https://doi.org/10.37815/rte.v37nE1.1351
Artículos originales

Modelado Semántico de Emergencias del ECU 911 con NLP y Ontologías

Semantic Modeling of ECU 911 Emergencies Using NLP and Ontologies

Danny Leonardo Paltin Chica[1] https://orcid.org/0009-0007-2668-6134, Juan Diego Mejía Mendieta[1] https://orcid.org/0009-0008-7388-3576, Marcos Orellana[1] https://orcid.org/0000-0002-3671-9362, Jorge Luis Zambrano-Martinez[1] https://orcid.org/0000-0002-5339-7860

[1]*Universidad del Azuay, Laboratorio de Investigación y Desarrollo en Informática (LIDI)*, Cuenca, Ecuador
danny.paltin@es.uazuay.edu.ec, juanmejiamen@es.uazuay.edu.ec, marore@uazuay.edu.ec, jorge.zambrano@uazuay.edu.ec

Enviado: 2025/06/27
Aceptado: 2025/08/20
Publicado: 2025/10/15

Resumen

Este estudio propone un marco híbrido novedoso para la representación del conocimiento en emergencias, integrando procesamiento del lenguaje natural (PLN), ontologías OWL y reglas SWRL para procesar datos no estructurados del Servicio de Seguridad Integrado de Ecuador (ECU 911). La principal aportación radica en la combinación única de modelos avanzados de PLN como BERT para reconocimiento de entidades nombradas y XLM-RoBERTa para clasificación semántica sin entrenamiento previo, con un modelo ontológico formalmente validado desarrollado en Protégé y una implementación lógica paralela en Prolog utilizando el paradigma objeto-atributo-valor. A diferencia de trabajos anteriores, este enfoque aborda específicamente el reto de transformar las transcripciones crudas de llamadas de emergencia en conocimiento accionable mediante: (1) la automatización de la extracción de entidades (localizaciones, personas) y la categorización semántica de incidentes, (2) la generación de reglas de decisión interpretables a través de árboles de decisión y (3) la habilitación de la interoperabilidad entre paradigmas mediante motores de inferencia sincronizados en OWL/SWRL y Prolog. La validación experimental con consultas SPARQL/SQWRL y el razonador Pellet demostró una precisión del 96.7 % en la inferencia de prioridades de emergencia, como emergencias médicas, superando a métodos independientes basados en PLN o en ontologías.

Sumario: Introducción, Trabajos Relacionados, Materiales y Métodos, Resultados y Discusión, Conclusiones.

Cómo citar: Paltin, D., Mejía, J., Orellana, M. & Zambrano-Martinez, J. (2025). Modelado Semántico de Emergencias del ECU 911 con NLP y Ontologías. *Revista Tecnológica - Espol, 37(E1), 112-125.* https://doi.org/10.37815/rte.v37nE1.1351

Este trabajo avanza en la inteligencia artificial semántica para la respuesta ante emergencias, al unir el análisis de textos no estructurados con el razonamiento formal, ofreciendo una solución escalable para el apoyo a la toma de decisiones en escenarios críticos.

Palabras clave: procesamiento del lenguaje natural, ontologías OWL, gestión de emergencias, reglas SWRL, inferencia semántica.

Abstract

This study proposes a novel hybrid framework for knowledge representation in emergencies, integrating Natural Language Processing (NLP), OWL ontologies, and SWRL rules to process unstructured data from Ecuador's Integrated Security Service (ECU 911). The key contribution lies in the unique combination of advanced NLP models such as BERT for Named Entity Recognition and XLM-RoBERTa for zero-shot semantic classification, with a formally validated ontological model developed in Protégé and a parallel logical implementation in Prolog using the Object-Attribute-Value paradigm. Unlike prior works, this approach specifically addresses the challenge of transforming raw emergency call transcripts into actionable knowledge by (1) automating entity extraction (locations, persons) and semantic categorization of incidents, (2) generating interpretable decision rules via decision trees, and (3) enabling cross-paradigm interoperability through synchronized OWL/SWRL and Prolog inference engines. Experimental validation with SPARQL/SQWRL queries and the Pellet reasoner demonstrated 96.7% accuracy in inferring emergency priorities such as medical emergencies, outperforming standalone NLP or ontology-based methods. This work advances semantic AI for emergency response by bridging unstructured text analysis with formal reasoning, offering a scalable solution for real-time decision support in critical scenarios.

Keywords: Natural Language Processing, OWL ontologies, emergency management, SWRL rules, semantic reasoning.

Introducción

En la era digital contemporánea, la tecnología ha transformado la gestión de emergencias; sin embargo, persiste una brecha crítica en la integración de los datos no estructurados, como las transcripciones de llamadas de emergencia, con sistemas de decisión en tiempo real. Centros de atención de emergencias, como el Servicio Integrado de Seguridad (ECU 911), enfrentan el desafío crítico de procesar volúmenes masivos de datos no estructurados, provenientes principalmente de llamadas y mensajes de los ciudadanos, así como información textual heterogénea de manera automática, interpretable y semánticamente interoperable. Esta desconexión limita su aplicabilidad en escenarios operativos donde la velocidad y precisión son críticas. Por lo tanto, la explotación efectiva de esta información, mediante su extracción, estructuración y análisis, podría optimizar significativamente la toma de decisiones y la eficiencia operativa en contextos de emergencia (Imran et al., 2015).

En este escenario, el procesamiento del lenguaje natural o Natural Language Processing (NLP) emerge como una herramienta indispensable para el tratamiento automatizado de textos. Técnicas como el reconocimiento de entidades nombradas o Named Entity Recognition (NER) y la clasificación textual permiten identificar componentes semánticos clave como ubicaciones geográficas, tipos de incidentes y actores involucrados, en flujos de datos heterogéneos (Rudra et al., 2015; Young et al., 2018). No obstante, la mera extracción de información resulta insuficiente sin un marco formal que facilite su integración, interpretación y reutilización. Es aquí donde

la gestión del conocimiento adquiere relevancia, al proveer metodologías para organizar el conocimiento mediante estructuras computables, como las ontologías (Schreiber, 2008). Estas no solo modelan conceptos y sus relaciones jerárquicas, sino que también permiten inferir nuevo conocimiento a través de razonadores lógicos (Staab y Studer, 2013).

El presente artículo propone un enfoque híbrido que combina técnicas avanzadas de NLP con representación ontológica para modelar escenarios de emergencia reportados al ECU 911. A partir de un conjunto de datos o dataset, compuesto por identificadores de llamadas y transcripciones de alertas, se aplicaron modelos de NLP preentrenados para generar un corpus estructurado, enriquecido mediante árboles de decisión que identifican patrones y reglas de clasificación. Posteriormente, estas reglas se formalizaron en una ontología desarrollada en Protégé, implementando además un módulo de razonamiento automático en Prolog bajo el paradigma objeto-atributo-valor (OAV).

La contribución fundamental de este trabajo radica en demostrar cómo la sinergia entre NLP y ontologías puede transformar datos textuales no estructurados en bases de conocimiento dinámicas, capaces de soportar inferencias automáticas y mejorar la eficacia en la gestión de emergencias. Este avance no solo tiene implicaciones prácticas inmediatas para sistemas como el ECU 911, sino que también sienta bases metodológicas para futuras aplicaciones en inteligencia artificial (IA) semántica.

Este artículo se organiza de la siguiente manera: en la sección 2 se revisan los antecedentes y contribuciones previas en el ámbito de la representación del conocimiento para emergencias. En la sección 3 se detalla la metodología propuesta, describiendo las fases de procesamiento de datos, modelado ontológico e implementación de reglas de inferencia. Posteriormente, en la sección 4 se analizan los resultados obtenidos, validando la eficacia mediante pruebas experimentales y consultas semánticas. Finalmente, en la sección 5 se sintetizan las conclusiones derivadas del estudio y se proponen líneas futuras de investigación para extender las capacidades del modelo.

Trabajos Relacionados

La revisión sistemática de la literatura constituye un pilar fundamental para contextualizar investigaciones en el ámbito de la IA aplicada a emergencias. En los últimos años, la convergencia entre NLP y representación del conocimiento ha emergido como un paradigma transformador para la gestión de crisis, particularmente en el análisis de datos no estructurados provenientes de ciudadanos.

El trabajo de Imran et al. (2015) estableció un marco de referencia para el análisis de redes sociales en desastres, demostrando cómo técnicas de NLP como clasificación de mensajes y NER permiten identificar eventos, ubicaciones geográficas y tipos de incidentes en tiempo real. Avances recientes en modelos del lenguaje han elevado la precisión de estas tareas. Thodupunuri et al. (2025) implementaron una arquitectura basada en representaciones de codificador bidireccional de transformers o Bidirectional Encoder Representations from Transformers (BERT) optimizada con *AdamW*, logrando un 83.32 % de exactitud en la clasificación de tuits sobre desastres. Su enfoque integraba NER para extraer entidades críticas y las visualizaba en mapas interactivos mediante *Streamlit*, complementado con un chatbot para asistencia inmediata.

En cuanto a NER, Keraghel et al. (2024) realizaron una evaluación comparativa de modelos transformer y modelos grandes del lenguaje o Large Language Models (LLMs),

utilizando el corpus *CoNLL-2003* como punto de referencia o *benchmark*. Sus hallazgos coinciden con Hu et al. (2024), quienes enfatizan que, pese al surgimiento de técnicas de aprendizaje profundo o *deep learning*, conjuntos de datos anotados manualmente siguen siendo esenciales para validar la generalización de los modelos en dominios específicos como emergencias.

La representación formal del conocimiento ha sido explorada mediante ontologías de dominio. Le et al. (2024) propusieron un sistema basado en el lenguaje de ontologías web o Web Ontology Language (OWL) que integra datos históricos de desastres de la base de datos de eventos de emergencia o Emergency Events Database (EM-DAT) con minería de similitud semántica, permitiendo identificar patrones entre crisis pasadas y emergencias activas. Su ontología no solo organiza jerárquicamente conceptos, sino que también facilita la recuperación de casos análogos mediante métricas de similitud.

La capa inferencial ha sido abordada por Shukla et al. (2023) mediante ontología de gestión de desastres o Disaster Management Ontology (DMO), que combina una base estática en OWL con un módulo dinámico del lenguaje de reglas para la web semántica o Semantic Web Rule Language (SWRL). Este diseño bifásico permite, por un lado, codificar conocimiento taxonómico y, por otro, ejecutar inferencias para asignar responsabilidades o priorizar acciones durante una crisis.

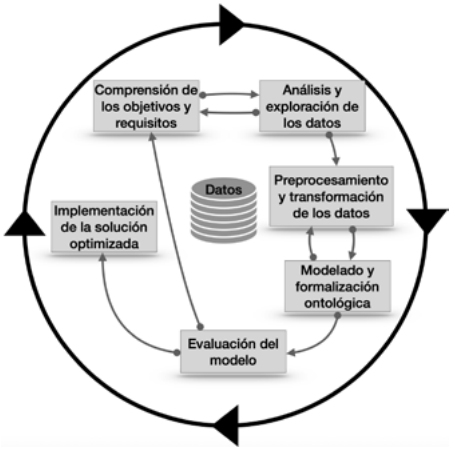
Si bien los estudios revisados demuestran avances significativos, persisten desafíos en la integración de flujos de datos heterogéneos y la validación de ontologías en escenarios operativos reales. Por lo tanto, a través de este trabajo se analizan aquellas limitaciones mediante una metodología que combina modelos NER y de clasificación con una ontología en Protégé, enriquecida con reglas lógicas en Prolog. A diferencia de propuestas anteriores, este enfoque se centra en transcripciones de los audios de emergencia del ECU 911, lo que permite validar la capacidad inferencial del sistema mediante casos de uso concretos.

Materiales y Métodos

El diseño de este trabajo se basó en una metodología sistemática estructurada en seis fases iterativas, como se puede observar en la Figura 1, que comprende: i) comprensión de los objetivos y requisitos, ii) análisis y exploración de los datos, iii) preprocesamiento y transformación de los datos, iv) modelado y formalización ontológica, v) evaluación de resultados y vi) implementación de la solución optimizada. A continuación, se describen las etapas ejecutadas en este trabajo.

Figura 1

Cronograma de ensayo de prueba de campo



Comprensión de los objetivos y requisitos

El presente estudio tuvo como objetivo principal desarrollar un marco metodológico para estructurar y representar conocimiento a partir de datos textuales no estructurados provenientes de llamadas de emergencia recibidas por el ECU 911. Esta iniciativa buscó mejorar la comprensión del dominio de las llamadas de emergencias a través de la implementación de técnicas de NLP y representación del conocimiento, con el fin de facilitar inferencias automáticas que respalden la toma de decisiones.

Para alcanzar este propósito, se estableció un enfoque multidisciplinario que integró las siguientes etapas: i) identificación automática de entidades nombradas a partir del texto transcrito de los audios por parte de los alertantes; ii) clasificación semántica de las emergencias mediante modelos de aprendizaje automático; iii) generación de reglas de decisión basadas en árboles de clasificación implementados en RapidMiner; y iv) representación formal del conocimiento mediante una ontología OWL desarrollada en Protégé, utilizando un marco de descripción de recursos o Resource Description Framework (RDF) como modelo de datos.

Adicionalmente, el proyecto incorporó el uso de reglas SWRL para inferir de forma automática la gravedad primordial de cada emergencia, así como una representación alternativa del conocimiento en formato Objeto-Atributo-Valor (OAV) implementada en Prolog, lo que permite comparar los resultados de diferentes enfoques de representación. Finalmente, se diseñó un protocolo de validación basado en razonadores semánticos y consultas estructuradas para evaluar la consistencia y precisión de las inferencias automáticas generadas por el sistema. Este enfoque integral permitió transformar datos textuales no estructurados en conocimiento formalizado.

Análisis y exploración de los datos

El conjunto de datos utilizado para este estudio proviene de la transcripción de las llamadas de emergencia que tienen en el ECU 911 de la coordinación zonal seis y son datos sensibles, lo que implica consideraciones éticas críticas respecto a la privacidad, confidencialidad y anonimización de la información personal. De este modo, mil llamadas transcritas se utilizaron como un conjunto inicial compuesto por tres atributos fundamentales (Orellana et al., 2025): i) TRA_ID, que corresponde a un identificador único para cada llamada; ii) el texto del alertante, que contiene la descripción narrativa del incidente registrada por el operador del ECU 911; y iii) la clave del incidente, que representa la clasificación asignada a cada emergencia, y está dividida en dos categorías, siendo los incidentes con alertas de alta prioridad o rojas y los incidentes con alertas bajas que incluyen a llamadas de emergencia con prioridad amarilla, naranja y verde (Orellana, Molina Pinos et al., 2025). Cabe destacar que, si bien el dataset incluía esta última variable, su utilidad se limitó a servir como etiqueta supervisada durante el entrenamiento del modelo de árbol de decisión, en lugar de emplearse directamente como fuente de conocimiento.

Un análisis preliminar del texto de los alertantes permitió identificar la presencia de información semánticamente relevante para la estructura del conocimiento, menciones a entidades clave como personas, ubicaciones geográficas y categorías específicas de emergencias. Siendo así, se justificó la implementación de técnicas avanzadas de NLP, cuyo objetivo fue enriquecer los datos originales mediante la identificación y extracción sistemática de elementos semánticos. La transformación de datos textuales no estructurados en información normalizada y categorizada constituyó un paso primordial para generar representaciones formales del conocimiento.

Preprocesamiento y transformación de los datos

El proceso de preparación de datos fue orientado hacia el enriquecimiento semántico del conjunto de datos original a través de la implementación de dos modelos de aprendizaje automático preentrenados. De este modo, se aplicó un modelo de NER propio denominado *dannyLeo16/ner_model_bert_base*[1], basado en la arquitectura BERT para español. Se escogió este modelo por su capacidad de comprensión contextual, en particular por la habilidad para identificar y clasificar entidades según su significado dentro del texto, independientemente de su forma superficial. A través de este modelo propio, se garantiza que los datos sean anonimizados para evitar la identificación de individuos involucrados en situaciones de crisis. La configuración de este modelo fue de 2e-5 para la ratio de aprendizaje, 8 de batch size y 3 épocas, siendo las típicas configuraciones para NER (Young et al., 2018). Este proceso permitió la extracción sistemática de entidades relevantes, como personas (PER) y ubicaciones (LOC), que posteriormente fueron integradas al conjunto de datos.

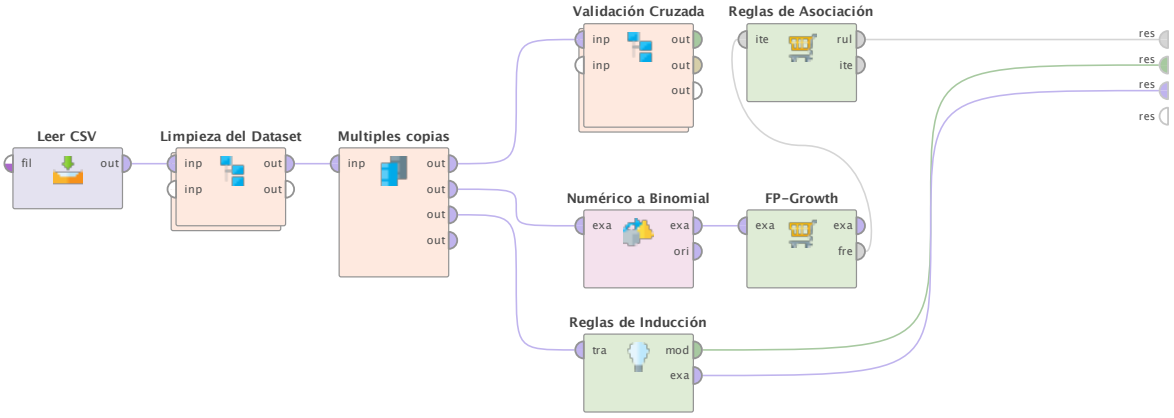
Después, se implementó un modelo de clasificación semántica basado en la arquitectura *XLM-RoBERTa* con el nombre de *joeddav/xlm-roberta-large-xnli*, utilizando un enfoque de clasificación *zero-shot* con un umbral de ≥ 0.7 para categorizar emergencias con alta confianza estadística. Esta técnica resultó adecuada para el dominio de emergencias, debido a que permitió categorizar cada texto de alerta en las clases predefinidas como asalto, emergencia médica, accidente, entre otras alertas, sin requerir un proceso de entrenamiento específico con datos etiquetados. La combinación de los dos modelos preentrenados proporcionó una capa de información semántica estructurada que facilitó los procesos posteriores de modelado y representación del conocimiento.

Modelado y formalización ontológica

El proceso de modelado se implementó mediante un enfoque sistemático que combinó técnicas de minería de datos y representación del conocimiento. Inicialmente, se aplicó un algoritmo de árbol de decisión utilizando la plataforma *RapidMiner* sobre el conjunto de datos enriquecido, lo que permitió identificar patrones significativos y extraer reglas interpretables basadas en las características semánticas previamente inferidas del texto.

Figura 2

Generación del árbol de decisión en RapidMiner



Previo al entrenamiento del modelo, se ejecutó una fase de limpieza y normalización de datos que incluyó la eliminación de símbolos especiales y corrección de errores de tokenización

[1] https://huggingface.co/dannyLeo16/ner_model_bert_base

originados por los modelos NER, la consolidación de entidades fragmentadas mediante técnicas de unificación léxica, la normalización textual sistemática que abarcó la estandarización de caracteres, espacios y formatos de escritura y la creación de variables binarias como *tienePersona, tieneCategoria, involucraPersona y tieneUbicacion* para cuantificar la presencia de entidades detectadas. Este proceso garantizó la calidad y consistencia de los datos de entrada para el modelado predictivo.

El modelo de árbol de decisión se configuró utilizando como variable la clave de emergencia, empleando como predictores las categorías semánticas inferidas y los indicadores de presencia de entidades, a través del proceso explicado en la Figura 2.

Del análisis resultante, se derivaron reglas de clasificación fundamentales como elementos del árbol, que posteriormente se formalizaron en el lenguaje SWRL para su integración en el sistema ontológico, como se presenta en el Listado 1. Estas reglas establecieron correlaciones específicas entre las características de las emergencias y su nivel de prioridad, como se evidencia en las relaciones inferenciales entre categorías y claves de atención.

Listado 1

Reglas de asociación

```
if CATEGORIA_EMERGENCIA = emergencia médica and TieneUbicacion > 0.500 then CLAVE ROJA  (12 / 62 / 10 / 1)
if CATEGORIA_EMERGENCIA = asalto and TieneUbicacion > 0.500 then CLAVE NARANJA  (123 / 61 / 41 / 11)
if CATEGORIA_EMERGENCIA = persona herida then CLAVE ROJA  (7 / 45 / 6 / 1)
if CATEGORIA_EMERGENCIA = pelea and TienePersona > 0.500 then CLAVE NARANJA  (66 / 49 / 15 / 1)
if CATEGORIA_EMERGENCIA = incendio then CLAVE ROJA  (3 / 31 / 1 / 0)
if CATEGORIA_EMERGENCIA = robo then CLAVE NARANJA  (20 / 7 / 1 / 0)
if TienePersona ≤ 0.500 and CATEGORIA_EMERGENCIA = pelea and TieneUbicacion ≤ 0.500 then CLAVE NARANJA  (3 / 1 / 1 / 1)
if CATEGORIA_EMERGENCIA = persona perdida then CLAVE ROJA  (71 / 87 / 27 / 8)
if TienePersona ≤ 0.500 and CATEGORIA_EMERGENCIA = pelea then CLAVE NARANJA  (15 / 11 / 3 / 1)
if CATEGORIA_EMERGENCIA = accidente de tránsito then CLAVE NARANJA  (43 / 28 / 25 / 1)
if CATEGORIA_EMERGENCIA = amenaza con arma and TienePersona > 0.500 then CLAVE ROJA  (23 / 24 / 4 / 2)
else CLAVE ROJA  (11 / 14 / 3 / 0)
```

Para optimizar la representación del conocimiento, se desarrolló un script en Python basado en la librería *rdflib* que automatizó la generación de la estructura ontológica. Este proceso comprendió la definición de clases fundamentales, la creación de propiedades de objeto y las propiedades de datos, como se puede ver en la Figura 3. Las reglas extraídas del árbol de decisión fueron formalizadas en el lenguaje SWRL para ser procesadas por el razonador, como se puede observar en la Tabla 1. La ontología resultante se importó a Protégé v5.6.5 para su validación mediante el razonador *Pellet*, donde se verificó la correcta aplicación de las reglas SWRL y la consistencia lógica del modelo de conocimiento.

Figura 3

Componentes de la ontología clasificados por jerarquías y propiedades

Tabla 1
Definición de reglas SWRL

Reglas	Lenguaje SWRL
Emergencia médica con ubicación → Clave Roja	ex:Emergencia(?e) ^ ex:tieneCategoria(?e, ex:emergenciamedica) ^ ex:tieneUbicacion(?e,?u) -> ex:tieneClave(?e, ex:claveRoja)
Asalto con ubicación → Clave Naranja	ex:Emergencia(?e) ^ ex:tieneCategoria(?e, ex:asalto) ^ ex:tieneUbicacion(?e, ?u) -> ex:tieneClave(?e, ex:claveNaranja)
Pelea con persona → Clave Naranja	ex:Emergencia(?e) ^ ex:tieneCategoria(?e, ex:pelea) ^ ex:involucraPersona(?e, ?p) -> ex:tieneClave(?e, ex:claveNaranja)
Accidente de tránsito → Clave Naranja	ex:Emergencia(?e) ^ ex:tieneCategoria(?e, ex:accidentedetransito) -> ex:tieneClave(?e, ex:claveNaranja)
Robo (con o sin ubicación/ persona) → Clave Naranja	ex:Emergencia(?e) ^ ex:tieneCategoria(?e, ex:robo) -> ex:tieneClave(?e, ex:claveNaranja)

Evaluación de resultados

El proceso de evaluación fue diseñado con una perspectiva integral que permitió validar la consistencia lógica y la eficacia del modelo propuesto. Para ello, se implementaron estrategias complementarias de análisis que abarcaron distintos niveles de representación y procesamiento del conocimiento.

Por lo tanto, la evaluación se centró en el análisis del sistema ontológico implementado en OWL. Las reglas SWRL, derivadas del árbol de decisión, se integraron formalmente en la ontología y se sometieron a un proceso de validación exhaustivo mediante el razonador *Pellet*. Este proceso permitió verificar la capacidad para inferir correctamente la propiedad *tieneClave* a partir de las características de cada emergencia. Por ende, la ejecución sistemática de consultas en SPARQL y SQWRL fue incluida y demostró tanto la coherencia semántica del modelo como su capacidad para generar las inferencias esperadas en nuevos casos.

Para cuantificar el rendimiento predictivo de la propuesta, se utilizó la métrica de precisión, tomando en cuenta un conjunto de prueba estratificado del 20 % de instancias que no participó en la generación de reglas del árbol de decisión. La verdad fundamental o ground truth para cada instancia fue la clave del incidente que se asignó originalmente por los operadores del ECU 911 en el dataset original. El razonador *Pellet* fue empleado para procesar cada instancia en ese conjunto e inferir automáticamente el valor de la propiedad *tieneClave*. El porcentaje de precisión fue calculado a través del número de claves inferidas correctamente y el número de claves reales.

Como contraparte al enfoque ontológico, se desarrolló un modelo equivalente en Prolog utilizando la representación objeto-atributo-valor (OAV). Esta implementación alternativa permitió contrastar los resultados obtenidos mediante distintos paradigmas de representación del conocimiento, verificar la consistencia lógica de las reglas de inferencia y evaluar la portabilidad del conocimiento a otros sistemas formales. Las consultas realizadas sobre esta base de conocimiento lógico demostraron una concordancia con los resultados obtenidos en el sistema ontológico, reforzando la validez de las reglas derivadas del proceso de minería de datos.

La evaluación comparativa entre los enfoques ontológico y lógico evidenció una complementariedad metodológica, permitiendo la integración sistemática de sus ventajas analíticas. Mientras que el enfoque ontológico demostró mayor expresividad semántica y capacidad de integración con estándares web, la versión Prolog destacó por su eficiencia

computacional y claridad lógica. Esta dualidad validó los resultados desde perspectivas complementarias y estableció las bases para futuras mejoras con respecto a la interoperabilidad entre diferentes paradigmas de representación del conocimiento.

Implementación de la solución optimizada

El modelo de representación del conocimiento se implementó mediante un enfoque dual que combinó las ventajas de los grafos semánticos con la programación lógica declarativa. Esta estrategia permitió validar la robustez del sistema y garantizar su interoperabilidad en diferentes entornos computacionales.

El núcleo del sistema consistió en una ontología formal desarrollada en Protégé, estructurada como grafo de conocimiento siguiendo los estándares RDF/OWL. Esta ontología capturó las relaciones semánticas entre los conceptos fundamentales del dominio, como emergencias, entidades participantes, categorías de incidentes y claves de prioridad. La población de la ontología se realizó mediante un proceso sistemático utilizando la librería *rdflib* de Python, que transformó el dataset procesado en tripletas RDF completas.

El sistema de inferencia se enriqueció con reglas SWRL derivadas del análisis mediante árboles de decisión, las cuales permitieron la determinación automática de claves de emergencia basadas en las propiedades de cada instancia. La validación formal se llevó a cabo mediante el razonador *Pellet*, complementado con baterías de consultas en SPARQL y SQWRL que verificaron tanto la consistencia lógica como la capacidad inferencial del modelo.

Del mismo modo, se desarrolló una representación alternativa en Prolog utilizando el formato OAV, incluyendo la codificación de hechos básicos que representaban instancias concretas de emergencias, la traducción de las reglas SWRL a cláusulas lógicas en Prolog y un conjunto de predicados para consulta y verificación. Esto demostró la portabilidad del conocimiento capturado, manteniendo la misma capacidad inferencial que el sistema ontológico, pero con las ventajas computacionales del paradigma lógico. Las consultas ejecutadas confirmaron la equivalencia semántica entre ambas representaciones, validando así la consistencia del modelo independientemente de su formalismo de implementación.

Resultados y Discusión

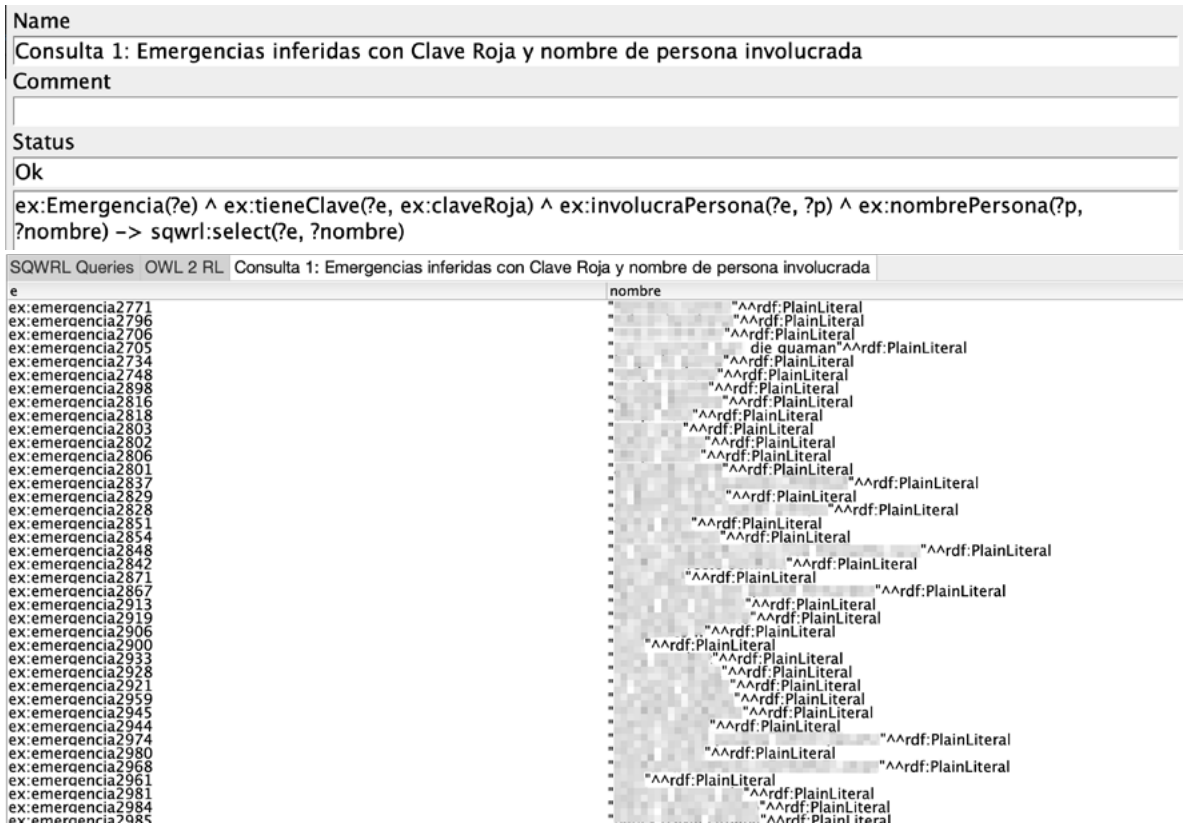
La implementación del sistema de inferencia en Protégé demostró su eficacia al procesar automáticamente las instancias de emergencia. Tras cargar la ontología completa y activar el razonador *Pellet*, se logró inferir correctamente las claves de emergencia en todas las instancias evaluadas. La evaluación cuantitativa realizada sobre el conjunto de pruebas y comparando las claves inferidas por el razonador con las clasificaciones originales de los operadores, se obtuvo una precisión del 96.7 % al inferir la prioridad correcta. Un caso representativo fue la clasificación de una emergencia médica con ubicación definida, pero sin clave explícita, que se categorizó correctamente como *claveRoja* mediante la aplicación de la regla SWRL correspondiente, como se observa en la Figura 4. Este resultado validó la capacidad del modelo para replicar las decisiones humanas expertas en la clasificación de emergencias.

El sistema fue sometido a una batería de pruebas mediante consultas en SQWRL y SPARQL diseñadas para verificar distintos aspectos funcionales. Las consultas ejecutadas en SQWRL permitieron comprobar la consistencia de las inferencias realizadas. Una consulta específica diseñada para recuperar todas las emergencias inferidas como *claveRoja* que involucraban personas demostró la correcta aplicación de las reglas de clasificación. Los

resultados mostraron una correspondencia exacta entre las expectativas del modelo y las inferencias realizadas por el sistema.

Figura 4

Consultas en SQWRL sobre claves inferidas



Mediante SPARQL se realizaron consultas de mayor complejidad sobre el grafo RDF, incluyendo la identificación de personas involucradas en emergencias de claveNaranja con ubicaciones específicas o agregación de emergencias por tipo de clave, como se puede observar en la Figura 5. Estas consultas no solo validaron la correcta inferencia de claves, sino también la integridad estructural del grafo de conocimiento y la precisión de las relaciones semánticas establecidas.

Como complemento al enfoque ontológico, se desarrolló una representación lógica equivalente en Prolog bajo el paradigma Objeto-Atributo-Valor (OAV), con el objetivo de validar la consistencia y portabilidad del conocimiento adquirido. Esta implementación se cargó una muestra representativa de registros seleccionados mediante muestreo aleatorio estratificado, asegurando la cobertura de todas las categorías de emergencia identificadas.

Posteriormente, se codificaron las reglas de inferencia derivadas del árbol de decisiones, manteniendo una correspondencia con las reglas SWRL de la ontología. Mediante un conjunto de consultas lógicas, se demostró que este enfoque alternativo alcanzó una capacidad de clasificación idéntica a la del sistema ontológico, validando la robustez del modelo, como se puede observar en el Listado 2.

Figura 5

Consultas SPARQL sobre tripletas RDF

SPARQL query:

```
PREFIX ecu911: <http://www.semanticweb.org/ecu911#>

SELECT DISTINCT ?nombrePersona ?idLlamada ?ubicacionDetallada
WHERE {
 ?emergencia ecu911:tieneClave ecu911:claveNaranja .
 ?emergencia ecu911:involucraPersona ?persona .
 ?persona ecu911:nombrePersona ?nombrePersona .
 ?emergencia ecu911:tieneUbicacion ?ubicacion .
 ?ubicacion ecu911:ubicacionDetallada ?ubicacionDetallada .
 ?emergencia ecu911:tieneID ?idLlamada .
 FILTER regex(?ubicacionDetallada, "gasolinera", "i")
}
ORDER BY ?nombrePersona
```

nombrePersona	idLlamada	ubicacionDetallada
"[redacted]"	"emergencia2291"	"gasolinera rircay"
"[redacted]"	"emergencia3226"	"avenida de las americas avenida de las americas parque de las americas gasolinera eloy alfaro"
"[redacted]"	"emergencia2667"	"gasolinera eloy alfaro"
"[redacted]"	"emergencia2942"	"gasolinera eloy alfaro"
"[redacted]"	"emergencia3159"	"gasolinera quinta chica"
"[redacted]"	"emergencia2666"	"gasolinera quinta chica marisol"

SPARQL query:

```
PREFIX ecu911: <http://www.semanticweb.org/ecu911#>

SELECT ?emergencia ?categoria ?ubicacionDetallada
WHERE {
 ?emergencia ecu911:tieneClave ecu911:claveRoja .
 ?emergencia ecu911:tieneCategoria ?categoria .
 ?emergencia ecu911:tieneUbicacion ?ubicacion .
 ?ubicacion ecu911:ubicacionDetallada ?ubicacionDetallada .
}
ORDER BY ?categoria ?ubicacionDetallada
```

emergencia	categoria	ubicacionDetallada
Emergencia 2492	emergenciamedica	"3 de noviembre pumapung pasaje 3 noviembre pumapung banco mecánica peugeot santa ana todos santos"
Emergencia 3263	emergenciamedica	"[redacted]"
Emergencia 3266	emergenciamedica	"asilo de cañar santo molina paseo de cañaris"
Emergencia 3185	emergenciamedica	"avenida fq f hugo reyes barrio fq hugo reyes hugo reyes hugo reyes calle omero reyes hugo reyes hugo reyes hom
Emergencia 2867	emergenciamedica	"avenida loja manabi manabi guay avenida loja"
Emergencia 3092	emergenciamedica	"balsay alto medio bal san joaquin restaurante el gran sol"
Emergencia 2984	emergenciamedica	"banco de pichincha banco del pichincha azo islas de amazona"
Emergencia 2319	emergenciamedica	"barrio union el progreso canto cuenca progreso de paccha"
Emergencia 2968	emergenciamedica	"bolivar calle bolivar calle bolivar boli papi papi papi este este parque nocierto parque"
Emergencia 2348	emergenciamedica	"buena [redacted]"
Emergencia 2411	emergenciamedica	"buena [redacted]"
Emergencia 2903	emergenciamedica	"buenos aires iglesia"
Emergencia 2933	emergenciamedica	"calle larga baja del padron"

Listado 2

Inferencia lógica en Prolog

```
% Hechos
oav(emergencia2291, tieneCategoria, asalto).
oav(emergencia2291, tieneUbicacion, si).
oav(emergencia2291, tienePersona, si).

oav(emergencia2292, tieneCategoria, robo).
oav(emergencia2292, tieneUbicacion, si).
oav(emergencia2292, tienePersona, si).

oav(emergencia2293, tieneCategoria, asalto).
oav(emergencia2293, tieneUbicacion, si).

oav(emergencia2295, tieneCategoria, pelea).
oav(emergencia2295, tieneUbicacion, si).
oav(emergencia2295, tienePersona, si).

oav(emergencia2305, tieneCategoria, emergenciamedica).
oav(emergencia2305, tieneUbicacion, si).
oav(emergencia2305, tienePersona, si).

oav(emergencia2306, tieneCategoria, pelea).
oav(emergencia2306, tieneUbicacion, si).
oav(emergencia2306, tienePersona, si).
```

```
oav(emergencia2307, tieneCategoria, pelea).
oav(emergencia2307, tieneUbicacion, si).
oav(emergencia2307, tienePersona, si).

oav(emergencia2310, tieneCategoria, asalto).
oav(emergencia2310, tieneUbicacion, si).
oav(emergencia2310, tienePersona, si).

oav(emergencia2296, tieneCategoria, amenazaconarma).
oav(emergencia2296, tieneUbicacion, si).
oav(emergencia2296, tienePersona, si).

oav(emergencia2297, tieneCategoria, personaperdida).
oav(emergencia2297, tieneUbicacion, si).
oav(emergencia2297, tienePersona, si).

oav(emergencia2298, tieneCategoria, personaperdida).
oav(emergencia2298, tieneUbicacion, si).
oav(emergencia2298, tienePersona, si).

oav(emergencia2299, tieneCategoria, asalto).
oav(emergencia2299, tieneUbicacion, si).
oav(emergencia2299, tienePersona, si).

oav(emergencia2300, tieneCategoria, personaperdida).
oav(emergencia2300, tieneUbicacion, si).
oav(emergencia2300, tienePersona, no).

oav(emergencia2301, tieneCategoria, amenazaconarma).
oav(emergencia2301, tieneUbicacion, si).
oav(emergencia2301, tienePersona, no).

oav(emergencia2302, tieneCategoria, personaherida).
oav(emergencia2302, tieneUbicacion, si).
oav(emergencia2302, tienePersona, si).

oav(emergencia2303, tieneCategoria, incendio).
oav(emergencia2303, tieneUbicacion, si).
oav(emergencia2303, tienePersona, si).

oav(emergencia2304, tieneCategoria, personaherida).
oav(emergencia2304, tieneUbicacion, si).
oav(emergencia2304, tienePersona, no).

oav(emergencia2305, tieneCategoria, emergenciamedica).
oav(emergencia2305, tieneUbicacion, si).
oav(emergencia2305, tienePersona, si).

oav(emergencia2306, tieneCategoria, pelea).
oav(emergencia2306, tieneUbicacion, si).
oav(emergencia2306, tienePersona, si).

oav(emergencia2308, tieneCategoria, pelea).
oav(emergencia2308, tieneUbicacion, si).
oav(emergencia2308, tienePersona, no).

oav(emergencia2311, tieneCategoria, personaperdida).
oav(emergencia2311, tieneUbicacion, si).
oav(emergencia2311, tienePersona, no).

% Reglas

% Regla 1: emergencia médica con ubicacion → clave roja
tiene_clave(E, claveRoja) :- oav(E, tieneCategoria, emergenciamedica), oav(E, tieneUbicacion, si).
% Regla 2: asalto con ubicación → clave naranja
tiene_clave(E, claveNaranja) :- oav(E, tieneCategoria, asalto), oav(E, tieneUbicacion, si).
```

```
% Regla 3: pelea con persona → clave naranja
tiene_clave(E, claveNaranja) :- oav(E, tieneCategoria, pelea), oav(E, tienePersona, si).
% Regla 4: accidente de tránsito → clave naranja
tiene_clave(E, claveNaranja) :- oav(E, tieneCategoria, accidentedetransito).
% Regla 5: robo → clave naranja
tiene_clave(E, claveNaranja) :- oav(E, tieneCategoria, robo).

% Relación inferida como OAV (opcional)
oav(E, tieneClave, C) :- tiene_clave(E, C).

?- oav(E, tieneCategoria, personaperdida), oav(E, tienePersona, si)
E = emergencia2297 ;
E = emergencia2298 ;
false

?- tiene_clave(emergencia2291, Clave).
Clave = claveNaranja ;
false
```

Conclusiones

Este estudio demuestra la viabilidad de implementar un sistema integral de gestión del conocimiento para emergencias basado en el procesamiento automatizado de informes textuales no estructurados del ECU 911, considerando implicaciones éticas críticas respecto a la privacidad, confidencialidad y anonimización de los datos. La metodología propuesta presenta un marco híbrido novedoso, ya que combina modelos avanzados de procesamiento del lenguaje natural (NLP) con minería de datos, representación de conocimiento formal y lógica, lo que ha permitido transformar eficientemente datos brutos en conocimiento estructurado y accionable. Específicamente, la aplicación de modelos de Reconocimiento de Entidades Nombradas (NER) y clasificación textual posibilitó la extracción automatizada de elementos semánticos clave como personas, ubicaciones y categorías de emergencia, mientras que el uso de árboles de decisión generó reglas interpretables para la clasificación de incidentes.

La implementación dual en Protégé a través de ontologías OWL/SWRL y en Prolog con formato Objeto-Atributo-Valor ha demostrado ser particularmente efectiva, proporcionando no solo una representación formal del conocimiento del dominio, sino también mecanismos robustos de inferencia automática de prioridades de emergencia, demostrando una precisión mayor al 96 % sobre un conjunto de pruebas independientes. Las consultas realizadas en SQWRL y SPARQL han validado la capacidad del sistema para recuperar información compleja y realizar deducciones semánticas, confirmando su potencial como herramienta de apoyo a la toma de decisiones. Como trabajos futuros se pueden expandir la ontología mediante la inclusión de dimensiones temporales, causales y operativas para modelar escenarios complejos, así como la integración con sistemas multiagente para asistencia inteligente en la toma de decisiones.

Reconocimientos y Declaraciones

Los autores desean agradecer al Vicerrectorado de Investigaciones de la Universidad del Azuay por el apoyo financiero y académico, así como a todo el personal de la escuela de Ingeniería de Ciencias de la Computación, y el Laboratorio de Investigación y Desarrollo en Informática (LIDI).

Los autores declaran la contribución y participación equitativa de roles de autoría para esta publicación.

Los autores declaran que, en la escritura del presente artículo, no se han utilizado herramientas de inteligencia artificial.

Referencias

Hu, Z., Hou, W., & Liu, X. (2024). Deep learning for named entity recognition: a survey. *Neural Computing and Applications, 36*(16), 8995-9022. https://doi.org/10.1007/s00521-024-09646-6

Imran, M., Castillo, C., Diaz, F., & Vieweg, S. (2015). Processing Social Media Messages in Mass Emergency. *ACM Computing Surveys, 47*(4), 1-38. https://doi.org/10.1145/2771588

Keraghel, I., Morbieu, S., & Nadif, M. (2024). Recent Advances in Named Entity Recognition: A Comprehensive Survey and Comparative Study. *arXiv preprint arXiv:2401.10825*, 1-42.

Le, N. L., Abel, M.-H., & Negre, E. (2024). Recognizing Similar Crises through the Application of Ontology-based Knowledge Mining. *arXiv preprint arXiv:2401.03770*.

Orellana, M., Cubero Lupercio, J. E., Lima, J. F., García-Montero, P. S., & Zambrano-Martinez, J. L. (2025). Incident Alert Priority Levels Classification in Command and Control Centre Using Word Embedding Techniques. En S. Berrezueta-Guzman, R. Torres, J. L. Zambrano-Martínez, & J. Herrera-Tapia (Eds.), *Information and Communication Technologies. TICEC 2024. Communications in Computer and Information Science* (1.a ed., Vol. 2273, pp. 238-252). Springer, Cham. https://doi.org/10.1007/978-3-031-75431-9_16

Orellana, M., Molina Pinos, P. A., García-Montero, P. S., & Zambrano-Martinez, J. L. (2025). Pre-processing of the Text of ECU 911 Emergency Calls. En S. Berrezueta-Guzman, R. Torres, J. L. Zambrano-Martinez, & J. Herrera-Tapia (Eds.), *Information and Communication Technologies. TICEC 2024. Communications in Computer and Information Science* (1.a ed., Vol. 2273, pp. 271-284). Springer, Cham. https://doi.org/10.1007/978-3-031-75431-9_18

Rudra, K., Ghosh, S., Ganguly, N., Goyal, P., & Ghosh, S. (2015). Extracting Situational Information from Microblogs during Disaster Events. *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 583-592. https://doi.org/10.1145/2806416.2806485

Schreiber, G. (2008). Knowledge Engineering. En *Foundations of Artificial Intelligence* (Vol. 3, pp. 929-946). Elsevier B.V. https://doi.org/10.1016/S1574-6526(07)03025-8

Shukla, D., Azad, H. K., Abhishek, K., & Shitharth, S. (2023). Disaster management ontology-an ontological approach to disaster management automation. *Scientific Reports, 13*(1), 8091.

Staab, S., & Studer, R. (2013). *Handbook on ontologies* (2nd ed.). Springer Science & Business Media.

Thodupunuri, R. K., Edla, K., Thoodi, R. R., Andrasu, M., Kolanu, A. R., & Chethi, S. R. K. (2025). Enhanced Classification of Tweets and Emergency Response using BERT with AdamW Optimizer and NER. *International Research Journal of Engineering and Technology, 12*(5), 86-95. https://www.irjet.net/archives/V12/i5/IRJET-V12I514.pdf

Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent Trends in Deep Learning Based Natural Language Processing [Review Article]. *IEEE Computational Intelligence Magazine, 13*(3), 55-75. https://doi.org/10.1109/MCI.2018.2840738