de la metodología de la ciencia basada en el diseño y para dar solución al objetivo específico número 4. "*Validar resultados del sistema experto propuesto comparado con los resultados del sistema de monitoreo existente*" se incluye la ejecución de las siguientes actividades:

a) Implementar indicadores de validación para evaluar el resultado de la inferencia.
b) Ejecución del sistema experto basado en ontologías y reglas SWRL.
c) Comparar los resultados obtenidos del sistema experto propuesto con los resultados del sistema de monitoreo existente (*gold standard*).
d) Identificar y analizar diferencias.
e) Ajustar el modelo, en caso de resultar necesario, de acuerdo con los resultados del análisis efectuado en el paso anterior.
f) Consolidar y analizar los resultados de la ejecución del sistema experto.
g) Elaboración del documento de tesis de maestría.

3. MARCO TEÓRICO

En este capítulo se presentan los conceptos teóricos a partir de los cuales se da el contexto requerido para el entendimiento del desarrollo del proyecto, iniciando desde el concepto de fraude junto con los tipos de clasificación y estrategias sugeridas para contrarrestarlo. Posteriormente, la definición de ontología, para luego abordar el Marco de Descripción de Recursos (del inglés *Resource Description Framework*, RDF), donde se describen los conceptos de tripleta y RDF-*Schema*. Además, se incluyen las definiciones junto con algunos ejemplos del lenguaje de consulta SPARQL. Finalmente, se aborda SWRL como el lenguaje de creación de reglas, utilizado en el desarrollo del sistema experto.

3.1. Fraude

El término fraude se refiere al abuso de un sistema de una organización lucrativa que no necesariamente conduce a consecuencias legales directas. En un entorno competitivo, el fraude puede convertirse en un problema crítico para las empresas si es muy frecuente y si los procedimientos de prevención no son a prueba de fallas. La detección de fraude, al ser parte del control general de fraude, automatiza y ayuda a reducir los procesos manuales en las tareas de revisión / verificación. De acuerdo con Asociación de Examinadores de Fraudes Certificados (ACFE) [11], existen dos tipos de fraude, interno y externo, donde el fraude interno se presenta cuando un empleado ocasiona el fraude en contra de esta u otra organización y el fraude externo que involucra una variedad de personas no pertenecientes a la organización como es el caso de los proveedores, clientes y robos de terceros.

Para el fraude externo, se han identificado tres tipos de defraudadores como son: 1) infractor promedio, 2) delincuente criminal, y 3) delincuente organizado fraude. Se pueden encontrar en la figura 6, los mecanismos de detección y prevención de fraude que comúnmente son utilizados para combatirlo.

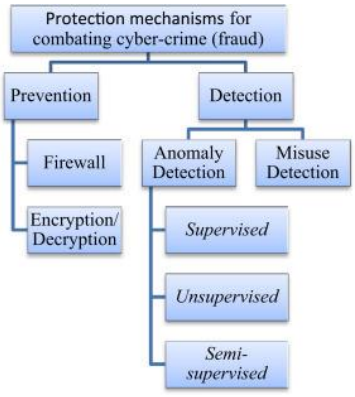


Figura 1. Mecanismos de detección y prevención de fraude [11]

Con el fin de lograr mecanismos efectivos de detección y prevención de fraude, se han desarrollado algunos sistemas como son: los Sistemas de prevención de fraude (FPS) [11], que permiten ser la primera capa de protección para asegurar los sistemas tecnológicos contra el fraude, buscando como primera medida poder restringir, suprimir, destruir, controlar, eliminar o evitar la ocurrencia de ataques cibernéticos en sistemas informáticos a nivel de hardware, software, redes o datos. Comúnmente entre estos sistemas se pueden encontrar los cortafuegos e internamente los sistemas de detección de intrusos. También se encuentran los Sistema de detección de fraude (FDS) [11], como una capa de protección; donde se intenta descubrir e identificar actividades fraudulentas a medida que ingresan a los sistemas que, en la actualidad, se hacen por medio de sistemas FDS computarizados y automatizados.

3.1.1. Fraude en tarjetas de crédito

En este tipo de fraude el término crédito es usado para hacer referencia al método de compra y venta de bienes sin tener dinero. Por tanto, la tarjeta de crédito juega un papel importante en el comercio electrónico y el área de transacciones de dinero en línea que crece cada año, donde los estafadores intentan encontrar más oportunidades para cometer fraudes que pueden causar grandes pérdidas a los titulares de tarjetas y bancos.

Conforme a [11], el fraude crediticio que aplica a las tarjetas de crédito se ha clasificado de la siguiente manera:

- Fraude de tarjeta de crédito fuera de línea. Este tipo de fraude ocurre cuando los estafadores roban la tarjeta plástica y proceden a usarla en los comercios como si fuera el propietario real. Este es un tipo de fraude resulta poco común, ya que las instituciones financieras realizan bloqueos inmediatamente los titulares de las tarjetas perdidas denuncian el robo.

- Fraude de tarjetas de crédito en línea (fraude no presente). Esta modalidad ocurre cuando los estafadores roban la información de las tarjetas de crédito para usarla en el futuro en transacciones en línea por Internet o por teléfono.

- Fraude de aplicaciones. Esta modalidad de fraude se presenta cuando los estafadores ingresan información y datos incorrectos en el formulario de solicitud para abrir una nueva tarjeta de crédito, donde lo que ocurre repetidamente es que los estafadores usan la información de otras personas para obtener tarjetas de crédito u obtener sus nuevas tarjetas de crédito mediante el uso de información personal falsa con la intención de nunca pagar las compras.

- Fraude conductual. Este fraude ocurre cuando los estafadores obtienen los detalles del titular de la tarjeta, para usarlos más tarde en las ventas que se realizan en base al presente

del titular de la tarjeta. Estas ventas incluyen ventas telefónicas y transacciones de comercio electrónico, donde solo se requieren los detalles de la tarjeta.

3.2. Sistema Experto

En la literatura aparecen diversas definiciones de sistema experto. Entre ellas, en este trabajo se destaca aquella que afirma que es un programa informático de resolución de problemas (software) de alto rendimiento, capaz de simular la experiencia humana en un dominio limitado [27]. Igualmente, como un sistema experto se puede definir como aquel programa de computación, ya sea un software o hardware, que contiene el conocimiento de un especialista humano acerca de un determinado campo de aplicación [28]. En este contexto, a nivel más específico, tenemos que un sistema experto basado en reglas es aquel que modela la base de conocimiento en un conjunto de reglas que se extraen como resultado de una serie de pruebas experimentales del tema estudiado [28].

Conforme a lo descrito en [28], un sistema experto está conformado por los siguientes elementos:

- Base de conocimiento. Es el equivalente a la memoria humana, en el sentido de que almacena toda la información disponible para realizar el proceso deductivo.
- Motor de inferencia. Reproduce las operaciones lógicas necesarias -algoritmos- para llegar a las conclusiones deseadas frente a un problema concreto.
- Interfaz de usuario. Consiste en el medio comunicacional a través del cual el sistema recibe información, sobre la base de conocimientos, y contiene los procesos lógicos a realizar y facilita los resultados de la actividad informática.

En la actualidad, se pueden encontrar a los sistemas expertos en una amplia área de aplicaciones para resolver problemas relacionados con: interpretación, predicción, diagnóstico, diseño, planificación, monitoreo, depuración, reparación, instrucción y control [29].

3.3. Ontologías

Una ontología desde la perspectiva informática es definida como una especificación formal y explicita de una conceptualización compartida, donde formal indica que puede ser computable e interpretada por una máquina, explicita debido a que todos sus componentes (conceptos, relaciones, propiedades, entre otros) están definidos explícitamente, conceptualización se refiere a que es un modelo abstracto del dominio para el cual fue construida y por compartida indica que la ontología debe ser consensuada por una comunidad [26].

Se puede decir que las ontologías son vocabularios formalizados de términos, que cubren un determinado dominio de interés, compartidos por una determinada comunidad de usuarios. Igualmente proporcionan un conjunto de supuestos explícitos con respecto al significado intencionado de los términos. Casi siempre incluyen conceptos y su clasificación, igualmente casi siempre incluyen las propiedades existentes entre objetos. Se expresan en OWL o RDF(S), ambos están basados en RDF.

Las ontologías están compuestas principalmente por elementos como clases, relaciones, atributos, funciones, instancias y axiomas [26], donde:

- Las clases hacen referencia a una entidad que puede ser descrita y que se asocia a un identificador único, pueden tener asociados atributos y, generalmente, son asociables a otras clases mediante relaciones.
- Las relaciones son las interacciones entre clases del dominio y, generalmente, estas interacciones son binarias entre dos clases. También se encuentran relaciones de tipo "es un" a "es parte de". Estas relaciones pueden cumplir algunas propiedades como la simetría, transitividad, asimetría, entre otras.
- Los axiomas son afirmaciones que siempre son ciertas dentro de la ontología.
- Las instancias son los elementos o representación de una clase en el mundo real.

Los tipos de ontologías de acuerdo con su grado de formalismo y semántica proporcionada se clasifican en [26]:

- Ontologías terminológicas: en estas ontologías, los conceptos y relaciones no se especifican mediante axiomas y definiciones que determinan las condiciones necesarias y suficientes de su uso.
- Ontologías formales/axiomatizadas: para estas ontologías, los conceptos y relaciones tienen axiomas y definiciones formuladas mediante lógica (o un lenguaje de programación que se puede traducir a lógica).

A partir del relacionamiento entre ontologías, se introduce el concepto de red de ontologías, el cual hace referencia a una colección de ontologías conectadas entre sí a través de diferentes tipos de correspondencia [26]. Durante la creación de una red de ontologías se tiene en cuenta lo siguiente:

- *priorVersionOf*: si la ontología a desarrollar es una nueva versión de una existente.
- *useImports*: si la ontología está importando cualquier otra ontología debido a que contiene en diferentes dominios de conocimiento.
- *extenderBy*: si la ontología está extendiendo una existente.
- *compositebyModules*: si la ontología a desarrollar se compone de una serie de módulos.
- *haveMapping*: si algunos componentes de la ontología tienen correspondencias (*mappings*) con otros componentes de ontologías existentes.

3.4. RDF y RDF-S

El Marco de Descripción de Recursos (RDF) [30] es un lenguaje para representar información sobre recursos en la Web. Está especialmente destinado a representar metadatos sobre recursos web, como el título, el autor y la fecha de modificación de una página web, los derechos de autor y la información de licencia sobre un documento web, etc.

Sin embargo, al generalizar el concepto de un "recurso web", RDF también se puede utilizar para representar información sobre cosas que se pueden identificar en la Web, incluso cuando no se pueden recuperar directamente en la red. Igualmente, RDF está destinado a situaciones en las que la información de los recursos web debe ser procesada por las aplicaciones, en lugar de mostrarse solo a las personas.

Dado que es un marco común, los diseñadores de aplicaciones pueden aprovechar la disponibilidad de analizadores RDF comunes y herramientas de procesamiento, con fines de intercambiar información entre diferentes aplicaciones, logrando que la información pueda estar disponible para aplicaciones distintas de aquellas para las que se creó originalmente.

Adicionalmente, RDF se basa en la idea de identificar cosas utilizando identificadores web (llamados identificadores uniformes de recursos o URI) y describir recursos en términos de propiedades simples y valores de propiedad. Esto permite que RDF represente declaraciones simples sobre recursos como un grafo de nodos y arcos que representan los recursos, sus propiedades y valores.

Como se ilustra en la Figura 2, la estructura central de la sintaxis abstracta de RDF es un conjunto de triples, cada uno de los cuales consta de un sujeto, un predicado y un objeto. Un conjunto de tales triples se llama un grafo RDF, el cual se puede visualizar como un diagrama de nodo y arco dirigido, en el que cada triple se representa como un enlace nodo-arco-nodo.

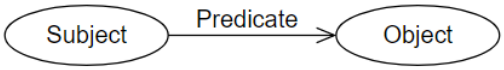


Figura 2. Estructura de un triple RDF [30]

El esquema RDF [31] proporciona un vocabulario de modelado de datos para datos RDF. El esquema RDF es una extensión del vocabulario básico de RDF, que proporciona mecanismos para describir grupos de recursos relacionados y relaciones entre recursos. Los recursos a su vez se utilizan para incluir características a otros recursos, tales como dominio, rango y propiedades. Entre los elementos que incluye RDF-S tenemos los siguientes:

- Clases: Los recursos pueden dividirse en grupos llamados clases y los miembros de una clase se conocen como instancias de la clase. Las clases son en sí mismas recursos, que se identifican por un IRI y se pueden describir utilizando propiedades RDF. La propiedad *rdf:type* se puede usar para indicar que un recurso es una instancia de una clase.

  Las clases se definen en RDF-S con la notación *rdfs:Class*. La propiedad *rdfs:subClassOf* se usa para indicar que una clase es una subclase de otra.

  En la Figura 3 se ilustran las principales clases dispuestas por RDF *Schema*:

Class name	comment
rdfs:Resource	The class resource, everything.
rdfs:Literal	The class of literal values, e.g. textual strings and integers.
rdf:langString	The class of language-tagged string literal values.
rdf:HTML	The class of HTML literal values.
rdf:XMLLiteral	The class of XML literal values.
rdfs:Class	The class of classes.
rdf:Property	The class of RDF properties.
rdfs:Datatype	The class of RDF datatypes.
rdf:Statement	The class of RDF statements.
rdf:Bag	The class of unordered containers.
rdf:Seq	The class of ordered containers.
rdf:Alt	The class of containers of alternatives.
rdfs:Container	The class of RDF containers.
rdfs:ContainerMembershipProperty	The class of container membership properties, rdf:_1, rdf:_2, ..., all of which are sub-properties of 'member'.
rdf:List	The class of RDF Lists.

Figura 3. Clases incluidas por RDF-S [31]

- Propiedades: La especificación RDF de conceptos y sintaxis abstracta describe el concepto de una propiedad RDF como una relación entre los recursos del sujeto y los recursos del objeto. En la Figura 4 se ilustran las principales propiedades dispuestas por RDF Schema:

Property name	comment	domain	range
rdf:type	The subject is an instance of a class.	rdfs:Resource	rdfs:Class
rdfs:subClassOf	The subject is a subclass of a class.	rdfs:Class	rdfs:Class
rdfs:subPropertyOf	The subject is a subproperty of a property.	rdf:Property	rdf:Property
rdfs:domain	A domain of the subject property.	rdf:Property	rdfs:Class
rdfs:range	A range of the subject property.	rdf:Property	rdfs:Class
rdfs:label	A human-readable name for the subject.	rdfs:Resource	rdfs:Literal
rdfs:comment	A description of the subject resource.	rdfs:Resource	rdfs:Literal
rdfs:member	A member of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:first	The first item in the subject RDF list.	rdf:List	rdfs:Resource
rdf:rest	The rest of the subject RDF list after the first item.	rdf:List	rdf:List
rdfs:seeAlso	Further information about the subject resource.	rdfs:Resource	rdfs:Resource
rdfs:isDefinedBy	The definition of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:value	Idiomatic property used for structured values.	rdfs:Resource	rdfs:Resource
rdf:subject	The subject of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:predicate	The predicate of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:object	The object of the subject RDF statement.	rdf:Statement	rdfs:Resource

Figura 4. Propiedades incluidas por RDF-S [31]

3.5. Linked Data

Es un conjunto de principios de diseño para compartir datos interconectados legibles por máquina en la Web y es uno de los pilares centrales de la Web Semántica, conocida como la Web de Datos [8]. La Web Semántica trata de hacer enlaces entre conjuntos de datos que sean entendibles no solo para los humanos, sino también para las máquinas para lo cual Linked Data proporciona las mejores prácticas para hacer posibles estos enlaces. Los principios asociados a Linked Data son las siguientes:

- Usar URI para nombrar las cosas.
- Utilizar HTTP y URI para que las personas puedan buscar esos nombres.
- Proporcionar información útil, utilizando los estándares RDF y SPARQL.
- Incluir enlaces a otros URI, para que se puedan descubrir más cosas.

Linked Open Data (LOD) [32] es Linked Data publicado con una licencia abierta, por lo cual no impide su reutilización y, adicionalmente, es de forma gratuita. Los datos vinculados, en general, no tienen que estar abiertos, sin embargo, para cumplir con que sean datos vinculados abiertos, entonces tienen que ser de acceso libre (abiertos). Para dar claridad con lo que deben cumplir los datos para ser abiertos, se ha diseñado un esquema de estrella donde se evalúa el nivel de apertura de los datos vinculados, donde el nivel de máxima apertura es 5 estrellas y el menor 1 estrella (ver Figura 4).

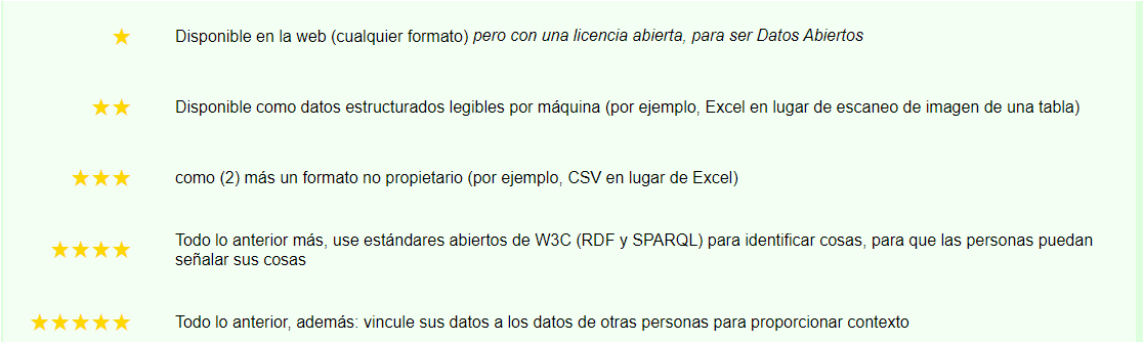


Figura 5. Esquema calificación LOD [32]

En la figura 5 se ilustra la nube LOD [30] conformada por el universo de datos abiertos vinculados, donde diversos conjuntos también presentan licencias de datos abiertas.

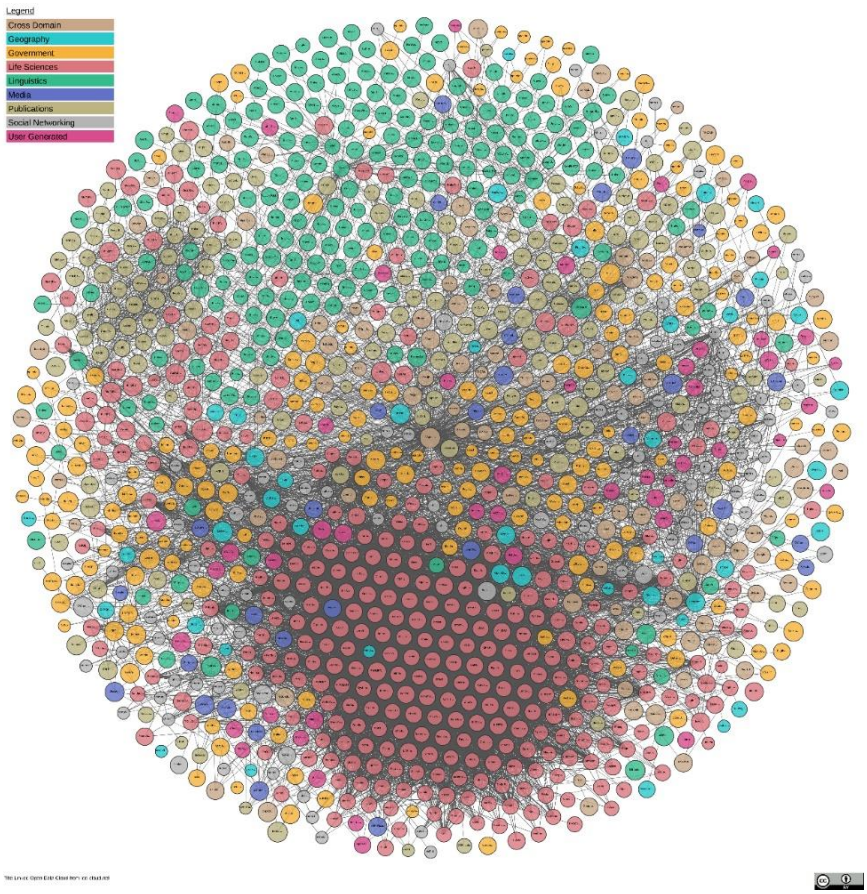


Figura 6. Linked Open Data Cloud [33]

3.6. SPARQL

SPARQL es un acrónimo recursivo de *SPARQL Protocol and RDF Query Language* [34], el cual es un lenguaje estandarizado para consulta de grafos RDF y clave en el desarrollo de la Web Semántica. Ese lenguaje se utiliza para expresar consultas que permiten hacer llamados a diversas fuentes de datos, si los datos han sido almacenados de forma nativa como RDF o si han sido definidos mediante vistas RDF a través de algún sistema *middleware*. SPARQL contiene las capacidades para la consulta de patrones obligatorios y opcionales de un grafo, junto con sus conjunciones y disyunciones [6]. Los resultados de las consultas SPARQL pueden ser conjuntos de resultados o grafos RDF.

Un ejemplo [6] de una consulta, sobre una tripleta de un libro, donde el sujeto es: **<http://example.org/book/book1>,** el predicado **<http://purl.org/dc/elements/1.1/title>** y el objeto **"SPARQL Tutorial"** (literal de cadena) y se quiere obtener el valor del título sería la siguiente:

Datos:

```
<http://example.org/book/book1> <http://purl.org/dc/elements/1.1/title>
"SPARQL Tutorial".
```

Consulta:

```
SELECT ?title

WHERE

{

  <http://example.org/book/book1> <http://purl.org/dc/elements/1.1/title>
?title .

}
```

El resultado de la consulta sería:

Tabla 1. Resultados consulta SPARQL sobre tripleta [6]

title
"SPARQL Tutorial"

3.7. SWRL

SWRL es un acrónimo en inglés de *Semantic Web Rule Language* [7] el cual está diseñado para ser el lenguaje de reglas de la web semántica. Este lenguaje incluye una sintaxis abstracta de alto nivel para reglas *Horn* y las reglas deben expresarse en términos de conceptos OWL (clases, propiedades, individuos).

Las reglas tienen la forma de una implicación entre un antecedente (cuerpo) y el consecuente (cabeza). El significado deseado puede leerse como: siempre que se cumplan las condiciones especificadas en el antecedente, también deben cumplirse las condiciones especificadas en el consecuente. Un ejemplo [36] de reglas SWRL se muestra a continuación:

- Sintaxis legible por humanos:

hasParent(?x1,?x2) ∧ hasBrother(?x2,?x3) ⇒ hasUncle(?x1,?x3)

- Sintaxis concreta XML:

La sintaxis concreta XML es una combinación de la sintaxis de presentación XML del lenguaje de ontología web OWL con la sintaxis XML RuleML, a continuación, se ilustra con un ejemplo la sintaxis XML RuleML:

```
<ruleml:imp>
  <ruleml:_rlab ruleml:href="#example1"/>
  <ruleml:_body>
    <swrlx:individualPropertyAtom  swrlx:property="hasParent">
      <ruleml:var>x1</ruleml:var>
      <ruleml:var>x2</ruleml:var>
    </swrlx:individualPropertyAtom>
    <swrlx:individualPropertyAtom  swrlx:property="hasBrother">
      <ruleml:var>x2</ruleml:var>
      <ruleml:var>x3</ruleml:var>
    </swrlx:individualPropertyAtom>
  </ruleml:_body>
  <ruleml:_head>
    <swrlx:individualPropertyAtom  swrlx:property="hasUncle">
      <ruleml:var>x1</ruleml:var>
      <ruleml:var>x3</ruleml:var>
    </swrlx:individualPropertyAtom>
  </ruleml:_head>
 </ruleml:imp>
```

4. TRABAJOS RELACIONADOS

En este capítulo se presentan los trabajos relacionados en el contexto del fraude en tarjetas de crédito, donde se muestra una tabla resumen con los detalles de las propuestas existentes que poseen un enfoque semántico para abordar el fraude de tarjetas de crédito. En dicha tabla (ver Tabla 2) se presentan los aspectos generales de cada uno de los trabajos relacionados, como son el nombre del artículo, año de publicación, el dominio al cual se aplicó, las fuentes de información sobre las cuales se realizó la investigación, los elementos ontológicos tenidos en cuenta, así como la técnica de desarrollo de las ontologías, para finalmente indicar el método de validación de los resultados de la investigación.

De forma particular, la propuesta descrita en [38] propone una representación ontológica de la criminología financiera para capturar los términos y temas comúnmente abordados en investigaciones y estudios de criminología financiera. Este trabajo realiza un análisis de texto para extraer e identificar los términos y a partir de ellos identificar la frecuencia de los términos utilizados en cada uno de los trabajos de investigación. Finalmente, los términos identificados se convierten a un lenguaje de representación ontológica (OWL) mediante Protégé, utilizando la metodología propuesta por Noy & McGuinness [39]. En [40] se ha propuesto la creación de un sistema de detección de fraude, que realiza el descubrimiento de patrones de fraude en estados financieros, apoyado en la ontología creada llamada *Fraud detection ontology* soportada bajo cinco tipos de variables del dominio de estados financieros y construida basada en la metodología de Noy & McGuinness [39]. Por último, se definen las reglas en lenguaje SWRL sobre el motor de inferencia Pellet.

Para el estudio [20], se propone un enfoque basado en ontología para la detección y clasificación de conflictos semánticos en sistemas expertos basados en reglas, el cual se centra en el caso de repositorios de reglas antifraude para la inspección de transacciones de tarjeta no presente (CNP) en entornos de comercio electrónico. Los experimentos mencionados en este estudio afirman que los enfoques ontológicos pueden descubrir y clasificar efectivamente conflictos en sistemas expertos basados en reglas en el campo de las aplicaciones antifraude. Igualmente mencionan que esta propuesta también se aplica a otros dominios donde están involucradas las bases de reglas de conocimiento. En este estudio los autores definen la ontología *afro.owl* (*Anti Fraud Rules Ontology*) utilizando la propuesta realizada *en Methodology For Building Fraud Ontologies* [41] donde la granularidad del estudio está planteado a nivel de Europa.

Según [42], se propone un enfoque basado en ontología para recopilar, integrar y almacenar datos de análisis web, de muchas fuentes de huellas digitales populares y comerciales, para el análisis del comportamiento del cliente en sitios de comercio electrónico, logrando diferentes perspectivas en el análisis del comportamiento del cliente. El artículo está planteado con granularidad a nivel de Europa y la ontología indicada por los autores es "*wao.owl*" (*Web Analytics Ontology*), en la cual utilizaron para su creación la metodología Noy & McGuinness [39]. Pare el caso del estudio [43] se especifica a la ontología *FinRegOnt* (FRO) como

una parte fundamental de un enfoque de Web Semántica para el cumplimiento normativo de las instituciones financieras. La cual está basada en los estándares de la industria: FIBO [44] y LKIF [45] y la aplicabilidad a nivel de Estados unidos. En el caso de estudio [44] se menciona a la ontología FIBO, la cual ha sido definida en dos formas: simple y compleja. En su forma simple ha sido empleada para variedad de usos, tales como glosario en inglés o un diccionario de datos de un banco o regulador. En su forma compleja, FIBO en su *Web Ontology Language* (OWL) nativo puede usarse como ontología operativa de un banco que tiene una planeación de granularidad a nivel mundial.

Para [46] se discute cómo las tecnologías semánticas podrían hacer que la investigación del cibercrimen sea más eficiente, tomando como ejemplo el fraude bancario en línea para proponer una ontología destinada a mapeo criminal organizaciones e identificar desarrolladores de malware. Igualmente, sugiere reglas de inferencia basadas en el conocimiento empírico que podrían abordar mejor las necesidades del analista humano. Finalmente, este estudio se propuso a nivel de Brasil y menciona la ontología creada para este se basó en la metodología NeOn. Luego de esto en [21] se aborda el desarrollo de un mecanismo efectivo para detectar transacciones sospechosas, el cual es un problema crítico para las instituciones financieras en su esfuerzo por prevenir actividades contra el lavado de dinero. Propone un sistema experto basado en la ontología para la detección de transacciones sospechosas llegando a la creación de *ontology based expert system to detect suspicious transactions*. En el caso de [22] se propone un método basado en la similitud de instancias ontológicas, para detectar el fraude, por medio de la comprobación de cambios sospechosos en el comportamiento del usuario, se dice que para esto los autores crean *Ontology-Based Fraud Detection.*

Se encuentra en [47] la descripción del concepto de ontología tópica, siendo una "ontología tópica" el conjunto de temas identificados para representar la estructura de conocimiento del experto de dominio. También se desarrolla una ontología tópica de fraude, la cual incluye en su alcance específico, las perspectivas y la granularidad de la conceptualización sobre estos temas. Se basa en una ontología básica que integra ontologías de dominios múltiples y sirve como el marco de conocimiento para ontologías de aplicaciones. Para esto, los autores, hacen uso de la metodología AKEM [48] y plantean la granularidad del estudio a nivel de Europa. En el estudio [49] es analizado el estado de la investigación sobre detección y prevención del fraude financiero, realizada como parte del proyecto financiado por la Comisión Europea IST FF POIROT [50] (Recursos de información orientados a la prevención del fraude financiero usando tecnología ontológica), especificando los requisitos del usuario que definen la funcionalidad de la ontología del fraude financiero diseñada por los socios de FF POIROT. Donde se afirma que modelar actividades fraudulentas implica una mezcla de leyes y hechos, así como inferencias sobre hechos presentes, hechos presuntos o hechos faltantes. El artículo está planteado con granularidad a nivel de Europa.

Por último, en [51] se menciona la creación de la Ontología Superior Sugerida (SUMO) utilizada para aplicaciones de búsqueda, lingüística y razonamiento, donde el autor indica que la granularidad es Estados unidos.

Tabla 2. Trabajos sobre propuestas semánticas relacionadas con fraude en tarjeta de crédito

Artículo	Año	Dominio	Fuentes de información	Elementos	Técnicas de desarrollo	Método validación
An Ontology-Based Representation of Financial Criminology Domain Using Text Analytics Processing [38]	2018	Criminología financiera	Veinticinco (25) revistas y artículos de investigación en criminología financiera han sido seleccionados para esta investigación con el fin de extraer los términos y temas abordados de los estudios de criminología financiera.	La investigación encontró que hay nueve (9) clases (Temas) que comúnmente se investigan en el campo de la Criminología Financiera: 1. Personas 2. Delitos 3. Sector de riesgo 4. Cumplimiento 5. Tecnología 6. Ubicación 7. Marco temporal 8. Recursos 9. Propiedad.	Análisis de texto.	Tres (3) expertos en Criminología Financiera y Auditores.
Knowledge-based Financial Statement Fraud Detection System: Based on an Ontology and a Decision Tree [40]	2017	Fraude en estados financieros	Un conjunto de 130 informes fue empleado, este conjunto de datos contenía 260 empresas (130 firmas comprometidas con el fraude y 130 firmas que no comprometían el fraude). En este conjunto de datos, 200 empresas fueron empleadas en la generación de reglas de detección de fraude y se utilizaron sesenta en la construcción de ontologías.	Variables financieras de cinco categorías: tamaño de la empresa, variables de rentabilidad, variables operacionales, variables de estructura y variables de actividad.	Árbol de decisiones. Ingeniería ontológica	Validación Cruzada - ten-fold cross-validation method
Enhancing semantic consistency in anti-fraud rule-based expert systems [20]	2017	Tarjeta Crédito, e-commerce	SME-Ecompass FP7 European Initiative http://www.sme-ecompass.eu/	ConditionGroups: ConditionGroups1, ConditionGroups2, ConditionGroups3, ConditionGroups4, ConditionGroups5, ConditionGroups6 and inconsistent conditiongroups are subclasses Rules: Rules1, Rules2, Rules3, Rules4, Rules5, Rules6, InconsistentRules and TautologicalRules are subclasses. RuleConditions Operators	Ingeniería ontológica	Se hizo uso del servicio AFRUSA para detectar los errores en las reglas antifraude (base de datos 2155 reglas antifraude) teniendo en cuenta las reglas SWRL definidas.
An ontology-based data integration approach for web analytics in e-commerce [42]	2016	E-commerce, Analítica Web	Datos de Google Analytics y las huellas digitales de Piwik asignadas en 15 tiendas electrónicas de diferentes sectores comerciales y países (Reino Unido, España, Grecia y Alemania)	Clases principales: Analytics_parameters E-shop Visitor Page Item	Ingeniería ontológica	Procedimientos de aprendizaje no supervisados. Algoritmo de agrupamiento. Árbol de decisiones con reglas.
Financial Regulation Ontology [43]	2016	Regulación Financiera	FIBO LKIF FRO FRO se completa con el texto completo de las leyes y regulaciones de EE. UU. Para la gestión bancaria y de inversiones:	No relacionados	No relacionados	No relacionado

What is the Financial Industry Business Ontology (FIBOTM)? [44]	2015	Industria Financiera	Actualmente, FIBO consta de 11 dominios centrales de la industria financiera, incluidos valores y acciones, préstamos y más, en 49 módulos y más de 400 archivos de ontología.	300 ontologías 30 Dominios	Ingeniería ontológica	Expertos en la materia en servicios financieros, gestión de procesos de proyectos y estándares, ontología y arquitectura.
Applying Semantic Technologies to Fight Online Banking Fraud [46]	2015	Fraude Bancario, Cibercrimen	Se evaluaron 30 informes forenses de dispositivos analizados a lo largo de 2013 en términos de: diversidad de autoría, riqueza del contenido y variedad de casos: los informes seleccionados deben cubrir tanto las quejas de fraude de cajeros automáticos de diferentes bancos como la computadora dispositivos incautados durante distintas operaciones.	No relacionados	Ingeniería ontológica	Validación de experto
Ontology Based Expert-System for Suspicious Transactions Detection [21]	2014	Lavado de activos. Transacciones débito y crédito	Conjunto de datos reales de más de 8 millones de transacciones de un banco comercial.	Clases Principales: Account Group suspiciousTxn TxnGroups TxnType	Ingeniería ontológica	Institutos financieros y organismos reguladores en función de su experiencia en el manejo de transacciones sospechosas.
Ontology-Based Fraud Detection [22]	2007	Detección de fraude, comportamiento, personalidad	No relacionada	No relacionada	No relacionada	No relacionada
Towards a Topical Ontology of Fraud [47]	2006	Prevención y detección de fraude	No relacionada	Los 9 clústeres son: Prevention Fraud Type Participan Profile Detection Fraud Configuration Motivation Actors Investigation Resolution	Ingeniería ontológica	Validación de experto
Towards a financial fraud ontology: A legal modelling approach [49]	2004	Prevención y detección de fraude Financiero. Modelo legar	Páginas de internet	Conceptos de: agent, role, intention, document, norm, right, and responsibility	Ingeniería ontológica	Leyes incluidas por la Commissione Nazionale per le Societa' e la Borsa (http://www.CONSOB.it). Reglas IOSCO.
The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications [51]	2002	Finanzas	Comunidad	No relacionada	Ingeniería ontológica	No relacionado

Estos trabajos relacionados, como se mencionó con anterioridad, presentan propuestas de ontologías especializadas en fraude financiero y la mayoría busca mediante su aplicación lograr la detección de anomalías y operaciones sospechosas para los campos de acción en que se diseñaron. Así mismo, cada una de las ontologías pretende cubrir un área de conocimiento específica y al evaluar en conjunto, no se observa correlación entre los trabajos revisados y tampoco iniciativas que surjan a partir de los trabajos pasados. Se destaca dentro de los trabajos que la mayoría han sido propuestos en Europa y a nivel de Suramérica se cuenta con una sola propuesta en Brasil. También se observa que con el paso del tiempo los trabajos se han enfocado en implementar algoritmos de aprendizaje automático como un valor agregado a los temas de prevención y detección de fraude.

Finalmente, en los trabajos relacionados, recogidos en la Tabla 2, se detecta como limitación que ninguno de los trabajos presenta formalmente una red de ontologías creada para transacciones de tarjeta crédito. Adicionalmente, los sistemas expertos que se mencionan tienen limitación en que no son especializados en transaccionalidad de tarjeta de crédito y no combinan inferencia a partir de dicha transaccionalidad. La mayoría de los incluidos se especializan en validación semántica, propuestas de reglas, pero no se observa una propuesta que integre los temas de un sistema experto basado en ontologías y reglas para la detección de fraude, razón por la cual surge este proyecto.

5. DESARROLLO

En este capítulo se realiza la descripción de la solución propuesta, presentando inicialmente la arquitectura para luego ir detallando lo desarrollado en la solución, abordando cada una de las fases de la metodología enmarcada en los principios de Linked Data, haciendo énfasis en las herramientas a partir de cada una de las tareas con las cuales se logró dar solución a lo propuesto.

5.1. Descripción de la solución

5.1.1. Arquitectura

Bajo el marco propuesto de la metodología de ciencia basada en el diseño, la cual fue acompañada por la metodología NeOn para diseño de la red de ontologías y de Linked Data para contemplar los estándares de datos conectados, se diseña la arquitectura con la cual se da cumplimento a los objetivos propuestos en este trabajo.

La arquitectura se esquematiza en la Figura 7 y posee las capas de:

i. Datos: en la cual se encuentran las fuentes de información con la cual se alimenta el sistema experto.

ii. ETL: en la cual se realizan los procesos de carga, adaptación y llamados a persistencia de la información al almacén de datos RDF;

iii. Publicación: en esta capa se tiene el almacén de datos RDF con la especificación TDB2 de Jena sobre el cual se realizan consultas con SPARQL y actualizaciones del modelo RDF

iv. Inferencia: en esta capa se aplican las reglas de detección de fraude con Jena y SWRL.

En la sección 5.1.2 del documento se encuentra el detalle de las capas de la arquitectura en mención.

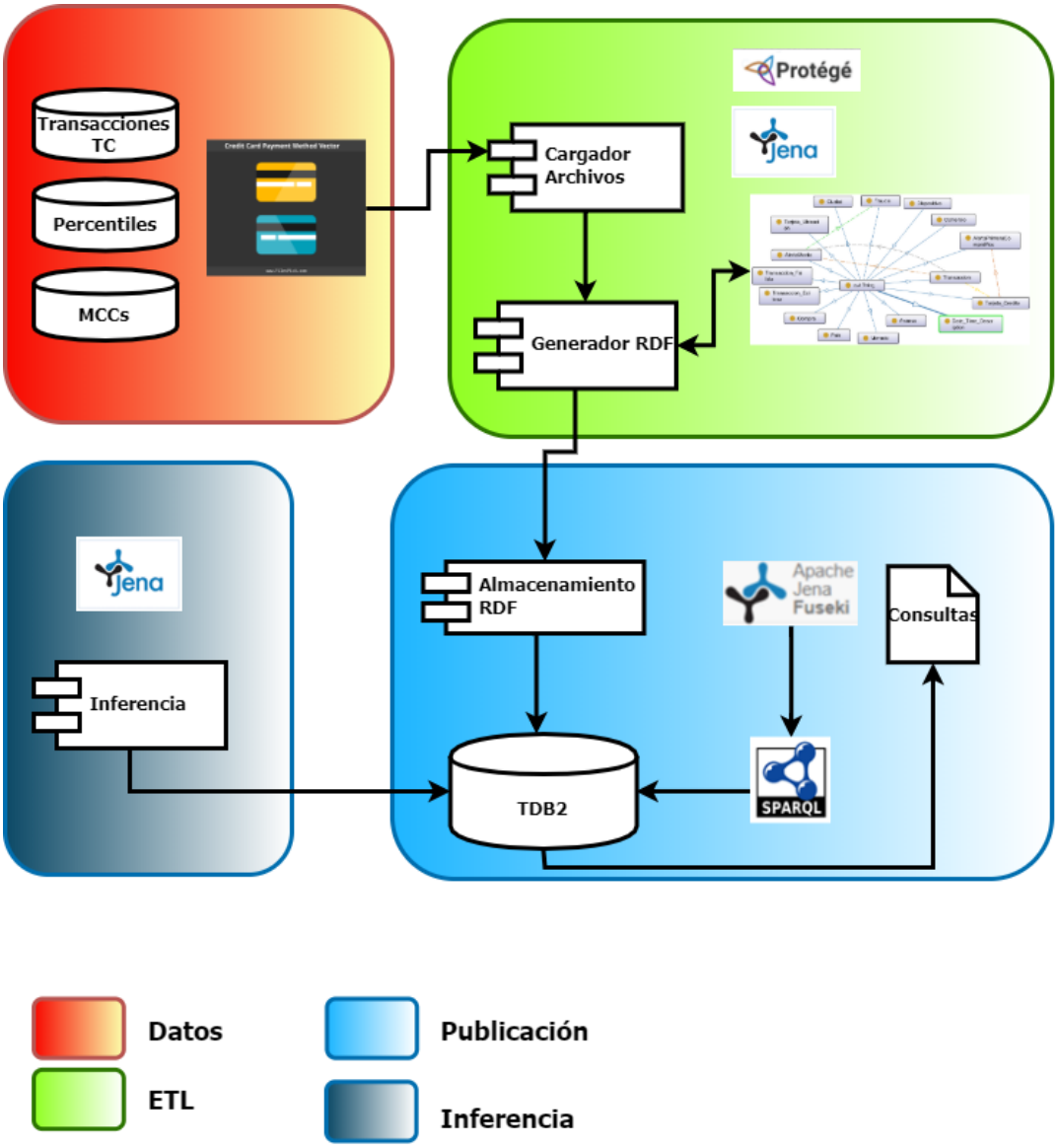


Figura 7. Arquitectura del sistema propuesta

5.1.2. Desarrollo de la solución

Como se ha venido mencionado en el documento, para el desarrollo de la solución propuesta se hace uso de la metodología Linked Data [8], apropiando las siguientes actividades de la metodología: Especificación, generación de RDF y publicación. Adicional a las etapas anteriores, se incluye la etapa de Inferencia que no hace parte de la metodología de Linked Data, pero será el valor agregado en este trabajo de grado.

Se relaciona a continuación el resultado esperado, para cada una de las fases de la metodología Linked Data adoptada:

- Modelado: El resultado de esta fase será la especificación de la ontología propuesta, la cual se logrará mediante la aplicación de los siguientes escenarios de la metodología NeOn:
    - Escenario 1: definición de los requerimientos de la red de ontologías de transacciones de tarjeta de crédito,
    - Escenario 2: reutilización y rediseño de recursos no ontológicos (NORs),
    - Escenario 3: reutilización de los recursos ontológicos.
    - Escenario 9: localización de recursos ontológicos.
- Generación de RDF: En esta etapa se realizarán dos procesos, el primero será la carga de la información transaccional y el segundo la generación de RDF.
- Publicación: Despliegue de un triple store que almacenará el RDF generado.

Para la fase de inferencia, como resultado se tendrá la aplicación de reglas de detección de fraude aplicadas sobre la información almacenada en el *triple store*.

En cuanto a los aspectos tecnológicos, como parte del desarrollo de la solución, se hace una evaluación de diversas herramientas que pueden permitir desarrollar la solución de acuerdo con la arquitectura propuesta. Para esto, se inició validando el *framework* de Apache Jena [53], el cual ofrece la posibilidad de implementar todos los componentes de la arquitectura, solo que en lugar de manejar SWRL como lenguaje de reglas, ofrece su propio lenguaje de reglas. De acuerdo con esto, se realiza la validación para integrar en el trabajo el API OWL [54], combinada con el razonador Pellet [55]. Estas tecnologías permitían lograr la definición y ejecución de las reglas conforme a SWRL. Sin embargo, esta solución tuvo una limitante y fue que no contaba con un repositorio robusto para el almacenamiento de las tripletas RDF de la información transaccional, encontrando como una solución el uso de repositorios de pago y ninguno de uso libre. Conforme a esta limitación, se evaluó el uso de Apache Marmotta [56] como repositorio RDF. Sobre esta tecnología se logró realizar el almacenamiento de tripletas RDF y la generación de consultas. Sin embargo, esta opción tuvo como desventaja el no contar con un lenguaje robusto de definición de reglas, así como no tener integración con SWRL.

Considerando las diversas opciones analizadas, se decidió utilizar el *framework* de Apache Jena [53], empleando las herramientas con las que cuenta dicho *framework* para cada una de las necesidades y componentes definidos en el proyecto. Así, como primera medida, se utilizó el API RDF, el cual se utiliza para la generación del RDF. También se hace uso del API Store, que permite el almacenamiento de las tripletas en el repositorio TDB. Seguidamente, el servidor Apache Fuseki se emplea para la publicación de resultados haciendo uso del lenguaje de consulta SPARQL. Adicionalmente, las reglas son diseñadas con al lenguaje definido por el API Reasoner y se emplea también el API Inference para obtener inferencia de las reglas desarrolladas. La utilización de los diversos componentes del *framework* de Apache Jena permite la integración de todos los elementos que componen la arquitectura definida para el sistema experto propuesto en este trabajo.

5.1.2.1. Datos

Esta fase del trabajo está asociada con la actividad de Especificación de la metodología propuesta en [8], centrándose en la recopilación de las fuentes de información con las que se va a trabajar. Así, en esta fase se tuvo en cuenta las fuentes de datos de información de percentiles de los montos de compra para cada una de las tarjetas de crédito y de información de transacciones de tarjeta de crédito. A continuación, se describe cada una de las fuentes de datos consideradas en este trabajo:

- **Información percentiles**

La fuente de información de percentiles contiene información de los percentiles 95 y 75 basados en los montos de las transacciones de compra para cada una de las tarjetas de crédito. El archivo de información de percentiles mostrado en la Figura 10 tiene un número aproximado de 25 mil registros. Cada registro (línea) que contiene este archivo tiene un tamaño aproximado de 90 caracteres y está compuesto por los siguientes campos:

  - *tarjeta*: número de la tarjeta de crédito cifrado con algoritmo de resumen SHA256.
  - *P95*: valor del percentil 95 calculado a partir de los montos de las transacciones de compra de la tarjeta de crédito.
  - *P75*: valor del percentil 75 calculado a partir de los montos de las transacciones de compra de la tarjeta de crédito.

Figura 8. Fragmento de archivo de percentiles

- **Información transaccional**

La fuente de información transaccional contiene las tramas que se generan desde los autorizadores transaccionales, a partir las transacciones de compras y avances de tarjeta de crédito, entendiendo como trama la información un registro de una transacción. Cada trama contiene los valores de las variables que detallan la transacción y tiene una longitud aproximada de 263 caracteres. El número de tramas por cada archivo es de aproximadamente 150 mil correspondientes al promedio de transacciones diarias. Las tramas (registros) de tarjeta de crédito mostradas en la Figura 11 presentan los siguientes campos:

- COD_TRX: indica el tipo de transacción que puede ser avance, compra, consulta, u otros significados a usos internos que requiera la entidad financiera.
- REVERSO: indica si se trata de una transacción reversada (no aplicada).
- MONTO_EN_MONEDA_LOCAL: valor en pesos colombianos de la transacción.
- NUMERO_TRX: número de la transacción generada por los sistemas autorizadores de transacciones de tarjeta de crédito.
- HORA_TRX: hora en la que se realizó la transacción de tarjeta de crédito.
- FECHA_TRX: fecha en la que se realizó la transacción de tarjeta de crédito.
- MCC: Actividad a la que se dedica el comercio al que se le hace la compra con tarjeta de crédito, por ejemplo, supermercados, aerolíneas, entre otras.
- ENTRY_MODE: datos de cómo se realizó la transacción, que puede ser de forma presente, por dispositivo bluetooth, por internet, entre otros.
- PIN_ENTRY_CAPABILLITY: en este campo se especifica si el dispositivo en el cual se registra la compra tiene capacidad de lectura de claveen caso de un cajero automático o de leer pin en un punto de venta.
- CONDICION_PTO_VENTA: identifica el entorno en el que se hizo la transacción, el cual puede ser compra por internet, por teléfono o por medio de un dispositivo físico.

- CANAL: red lógica a la que se encuentra conectada la terminal y por donde curso la transacción.
- ID_ADQ: código que Identifica la entidad donde el comercio tiene la cuenta y donde se abona el valor de las ventas.
- COD_RPTA: código de respuesta de la transacción, donde se puede reconocer si fue una transacción aprobada o rechazada.
- ID_TERMINAL_DIRECCION_IP: código del dispositivo o ATM desde el cual se hizo uso de la tarjeta de crédito.
- COD_CIO_AGENCIA_OFICINA_ORIGEN: en este campo llega el *nit* con el cual una entidad financiera reconoce un comercio, el cual puede ser diferente por cada entidad.
- NOMBRE_LOCALIZACION_COMERCIO: descripción con la cual una entidad financiera reconoce un comercio, el cual puede ser diferente por cada entidad.
- LOCALIDAD_COMERCIO: Ciudad donde se hace la transacción.
- PAIS_ORIGEN: país donde se hace la transacción.
- ID_COMERCIO: complementa el valor del campo COD_CIO_AGENCIA_OFICINA_ORIGEN ya que aquí indica el número del NIT.
- INDICADOR_E_COMERCE: indica si una transacción se hace o no por Internet.
- NO_TARJETA_ENCRIPTADO_SHA256: número de Tarjeta con la que se realizó la transacción cifrado con algoritmo de resumen SHA256.
- INDICADOR_DE_FRAUDE: indicador de resultado de Investigación: Buena [G], Descartada [D], Fraude [F] y Pendiente [P].
- CODIGO_DE_COMERCIO_P48_A27: mismo código de agencia COD_CIO_AGENCIA_OFICINA_ORIGEN.
- REGION: indica la región donde se realiza la transacción.
- L_I_E: indica si la transacción se hizo local, internacional.
- MCG_P: código de agrupamiento de los *mcc* [37].

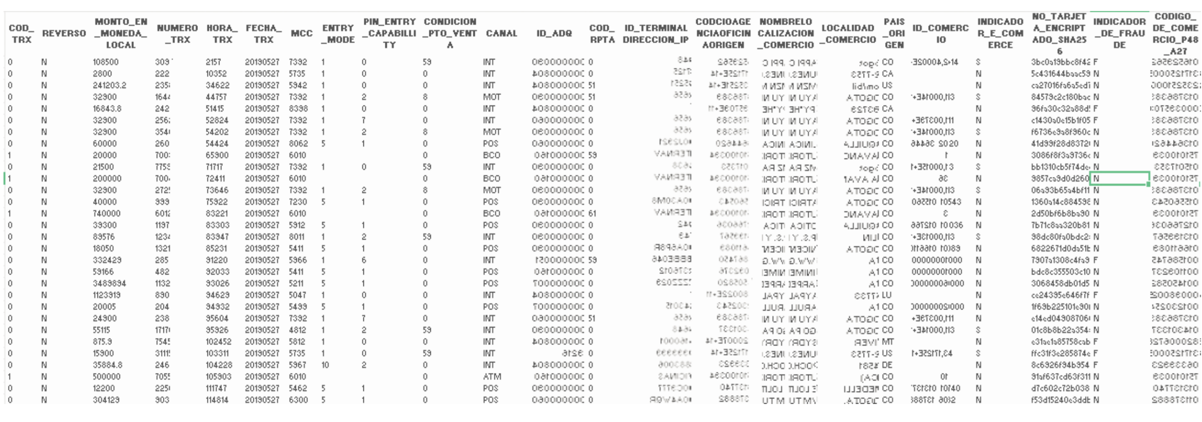
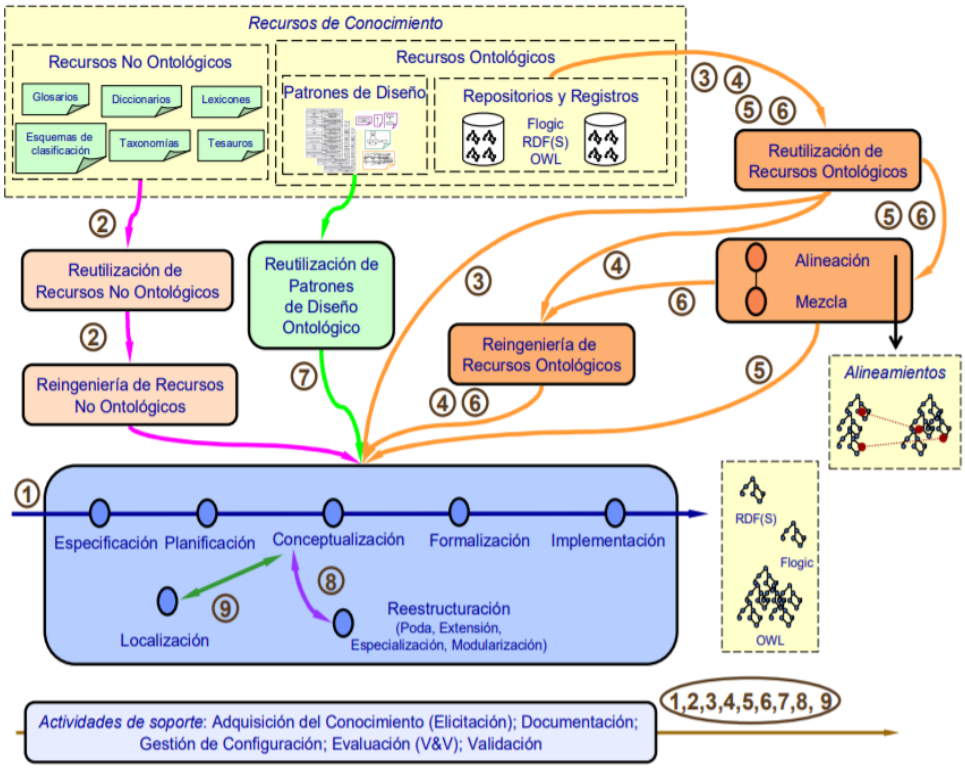


Figura 9. Fragmento de la fuente de información transaccional

5.1.2.2. Modelado

Para el desarrollo de la red de ontologías se abordó la ejecución de los escenarios de las metodologías NeOn que aplicaban para esta solución y que se detallan a continuación. Asimismo, la construcción de la red de ontologías parte de un enfoque *bottom-up* desde las fuentes de información de transacciones de tarjetas de crédito, así como de los catálogos de información que dan detalle a los valores que viajan en una transacción. Igualmente, la propuesta se complementa con un enfoque *top-down* a partir de la reutilización de diversas ontologías existentes que pasarán a formar parte de la red de ontologías desarrollada. A continuación, se recogen los detalles del proceso de construcción:



Conforme al Escenario 1 de la metodología NeOn se aborda la definición de los requerimientos de la red de ontologías de transacciones de tarjeta de crédito. Para ello, con el fin de describir el propósito, alcance y requisitos de la red de ontologías, se hace uso de la plantilla para el documento de especificación de requerimientos de la ontología [26], cuyo detalle se recoge en la tabla 3.

Tabla 3. Especificación de requisitos de la red de ontologías de tarjetas de crédito

1	**Propósito**
	El propósito de la red de ontologías es modelar las transacciones de tarjetas de crédito, desde una visión enriquecida por la consideración de información desde diferentes fuentes y así permitir realizar un análisis del comportamiento transaccional de los clientes, con la cual se pueda llegar a identificar transacciones atípicas que ameriten generar un alertamiento por riesgo de fraude.
2	**Alcance**
	La red de ontologías se conformará a partir de la información de transacciones de tarjetas de crédito extraídas de la base de transacciones de los sistemas autorizadores de tarjeta de crédito de una entidad bancaria. Esta fuente de información incluye el detalle de los movimientos de compra o avance. Así mismo, la red representará la información de la fuente de información que incluye los valores de los percentiles calculados sobre los montos transaccionales de tarjetas de crédito.
3	**Lenguaje de implementación**
	El lenguaje de esta red de ontologías estará implementado en OWL.
4	**Usuarios Finales previstos**
	Los usuarios previstos principalmente que harán uso de esta ontología serán; o Organismos científicos o Usuarios de áreas de fraude de entidades bancarias o Grupos de investigación interdisciplinarios que estén interesados en temas de fraude
5	**Usos previstos**
	Los usos previstos para esta red de ontologías serán: o Uso investigativo o pedagógico por parte de organismos científicos interesados en integración de ontologías de fraude. o Uso en entidades bancarias que requieran un punto de partida en integración de datos relacionados con tarjetas de crédito y análisis de riesgo de fraude.
6	**Requerimientos**
	6.1. Requerimientos no funcionales
	o La ontología debe ser modular. o La ontología presentará comentarios de cada elemento que la conforma en su definición en idioma español

	6.2. Requerimientos funcionales. Preguntas de competencia
	○ ¿Con qué oportunidad (número de transacciones) se puede detectar un fraude? ○ ¿Cuáles son las fechas de mayor fraude? ○ ¿Cuál es el *top* de regiones desde donde se realizan más ataques de fraude? ○ ¿Cuáles segmentos de tarjetas que más atacan con mayor frecuencia y por qué montos?

- El Escenario 2 de la mencionada metodología se refiere a la reutilización y rediseño de recursos no ontológicos (NORs). Este escenario se tuvo en cuenta, ya que se tiene como insumo la información de transacciones de tarjeta de crédito extraídas a partir de la base de transacciones de los sistemas autorizadores de tarjeta de crédito y la información de catálogos de valores (significado de valores numéricos o valores resumen) de los campos de las transacciones de tarjeta de crédito. Estos recursos no ontológicos fueron proporcionados por una entidad bancaria. A continuación, se indican los recursos no ontológicos con los que se incorporan a la red de ontologías:

  - Registro TC: Para la construcción de la red de ontologías, se hizo descomposición de la información que contiene una transacción de tarjeta crédito, en la cual viajan los valores que detallan la misma. Entre los campos que se encuentran en este registro, se tienen como ejemplo: número de transacción, fecha, hora, monto, región, nombre del comercio, entre otros.

  - Percentiles: esta fuente de información contiene los valores de los percentiles 95 y 75 basados en los montos de las transacciones de compra para cada una de las tarjetas de crédito. Los campos que conforman esta fuente son tarjeta, p95 y p75.

  Una vez se contó con las definiciones de los recursos no ontológicos con la información que proviene de una transacción de tarjeta de crédito y percentiles, se procedió a realizar un modelado inicial en la herramienta seleccionada Protégé [52] tal como se muestra en la Figura 8.

  Como producto del Escenario 2 de recursos no ontológicos, se crearon las clases asociadas para las 5 alertas propuestas en el sistema experto, como son *AlertaSuperaMontoTrx*, *AlertaSuperaAVGMontoDiario*, *AlertaPrimeraVezMCC*. *AlertaVelocidadTarjeta* y *AlertaAtaque7d*, igualmente se crearon las clases de *Avance*, *Comercio*, *Fraude* y *Tarjeta_Ubicación*.

De igual forma con base en el escenario 2, se crearon todos los Predicados (*Object Properties*) que permitirán la correlación entre las Clases y los *Data Properties* relacionados en la figura 8 y que en la Figura 11 aparecen como cajas blancas con borde verde, donde cada *Data Property* tiene la equivalencia a un campo de la transacción de tarjeta de crédito. Entre algunos de los *Data Properties* creados, se tienen: *numero_transacción*, *monto*, *fecha_transaccion*, *hora_transaccion*, etc.

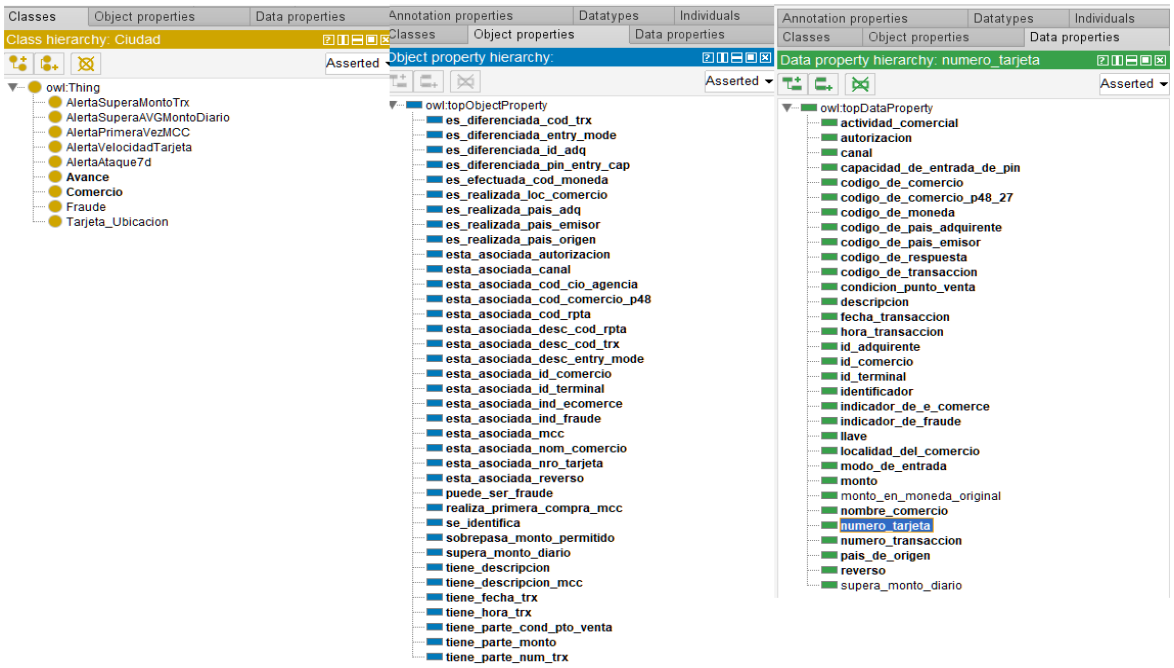


Figura 10. Modelo con NOR de la ontología en Protégé

- Con respecto al Escenario 3 “reutilización de los recursos ontológicos”, se procede a determinar los recursos ontológicos o declaraciones ontológicas que van a ser reutilizadas en el proceso de construcción de la red de ontologías. Esto se realiza a partir del estudio del estado del arte, donde se identifican diversos trabajos que proponen recursos ontológicos para la detección de fraude en distintas áreas interdisciplinarias, tales como: gobierno, ecommerce, etc. A partir de estos recursos se identifican diversas clases que se pueden ser incorporados en la red de ontologías propuesta, los cuales se detallan a continuación:

  Específicamente de la ontología Visitor_Behaviour se identifican 8 clases que se incorporan en la red desarrollada. Estos elementos son: *City*, *Successful_transaction*, *Failed _transaction*, *Device*, *Credit_card*, *Location*, *Transaction* y *Purchase*.

Para la ontología OBMO Ontology, se reutilizan la clase *Account* y de la ontología FIBO se reutilizan las clases *Currency* y *Country*.

- Con base en el Escenario 9 "Localización de recursos ontológicos", en el cual se hace adaptación a otros idiomas de los recursos ontológicos. A partir de los recursos reutilizados se propone traducirlas para añadir elementos de localización a la red de ontologías propuesta, los cuales se detallan a continuación.

  En Tabla 4 se detallan las traducciones 8 clases de la ontología Visitor_Behaviour, de la siguiente manera:

  - Se agrega en la sección de anotaciones la etiqueta *rdfs:label* en lenguaje español a *Ciudad* para el recurso *City*. Igualmente se incluye la etiqueta *rdfs:comment* en lenguaje español describiendo al campo como *"Ciudad origen de la transacción".*

  - En la sección de anotaciones la etiqueta *rdfs:label* en lenguaje español a *Transacción_Exitosa* para el recurso *Successful_transaction*. Igualmente se incluye en la etiqueta *rdfs:comment* en lenguaje español el mismo comentario.

  - Igual que en los pasos anteriores, se agregan anotaciones de etiqueta *rdfs:label* en lenguaje español a *Transacción_Fallida* para el recurso *Failed _transaction*. Igualmente se incluye en la etiqueta *rdfs:comment* en lenguaje español el comentario *"Transacción que no se logra completar".*

  - Se incluye en la sección de anotaciones la etiqueta *rdfs:label* en lenguaje español a *Dispositivo* para el recurso *Device*, incluyendo la etiqueta *rdfs:comment* en lenguaje español como *"Dispositivo desde el cual se realiza la transacción con Tarjea de Crédito".*

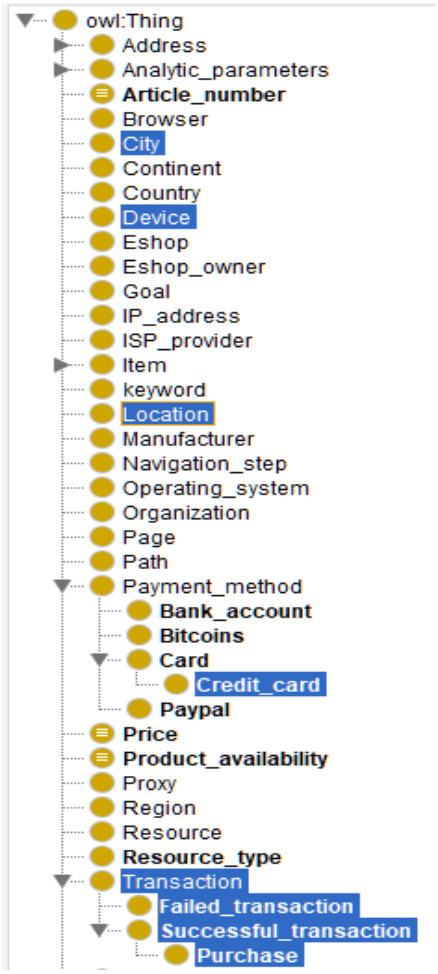
  - *Credit_card:* se modifica el recurso incluyendo las anotaciones *rdfs:label* en lenguaje español a *Tarjeta_Crédito*, luego se etiqueta con *rdfs:comment* en lenguaje español como *"Tarjeta de crédito con la cual se realiza la transacción.".*

  - *Location:* se incluyeron anotaciones como *rdfs:label* en lenguaje español a *Ubicación,* donde el valor de *rdfs:comment* es "Ubicación geográfica de la transacción."

  - Se incluye en la sección de anotaciones la etiqueta *rdfs:label* en lenguaje español a *Transaction* quedando como *Transacción,* igualmente se incluye la anotación *rdfs:comment* para indicar lo siguiente "Transacción realizada de compra o avance".

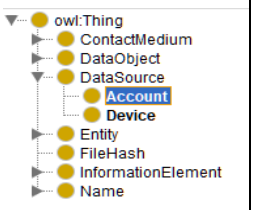
- Se agrega en la sección de anotaciones la etiqueta *rdfs:label* en lenguaje español a *Purchase* de forma que quede también *Compra*, luego de esto se incluye *rdfs:comment* en lenguaje español como "Transacción de compra de un producto o servicio con Tarjeta de Crédito."

Tabla 4. Traducción de 8 clases de la ontología *Visitor_Behaviour*

Antes	Después
	

Para la ontología OBMO Ontology, se puede observar en la Tabla 5 la traducción de la clase *Account*, para el cual se agrega en la sección de anotaciones la etiqueta *rdfs:label* en lenguaje español a *Account* de forma que quede también *Cuenta*, luego de esto se incluye *rdfs:comment* en lenguaje español como "Cuenta asociada a la transacción"

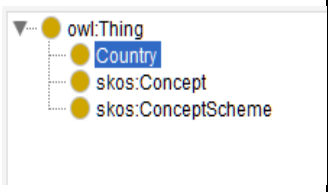
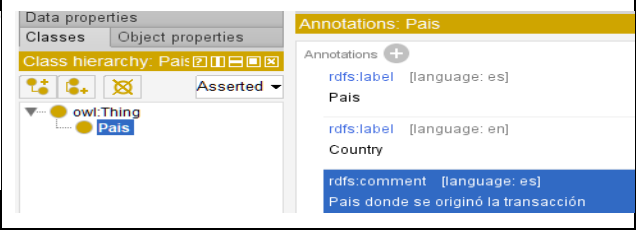
Tabla 5. Traducción de la clase *Account* ontología OBMO Ontology

Antes	Después
	

Al igual que en los anteriores casos, para la ontología FIBO, se realizó el proceso de traducción observado en la Tabla 6 para las clases *Currency* y *Country*

- Se incluye en la sección de anotaciones la etiqueta *rdfs:label* en lenguaje español a *Country* quedando como *País*, igualmente se incluye la anotación *rdfs:comment* para indicar lo siguiente "País donde se realiza la transacción".

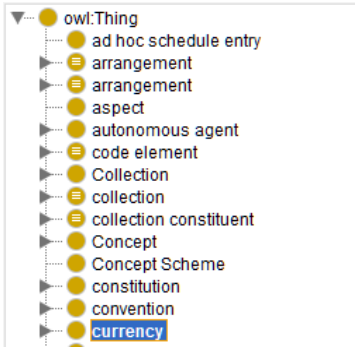
Tabla 6. Traducción de la clase *Country* de ontología FIBO

Antes	Después
	

Finalmente, para la ontología FIBO, se realizó el proceso de traducción observado en la Tabla 7 para *Country*

- En la sección de anotaciones la etiqueta *rdfs:label* en lenguaje español a *Moneda* para el recurso *Currency*. Igualmente se incluye en la etiqueta *rdfs:comment* en lenguaje español el mismo comentario "Moneda con la cual se realiza la transacción".

Tabla 7. Traducción de la clase *Country* de la ontología FIBO

Antes	Después
	

Tras la aplicación de los diferentes escenarios de la metodología NeOn, se produce como resultado el modelo de la red de ontologías presentado en la Figura 9, donde se utilizan tanto recursos de conocimiento ontológicos como no ontológicos.

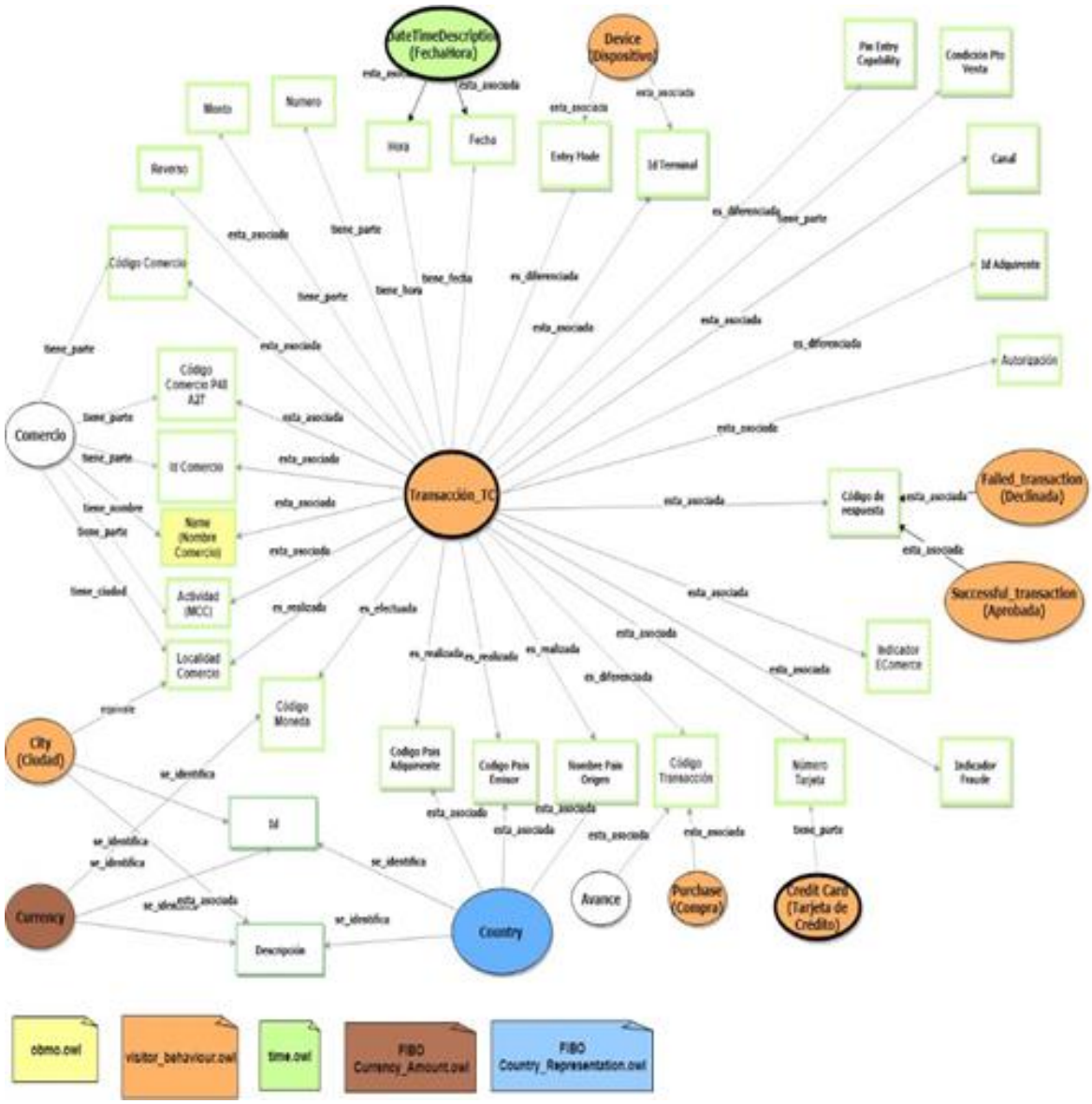


Figura 11. Red de ontologías del sistema propuesta

5.1.2.3. Generación de RDF

5.1.2.3.1. Proceso de carga de datos

Dentro de la fase de Generación de RDF de la metodología de Linked Data considerada en este trabajo, se tuvo en cuenta las fuentes de datos de información de percentiles de los montos de compra para cada una de las tarjetas de crédito y de información de transacciones de tarjeta de crédito. Para esto se desarrolló un componente de carga de archivos para este proyecto, construido en el lenguaje de programación Java con el IDE Eclipse.

En primera instancia se utilizó el componente para la carga del archivo de percentiles, haciendo un recorrido línea a línea para almacenar cada registro como un objeto de tipo *Percentil* e inmediatamente se almacena como tripletas en el repositorio RDF. La información de los percentiles será el insumo para las reglas que se describirán en el capítulo 5.1.2.4. Inferencia.

Luego del almacenamiento de los percentiles, se utiliza el componente para la carga de la información del archivo de transacciones, haciendo un recorrido línea a línea para almacenar cada transacción como un objeto de tipo *Transaction*, donde por cada carácter de separación ("|") se encuentran los valores de los campos de la transacción descritos en la sección 5.1.2.1. Información transaccional. Una vez se almacena cada transacción en un objeto de tipo *Transaction*, queda disponible para ser transportado a los demás componentes de los módulos del sistema experto.

El objeto *Transaction* mostrado en la Figura 12 posee funciones miembro públicas (*getters* y *setters*) para el acceso a la información de los campos de la transacción que en este caso son atributos de la clase. La salida del componente de carga va a ser un objeto *Transaction* el cual será transportado a la siguiente etapa del proceso generación de RDF en el cual se hace la adaptación a tripletas RDF a ser persistidas.

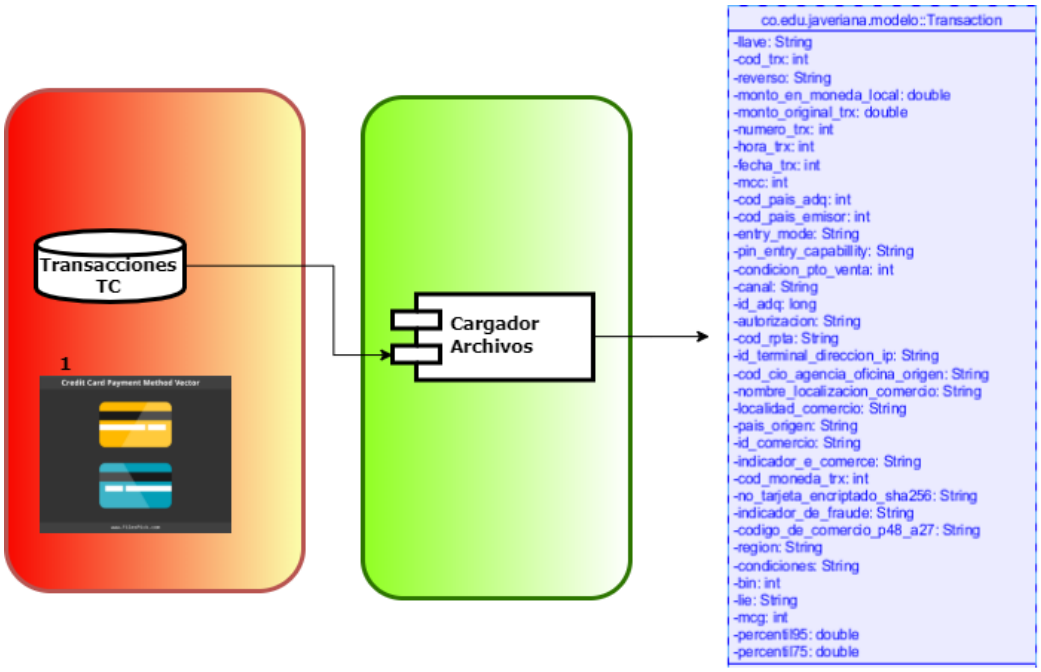


Figura 12. Objeto t*ransaction* para carga de transacciones

5.1.2.3.2. Proceso de generación de RDF

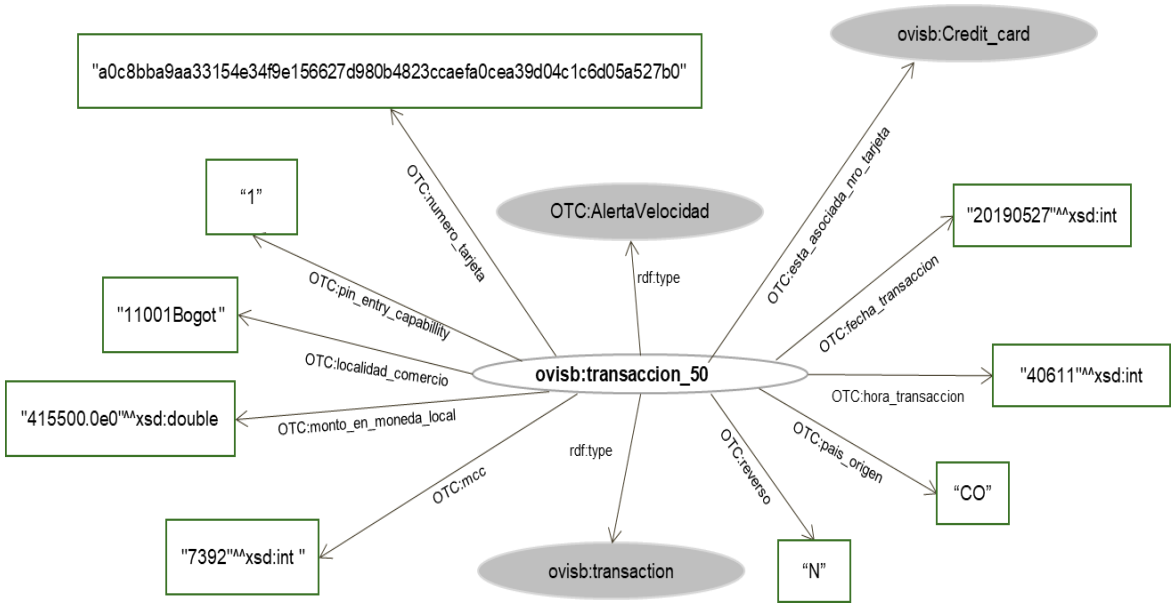
El componente de generación se diseñó y construyó para lograr convertir los objetos *Transaction* y *Percentil*, obtenidos durante el proceso de carga y descritos en la sección 5.1.2.3.1, en tripletas RDF. La creación de las tripletas se realiza mediante el uso del API RDF que ofrece Apache Jena [53], por medio de la creación de los objetos siguientes:

- *Resource:* este objeto es utilizado para la creación del **sujeto**
- *Property:* este objeto se crea para las relaciones que en el modelo de la ontología de transacciones de tarjeta de crédito fueron declaradas como **predicados**.
- *Statement*: objeto con el cual se crea una **tripleta** RDF.
- *createStatement / createLiteralStatement:* función que permite asociar la relación sujeto, predicado, **objeto**. Donde **objeto** puede ser un literal o una instancia de clase (*Resource*).

Para el caso del sistema experto y con el fin de dar claridad en el proceso, se muestra a continuación un ejemplo de la creación de una tripleta con sentencias Java del API RDF, basada en la información transaccional mediante la declaración y asignación de valores a los objetos *Resource*, *Property y Statement.*

```
/*Resource crea un Sujeto*/
Resource transaccion = JenaManager.getModel().createResource(transaccionUri);
/*Property crea un Predicado*/
Property numero_tarjeta = JenaManager.getModel().getProperty(Principal.otc + "numero_tarjeta");
/*Statement crea una tripleta RDF, t.getNumero_trx() contiene el número de la tarjeta en este caso*/
Statement st = JenaManager.getModel().createLiteralStatement(transaccion, numero_transaccion,
t.getNumero_trx());
```

En la Figura 13 que se muestra a continuación, se presenta un grafo para una transacción creada y un fragmento del RDF descrito en los pasos anteriores.



```
@prefix :      <http://www.javeriana.edu.co/ontologies/transacciones_tc#> .
@prefix ovisb: <http://www.sme-ecompass.eu/ontologies/visitor_behaviour_jqrs.owl#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .
@prefix swrlb: <http://www.w3.org/2003/11/swrlb#> .
@prefix swrl:  <http://www.w3.org/2003/11/swrl#> .
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
@prefix terms: <http://purl.org/dc/terms/> .
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xml:   <http://www.w3.org/XML/1998/namespace> .
@prefix ocurc: <http://www.javeriana.edu.co/ontologies/CurrencyCodes_JGRS#> .
@prefix otc:   <http://www.javeriana.edu.co/ontologies/transacciones_tc#> .
ovisb:transaccion_2  a                  "<http://www.javeriana.edu.co/ontologies/transacciones_tc#AlertaVelocidad" ;
        otc:bin                         "441080"^^xsd:int ;
        otc:canal                       "INT" ;
        otc:cod_cio_agencia_oficina_origen
                "015017353      " ;
        otc:codigo_de_comercio_p48_a27  "015017353          00000003" ;
        otc:codigo_pais_adquirente      "0"^^xsd:int ;
        otc:codigo_pais_emisor          "0"^^xsd:int ;
        otc:codigo_respuesta            "0" ;
        otc:codigo_transaccion          "2"^^xsd:int ;
        otc:condicion_punto_venta       "59"^^xsd:int ;
        otc:entry_mode                  "01             " ;
        otc:esta_asociada_nro_tarjeta   ovisb:Credit_card ;
        otc:fecha_transaccion           "20190527"^^xsd:int ;
        otc:hora_transaccion            "40611"^^xsd:int ;
        otc:id_adquirente               "10000000090"^^xsd:long ;
        otc:id_comercio                 "010001501730000" ;
        otc:id_terminal_direccion_ip    "00024638" ;
        otc:indicador_e_comerce         "S" ;
        otc:indicador_fraude            "F" ;
        otc:lie                         "E" ;
        otc:localidad_comercio          "11001Bogot    " ;
        otc:mcc                         "7392"^^xsd:int ;
        otc:mcg                         "800"^^xsd:int ;
        otc:monto_en_moneda_local       415500.0e0 ;
        otc:monto_en_moneda_original    0.0e0 ;
        otc:numero_tarjeta              "a0c8bba9aa33154e34f9e156627d980b4823ccaefa0cea39d04c1c6d05a527b0" ;
        otc:numero_transaccion          "738"^^xsd:int ;
        otc:pais_origen                 "CO" ;
        otc:pin_entry_capabillity       "0              " ;
        otc:region                      "CO             " ;
        otc:reverso                     "N" .

ovisb:transaccion_1  a                  "<http://www.javeriana.edu.co/ontologies/transacciones_tc#AlertaSuperaAVGMontoDiario"
        otc:bin                         "441080"^^xsd:int ;
        otc:canal                       "INT" ;
        otc:cod_cio_agencia_oficina_origen
                "015017353      " ;
```

Figura 13. Fragmento del RDF generado

5.1.2.4. Publicación

Como tercer paso de la arquitectura se considera la publicación a partir de consultas de la información de tripletas RDF. Para lograr la publicación, se hizo uso de TDB como repositorio RDF de alto rendimiento, el cual es provisto por Apache Jena [53] puntualmente por el API *Store*. Se escogió a TDB como repositorio RDF ya que se recomienda para hacer uso en aplicaciones que manejen información transaccional [53], como es el caso de este proyecto.

Adicional al API *Store*, se incluyó como parte del proceso de publicación el uso del servidor Fuseki suministrado por Apache Jena, para ser utilizado como un SPARQL *endpoint* a partir del cual realizan las consultas de la información de las tripletas que se encuentran en el repositorio RDF de TDB. En la figura 14 se muestra el proceso de una consulta, la cual parte desde la interfaz gráfica del servidor Fuseki, accediendo desde un navegador Web a la URL *localhost:3030*; donde se ingresan las sentencias SPARQL de la información a visualizar, una vez se ejecuta la consulta se muestra la salida con las tripletas RDF coincidentes.

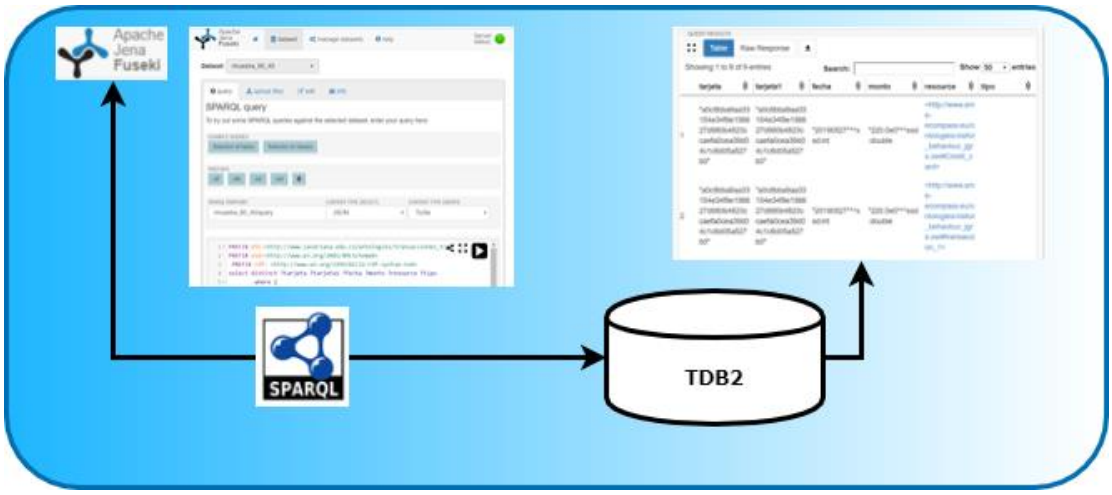


Figura 14. Proceso de publicación al almacén RDF TDB

5.1.2.5. Inferencia

El proceso de inferencia incluido en este proyecto y que hace parte del sistema experto, no está incluido dentro de las fases propuestas por la metodología de Linked Data, por lo cual es una fase propia de esta propuesta. En el proceso de inferencia del sistema experto, con base en las reglas de alertamiento de fraude transaccional de tarjeta de crédito, se crearon 5 reglas de detección de fraude, que abordan tipos de ataques estándar sobre los productos de tarjetas de crédito. Entre los tipos de ataques estándar sobre los que actúan las reglas creadas se encuentran: ataques de velocidad, ataques por montos superiores a montos definidos (promedio diario y promedio histórico), ataques en comercios no comunes de los clientes y ataques por patrones de tiempo, en este caso semanales. Las reglas creadas se apoyan en la inferencia sobre el motor de inferencia de Apache Jena y son descritas a continuación:

- **Regla 1 AlertaSuperaMontoTrx:** Esta alerta se creó con el fin de encontrar las transacciones que superan un monto establecido individualmente por cada tarjeta de crédito. El monto individual que se definió fue el percentil 95, calculado a partir de los montos de las transacciones históricas realizadas sobre cada tarjeta. Por lo tanto, la regla genera la alerta si el monto de la transacción evaluada supera el percentil 95.

  Los elementos de la red de ontologías que hacen parte de la regla son: clase *transacción* que se asocia al tipo de clase *AlertaSuperaMontoTrx* cuando la regla cumple, los *object properties fecha_transacción* y *monto_en_moneda_local* de la clase *transacción* y, finalmente, el *data property p95* que corresponde a una clase *tarjeta*, el cual se usa para comparar contra el valor de *monto_en_moneda_local*. A continuación, se detalla el código de construcción de la regla y los parámetros tenidos en cuenta:

```
parametros.get(0)= Transacción evaluada
parametros.get(1)= Valor del Percentil 95 para la tarjeta evaluada
parametros.get(2)= Fecha de la transacción


AlertaSuperaMontoTrx ="(?s ?p ?o), "
       + "(?s equal(?s," + parametros.get(0) + ")) , "
       +"(?s <http://www.javeriana.edu.co/ontologies/transacciones_tc#fe-
cha_transaccion> ?fecha_transaccion) , "
       + "(?o equal(?fecha_transaccion," + parametros.get(2) + ")) , "
       + "(?s <http://www.javeriana.edu.co/ontologies/transac-
       ciones_tc#monto_en_moneda_local> ?p95) , "
       + "greaterThan(?p95," + parametros.get(1) + ") "
       + "-> (?s  rdf:type  <http://www.javeriana.edu.co/ontologies/transac-
       ciones_tc#AlertaSuperaMontoTrx>). ";
```

- **Regla 2 AlertaSuperaAVGMontoDiario. S**e creó esta alerta con el fin de encontrar las transacciones que superan un monto promedio diario por cada tarjeta de crédito. Para lo cual como insumo de la regla se hace una consulta de las transacciones previas del día evaluado y se les calcula el valor promedio de las compras del día. Si el valor del monto promedio diario es superado por el valor de la transacción evaluada, se genera la alerta.

  Los elementos de la red de ontologías que conforman la regla son: clase *transacción* que se asocia al tipo de clase *AlertaSuperaAVGMontoDiario* cuando la regla cumple, los *object properties fecha_transacción* y *monto_en_moneda_local*. El código de creación de la regla y los parámetros tenidos en cuenta, se detallan a continuación:

```
parametros.get(0)= Transacción evaluada
parametros.get(1)= Fecha de la transacción
AVG_monto= Valor promedio diario de las transacciones previas a la evaluada.
```

```
Regla2 = "(?s ?p ?o), " + "(?s equal(?s," + parametros.get(0) + ")) , "
                              + "(?s <http://www.javeriana.edu.co/ontolo-
gies/transacciones_tc#fecha_transaccion> ?fecha_transaccion) , "
                              + "(?o equal(?fecha_transaccion," + parame-
tros.get(1) + ")) , "
                              + "(?s <http://www.javeriana.edu.co/ontolo-
gies/transacciones_tc#monto_en_moneda_local> ?montoAVGDiario) , "
                              + "greaterThan(?montoAVGDiario," + AVG_monto + ") "

                              + "-> (?s  rdf:type  <http://www.javeriana.edu.co/on-
tologies/transacciones_tc#AlertaSuperaAVGMontoDiario>). ";
```

- **Regla 3 AlertaPrimeraVezMCC.** Esta alerta de habitualidad se creó con el fin de encontrar las transacciones que se hacen en un comercio (*mcc*) donde el cliente nunca ha comprado. Como insumo de la regla, se hace una consulta sobre el universo de transacciones de la tarjeta de crédito, devolviendo un contador de ocurrencias del *mcc* de la transacción actual, si el resultado es 0 se procede a generar la alerta.

  Los elementos de la red de ontologías que hacen parte de la regla son: clase *transacción* que se asocia al tipo de clase *AlertaPrimeraVezMCC* cuando la regla cumple y el *object property numero_tarjeta* utilizado para conocer a qué transacción se le debe asociar la alerta. A continuación, se detalla el código de construcción de la regla y los parámetros tenidos en cuenta:

```
parametros.get(0)= Transacción evaluada
parametros.get(1)= Tarjeta de crédito evaluada
```

```
Regla3 = "(?s ?p ?o), " + "(?s equal(?s," + parametros.get(0) + ")) , (?s
<http://www.javeriana.edu.co/ontologies/transacciones_tc#numero_tarjeta> " + par-
ametros.get(1) + " ) -> (?s  rdf:type  <http://www.javeriana.edu.co/ontolo-
gies/transacciones_tc#AlertaPrimeraVezMCC>).";
```

- **Regla 4 AlertaVelocidad.** Esta alerta se construyó con el fin de encontrar las transacciones que se realizan de forma consecutiva sobre la misma tarjeta de crédito, en un intervalo de tiempo inferior al estándar en una operación de compra con tarjeta de crédito. Este comportamiento, se obtiene a partir de una consulta sobre las transacciones anteriores a la evaluada de la tarjeta de crédito, colocando como parámetro de búsqueda la variable hora de la transacción y fecha de la transacción. Si el resultado de esta consulta resulta ser mayor que 1 la transacción generar la alerta.

  Los elementos de la red de ontologías, que hacen parte de la regla son: clase *transacción* que se asocia al tipo de clase *AlertaVelocidadTarjeta* cuando la regla cumple y los *object properties fecha_transaccion* y *hora_transaccion* utilizados para conocer a cuál transacción se le debe asociar la alerta. El código con el que se creó la regla y los parámetros tenidos en cuenta se muestran a continuación:

```
parametros.get(0)= Transacción evaluada
parametros.get(1)= Tarjeta de crédito evaluada
parametros.get(2)= Fecha de la transacción
parametros.get(3)= Hora de la transacción
```

```
Regla4 = "(?s ?p ?o), " + "(?s equal(?s," + parametros.get(0) + ")) , (?s
<http://www.javeriana.edu.co/ontologies/transacciones_tc#numero_tarjeta> " + para-
metros.get(1) + " ) , (?s <http://www.javeriana.edu.co/ontologies/transaccio-
nes_tc#fecha_transaccion> ?fecha_transaccion) , (?o equal(?fecha_transaccion," +
parametros.get(2) + ")), (?s <http://www.javeriana.edu.co/ontologies/transaccio-
nes_tc#hora_transaccion> ?hora_transaccion) , (?o equal(?hora_transaccion," + pa-
rametros.get(3) + ")) -> (?s  rdf:type  <http://www.javeriana.edu.co/ontolo-
gies/transacciones_tc#AlertaVelocidadTarjeta>).";
```

- **Regla 5 AlertaAtaque7d.** La alerta *AlertaAtaque7d* tiene como fin detectar ataques semanales, basados en un patrón que se repite en las mismas horas, comercios, tarjetas, entre otras variables. Este patrón de comportamiento se obtiene a partir de una consulta sobre las transacciones en la semana anterior (fecha actual – 7 días) a la evaluada de la tarjeta de crédito. Si el resultado de esta consulta resulta ser mayor que 1 indica que la transacción genera alerta.

  Los elementos de la red de ontologías, que hacen parte de la regla son: clase *transacción* que se asocia al tipo de clase *AlertaAtaque7d* cuando la regla cumple y el *object property numero_tarjeta* utilizado para conocer a cuál transacción se le debe asociar la alerta. El código con el que se creó la regla y los parámetros tenidos en cuenta se muestran a continuación:

```
parametros.get(0)= Transacción evaluada
parametros.get(1)= Tarjeta de crédito evaluada
```

```
Regla5 = "(?s ?p ?o), " + "(?s equal(?s," + parametros.get(0) + ")) , (?s
<http://www.javeriana.edu.co/ontologies/transacciones_tc#numero_tarjeta> " + par-
ametros.get(1) + " ) -> (?s  rdf:type  <http://www.javeriana.edu.co/ontolo-
gies/transacciones_tc# AlertaAtaque7d>).";
```

6. VALIDACIÓN

En este capítulo se da detalle del proceso de validación realizado para medir los resultados de las reglas de detección de fraude implementadas en el sistema experto. La validación se realizó a partir de los indicadores *Precision, Recall y F-Measure* utilizando como referencia un *gold standard* provisto por la entidad financiera, el cual contiene las transacciones marcadas como fraudes y las que no lo son.

Para las validaciones se contó con 3 muestras de transacciones proporcionadas por una entidad financiera, donde cada una de ellas cuenta con un total de 10.000 transacciones, donde: i) la primera muestra presentaba un 60% de transacciones marcadas como no fraude y 40% marcadas como fraude, ii) la segunda muestra presenta un porcentaje de 70% transacciones no fraudulentas y 30% marcadas como fraude y, finalmente, iii) la tercera muestra cuenta con un 80% de transacciones no fraudulentas y un 20% marcadas como fraude. Todas las muestras pertenecen al periodo de tiempo comprendido entre el 27 de mayo y 05 de agosto de 2019, periodo donde se produjeron gran número de fraudes según información de la entidad. Cabe mencionar adicional a lo anterior, que se observó un solape no significativo entre las muestras, debido a que en el muestreo no se descartaron las transacciones comunes entre sí.

Para poder abordar con mayor claridad los cálculos de los indicadores de validación que se presentan en este capítulo, se describen algunos términos que se mencionan en la formulación de los indicadores, tales como:

- *Verdaderos Positivos* (TP). Este término [57] indica aquellos valores positivos identificados correctamente, lo que significa que la transacción en el *gold standard* se marcó como fraude y el resultado de la ejecución de la regla también es un fraude.

- *Verdaderos negativos* (TN). Los verdaderos negativos [57] son aquellos valores negativos que se predijeron correctamente, indicando que la transacción se refleja en el *gold standard* como un “no fraude” y el resultado de las reglas construido también es un “no fraude”.

- *Falsos Positivos* (FP). Este tipo de valores [57] son los que se obtienen cuando en el *gold standard* se recoge un “no fraude” y las reglas construidas dan como resultado un fraude.

- *Falsos Negativos* (FN). Los valores falsos negativos [57] ocurren cuando el indicador de fraude de la transacción del *gold standard* es recogida como fraude, pero las reglas del modelo indicaron que es un “no fraude”.

De acuerdo con las definiciones anteriores, se formula el indicador *Precision* [57], como la relación entre las observaciones positivas predichas correctamente y el total de observaciones positivas predichas. La fórmula del indicador *Precision* se presenta a continuación:

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Al aplicar el indicador *Precision* sobre los resultados de las reglas definidas, se obtienen los valores presentados en la Tabla 8, donde se observa un comportamiento de disminución en el indicador de *Precision* entre la muestra 1 (60% - 40%) y la muestra 2 (70% - 30%) que puede deberse a la disminución de las coincidencias de transacciones que cumplen alguno de los patrones de fraude, pero que se acentúa más entre la muestra 2 (70% - 30%) y muestra 3 (80% - 20%), lo que confirma esta afirmación, ya que al presentarse menos comportamientos similares y volverse ataques aislados la predicción de las reglas baja considerablemente. Adicional a lo anterior, cabe aclarar que las reglas detectan comportamientos de fraude a partir del segundo ataque.

Tratando de llevar a cabo una comparativa con las cifras de la entidad financiera, se muestra que para la *AlertaAtaque7d* existen similitudes con las reglas de *Patrón de Fraude 1 y Patrón de Fraude 2*, mostrando que la precisión en el caso de las reglas de patrones es en promedio de 0,06 para el universo de fraudes alertados y para la regla de *AlertaAtaque7d* para las tres muestras es de 0,29 lo que indica que se tiene una efectividad casi de 5 veces en la regla propuesta.

Para el caso de la regla *AlertaSuperaAVGMontoDiario*, se hace la comparación con la regla llamada *Monto Acumulado Alto*, encontrando que, aunque la precisión de la regla propuesta no es muy alta, es realmente significativa con la precisión de 0,018 que tiene la regla de la entidad financiera, donde se mejora la precisión en 0,122.

Comparando la regla *AlertaSuperaMontoTrx* propuesta que tiene una precisión de 0,24 y la regla de la entidad financiera llamada *Monto Alto*, se observa que la precisión mejora considerablemente en 0,22, garantizando un mejor desempeño de acuerdo con su formulación. Finalmente, para el caso de la alerta creada llamada *AlertaVelocidad*, se encuentran dos reglas que propone la entidad financiera llamadas *Velocidad 1 y Velocidad* 2 creadas para cubrir ataques de velocidad, siendo muy baja la precisión y demostrando que la regla propuesta propone una efectividad 30 veces mejor a las utilizadas por la entidad financiera.

Tabla 8. Indicador *Precision* aplicado a resultado de reglas

ALERTA	MUESTRA	PRECISION	PROMEDIO		ALERTA ENTIDAD	PRECISION ENTIDAD	PROMEDIO ENTIDAD
AlertaAtaque7d	60_40	0,38	0,29		Patrón de Fraude 1	0,1016	0,06
	70_30	0,30			Patrón de Fraude 2	0,0122	
	80_20	0,19					
AlertaSuperaAVGMontoDiario	60_40	0,23	0,14		Monto Acumulado Alto	0,0180	0,018
	70_30	0,15					
	80_20	0,06					
AlertaSuperaMontoTrx	60_40	0,33	0,24		Monto Alto	0,0156	0,02
	70_30	0,29					
	80_20	0,09					
AlertaVelocidad	60_40	0,41	0,30		Velocidad 1	0,0084	0,01
	70_30	0,35			Velocidad 2	0,0074	
	80_20	0,14					

Continuando con la validación, se presenta el indicador *Recall* [57], el cual permite calcular la sensibilidad de la predicción y que es formulado como la proporción de observaciones positivas predichas correctamente con respecto a todas las observaciones de la realidad. La fórmula con la que se calcula el indicador de *Recall* es el siguiente:

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

De acuerdo con la definición del indicador *Recall* y al no conocer los resultados de falsos negativos de cada una de las reglas aplicadas por la entidad financiera por temas de confidencialidad en su modelo de fraude, el indicador *Recall* se aplicó de forma global. De esta manera, la comparación se realiza con base en el resultado de la inferencia del universo de reglas creado contra el total de transacciones marcadas en el *gold standard.*

Según lo anterior, los resultados del indicador *Recall* aplicado a cada una de las muestras se presentan en la Tabla 9, donde en comparación con el indicador de *Precision*, el indicador *Recall* no presenta variaciones significativas entre muestras al aplicarse globalmente, siendo la variación más fuerte de 0,02 entre la muestra 2 (70%-30%) y la muestra 3 (80%-20%).

Tabla 9. Indicador *Recall* aplicado por muestra

MUESTRA	RECALL	PROMEDIO
60_40	0,28	0,28
70_30	0,27	
80_20	0,29	

Como apoyo a los anteriores indicadores y con el fin de validar la relación entre *Precision* y *Recall*, se aplica el indicador *F-Measure* [57], el cual tiene en cuenta tanto los falsos positivos como los falsos negativos y es formulado de la siguiente manera:

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

Al igual que ocurre para el cálculo del indicador *Recall*, el indicador *F-Measure* requiere de conocer los resultados de las reglas que se aplicaron por la entidad financiera con el fin de encontrar el valor de los falsos negativos al ser comparados con las reglas creadas, valor que como se explicó anteriormente no se pudo obtener. Según esto, el indicador *F-Measure* fue posible aplicarlo de forma global, con base en el resultado de la inferencia del universo de reglas creado contra el total de transacciones marcadas en el *gold standard.*

De acuerdo con lo anterior, los resultados del indicador *F-Measure* aplicado a cada una de las muestras, se presentan en la Tabla 10, donde ocurre algo similar en comparación con el indicador de *Recall*, el cual no presenta variaciones significativas entre muestras, la variación más fuerte es de 0,02 entre la muestra 2 (70_30) y la muestra 3 (80_20).

Tabla 10. Indicador *F-Measure* aplicado por muestra

MUESTRA	MEDIDA F	PROMEDIO
60_40	0,44	0,44
70_30	0,43	
80_20	0,45	

Adicional a los indicadores mostrados anteriormente, se presentan las tablas 11 y 12 que contienen dos indicadores que utiliza la entidad financiera para conocer la efectividad de las

reglas creadas y que se consideraron dentro de este proyecto como un plus a los indicadores ya evaluados.

De acuerdo con lo observado en la tabla 11, la regla que presenta un mejor comportamiento entre muestras es la regla de *AlertaAtaque7d*, donde a pesar de disminuir los casos de fraude va aumentando en la efectividad de monto alertado. Para las demás reglas, ocurre un comportamiento de disminución a través de las muestras, siendo la más significativa la regla de *AlertaSuperaMontoTrx* que de la muestra 2 a la 3 disminuye en la efectividad por monto alertado en aproximadamente 100 millones de pesos.

Tabla 11. Montos alertados por reglas construidas

ALERTA	MUESTRA	MONTO ALERTADO	PROMEDIO
AlertaAtaque7d	60_40	$ 96.231.939,6	$ 86.268.113,10
	70_30	$ 56.351.568,4	
	80_20	$ 106.220.831,3	
AlertaSuperaAVGMontoDiario	60_40	$ 76.785.530,2	$ 44.756.638,83
	70_30	$ 32.385.855,2	
	80_20	$ 25.098.531,1	
AlertaSuperaMontoTrx	60_40	$ 187.235.155,6	$ 135.018.035,27
	70_30	$ 158.996.985,1	
	80_20	$ 58.821.965,1	
AlertaVelocidad	60_40	$ 79.573.293,5	$ 67.762.639,00
	70_30	$ 67.431.140,2	
	80_20	$ 56.283.483,3	

Según lo observado en la tabla 12, la regla que presenta un mejor Falso Positivo y que se mantiene al cambiar entre las muestras es la regla de *AlertaAtaque7d*. También se observa que las demás reglas presentan una variación significativa entre las muestras 2 y 3, como es el caso de las reglas de *AlertaSuperaAVGMontoDiario*, *AlertaSuperaMontoTrx* y *AlertaVelocidad*. Este comportamiento puede deberse a que de una muestra a otra varían significativamente las ocurrencias de transacciones diarias, con las que se predice el alertamiento de estas, mostrando un aumento del indicador de falso positivo de más del doble, que indica una degradación de la efectividad de las reglas entre muestras.

Igualmente, de acuerdo con la tabla 12, se trata de llevar a cabo una comparativa entre las reglas de la entidad financiera y las propuestas por el sistema experto. Para el primer caso, la entidad financiera propone las reglas de *Patrón de Fraude 1 y Patrón de Fraude 2*, la cual es

comparada con la regla *AlertaAtaque7d* propuesta, mostrando alta efectividad en la regla propuesta que logra detectar en 4 transacciones alertadas 1 fraude y las reglas de la entidad financiera en promedio detectan por cada 55 alertas generadas 1 fraude.

Al hacer la comparación de la regla *AlertaSuperaAVGMontoDiario* propuesta, contra la regla de la entidad financiera llamada *Monto Acumulado Alto*, se observa que en la detección de la regla propuesta es de cada 10 alertas generadas se encuentra un fraude, lo cual es un buen indicador, comparado con la regla de la entidad financiera donde por cada 57 alertas solamente detecta un fraude.

Para el caso de la regla *AlertaSuperaMontoTrx* se observa que por cada 6 alertas generadas una es fraude y que la regla *Monto Alto* propuesta por la entidad financiera tiene un muy alto falso positivo ya que solo se detecta un fraude por cada 65 alertas. Por último, se evidencia el peor escenario de alertamiento de las reglas de la entidad financiera en las reglas *Velocidad 1 y Velocidad* 2 siendo el falso positivo de 129 alertas a 1 fraude, en su lugar la regla propuesta de *AlertaVelocidad* creada, tiene un falso positivo cercano a 4 alertas por 1 fraude detectado.

Tabla 12. Falso positivo por regla construida

ALERTA	MUESTRA	FALSO POSITIVO	PROMEDIO	ALERTA ENTIDAD	FALSO POSITIVO ENTIDAD	PROMEDIO ENTIDAD
AlertaAtaque7d	60_40	2,66	3,75	Patrón de Fraude 1	27,89	55,50
	70_30	3,30		Patrón de Fraude 2	83,12	
	80_20	5,29				
AlertaSuperaAVGMontoDiario	60_40	4,29	9,61	Monto Acumulado Alto	56,50	56,50
	70_30	6,89				
	80_20	17,66				
AlertaSuperaMontoTrx	60_40	2,99	5,67	Monto Alto	65,00	65,00
	70_30	3,43				
	80_20	10,59				
AlertaVelocidad	60_40	2,42	4,17	Velocidad 1	120,00	128,50
	70_30	2,87		Velocidad 2	137,00	
	80_20	7,21				

7. CONCLUSIONES, APORTES Y TRABAJO FUTURO

Conclusiones y aportes

El presente trabajo de grado en modalidad de profundización ha abordado el estudio y aplicación de las metodologías de la ciencia basada en el diseño, NeOn y Linked Data en cada una de sus fases y escenarios, así como la utilización de diversas tecnologías de la Web Semántica aplicadas a la detección de fraudes en tarjetas de crédito.

Se logra la creación de un sistema experto con aplicabilidad en el área de monitoreo de fraude para una entidad financiera, el cual cumple de forma completa, con los elementos exigidos por un sistema experto como son la base de conocimiento (repositorio RDF), componente de inferencia apoyado en reglas de la Web Semántica y el componente de interfaz (SPARQL *Endpoint*).

De la misma forma, a partir de este proyecto, se logra poner a disposición de la comunidad educativa, una red de ontologías que modela la información transaccional de tarjeta de crédito, base para futuras integraciones con otras redes ontológicas y fuentes de información que realimenten la red creada.

Adicionalmente, se logra desarrollar un proceso de integración semántica basada en los principios de Linked Data, conectando la información de las diferentes fuentes transaccionales de tarjeta de crédito y de información de apoyo para la detección de fraude, producto del cual es generado un RDF que permite ser utilizado como la base de conocimiento del sistema experto.

Haciendo uso del lenguaje de reglas de la Web Semántica (SWRL) fue posible realizar la creación de reglas de alertamiento de fraude de tarjeta de crédito genéricas, las cuales produjeron resultados de alertamiento, principalmente, a nivel de montos de las transacciones y a nivel de velocidad de transacciones.

Asimismo, se realizó una validación de los resultados del modelo propuesto incorporado en el sistema experto, comparado con la base de un *gold standard* que contenía los resultados de la marcación de fraude de la entidad financiera. Esta validación fue realizada utilizando las métricas *precision*, *recall* y *F-Measure*, obteniendo una mejor efectividad para los indicadores de falso positivo y *precisión* por cada una de las reglas de alertamiento construidas.

Trabajos futuros

Teniendo presente que muchas herramientas que se utilizaron para el desarrollo del proyecto no ofrecen soporte para cargas transaccionales altas que impliquen inferencia inmediata, se propone como trabajo futuro ahondar en propuestas para mejorar la estrategia de inferencia en modelos de conformes con los principios de Linked Data. El objetivo se centrará en que las propuestas de este contexto puedan tratar con grandes volúmenes de datos y de inferencias de forma simultánea.

Asimismo, se trabajará en la incorporación de nuevas fuentes de información en el sistema experto propuesto, con el objetivo de contar con una visión integral de los clientes, lo cual daría una gran ventaja para lograr la creación de reglas más efectivas y oportunas en el proceso de alerta de posibles fraudes.

De igual manera, se propone seguir generando nuevas reglas dentro del sistema experto propuesto, que permitan la detección no solo de ataques estándar, sino de ataques más especializados a tarjetas de crédito.

Finalmente, se propone añadir algoritmos de *machine learning* al sistema experto desarrollado, con el fin de lograr identificar patrones presentes en diversos y heterogéneos conjuntos de datos de transacciones de tarjetas de crédito, que permitan retroalimentar la base de conocimiento del sistema construido.

REFERENCIAS

[1] Reporte trimestral de inclusión financiera (s. f.). http://bancadelasoportunidades.gov.co/sites/default/files/2019-07/REPORTE%20TRIMESTRAL%20DE%20INCLUSI%C3%93N%20FINANCIERA_MAR2019.pdf. Recuperado 13 de noviembre de 2019

[2] Debit / Credit Card Fraud. (s. f.). https://criminal.findlaw.com/criminal-charges/credit-debit-card-fraud.html. Recuperado 29 de mayo de 2018.

[3] El software de procesamiento de pagos BASE24-eps gana el premio a la solución financiera más innovadora - Innovando.biz - Noticias, Banca y Finanzas, Pagos y Transacciones. (s. f.). http://www.consultor-it.com/articulo/69552/pagos-y-transacciones/banca-y-finanzas/el-software-de-procesamiento-de-pagos-base24-eps-gana-el-premio-a-la-solucion-financiera-mas-innovadora. Recuperado 29 de mayo de 2018.

[4] ISO 8583-1:2003 (2003) Financial transaction card originated messages — Interchange message specifications — Part 1: Messages, data elements and code values. https://www.iso.org/obp/ui/#iso:std:iso:8583:-1:ed-1:v1:en

[5] Ecommerce Fraud Prevention Poised for New Wave of Machine Learning Disruption, According to Signifyd's New Chief Scientist. (2017) Marketing Weekly News, 19 Aug. 2017, p. 428. Infotrac Newsstand, http://link.galegroup.com/apps/doc/A500446610/STND?u=metrial&sid=STND&xid=3c892d1e. Accessed 29 May 2018.

[6] SPARQL Query Language for RDF. de https://www.w3.org/TR/rdf-sparql-query/

[7] SWRL: A Semantic Web Rule Language Combining OWL and RuleML. https://www.w3.org/Submission/SWRL/

[8] Villazón-Terrazas, B., L. M. Vilches-Blázquez, O. Corcho, and A. Gómez-Pérez. (2011).‘‘Methodological Guidelines for Publishing Government Linked Data.’’ In Linking Government Data, edited by D. Wood, 27_49, Springer. http://www.springer.com/computer/database+management+%26+information+retrieval/book/978-1-4614-1766-8.

[9] Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. Expert Systems with Applications, 41(10), 4915-4928. https://doi.org/10.1016/j.eswa.2014.02.026

[10] The Nilson Report | News and Statistics for Card and Mobile Payment Executives. (s. f.). https://nilsonreport.com/ Recuperado 29 de mayo de 2018.

[11] Abdallah, A., Maarof, M. A., & Zainal, A. (2016). Fraud detection system: A survey. Journal of Network and Computer Applications, 68, 90-113. https://doi.org/10.1016/j.jnca.2016.04.007

[12] About CyberSource - US & Canada - CyberSource. (s. f.). https://www.cybersource.com/about/ Recuperado 29 de mayo de 2018.

[13] En Colombia, el 10% de órdenes online se rechazan por posible fraude • ENTER.CO. (s. f.). http://www.enter.co/especiales/empresas-del-futuro/en-colombia-el-10-de-ordenes-online-se-rechazan-por-posible-fraude/ Recuperado 29 de mayo de 2018.

[14] Radio, C. (2017). En Colombia cerca del 80% del fraude que se realiza son con tarjetas de crédito: Asobancaria. Caracol Radio website: https://caracol.com.co/radio/2017/10/29/economia/1509306065_940482.html Recuperado 13 de noviembre de 2019.

[15] Rushin, G., Stancil, C., Sun, M., Adams, S., & Beling, P. (2017). Horse race analysis in credit card fraud #x2014;deep learning, logistic regression, and Gradient Boosted Tree. En 2017 Systems and Information Engineering Design Symposium (SIEDS) (pp. 117-121). https://doi.org/10.1109/SIEDS.2017.7937700

[16] Pumsirirat, A., & Yan, L. (2018). Credit Card Fraud Detection using Deep Learning based on Auto-Encoder and Restricted Boltzmann Machine. International Journal of Advanced Computer Science and Applications (Ijacsa), 9(1). https://doi.org/10.14569/IJACSA.2018.090103

[17] Fraud detection with Linked Data | Semantics 2016. https://2016.semantics.cc/fraud-detection-linked-data Recuperado 30 de mayo de 2018.

[18] Potamitis, G. (2013). Design and Implementation of a Fraud Detection Expert System using Ontology-Based Techniques. https://studentnet.cs.manchester.ac.uk/resources/library/thesis_abstracts/MSc13/FullText/Potamitis-Giannis-fulltext.pdf

[19] Hu, Bo., Naseer, Aisha., Matsutsuk, Takahide. (2014). Anomaly Detection Technology Using BigGraph. FUJITSU Sci. Tech. J., Vol. 50, No. 1, pp. 9–15.

[20] Roldán-García, M. del M., García-Nieto, J., & Aldana-Montes, J. F. (2017). Enhancing semantic consistency in anti-fraud rule-based expert systems. Expert Systems with Applications, 90, 332-343. https://doi.org/10.1016/j.eswa.2017.08.036

[21] Rajput, Q., Khan, N. S., Larik, A., & Haider, S. (2014). Ontology Based Expert-System for Suspicious Transactions Detection. Computer and Information Science, 7(1). https://doi.org/10.5539/cis.v7n1p103

[22] Fang L., Cai M., Fu H., Dong J. (2007). Ontology-Based Fraud Detection. In: Shi Y., van Albada G.D., Dongarra J., Sloot P.M.A. (eds) Computational Science – ICCS 2007. ICCS 2007. Lecture Notes in Computer Science, vol 4489. Springer, Berlin, Heidelberg

[23] Kazemi, Z., & Zarrabi, H. (2017). Using deep networks for fraud detection in the credit card transactions. En 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI) (pp.0630-0633). https://doi.org/10.1109/KBEI.2017.8324876

[24] Pouyanfar, S., & Chen, S.-C. (2016). Semantic event detection using ensemble deep learning. En Multimedia (ISM), 2016 IEEE International Symposium on (pp. 203–208). IEEE.

[25] González, Rafael & Pomares Quimbaya, Alexandra. (2012). La investigación científica basada en el diseño como eje de proyectos de investigación en ingeniería.

[26] Suárez-Figueroa, M.C. (2010). Neon Methodology for Building Ontology Networks: Specification, Scheduling and Reuse" Universidad Politécnica de Madrid, España.

[27] Kaisler, S. (1986). "Expert systems: An overview," in IEEE Journal of Oceanic Engineering, vol. 11, no. 4, pp. 442-448.

[28] Ponce, C. V., Rojas, B., Ponce, C. V., & Rojas, B. (2019). Diseño de un Sistema Experto Difuso para la Determinación de la Densidad de Corriente en una Planta de Cromado. Información tecnológica, 30(2), 157-170. https://doi.org/10.4067/S0718-07642019000200157

[29] Tzafestas S.G., Kokkinaki A.I., Valavanis K.P. (1993). An Overview of Expert Systems. In: Tzafestas S. (eds) Expert Systems in Engineering Applications. Springer, Berlin, Heidelberg

[30] RDF Primer. https://www.w3.org/TR/rdf-primer/ Recuperado 4 de noviembre de 2019.

[31] RDF Schema 1.1. https://www.w3.org/TR/rdf-schema/ Recuperado 4 de noviembre de 2019

[32] Linked Open Data—W3C eGovernment Wiki. https://www.w3.org/egov/wiki/Linked_Open_Data. Recuperado 5 de noviembre de 2019

[33] The Linked Open Data Cloud. https://lod-cloud.net/ Recuperado 5 de noviembre de 2019.

[34] SPARQL Lenguaje de consulta para RDF. http://skos.um.es/TR/rdf-sparql-query/#docOutline. Recuperado 5 de noviembre de 2019.

[35] Protegeproject/swrlapi. (s. f.). https://github.com/protegeproject/swrlapi Recuperado 5 de noviembre de 2019.

[36] Semantic Web Rule Language. (2018). https://www.w3.org/Submission/SWRL/

[37] Merchant category codes (MCC) definition—Glossary. (s. f.). CreditCards.com website: https://www.creditcards.com/credit-card-news/glossary/term-merchant-category-codes-mcc.php Recuperado 8 de noviembre de 2019.

[38] Sahri, Zulazeze. (2018). An Ontology-Based Representation of Financial Criminology Domain Using Text Analytics Processing. 18.

[39] Noy, N. F., & McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology.

[40] Tang, Xiao-Bo & Liu, Guang-Chao & Yang, Jing & Wei, Wei. (2018). Knowledge-based Financial Statement Fraud Detection System: Based on an Ontology and a Decision Tree. KNOWLEDGE ORGANIZATION. 45. 205-219. 10.5771/0943-7444-2018-3-205.

[41] Alexopoulos, P., Kafentzis, K., Benetou, X., Tagaris, T., & Georgolios, P. (2007). Towards a Generic Fraud Ontology in e-Government. In ICE-B (pp. 269-276).

[42] García, M. del M. R., García-Nieto, J., & Aldana-Montes, J. F. (2016). An ontology-based data integration approach for web analytics in e-commerce. Expert Systems with Applications, 63, 20-34. https://doi.org/10.1016/j.eswa.2016.06.034

[43] Semantic Compliance in Finance. (s. f.). Financial Regulation Ontology website: https://finregont.com/ Recuperado 11 de noviembre de 2019.

[44] FIBO - EDM Council. (s. f.). https://edmcouncil.org/page/aboutfiboreview Recuperado 11 de noviembre de 2019.

[45] LKIF-Core Ontology: A Commonsense-based Legal Ontology. (s. f.). http://www.estrellaproject.org/lkif-core/ Recuperado 17 de noviembre de 2019.

[46] Carvalho, R., Goldsmith, M., and Creese, S. (2015). "Applying Semantic Technologies to Fight Online Banking Fraud," 2015 European Intelligence and Security Informatics Conference, Manchester, pp. 61-68.

[47] Zhao, G., & Meersman, R. (2006). Towards a Topical Ontology of Fraud. 4185. 566-572. 10.1007/11836025_54.

[48] Application Knowledge Engineering Methodology (s. f.). http://theses.univ-lyon2.fr/documents/getpart.php?id=lyon2.2009.sureephong_p&part=225274 Recuperado 17 de noviembre de 2019.

[49] Kingston, J. & Schafer, B., & Vandenberghe, W. (2004). Towards a Financial Fraud Ontology: A Legal Modelling Approach. Artif. Intell. Law. 12. 419-446. 10.1007/s10506-005-4163-0.

[50] Financial Fraud Prevention-Oriented Information Resources using Ontology Technology | FF POIROT Project | FP5. (s.f.). CORDIS | European Commission website: https://cordis.europa.eu/project/rcn/64012/factsheet/en Recuperado 17 de noviembre de 2019.

[51] Pease, A. Niles, I. & Li, J. (2002). The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications.

[52] Protégé. https://protege.stanford.edu/products.php#web-protege Recuperado 12 de noviembre de 2019.

[53] Apache Jena. https://jena.apache.org/index.html Recuperado 11 de noviembre de 2019.

[54] OWL API by owlcs. (s. f.). Recuperado 24 de noviembre de 2019, de http://owlcs.github.io/owlapi/

[55] Pellet—Semantic Web Standards. (s. f.). Recuperado 24 de noviembre de 2019, de https://www.w3.org/2001/sw/wiki/Pellet

[56] Apache Marmotta—Home. (s. f.). Recuperado 24 de noviembre de 2019, de https://marmotta.apache.org/

[57] Shung, K. P. (2018). Accuracy, Precision, Recall or F1? Recuperado 11 de noviembre de 2019, de Medium website: https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9