UNIVERSIDAD DE MEDELLIN

EXTRACCION DE INFORMACIÓN DE DOCUMENTOS DE NEGOCIO ESCRITOS EN LENGUAJE NATURAL EN IDIOMA ESPAÑOL Y SU REPRESENTACIÓN EN UN MODELO CONCEPTUAL

AUTOR:
Ing. Diego Alejandro Marín Álvarez
CC 1036605136
dialmarin17@gmail.com
3011983 - 3206192647

DIRECTORES:
Dra. Bell Manrique Losada
Dr. Juan Bernardo Quintero

LÍNEA TEMÁTICA:
Natural Language Processing

MAESTRÍA EN INGENIERÍA DE SOFTWARE
UNIVERSIDAD DE MEDELLÍN

MEDELLÍN
2019

AGRADECIMIENTOS

En primera instancia a la Dra Bell Manrique, docente y directora de este trabajo, quien desde un principio me dirigió en la línea de investigación aportó mucho desde su experiencia, además al Dr Juan Bernardo Quintero, docente y co-director, entre ambos me supieron llevar por el proceso investigativo con gran carisma y siendo extremadamente asertivos en sus instrucciones y recomendaciones.

A las docentes María Clara Gómez y Luisa Fernanda Villa, quienes me dieron los primeros pinos en el proceso investigativo y permitieron iniciar las exploraciones iniciales que a la postre dieron las bases para el presente trabajo.

A los ingenieros que participaron del experimento, en especial a Juan Sebastián Morales; su aporte es muy valioso teniendo en cuenta que no solo permitieron poner a prueba el método propuesto, sino que dedicaron tiempo en medio de sus exigentes compromisos laborales y familiares para poner su granito de arena a este trabajo.

TABLA DE CONTENIDO

LISTA DE FIGURAS

LISTA DE TABLAS

PARTE I:

INTRODUCCIÓN

CAPÍTULO 1: INTRODUCCIÓN

La documentación del conocimiento se realiza mediante modelos o diagramas construidos por un profesional que debe identificar las fuentes de información que le sean útiles y de allí extraer los datos que requiere para documentar dichos modelos, todo esto basado en sus habilidades para obtener información de diferentes fuentes, así como de su experiencia [1] .

Durante el proceso de educción de requisitos se ejecutan entrevistas, sin embargo los *stakeholders* pueden estar obviando información importante [2]. Como complemento, se puede obtener información desde otras fuentes, como documentos de negocio [3]. Algunos de los documentos de negocio son escritos en lenguaje natural, como documentos de procesos, documentos de requisitos, documentos de diseño, manuales de usuario, escenarios de casos de uso, entre otros, y éstos contribuyen al conocimiento del dominio [4]. Obtener y analizar la información presente en estos documentos es importante para comprender la evolución de un software, así como de su dominio [4].

Los principales inconvenientes que se presentan al procesar documentación dentro de una organización radican en la extracción misma de la información, ya que esta no se encuentra estructurada bajo marcos previamente definidos, lo cual conlleva a procesos de extracción de información repetitivos que dan como resultado requisitos que no cumplen las necesidades planteadas [5].

Con todo lo anterior, cualquier mecanismo que permita crear modelos que expliquen un dominio son bienvenidos, además en casos donde un analista de negocio sea nuevo en un proyecto o área funcional, estos modelos ayudan a educir y analizar requisitos [6].

Gracias al incremento en tamaño y complejidad de los sistemas y de la información misma, hay una gran demanda por enfoques inteligentes que aporten en el proceso de Ingeniería de requisitos [7], además, se cuenta con gran cantidad de información textual, pero los ingenieros de requisitos no tienen suficiente tiempo para leerla y analizarla; de ahí que las tecnologías para la extracción automática capaces de presentar la información de manera concisa, estén adquiriendo gran importancia [8]. Así, puede considerarse útil el uso de Procesamiento de Lenguaje Natural (NLP por sus siglas en inglés) dado que este tiene como entrada textos o lenguaje hablado [9] y utiliza diferentes técnicas con el propósito de lograr un procesamiento de lenguaje similar al humano para una gama de tareas o aplicaciones [10], lo que podría permitir agilizar el tiempo que toma leer y entender la información contenida en los textos.

La ejecución de una revisión sistemática de literatura evidencia que en los últimos años han surgido diversos estudios que proponen diferentes enfoques para la extracción y representación del conocimiento, los cuales ejecutan procesos de NLP para procesar información escrita en lenguaje natural con una preferencia muy marcada a procesar textos escritos en idioma inglés; además se destaca que dentro de los métodos de NLP usados para la extracción de la información hay una clara tendencia a la implementación del método *Part of Speech Tagging o PoS Tagging*. El conocimiento extraído es representado en la mayoría de trabajos mediante ontologías, las cuales requieren conocimientos previos para ser interpretadas, sin embargo un número importante de propuestas representan el conocimiento mediante

modelos UML, los cuales son bastante populares gracias a su baja curva de aprendizaje y sus notaciones visuales [11].

Con base en las tendencias antes descritas, el presente trabajo adopta *PoS Tagging* como método de extracción y modelos UML para la representación del conocimiento, planteando que aplicar técnicas de PLN a documentos de negocio escritos en idioma español permite extraer información y generar un modelo conceptual de una manera más rápida que el proceso manual realizado por un ingeniero de requisitos. Este proceso asegura que se mantenga la comprensión de la información del negocio contenida en los textos.

Este trabajo tiene como objetivo proponer un método de extracción de información desde documentos de negocio escritos en lenguaje natural en idioma español que permita generar un modelo conceptual a partir del conocimiento contenido en dichos documentos, para lo cual se adaptan varias herramientas que permiten realizar el pre-procesamiento de documentos de negocio para extraer elementos relevantes del texto, definir reglas de mapeo entre elementos identificados en el pre-procesamiento y elementos de un modelo conceptual, desarrollar una herramienta que permita transformar los elementos extraídos en elementos de un modelo conceptual y evaluar el método de extracción propuesto aplicándolo a documentos de negocio de una organización para identificar tiempo de procesamiento y nivel de interpretación de la información.

El documento presenta en el capítulo 2 el contexto general de la investigación describiendo el problema planteado, la pregunta de investigación, la hipótesis y los objetivos. Luego, en el capítulo 3, se muestra un análisis del estado del arte mostrando tendencias para solucionar el problema planteado. En el capítulo 4 se da el marco teórico de la investigación. En el capítulo 5 se plantea la propuesta de solución de este trabajo en la cual se describe en detalle el método planteado. En el capítulo 6 se plantea el diseño de la validación. Finalmente, en el capítulo 7 se presentan los resultados de la experimentación con su respectivo análisis y en el capítulo 8 las conclusiones y descripción del trabajo futuro derivado de esta investigación.

CAPÍTULO 2: CONTEXTO DE LA INVESTIGACIÓN

2.1. Planteamiento del problema

2.1.1. Contexto del problema

En el proceso de ingeniería de software, la fase de ingeniería de requisitos cobra gran importancia, teniendo en cuenta que la baja calidad en los requisitos representa una las mayores razones de fallas en un proyecto [7] [12] [2].

Un inconveniente mayor con los requisitos son los requisitos no declarados o implícitos, aquellos que los *stakeholders* asumen que los analistas quienes ejecutan el proceso de educción conocen [13], cobrando relevancia la obtención de información de negocio diferente a la declarada por los *stakeholders* desde fuentes textuales. Se consideran las siguientes como fuentes de texto de negocio escritos en lenguaje natural:

- Documentos de negocio: Muchos sistemas empresariales manejan un gran pool de documentos que forman parte de su negocio, los cuales típicamente contienen varias piezas de información que pueden tener un formato estructurado, pero generalmente contienen datos sin estructura en lenguaje natural [14].

- Documentos con información textual de software existente: Estos textos pueden ser documentos de requisitos, documentos de diseño, manuales de usuario, casos de uso, reportes de errores, mensajes de desarrolladores, etc. La obtención y análisis de esta información es extremadamente importante en la comprensión del programa y su soporte, y la evolución en las tareas de evolución del software [15]

La identificación y captura de los datos en el tiempo correcto, en el lugar correcto y en el formato correcto se ha convertido en una necesidad de las organizaciones modernas, lo cual es significativo para alcanzar el éxito del negocio [16].

2.1.2. Problema

Durante la fase de ingeniería de requisitos en el ciclo de vida de desarrollo de software se evidencian grandes retos dado que se debe obtener información de diferentes fuentes [17]; obtener esta información en forma parcial o no interpretarla en forma adecuada, puede derivar en baja calidad de los requisitos [17] [18]. El inconveniente principal radica en que las organizaciones cuentan con gran cantidad de información escrita en lenguaje natural, pero las personas no tienen suficiente tiempo para leerla y analizarla dado que los textos en lenguaje natural pueden ser muy extensos [19].

La utilización de técnicas de procesamiento del lenguaje natural (PLN o NLP por sus siglas en inglés) para el procesamiento de este tipo de documentos puede ayudar a mejorar el tiempo de los ingenieros de requisitos en leer y comprender la información allí contenida sin que se comprometa el nivel de interpretación de la misma.

2.2.Pregunta de investigación

¿Cómo se pueden aplicar técnicas de Procesamiento de Lenguaje Natural para la extracción de información textual desde documentos de negocio escritos en idioma español y hacer la especificación del conocimiento contenido en estos documentos en un modelo de representación de conocimiento?

2.3. Hipótesis

Aplicar técnicas de PNL a documentos de negocio escritos en idioma español permite extraer información y generar un modelo conceptual de una manera más rápida que el proceso manual realizado por un ingeniero de requisitos. Este proceso asegura que se mantenga la comprensión de la información del negocio contenida en los textos.

2.4. Objetivos

2.4.1. Objetivo general

Proponer un método de extracción de información desde documentos de negocio escritos en lenguaje natural en idioma español que permita generar un modelo conceptual a partir del conocimiento contenido en dichos documentos.

2.4.2. Objetivos específicos

1. Explorar la técnicas o métodos de NLP adecuados para extraer información de negocio desde documentos de negocio escritos en lenguaje natural.
2. Desarrollar o adaptar una o varias herramientas que permitan realizar el pre-procesamiento de documentos de negocio para extraer y etiquetar los elementos relevantes del texto.
3. Definir unas reglas de mapeo entre los elementos identificados en el pre-procesamiento y los elementos de un modelo conceptual.
4. Desarrollar una herramienta que permita transformar los elementos extraídos, etiquetados y relacionados, en elementos del modelo conceptual apoyado en las reglas de mapeo definidas.
5. Evaluar el método de extracción propuesto aplicándolo a documentos de negocio de una organización para identificar el tiempo de procesamiento y el nivel de interpretación de la información contenida en los mismos.

2.5. Alcances del trabajo

Con este trabajo se pretende procesar documentos de organizaciones escritos en lenguaje natural, para extraer los elementos más importantes y sus relaciones, utilizando técnicas que permitan obtener los elementos, etiquetarlos y relacionarlos, para luego mapear estos elementos y relaciones con elementos de un modelo conceptual en UML. El estado del arte demuestra que es posible el uso de modelos UML para representación de conocimiento de negocio, además, los modelos UML son bastante populares gracias a su baja curva de aprendizaje y sus notaciones visuales [11], por lo que la propuesta acá presentada adopta este enfoque. Finalmente, el uso de una herramienta permite mostrar gráficamente el modelo conceptual generado.

PARTE II:

FUNDAMENTOS TEÓRICOS

CAPÍTULO 3. REVISIÓN DE LITERATURA

3.1. Estado del arte

Esta revisión de estado del arte se ejecutó basada en la metodología de revisión sistemática presentada por Beltrán [20], la cual propone los siguientes pasos básicos:

- Definición clara del problema.
- Especificación de los criterios de inclusión y exclusión de los estudios.
- Formulación del plan de búsqueda de la literatura.
- Registro de los datos y evaluación de la calidad de los estudios seleccionados.
- Interpretación y presentación de los resultados.

Partiendo del problema planteado anteriormente, se incluyen artículos que cumplen con los siguientes criterios de inclusión:

- Solo se considera los artículos de trabajo primario basados en procesamiento del lenguaje natural.
- Se incluyen únicamente los artículos publicados a partir del año 2012.
- Los trabajos considerados deben presentar algún modelo de representación de conocimiento por medio de cualquier modelo, preferiblemente en el área de ingeniería de software, sin embargo, no se descartan enfoques en otras áreas de investigación.
- Los artículos seleccionados deben utilizar fuentes de información desde fuentes organizacionales.

Los artículos se buscaron en las siguientes bases de datos con las cadenas de búsqueda que se presentan en la Tabla 1:

Tabla 1. Bases de datos y cadenas de búsqueda

Base de datos	Cadena de búsqueda
ACM	*natural laguage processing AND business process*
	natural laguage processing AND spanish
IEEE XPLORE	*natural language process AND business process*
	natural language process AND spanish
Science direct - elsevier	*natural Language Processing AND business (Computer Science)*
	natural Language Processing AND spanish (Computer Science)
Spriner Link	*natural language processing AND business process*
	natural Language Processing AND spanish

Además de las búsquedas en las bases de datos mostradas en la tabla 1, se ejecutó una búsqueda manual en las publicaciones de la sociedad española de procesamiento de lenguaje natural –SEPNL-, y en algunos casos, se tomaron artículos referenciados por trabajos primarios o secundarios resultado de las búsquedas ejecutadas, siempre garantizando que los trabajos seleccionados cumplen con los criterios de inclusión.

En la primera búsqueda de documentos se hizo lectura de los títulos de los trabajos resultantes de las búsquedas en cada motor y se obtuvo un número de documentos, que luego fueron depurados mediante lectura de resumen para obtener la primera pre-selección, posteriormente se hizo una revisión de las secciones Introducción y Conclusiones para llegar a la pre-selección, finalmente se hizo una lectura completa de documentos para llegar a una cantidad final. En la tabla 2 se evidencian las cantidades de documentos obtenidos en la depuración antes descrita.

Tabla 2. Cantidad de trabajos resultantes en revisión de literatura

Inicial	Pre-selección 1	Pre-selección 2	Finalmente
64	38	21	17

A continuación, se presentan los resultados obtenidos de la revisión, organizados en dos categorías, la primera muestra trabajos que presentan un modelo de representación, en la segunda trabajos que procesan documentos de negocio sin entregar un modelo específico de representación:

- **Propuestas que entregan un modelo de representación de conocimiento desde información de negocio escrita en lenguaje natural.**

  Diamantopoulos, Roth y Symeonidis [21] presentan la generación de una ontología a partir de requisitos funcionales escritos en inglés generando un corpus lingüístico representado mediante una ontología. Aplica un análisis semántico usando *tokenization, POS tagging, lemmatization* y *dependency parsing*.

  Iqbal y Bajwa [22] proponen la generación de diagramas de actividades UML a partir de requisitos transformados manualmente de lenguaje natural a notación SBVR (*Semantic of Business Vocabulary and Rules*) mediante *PoS (Part of Speech)*. Su fuente de información son reglas en notación SBVR para extraer información ingresada en idioma inglés. Esta propuesta requeriría intervención manual para el pre-procesamiento de los documentos.

  En el trabajo de Bhala y otros[13] se propone la generación automática de un modelo conceptual desde requisitos escritos en lenguaje natural en idioma inglés. Se pre-procesa el texto "tokenizando" las frases, luego se ejecuta el análisis sintáctico: se ejecuta *Deep parse* y P*OS tagging*, para luego, apoyados en el corpus de *WordNet*, extraer ocurrencias (pronombres, verbos, adjetivos, adverbios, etc.). Este trabajo es una buena aproximación a la solución del problema planteado, sin embargo, requeriría intervención para su aplicación al idioma español.

  El aporte de Annervaz, Kaulgud, Sengupta y Savagaonkar [6] se orienta hacia procesar requisitos escritos en lenguaje natural y un modelo de dominio en Excel escrito en forma manual, se ejecuta análisis semántico y sintáctico para hacer mapeo entre los requisitos descritos y el modelo de dominio; estas similitudes se llevan a una ontología que finalmente es graficada. El mayor problema de este trabajo es que depende de intervención manual previa para poder usar el modelo presentado.

Lo más interesante de los trabajos anteriores es que permitirían procesar documentos de requisitos de software existente para entender cómo funciona dicho software.

De Araujo, Rigo, Muller y Chrishman [23] generan una propuesta de extracción de información de texto escrito en lenguaje natural que combina conocimiento de negocio con conocimiento lingüístico. El método permite poblar una ontología con instancias de eventos representada mediante OWL, aplicando la técnica de análisis semántico a partir de texto escrito en lenguaje natural, para extraer información ingresada en idioma portugués.

El aporte de Roychoudhury [14] consiste en la verificación automática de documentos de dominios de negocio, enfocado en: 1) crear conocimiento a través de una ontología y 2) proveer un framework de automatización para extraer definición formal del texto escrito en lenguaje natural. Se aplica una técnica de Análisis Gramatical desde *Machine Learning* a reglas específicas escritas en lenguaje natural en idioma inglés para generar reglas en una ontología del dominio. En este caso se requiere intervención manual para hacer una pre-transformación del documento de negocio, también en lenguaje natural, pero con reglas específicas.

Xiao [19] propone la extracción de reglas de ACP (*Access Control Policy*) a partir de documentos en lenguaje natural aplicando análisis sintáctico, para representarlas al final en XACML (*eXtensible Access Control Markup Language*).

El aporte de Lee [24] consiste en un framework para generación de restricciones en OCL (*Object Constraint Language*) desde el texto en lenguaje natural entregado por el usuario, aplicando *POS Tagging*. Se procesa el texto para transformarlo en un modelo de restricciones (PoS a restricciones en SBVR), las que luego se convierten en expresiones OCL.

Los dos aportes anteriores permitirían tomar documentación de reglas negocio para ser transformadas en un lenguaje etiquetado, el principal inconveniente es que requiere que el texto se escriba en forma controlada, lo que podría requerir pre-procesar los documentos para ser usados por los modelos presentados.

Pinquie [25] propone la utilización de técnicas de NLP a requisitos escritos en lenguaje natural en idioma inglés, para generar PPBR (*Parametric Property-Based Requirements*). El método procesa documentos en diferentes extensiones, aplicando técnicas de análisis sintáctico y semántico para generar un XML con la especificación de *PPBRs*. Este trabajo en particular se acerca mucho a solucionar el problema planteado en esta propuesta, sin embargo, está aplicado al idioma inglés y el modelo de representación entregado puede requerir ciertos conocimientos adicionales para ser interpretado por los analistas.

El trabajo publicado por Sawant y otros [15] consiste en forzar la estructura en casos de uso potencialmente sin estructura extrayendo el modelo de oraciones comúnmente encontradas en casos de uso de la industria. La fuente de información son casos de uso escritos en inglés, en un formato con reglas

específicas. Finalmente, se representa lo extraído en un modelo anotado en XML.

Fernandes, Furquim y Lopes [26] presentan un método para extraer términos específicos de un dominio específico, este procesa texto escrito en lenguaje natural en idioma portugués. Al final, luego de aplicar análisis sintáctico y semántico se genera una ontología del dominio procesado, entregando léxico propio de este.

Tabla 3. Idioma y modelo de representación presentado

Referencia	Idioma del texto procesado	Modelo presentado
[23]	Portugués	Ontología
[26]	Portugués	Ontología
[6]	Inglés	Ontología
[14]	Inglés	Ontología
[21]	Inglés	Ontología
[13]	Inglés	Modelo UML
[22]	Inglés	Modelo UML
[24]	Inglés	Modelo UML
[15]	Inglés	Modelo propio
[19]	Inglés	Lenguaje etiquetado
[25]	Inglés	Lenguaje etiquetado

Como se evidencia en la tabla 3, el modelo de representación dominante son las ontologías y/o lenguajes etiquetados, con siete trabajos. Tres de los trabajos representan la información extraída en modelos UML, sin embargo, ninguno de estos se procesa desde el idioma español.

Si bien el uso de ontologías marca una tendencia, los modelos UML son bastante populares gracias a su baja curva de aprendizaje y sus notaciones visuales [11]. Tomando en cuenta lo anterior y basados en que, como se ve en la Tabla 2, es posible representar información de negocio en modelos UML, la propuesta acá presentada adopta este enfoque, de modo que se genere un modelo de dominio representado en un modelo conceptual.

- **Propuestas que aplican NLP para extracción de conocimiento desde información de negocio escrita en lenguaje natural, sin generar modelo de representación.**

  Los trabajos que se presentan a continuación, si bien no generan un modelo de representación como tal, procesan información de texto en lenguaje natural desde documentos de negocio o se procesa información en español, por lo que, en cierta medida, estarían aportando a la solución del problema planteado en el presente trabajo.

  El aporte de Escartín [27] consiste en la comparación de diferentes aplicaciones *PoS tagger* comparando los resultados arrojados por cada uno

para proponer el que mejor resultado arroje como un estándar. Se aplica los *PoS tagger* a documentos escritos en lenguaje natural en idioma español para extraer información y comparar los resultados. Si bien este trabajo no presenta un modelo de representación específico, es el único que aplica NLP directamente a textos escritos en idioma español.

El aporte de Gaw [28] consiste en el procesamiento de reglas de negocio expresadas en lenguaje natural controlado, validando si los métodos generados en código fuente corresponden a una regla de negocio. Aplica análisis semántico usando como fuente de información reglas de negocio escritas en lenguaje natural controlado (*SBVRSE*) para extraer información ingresada en idioma inglés y generar su representación en Lenguaje DRL, que puede ser comparado con los métodos generados en código fuente en un programa. Lo más relevante de este trabajo es que procesa documentos de reglas de negocio, sin embargo, requiere que estas se escriban en un lenguaje natural controlado, por lo que no cualquier documento de reglas puede aplicarse.

El trabajo de Aa, Leopold y Reijers [29] presenta la detección automática de inconsistencias entre descripciones textuales de procesos y modelos de procesos, combinando análisis lingüístico, medidas de similitud semántica y orden de relaciones, aplicando análisis semántico.

Miranda [30] propone un nuevo modelo para la generación de resúmenes de un documento basado en grafos conceptuales aplicando el analizador *Parser* de Stanford, su fuente de información son textos escritos en idioma inglés. Se generan árboles de dependencia usando un analizador sintáctico y aplicando relaciones conceptuales se generan en forma manual los grafos conceptuales, a partir de los que se genera texto resumen.

Movshovitz y Cohen [31] proponen el análisis del uso de lenguajes de modelo para predecir comentarios de documentos que contienen mezcla de código fuente y texto, aplicando modelos estadísticos y análisis semántico a código fuente comentado para extraer información ingresada en idioma inglés. Los autores no especifican cómo representar los resultados obtenidos.

El trabajo de Naeem [16] consiste en la generación de *Queries OLAP* a partir de texto escrito en lenguaje natural controlado aplicando Stanford POS *tagger*, *Stanford parser*, y un analizador semántico propio basado en vocabulario SBVR (*Semantic of Business Vocabulary and Business Rules*).

Tabla 4. Idioma y método de extracción de información empleado

Referencia	Idioma del texto procesado	Método extracción principal
[19]	Ingles	*PoS Tagging*
[16]	Ingles	*PoS Tagging*
[13]	Ingles	*PoS Tagging*
[15]	Ingles	*PoS Tagging*
[21]	Ingles	*PoS Tagging*
[22]	Ingles	*PoS Tagging*
[24]	Ingles	*PoS Tagging*

[25]	Ingles	*PoS Tagging*
[27]	Español	*PoS Tagging*
[6]	Ingles	*Parsing*
[30]	Ingles	*Parsing*
[29]	Ingles	*Parsing*
[23]	Portugués	*Parsing*
[26]	Portugués	Herramienta de mapeo
[28]	Ingles	Herramienta de mapeo
[31]	Ingles	Modelo estadístico
[14]	Ingles	*Machine Learning*

Es interesante que, como se muestra en la tabla 4, 14 de las propuestas se aplican al idioma inglés, dos se aplican a portugués y solo uno al español. Teniendo en cuenta que el trabajo aplicado a español no entrega modelo de representación, como aporte a la solución del problema acá planteado no fue encontrado un trabajo que procese lenguaje español y que presente un modelo de representación.

De la Tabla 3 también se puede inferir que el método principal utilizado para el procesamiento de lenguaje es el método de separación en *PoS Tagging* con 9 trabajos, 4 de ellos utilizan *Parsing*, 3 de ellos una herramienta de mapeo, solo uno de los trabajos aplica *machine learning* y otro aplica un modelo estadístico. Esto nos da una tendencia importante de lo que se puede aplicar para la solución del problema, además, el único trabajo en español precisamente utiliza el método *PoS Tagging*.

CAPÍTULO 4. MARCO TEÓRICO

4.1.Marco teórico

4.1.1. Natural Language Processing

Procesamiento del lenguaje natural (PNL o NLP por sus siglas en inglés) [9] se refiere a sistemas informáticos que analizan, intentan comprender o producen uno o más lenguajes humanos, como inglés, japonés, italiano o ruso. La entrada puede ser texto, lenguaje hablado o entrada de teclado. La tarea podría ser traducir a otro idioma, comprender y representar el contenido del texto, construir una base de datos o generar resúmenes, o mantener un diálogo con un usuario como parte de una interfaz para la recuperación de información de una base de datos.

Un método explicativo para presentar lo que realmente sucede en un sistema de Procesamiento de Lenguaje Natural es mediante el enfoque de 'niveles de lenguaje' [10]:

- Fonología: Este nivel se ocupa de la interpretación de los sonidos del habla dentro y a través de las palabras.
- Morfología: Este nivel trata de la naturaleza componencial de las palabras, que se componen de morfemas -las unidades más pequeñas de significado.
- Lexical: En este nivel, los seres humanos, así como los sistemas de NLP, interpretan el significado de las palabras individuales.
- Sintáctico: Este nivel se centra en analizar las palabras en una oración para descubrir la estructura gramatical de la oración.
- Semántica: El procesamiento semántico determina los significados posibles de una oración centrándose en las interacciones entre los significados de nivel de palabra en la oración.
- Discurso: Mientras que la sintaxis y la semántica funcionan con unidades de longitud de oración, el nivel de discurso de NLP trabaja con unidades de texto más largas que una oración
- Pragmático: este nivel se refiere al uso intencional del lenguaje en situaciones y utiliza el contexto más allá de los contenidos del texto para la comprensión.

La dificultad principal en el procesamiento del lenguaje natural es la ambigüedad omnipresente que se encuentra en todos los niveles del problema. Por ejemplo, todos los lenguajes naturales implican [9] :

- La ambigüedad léxica simple (por ejemplo, "*duck*", en idioma inglés puede ser un sustantivo [el animal] o un verbo [para evitar algo lanzado]).
- La ambigüedad estructural o sintáctica (por ejemplo, en "Vi al hombre con un telescopio", el telescopio podría ser usado para la observación o podría ser sostenido y el hombre observado).
- La ambigüedad semántica (por ejemplo, "ir" como un verbo tiene más de 10 significados distintos en cualquier diccionario).
- La ambigüedad pragmática (por ejemplo, "¿Se puede levantar la roca del sombrero?" Puede ser una pregunta de SÍ / NO o una petición para levantar la roca).

- La ambigüedad referencial (por ejemplo, "Jack conoció a Sam en la estación, se sentía enfermo...", no está claro quién está enfermo, aunque el resto de la oración podría sugerir una interpretación preferida).

4.1.2. *Part-of-speech tagging* (POS *tagging*)

Es un proceso en el cual cada palabra en un texto se asigna a su categoría morfosintáctica apropiada (por ejemplo sustantivo-singular, verbo-pasado, adjetivo, pronombre, y similares) [32]. Por lo tanto, proporciona información sobre la morfología (estructura de las palabras) y la sintaxis (estructura de las oraciones). Este proceso de desambiguación se determina tanto por las restricciones del léxico (¿cuáles son las posibles categorías para una palabra?), como por las restricciones del contexto en el que se produce la palabra (¿cuál de las categorías posibles es la correcta en este contexto?). En una oración como "Ponlo sobre la mesa", el hecho de que la tabla esté precedida por el determinador, es una buena indicación de que se usa como un sustantivo aquí. Los sistemas que asignan automáticamente partes del discurso a las palabras en el texto deben tener en cuenta tanto las restricciones léxicas como contextuales, y se encuentran típicamente en implementaciones como un módulo de búsqueda y un módulo de desambiguación.

4.1.2.1. Aplicaciones.

Un *POS tagger* es el primer módulo de desambiguación en sistemas de análisis de texto. Para determinar la estructura sintáctica de una oración (y su semántica), debemos conocer las partes de la oración de cada palabra. En enfoques anteriores al análisis sintáctico (*parsing*), el *POS tagging* fue parte del proceso de análisis sintáctico, sin embargo, los *POS tagger* individualmente entrenados y optimizados se han convertido cada vez más en un módulo separado en sistemas de análisis sintácticos poco profundos o profundos. Como extensión, el *POS tagging* es también un módulo fundamental en aplicaciones de minería de textos que van desde la extracción de información y la extracción de terminología / ontología hasta el resumen y la respuesta a preguntas.

Aparte de ser uno de los primeros módulos de cualquier sistema de análisis de texto, el *POS tagging* también es útil en estudios lingüísticos (lingüística de corpus), por ejemplo, para calcular frecuencia de palabras desambiguadas y estructuras superficiales sintácticas. En la tecnología del habla, conocer la parte del discurso de una palabra puede ayudar en la síntesis del habla y en el reconocimiento de voz. En la corrección ortográfica y gramatical, el *POS tagging* juega un papel en el aumento de la precisión de tales sistemas.

4.1.2.2. Set de etiquetas Part-of-speech

El inventario de etiquetas POS puede variar de decenas a cientos dependiendo de la riqueza de la morfología y la sintaxis que se representa y de la complejidad morfológica inherente de un idioma. Los conjuntos de etiquetas se desarrollan con mayor frecuencia en el contexto de la construcción de corpus anotados. Ha habido esfuerzos para estandarizar la construcción de conjuntos de etiquetas para aumentar la traducción entre diferentes conjuntos de etiquetas.

4.1.3. Análisis léxico y *Parsing*

El análisis léxico es un proceso de conversión de una secuencia de caracteres a una secuencia de tokens; estos tokens son analizados por un *parser* para encontrar significados [33]. Por ejemplo, para la expresión (producto = 4 * 5), en la tabla 5 se presentan los tokens para esta expresión:

Tabla 5. Ejemplo de tokenización

Token	**Categoría**
producto	identificador
=	operador de asignación
4	Entero
*	operador de multiplicación
4	Entero

Siguiendo con el ejemplo, en el proceso de análisis léxico que identifica los cinco tokens que se muestran en la Tabla 4; lo siguiente que se debe identificar es si se trata de una expresión aritmética, una expresión lógica, o algo más, las herramientas de *parsing* se pueden usar para hacer tales identificaciones de modo que se puede encontrar el significado de los datos dados. Una vez encontrados los significados se puede procesar más, por ejemplo, en este caso al saber que es una expresión aritmética se podría procesar la expresión misma para encontrar el resultado.

4.1.4. Modelado Conceptual

El modelado conceptual consiste en describir la semántica de las aplicaciones de software en un alto nivel de abstracción [34]. Específicamente, los modeladores conceptuales (1) describen modelos de estructura en términos de entidades, relaciones y restricciones; (2) describen modelos de comportamiento o funcionales en términos de estados, transiciones entre estados y acciones realizadas en estados y transiciones; y (3) describen interacciones e interfaces de usuario en términos de mensajes enviados y recibidos, intercambio de información y apariencia [34].

En su uso típico, los diagramas de modelos conceptuales son abstracciones de alto nivel que permiten a los clientes y analistas entenderse entre sí y permitir a los analistas comunicarse con éxito con los programadores de aplicaciones. Es un desafío proporcionar con éxito el conjunto correcto de constructos de modelado en el nivel correcto de abstracción para permitir esta comunicación.

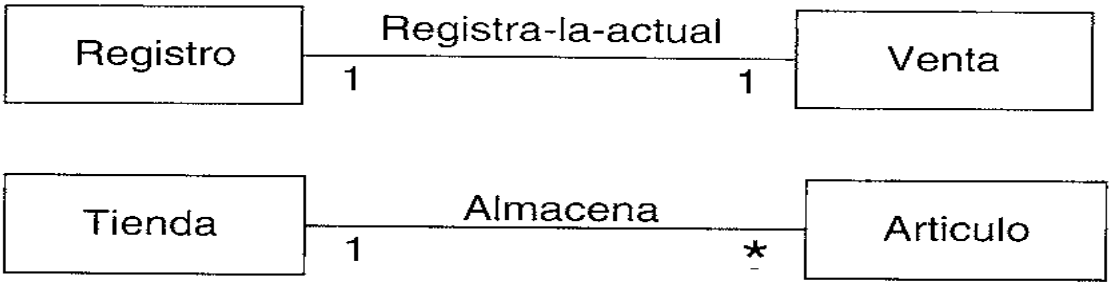


Figura 1. Ejemplos de modelo conceptual [35]

Es un desafío añadido formalizar estas abstracciones de modelado para que conserven su propiedad de facilidad de comunicación y, sin embargo, sean capaces de (parcial o totalmente) generar software de aplicación en funcionamiento.

También es un reto impulsar el modelado conceptual hacia el servir como herramientas de análisis y desarrollo para aplicaciones exóticas, como modelar las características computacionales de la vida a nivel de ADN o modelar la capacidad de leer y extraer información de texto de forma libre. Un desafío central del modelado conceptual es facilitar el sueño de largo plazo de ser capaz de desarrollar sistemas de información estrictamente mediante el modelado conceptual.

4.1.5. Metodología de investigación "*Design Science in Information Systems Research*"

La ciencia basada en el diseño [36] es un paradigma de resolución de problemas en investigaciones de informática y ciencias de la computación cuyo objetivo es contribuir en la solución de problemas relevantes al mismo tiempo que hacer aportes significativos a un área del conocimiento, mediante el análisis de problemas aún no resueltos en un ambiente del mundo real y su resolución de una manera novedosa.

Este paradigma establece el desarrollo de un proceso de investigación basado en tres ciclos, así:

- El ciclo de *relevancia*, establece un problema del mundo real sobre el cual se pueda aplicar la solución que se plantea diseñar, dando como insumo al diseño los requerimientos del problema.
- El ciclo de *rigor*, brinda el conocimiento existente, tomado en su mayoría de la literatura científica, al diseño de la solución, aportando conocimiento existente o metodologías que puedan ser aprovechadas.
- El ciclo de *diseño* toma como insumo los resultados de las fases de rigor y relevancia para generar nuevo conocimiento pertinente a un problema real.

4.2. Diseño Metodológico

El presente trabajo se desarrolla basado en la metodología propuesta por Herver y compañía [36], el cual define tres fases, así:

4.2.1. Fase 1. Análisis del entorno (Relevancia)

- Identificar el problema a abordar teniendo en cuenta el planteamiento de diversos investigadores.
- Plantear una pregunta de investigación que dirige la forma en que se ejecutan las demás actividades.
- Plantear una hipótesis que, una vez validada, permita responder a la pregunta de investigación.

4.2.2. Fase 2. Análisis de la base del conocimiento (Rigor)

- Realizar una revisión del estado del arte que permite ver los trabajos que han intentado resolver un problema similar al planteado en este trabajo, e identificar herramientas y metodologías que se pueden usar para solucionar el problema específico acá planteado.
- Explorar técnicas o métodos de NLP para realzar la extracción de información de documentos de negocio escritos en lenguaje natural en idioma español, de acuerdo a los resultados de la revisión del estado del arte. Con esta actividad se cumple el objetivo específico 1.
- Una vez identificadas esas herramientas y metodologías, describir las teorías que las sustentan y que sirven como base para el diseño.

4.2.3. Fase 3. Diseño y Validación del modelo (validación)

Las actividades a implementar para responder a los objetivos propuestos son las siguientes:

- Objetivo específico 2:
  - Implementar y utilizar una herramienta que permita leer documentos en formato .pdf, tomando sólo texto escritos en lenguaje natural.
  - Implementar, adaptar y utilizar una o varias herramientas que permitan usar el método *PoS*.
  - Desarrollar o utilizar una herramienta que permita generar relaciones entre los elementos extraídos.

- Objetivo específico 3:
  - Definir una serie de reglas que permitan realizar un mapeo entre los elementos extraídos y etiquetados del texto con elementos en un modelo conceptual.

- Objetivo Específico 4:
  - Desarrollar una herramienta que permita transformar los elementos extraídos, etiquetados y relacionados en elementos de modelo de dominio, apoyado en el modelo o meta-modelo que se construya.
  - Implementar, adaptar y utilizar una herramienta para la representación gráfica del modelo transformado.

- Objetivo Específico 5:
  La validación comprende las siguientes actividades:
  - Definir el experimento o caso de estudio. En esta se define el dominio en el cual aplicar el método, los documentos que conformarán el corpus inicial que requiere el método y se eligen los analistas que ejecutarán los ejercicios para evaluar el método.

- Diseñar los instrumentos. En esta se genera un cuestionario basado en cada uno de los documentos entregados a los dos grupos de ingenieros de requisitos.
- Ejecutar un experimento con dos grupos de ingenieros de requisitos a los cuales se les entrega varios documentos de negocio escritos en leguaje natural, así:
  - El primero realiza lectura directa y genera un modelo conceptual basado en lo que leyeron.
  - El segundo utiliza las herramientas que se desarrollen para procesar los documentos y realizan lectura del modelo resultante.
- Medir el tiempo que les toma hacer toda la operación en ambos grupos, además al final contestan el cuestionario acerca del texto, para obtener:
  - Comparativo de Tiempo de interpretación aplicando técnica de PLN vs Tiempo de interpretación aplicando lectura directa.
  - Comparativo del nivel de interpretación del texto aplicando técnicas de NLP vs Nivel de interpretación aplicando lectura directa. Se mide contestando el cuestionario acerca del texto, tomando las respuestas correctas.

  De acuerdo a la hipótesis se espera que el tiempo sea menor procesando el documento, que leyéndolo. Además, que la cantidad de respuestas correctas al final sea similar en ambos casos.

Estas fases y cada una de las actividades se ejecutan en forma iterativa y no precisamente en el orden acá establecido, así, en cada parte del trabajo se puede refinar ejecutando nuevamente actividades previamente ejecutadas.

PARTE III:

PROPUESTA DE SOLUCIÓN

CAPÍTULO 5. PROPUESTA DE SOLUCIÓN

5.1. Consideraciones de la Propuesta

A partir de la hipótesis planteada, la solución propuesta se centra en un método de extracción de información desde documentos de negocio escritos en lenguaje natural en idioma español y su representación en un modelo conceptual. El método se presenta a partir de las siguientes consideraciones:

- Teniendo en cuenta la tendencia evidenciada en la revisión de literatura, donde la técnica base de PLN a implementar más apropiada es el *Part Of Speech Tagging (POS Tagging)*, el método aplica esta técnica para extraer elementos específicos del texto como sustantivos, verbos, y adverbios. Así, es posible realizar un mapeo entre los elementos extraídos y un modelo de representación.

- La representación de la información se plantea mediante el uso de modelos conceptuales, teniendo en cuenta que estos muestran conceptos significativos en un dominio [35] y permiten ser representados en diagramas UML. Estos diagramas son bastante populares gracias a su baja curva de aprendizaje y sus notaciones visuales [11]. Es importante anotar que los modelos resultantes se presentan típicamente como modelos de estructura, dado que se dan en términos de entidades y relaciones [34], sin embargo puede darse que, de acuerdo al contenido de los documentos procesados, se presenten como modelos de comportamiento o modelos de interacciones e interfaces de usuario.

- Finalmente, el método se sustenta en varias herramientas de software que permiten, apoyadas en herramientas existentes, aplicar la técnica definida y generar el modelo de representación antes especificado.

5.2 Marco metodológico de referencia

La propuesta de solución planteada en este trabajo se orienta bajo el marco metodológico propuesto por Manrique y otros [37] para desarrollar proyectos de analítica de texto desde documentos de requisitos. La propuesta metodológica se especifica y representa en la figura 2. De dicha especificación se perciben tres tipos de artefactos que podrían ser los posibles insumos (documentos de entrada en el proceso de ingeniería de requisitos): documento de negocio, documento de requisitos e historias de usuario. Los artefactos resultantes del proceso son el documento de contexto del proyecto, el corpus, el modelo, el análisis de resultados y el informe final del proyecto. En cuanto a las herramientas a usar se perciben tres tipos fundamentales, los cuales para este tipo de proyectos son automáticas:

- Preparador (pre-procesador): se encarga de tomar todos los documentos que conforman el corpus y unificarlos en un solo documento. En este caso se usa o adapta herramientas de software que permiten compilar documentos, además del entrenamiento de algoritmos de NLP o *Text Mining*, usando el corpus generado transformándolo en uno propio para cada caso según lo requiera el compilador.

- Compilador (procesador): se encarga de hacer el análisis sintáctico, gramatical, semántico, entre otros; y extraer los conceptos más relevantes del negocio. Entrega como resultado una especificación en lenguaje controlado (i.e. ontología). En este se configuran herramientas de software que realizan procesamiento de lenguaje natural o *Text Minig*, como Freeling, *Stanford Core NLP* y otros, las cuales permiten ejecutar procesos como *POS Tagging*, *Lemmatization*, y demás.

- Modelador: se encarga de tomar los conceptos de la especificación de salida y representarlos en un modelo más estructurado. Normalmente corresponde herramientas de software construidas en cada proyecto, las cuales permiten transformar la especificación en diagramas que pueden ser vistos gráficamente desde herramientas como *Enterprise Architect* o *yEd Graph Editor*.

Los roles implicados son:

(i) director del proyecto, que se encarga del 'documento de contexto del proyecto' para definir los objetivos y el alcance;

(ii) documentador, que se encarga de las actividades centrales que permiten generar el modelo a partir de los documentos, y

(iii) implementador, que se encarga de verificar que el modelo se ajuste a las necesidades de los desarrollos en los que se utiliza y hacer el respectivo seguimiento del uso de los modelos resultantes.

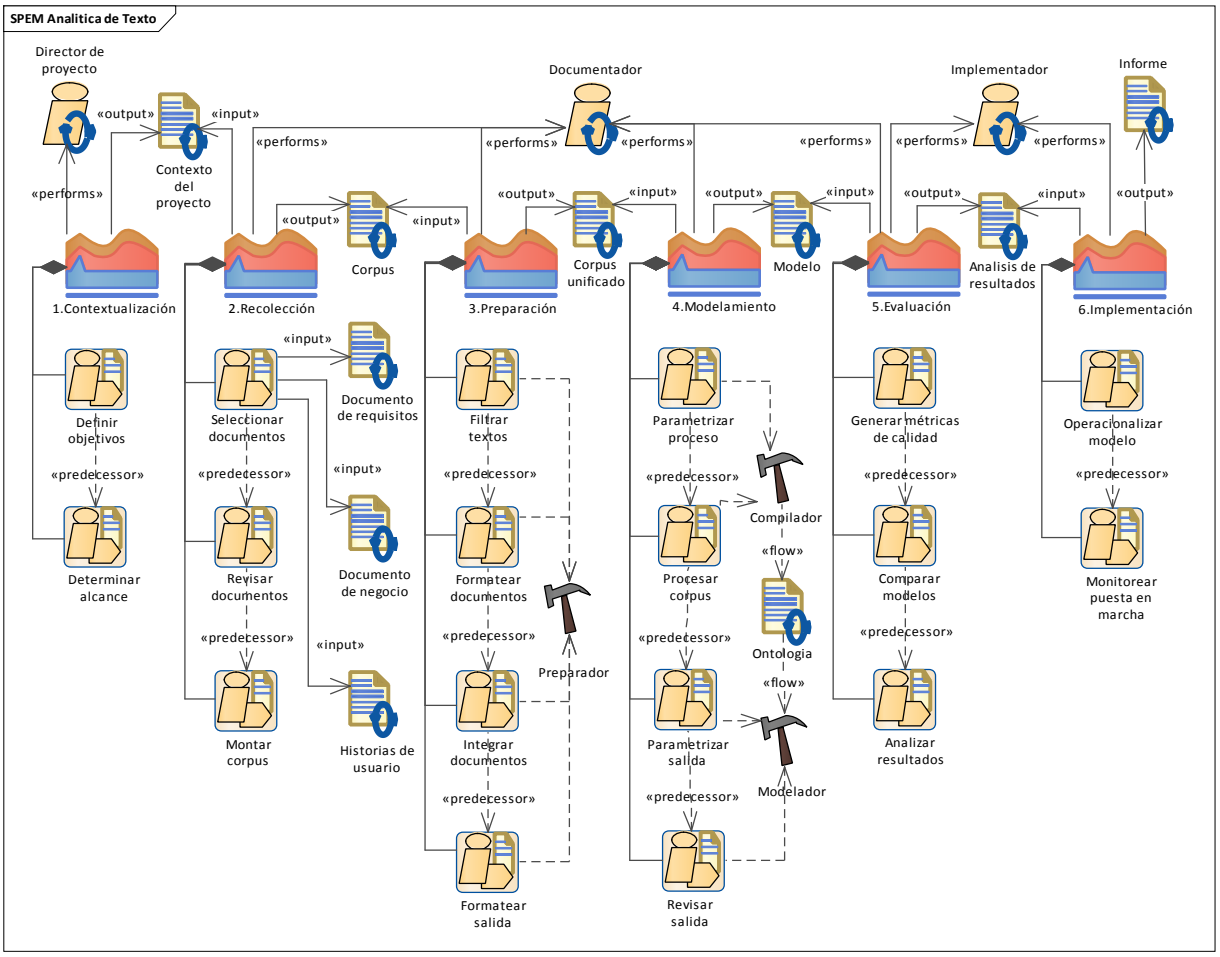


Figura 2. Propuesta metodológica para desarrollo de proyectos de analítica de texto

5.3 Método de extracción propuesto

El marco de referencia es tomado como base gracias a que permite organizar las actividades propias de proyectos que responden a la línea de investigación seguida en el presente trabajo, además es un marco presentado en el contexto local, donde, de acuerdo a lo determinado en el estado del arte, hay poco desarrollo de proyectos de NLP, lo que permite que el caso de estudio acá presentado sirva como validación del marco mismo.

Teniendo en cuenta que este se trata de una propuesta experimental, es necesario establecer las siguientes consideraciones con respecto a la propuesta metodológica en la cual se basa:

- No se cuenta con roles claramente definidos, en este caso existe un único rol que ejecuta todo el proyecto.
- Al ser experimental, no se implementará la etapa 6 de Implementación. En la fase 5 de Evaluación se tomarán los datos necesarios para este trabajo.
- El presente trabajo está planteado de forma que se preparan las herramientas de software para procesar un documento de negocio cada vez, mientras la

propuesta metodológica base plantea el procesamiento de muchos documentos a la vez.

En la figura 3 se muestra el modelo general del método antes explicado. A continuación, se describe el método de extracción usado en el presente trabajo, organizado en las fases definidas en el marco metodológico de referencia.

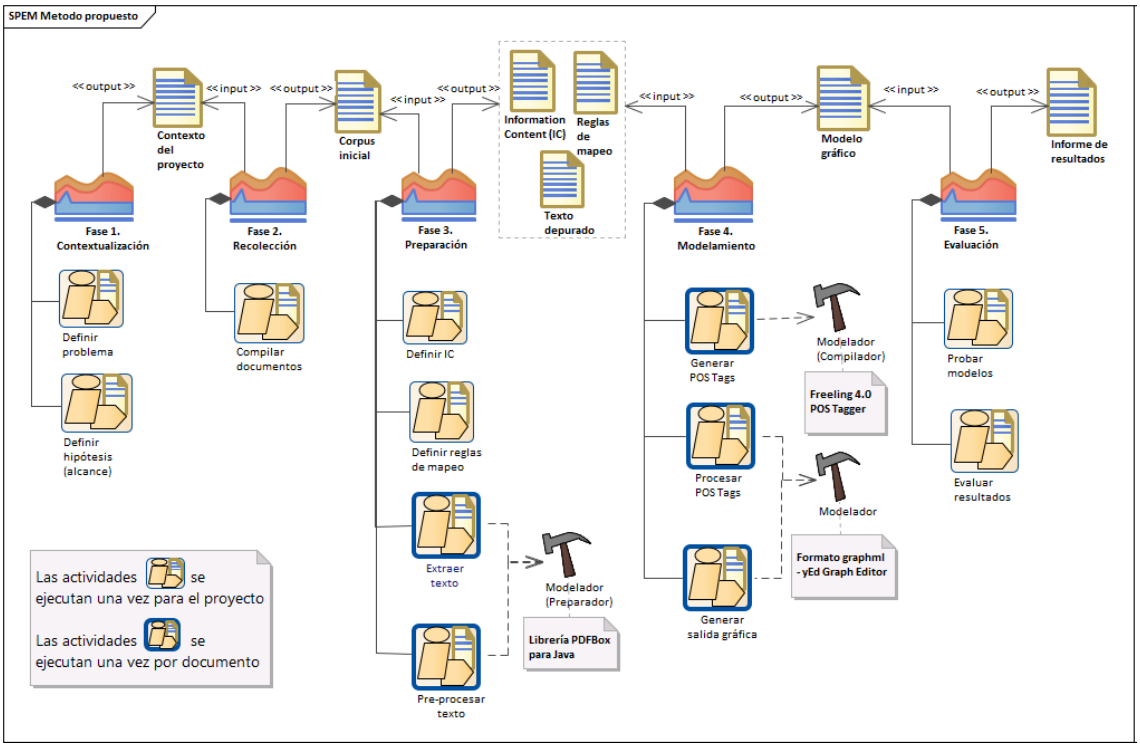


Figura 3. Método propuesto para generar un modelo conceptual a partir de un documento de negocio

Fase 1. Contextualización:

Esta base está basada en la definición del problema y la definición de la hipótesis. El objetivo consiste en el análisis automático de documentos de negocio para la extracción de información relevante y su representación en un modelo conceptual.

En el caso de estudio abordado en este trabajo en particular se toma como artefacto output lo contenido en el CAPÍTULO 2: CONTEXTO DE LA INVESTIGACIÓN.

Fase 2. Recolección:

En esta fase se realiza la compilación de "documentos de negocio". En esta se determina los documentos que sirven como base para el modelo y desde los cuales se obtiene términos relevantes para el dominio en particular, y se identifican los documentos que se usarán en la fase de evaluación.

El caso de estudio en particular, y tomando como input el contexto mencionado en la fase anterior, determinó el uso de documentos de resolución emitidos por la Comunicación de Regulación de Comunicaciones CRC, los cuales son de dominio público, estos son considerados como el corpus de documentos output, conformado así:

- Resolución 5111 de 2017
- Resolución 5199 de 2016
- Resolución 4841 de 2015
- Resolución 4891 de 2016
- Proyecto de resolución 5397 de 2018
- Proyecto de resolución 4930 de 2016
- Proyecto de resolución 5161 de 2016

Fase 3. Preparación:

En esta se ejecutan dos labores fundamentales, la primera aplica para la preparación de las herramientas de software para procesar cada documento, así:

- Determinar las herramientas más idóneas para la implementación.
- A partir de los documentos base para el modelo, se toman los términos de negocio más importantes y se les da un peso de negocio de 1 a 10, donde 1 es menos importante y 10 es más importante. Estos términos servirán posteriormente para priorizar elementos en el modelo conceptual resultante. Esta clasificación es conocida como *Information content (IC),* siendo una métrica que denota la importancia de un término en un corpus o en un dominio [38], enmarcada en la técnica de procesamiento de texto *Bag Of Words*. La información de los IC de cada término es almacenada en un archivo plano de modo que éste pueda ser usado en la herramienta de software.

Luego inicia la ejecución de los pasos 1 y 2 para la generación del modelo conceptual para cada documento de negocio y el uso de las herramientas de software. En esta se realiza el filtrado y formateo del texto, de modo que se eliminen imágenes, encabezados y pie de páginas, tablas, entre otros.

En el caso de estudio particular se tomó los siguientes documentos para determinar los sustantivos que más se repiten en el dominio específico.

- Resolución 5111 de 2017. Tomado a partir de la página 9
- Resolución 5199 de 2016
- Resolución 4841 de 2015
- Resolución 4891 de 2016

NOTA: Se toma los sustantivos como términos de negocio dado que para el método propuesto estos hacen parte fundamental de la primera regla de mapeo, lo cual se explica en la Fase 4 del método.

Una vez obtenidos los sustantivos se hizo una definición de peso de negocio (*Information content (IC)*) para cada término, labor ejecutada por un analista que, a la fecha en que se hizo la identificación, llevaba aproximadamente 3 años trabajando

con documentos de resolución de la CRC y la ANTV (Autoridad Nacional de Televisión) como insumo para elicitación de requisitos.
Es de anotar que se toma los 130 términos con mayor peso de negocio, cifra arbitraria elegida por consenso entre el analista que dio los pesos de negocio y la persona responsable del presente trabajo.
Esto refina el corpus inicial, quedando así:

- Definición de pesos de negocio.
- Fragmento de Resolución 5111 de 2017. Primeras 8 hojas.
- Proyecto de resolución 5397 de 2018
- Proyecto de resolución 4930 de 2016
- Proyecto de resolución 5161 de 2016

Una vez finalizado lo anterior, se procede a procesar cada documento (sea resolución o fragmento de resolución) ejecutando los pasos 1 y 2 de la herramienta de software explicados en la sección 5.4. Implementación de herramienta.
En este caso, como artefacto de output se tiene la definición de pesos de negocio para el dominio específico y el texto depurado del documento que se esté procesando.

Fase 4. Modelamiento:

En esta fase se procesa el texto y se obtiene de este un modelo conceptual ejecutando los demás pasos, así:

- Paso 3. Generar POS Tags:
  Mediante herramientas de clasificación como *Freeling,* se obtiene una clasificación de partes de oración de los términos del texto extraídos, esto con una salida en XML.

- Paso 4. Procesar POS Tags:
  Luego, se analiza el XML resultante de modo que se transforme desde éstos a elementos de un modelo conceptual, específicamente conceptos y relaciones. Esta identificación se hace de acuerdo a lo identificado en la literatura, donde se especifica que:

  - Las clases de un modelo conceptual se pueden obtener usando nombres y frases nominales [35].
  - Un dominio consiste en conceptos y relaciones, así, en un modelo conceptual los conceptos se identifican como entidades. El nombre de una entidad debe ser un sustantivo [39].
  - Cuando se crea un modelo de dominio, una buena fuente de clases de dominio incluye requisitos de alto nivel, los que normalmente están (pero no siempre) escritos en forma de "El sistema hará esto; el sistema no hará eso". Es útil analizar estos requisitos, extrayendo los sustantivos y las frases nominales. A continuación, puede refinarlos para crear el modelo de dominio inicial [40].

  Con lo anterior se define entonces la primera regla de transformación de elementos extraídos del texto en elementos de un modelo conceptual, la cual determina que un elemento identificado como sustantivo se convierte en un concepto del modelo. En términos del resultado de *Freeling* se identifican con el valor "noun" en la propiedad "pos" de cada termino en el XML resultante.

Esto queda parametrizado en un archivo plano de modo que pueda ser leído por la herramienta y pueda ser modificado a posteriori sin requerir cambios a nivel do código fuente.

La cantidad de conceptos identificados en cada documento puede ser muy alta, esto genera un modelo confuso que se ve saturado por conceptos y dificulta su lectura y comprensión; por lo que es necesario no mostrar en el modelo final todos los conceptos, sino los más importantes, por esto se deben priorizar. Para lo anterior se siguió la técnica de procesamiento de texto *Bag of Words* con *Information Content (IC)* [38]. La técnica permite obtener una lista de palabras con la cantidad de veces que se repite en el texto, lo cual se conoce como *Term Frequency (TF),* este se multiplica por el *Information content (IC)* de cada término (explicado en la Fase 3 preparación), obteniendo una medida que para este trabajo llamaremos TFIC.

En este caso se obtienen los TFIC de los conceptos identificados y se ordenan de mayor a menor.

La cantidad final de conceptos a mostrar puede variar, así, basados en la experimentación, se obtiene el número óptimo de conceptos a mostrar en el modelo dependiendo de qué tan grande sea el documento, en función del número de páginas o del número de palabras.

Las demás reglas de mapeo sirven para identificar las relaciones entre conceptos, para lo cual se definen unos patrones de *POS Tags*. Dado que durante la experimentación estos patrones pueden variar, se parametrizan en el mismo archivo plano donde se definió la regla de identificación de conceptos.
En este caso se hizo una generación manual de modelos conceptuales a partir de textos de modo que se identificó los patrones que utiliza un analista para determinar relaciones, la herramienta emula esto y toma los *POS Tags* que se encuentran entre dos sustantivos (de acuerdo a la regla de mapeo de conceptos) y cumplen con la regla de mapeo; así, si, por ejemplo, en el archivo se configuran los siguientes patrones:

a) pos=adjective,pos=adposition,pos=verb
b) pos=verb,pos=adposition

Se obtienen los siguientes resultados como una relación, estando estos entre dos sustantivos:

a) “múltiples para analizar”, “suficiente para permitir”
b) “dispuesto en”, “prestados a”

NOTA: “pos” corresponde a una propiedad en el resultado de *POS Tagging.* Esto se puede ver más adelante en el detalle de pasos para la generación del modelo.

Las reglas de mapeo definidas para el caso de estudio se presentan en la tabla 6.

Tabla 6. Reglas de mapeo definidas para el caso de estudio.

ELEMENTO EN MODELO	ELEMENTO EN TEXTO	NOTA
Concepto	noun	Basadas en la propiedad POS en el XML
Relación	pos=adjective,pos=adposition,pos=verb pos=verb,pos=adposition pos=adposition,pos=verb pos=adposition form=que verboconjugado	Patrones que se identifican en el XML. Se definen propiedad y valor del el XML Patrón "verboconjugado" genera una acción codificada en la herramienta

- Paso 5. Generar salida gráfica:
  Identificados los conceptos a mostrar y sus relaciones, se pasan en forma automatizada a un formato gráfico, de modo que pueda ser visualizado en una herramienta de software.

En el caso de estudio en particular se obtiene, para cada documento, un modelo gráfico resultante como output. Los modelos finales se presentan en la sección 7.2 Resultados.

Fase 5. Evaluación:

En esta fase se evalúan los resultados del trabajo siguiendo los parámetros definidos en el CAPÍTULO 6. VALIDACIÓN DE LA PROPUESTA DE SOLUCIÓN.
El caso particular presenta como output el Análisis de Resultados, que corresponde a las seccione 7.2 Análisis de resultados y 7.3. Análisis de hallazgos.

5.4 Implementación de herramienta para automatizar el método

- **Consideraciones previas.**

A continuación, se presenta los pasos que ejecuta cada que se procesa un documento para generar un modelo conceptual a partir de su información. Durante cada una de estas ejecuciones se pone en práctica parte de la fase 3 preparación con los pasos uno y dos, dado que el resto de la fase corresponde a la configuración misma de la herramienta y se debió ejecutar previo al uso de la misma. Con los demás pasos se pone en práctica por completo la fase 4 Modelamiento

En adelante llamaremos a la herramienta principal "Modelador", esta es la encargada de orquestar cada paso, ejecutar procesos varios y hacer uso de las demás herramientas de software, así, el usuario sólo tendrá que ejecutar esta.

El computador en el cual se ejecute el programa "Modelador" deberá cumplir con los siguientes requisitos:

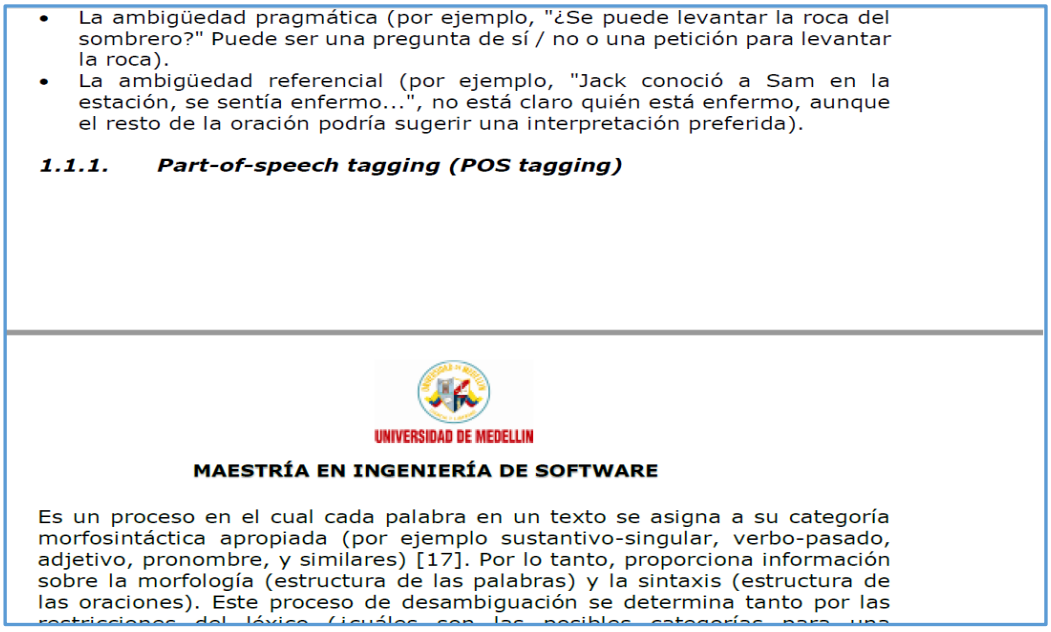
- Freeling 4.0 Instalado
- yEd Graph Editor Instalado.

- Variable de entorno FLINSTALL. Esta corresponde a la ruta de instalación de Freeling. Normalmente por defecto la ruta es "C:\freeling". En adelante usaremos FLINSTALL como una referencia a esta ruta.
- Variable de entorno MODELADOR. Esta corresponde a la ruta de ejecución de Modelador. En adelante usaremos MODELADOR como una referencia a esta ruta.
- Cada documento a procesar debe estar en formato PDF. Si está en otro formato debe ser convertido a PDF.

El lenguaje de programación usado para construir Modelador es Java, más adelante se explica el criterio de selección.

- **Paso 1. Extraer texto de PDF**

El texto se extrae mediante el uso de la librería PDFBox 2.0.11 para Java. En este sentido se exploró varias herramientas gratuitas, (PDFBox para Java y PDFMiner.six para Python), sin embargo, la primera entrega texto más limpio, en la medida que se obtuvo con menos caracteres especiales y saltos de página. En la figura 4 se puede ver un documento que luego es procesado por ambas herramientas, las figuras 5 y 6 muestran las diferencias en el resultado de una y otra:


- La ambigüedad pragmática (por ejemplo, "¿Se puede levantar la roca del sombrero?" Puede ser una pregunta de sí / no o una petición para levantar la roca).
- La ambigüedad referencial (por ejemplo, "Jack conoció a Sam en la estación, se sentía enfermo...", no está claro quién está enfermo, aunque el resto de la oración podría sugerir una interpretación preferida).

1.1.1. Part-of-speech tagging (POS tagging)

UNIVERSIDAD DE MEDELLIN

MAESTRÍA EN INGENIERÍA DE SOFTWARE

Es un proceso en el cual cada palabra en un texto se asigna a su categoría morfosintáctica apropiada (por ejemplo sustantivo-singular, verbo-pasado, adjetivo, pronombre, y similares) [17]. Por lo tanto, proporciona información sobre la morfología (estructura de las palabras) y la sintaxis (estructura de las oraciones). Este proceso de desambiguación se determina tanto por las

Figura 4. PDF original

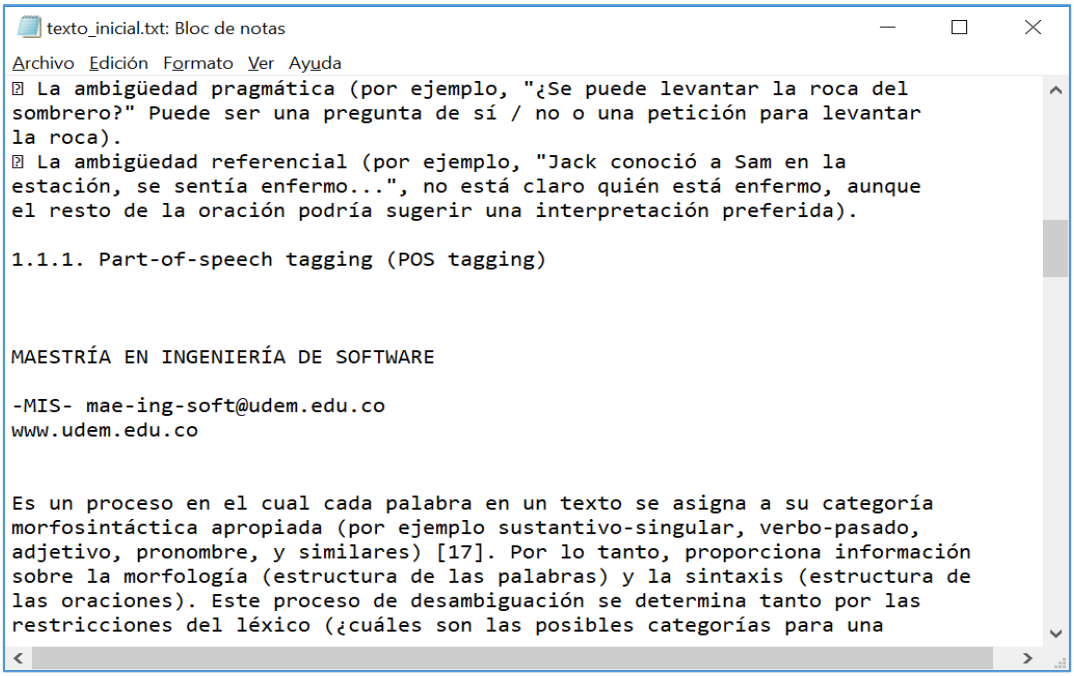


Figura 5. Texto obtenido mediante PDFBox para Java

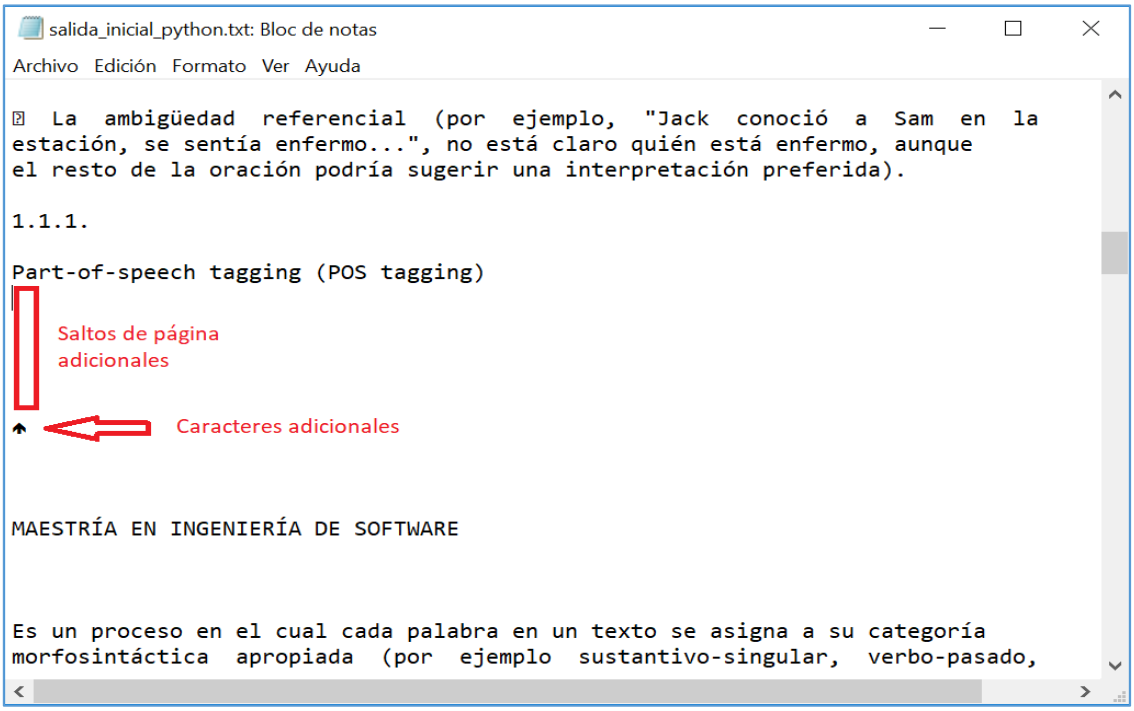


Figura 6. Texto obtenido mediante PDFMiner.six para Python

Lo anterior también motivó el uso de Java como lenguaje de programación para la implementación de Modelador, en tanto el lenguaje no dificulta las demás operaciones y permite usar la librería PDFBox.

En adelante se utiliza el siguiente fragmento de la resolución 5199 de 2017 de la CRC como texto de entrada para los ejemplos:

"*Las disposiciones previstas en esta resolución deberán cumplirse a partir del 1o de enero de 2018. Los operadores que puedan dar cumplimiento a la totalidad de estas disposiciones antes de dicha fecha, podrán implementar el Nuevo Régimen de Protección de los Derechos de los Usuarios de Servicios de Comunicaciones, previa remisión de una comunicación a la CRC y a las autoridades de vigilancia y control, informando dicha situación.*"

- **Paso 2. Pre-procesar texto**

En este paso se toma el texto obtenido del PDF y se le retira texto innecesario, así:

- Encabezados y pies de página
- Líneas de texto de Tablas
- Líneas de texto de Figuras

- **Paso 3. Generar POS Tags**

En este se envía el texto a la herramienta Freeling 4.0, esta herramienta este retorna un XML con cada palabra categorizada en una parte de oración (como sustantivo, verbo, adjetivo, etc.)
Freeling para su ejecución exige la definición de parámetros, en este caso la herramienta envía los definidos en la tabla 7:

Tabla 7. Parámetros usados por Modelador para ejecutar *Freeling*

Nombre	Parámetro	Valor
Ruta de .bat de Freeling		RUTA_MODELADOR
Archivo de configuración (propio de freeling)	-f	es.cfg
Tipo de salida	--outlv	tagged
Formato de salida	--output	xml
Ruta del archivo de entrada	<	RUTA_MODELADOR\salidas\texto_procesar.txt
Ruta del archivo de salida	>	RUTA_MODELADOR\salidas\tags_xml.txt

En la figura 7 presenta un ejemplo de salida XML de *Freeling* para *POS Tagging*, cuyo texto de origen es el presentado en el Paso 1 Extraer Texto de PDF

Figura 7. Ejemplo de XML de salida

El *POS Tagger* fue seleccionado tomando como referencia los ya usados por los trabajos presentados en el estado del arte que usaban *POS Tagging*, usando los siguientes:

- Stanford Log-linear Part-Of-Speech Tagger
- Paquete NLTK para Python
- Freeling
- TreeTagger CIS

Luego, se hizo una comparación tomando como referencia dos textos de modo que se verificó la precisión de cada una de las herramientas *POS Tagger*. Se hizo un proceso de *POS Tagging* manual con el apoyo del diccionario de la Real Academia Española y se comparó con cada herramienta, obteniendo los siguientes resultados, los cuales se comparan en la tabla 9.

Texto 1: "*El Gestor del requerimiento es el responsable de gestionar los controles de cambio y garantizar que surtan el debido proceso. Un control de cambios solo puede ser solicitado por cambios de alcance del requerimiento, cambios en el presupuesto del requerimiento, o cambios en las restricciones de tiempo del requerimiento. Si el esfuerzo de realizar el control de cambios es inferior al 10% del esfuerzo total del requerimiento no se requiere aprobación para ejecutarlo. En caso contrario debe volver al proceso de priorización y cumplir con las aprobaciones allí establecidas El Administrador de cumplimiento debe llevar el registro de todos los controles de cambio solicitados y aprobados/rechazados.*

Posterior a la aprobación el Gestor del requerimiento debe comunicar el control de cambios al solicitante"

Texto 2: "*La siguiente mejora tiene como objetivo fidelizar a nuestros clientes que tienen o adquieran una línea telefónica con la compañía, dándoles la posibilidad de NO perder su número número/identificador telefónico.*
La Portabilidad Numérica consiste en conservarle al cliente su número telefónico cuando realice Transacciones de traslado (lineal, retiro / nuevo), cambio de tecnología y migraciones de clientes entre sistemas de información.
Se requiere que, desde los sistemas de información actuales de la empresa, se pueda conservar el número telefónico del cliente, independiente del tipo de tecnología en la que quede, zona geográfica o localidad donde se traslade siempre y cuando aplique la misma numeración telefónica según asignación de la CRC.
Los sistemas de información y las plataformas deben controlar que el cliente no pierda su número telefónico en las transacciones de traslado (lineal, retiro / nuevo), cambio de tecnología y migraciones de clientes entre sistemas de información en cualquiera de los siguientes escenarios"

Tabla 8. Resultado de comparación de *POS Taggers*

		Stanford	***NLTK***	***Freeling***	***TreeTagger***
Cantidad de palabras correctas	Texto 1	120	46	120	120
	Texto 2	149	61	161	151
Porcentaje con respecto al total de palabras	Texto 1	85%	32%	85%	85%
	Texto 2	85%	35%	91%	86%

Dado que en el Texto 1 se presenta un rendimiento similar entre *Stanford*, *Freeling* y *TreeTagger*; y en el Texto 2 se presenta un rendimiento superior de *Freeling*, se eligió esta última como herramienta de *POS Tagging* para el presente trabajo.

- **Paso 4. Procesar POS Tags**

En este paso se toma el XML y se recorre para identificar en primera instancia los conceptos y luego las relaciones entre los mismos (de acuerdo a lo definido en la sección anterior Método Usado). Para esta identificación se utiliza como insumo el archivo de entrada "*MODELADOR\ archivos proceso\ReglasTransformacion.txt*". En este se define primero la regla de identificación de conceptos (componentes para el modelador) y luego las reglas de identificación de relaciones, definidas como los patrones de identificación de Relaciones, así, al procesar el XML se identifican las secuencias de palabras que cumplen con estos patrones y se toman como elementos válidos. En la figura 8 se muestra la configuración de relaciones usada en el caso de estudio.

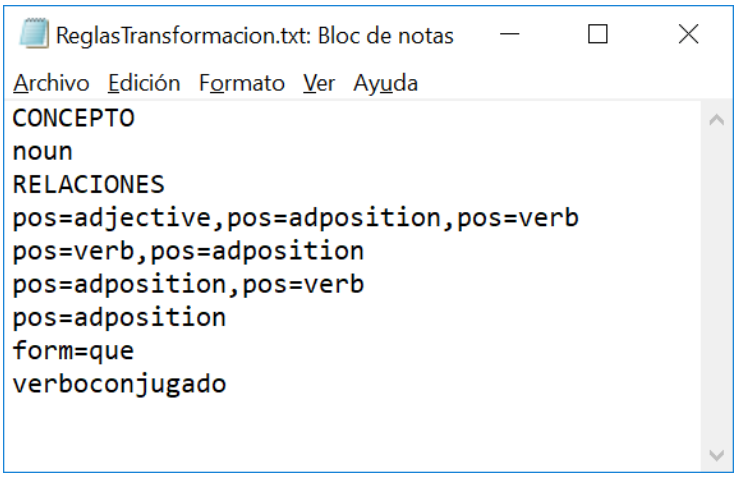


Figura 8. Ejemplo archivo de reglas de transformación

Además de lo anterior, se obtiene el *Term Frequency (TF)* de cada concepto contando el número de ocurrencias en el XML. Se toma su *Information content (IC)* del archivo insumo "*MODELADOR\ archivos proceso\PesoNegocio.txt*", el cual tiene la estructura mostrada en la figura 9, se obtiene el TFIC para cada concepto, así TFIC = TF * IC.

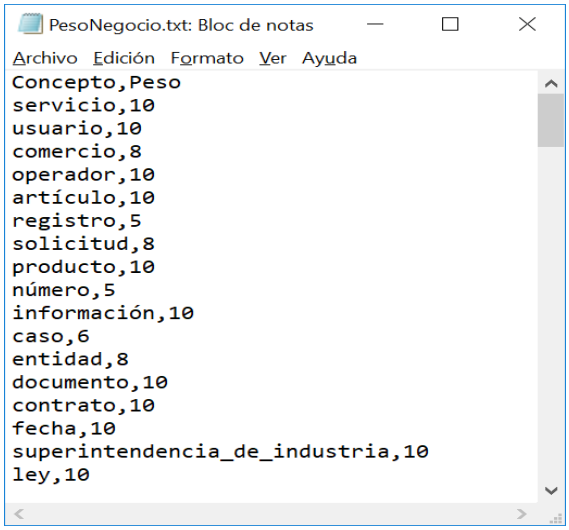


Figura 9. Ejemplo archivo de pesos de negocio (*Information Content*)

Obtenido este, se ordenan los conceptos por TFIC de mayor a menor, y se identifican como "mostrables en el modelo gráfico" los primeros *N* conceptos. El valor de *N* se obtiene así:

En primera instancia se toma el número de páginas y se multiplica por un número X, Este número X corresponde a un valor arbitrario que se especifica de acuerdo a la experimentación con el modelo, identificando en forma visual que tan cargados de conceptos quedan los modelos resultantes con cada cambio en el valor X, buscando un equilibrio entre legibilidad y completitud de la información contenida por los modelos. Para el caso particular se determinó el número 15.
La cantidad de conceptos resultantes es sugerida al usuario, quien finalmente decide si deja este valor sugerido o ingresa un valor diferente. En la figura 10 se presenta un ejemplo de un documento con una extensión de seis páginas.

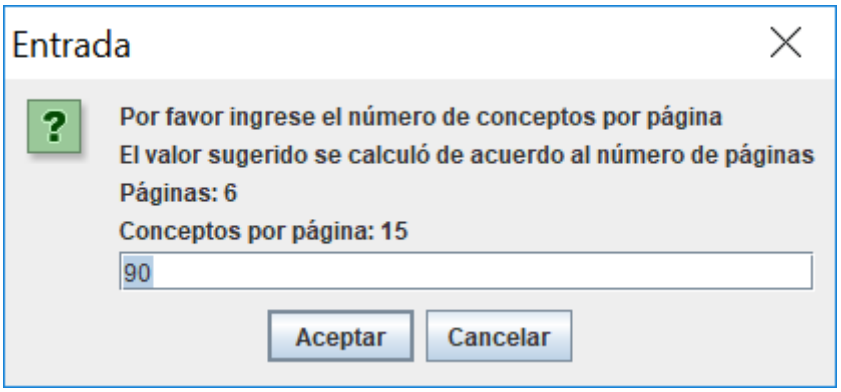


Figura 10. Mensaje para captura de cantidad de conceptos a mostrar en el modelo

- **Paso 5. Generar salida Gráfica**

Identificados los conceptos y relaciones, se convierten éstos a salidas gráficas. Se hizo una exploración de herramientas gratuitas que permiten generar modelos conceptuales y que además permitan recibir un formato escrito en texto, en lugar de o además de generación gráfica directamente por un usuario.

Con lo anterior se identificó la herramienta yEd Graph Editor, que permite mostrar gráficamente modelos escritos en formato graphml, así, el modelador genera el modelo en la siguiente ruta: "MODELADOR\salidas\modelo_graphml.graphml". En la figura 11 se puede visualizar el ejemplo del texto generado.

Esta herramienta permite hacer zoom al modelo generado, lo que resulta beneficioso para modelos resultantes de textos extensos, además habilita al usuario para editar gráficamente el modelo, modificando, eliminando o adicionando tanto conceptos (clases para la herramienta) como relaciones.

Figura 11. Ejemplo de graphml de salida

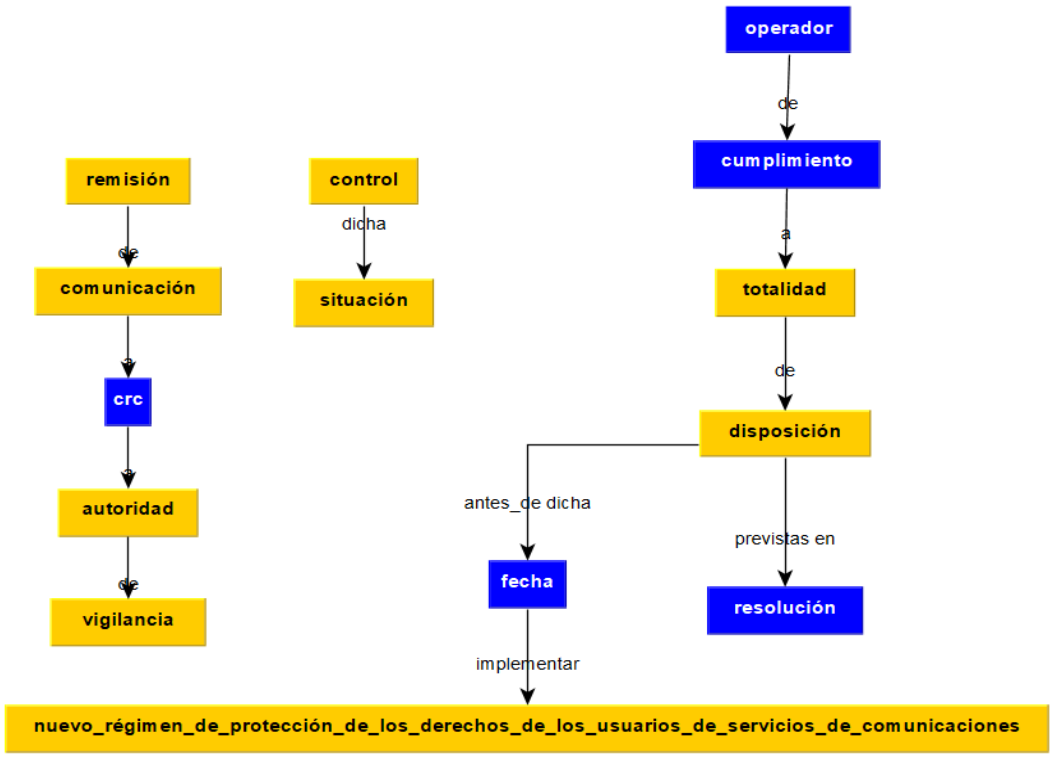


Figura 12. Ejemplo de visualización de graphml en yEdGraphics

La figura 13 muestra el ejemplo de visualización, Los conceptos en fondo azul y letra blanca corresponden a los cinco conceptos con mejor TFIC de acuerdo al ordenamiento explicado anteriormente. Esto permite al usuario identificar los conceptos con mejor peso para el modelo y a partir de ahí hacer el análisis del gráfico. Esto es útil en modelos resultantes con gran cantidad de conceptos.

Además de la herramienta antes mostrada, se puede usar la herramienta *Mermaid Live Editor*, este es un proyecto que se encuentra publicado en la *GitHub* y permite, a partir de un código específico, generar modelos. En este sentido, el Modelador también genera salida para esta herramienta en la ruta "MODELADOR\salidas\modelo_mermaid.graphml" En la figura 13 se puede apreciar el código generado, el cual una vez es usado se transforma en el modelo gráfico que se puede apreciar en la figura 14.

Esta, a diferencia de yEd Graph Editor no permite hacer un zoom consistente que permita leer claramente el modelo resultante de textos extensos, además, para su edición es absolutamente necesario modificar el código específico. Otra desventaja es que no permitiría adicionar Atributos a los conceptos (clase para un modelo conceptual), así, en un hipotético trabajo futuro en el que se adicionara Atributos a los Conceptos no se podría usar esta herramienta.

modelo_mermaid.txt: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
graph TD
disposición[disposición] -->|previstas en| resolución(resolución)
operador[operador] -->|de| cumplimiento(cumplimiento)
cumplimiento[cumplimiento] -->|a| totalidad(totalidad)
totalidad[totalidad] -->|de| disposición(disposición)
disposición[disposición] -->|antes_de dicha| fecha(fecha)
fecha[fecha] -->|implementar| nuevo_régimen_de_protección_de_los_derechos_de_los_usuarios
remisión[remisión] -->|de| comunicación(comunicación)
comunicación[comunicación] -->|a| crc(crc)
crc[crc] -->|a| autoridad(autoridad)
autoridad[autoridad] -->|de| vigilancia(vigilancia)
control[control] -->|dicha| situación(situación)
```

Figura 13. Ejemplo de texto para Mermaid de salida

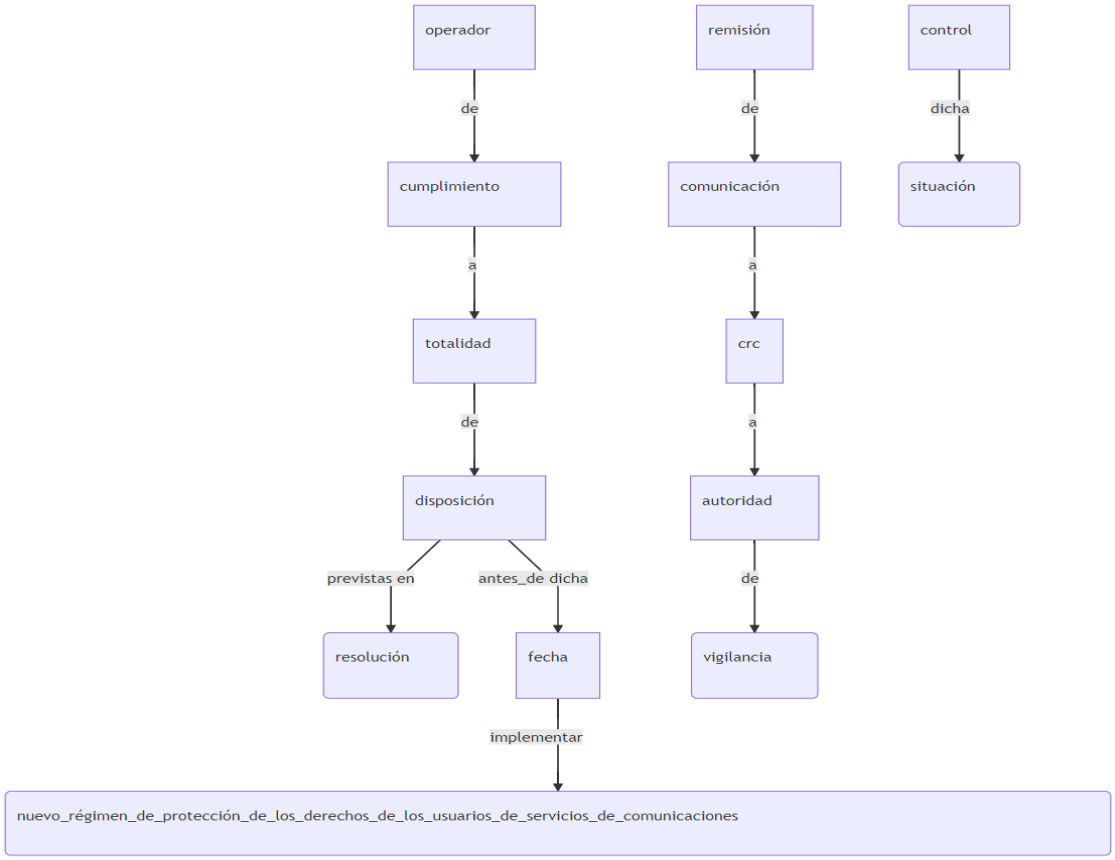


Figura 14. Ejemplo de visualización en Mermaid

En la tabla 9 se presenta un resumen de herramientas usadas, herramientas comparadas y su criterio de selección:

Tabla 9. Herramientas de software utilizadas

Herramienta	Comparada con	Diferenciador
PDFBox 2.0.11 para Java	PDFMiner.six para Python	Textos resultantes más limpios
Freeling 4.0	Stanford Log-linear Part-Of-Speech Tagger Paquete NLTK para Python TreeTagger CIS	Mejor efectividad en comparación con Taggeo manual
yEd Graph Editor	Mermaid Live Editor	Permite edición manual Permite adicionar atributos, pensando en el trabajo futuro

En resumen, para el método es necesario realizar una identificación manual de conceptos más importantes para el dominio específico a evaluar, como se explicó en la identificación de los pesos de negocio (*Information Content (IC)*), lo que permite al modelo refinar la identificación de conceptos mediante la identificación del TFIC (*Term Frequency(TF) * Information Content(IC)*), además se definen unas reglas específicas para identificación de conceptos y relaciones. Luego, se pueden procesar documentos uno a uno, demo que la herramienta lo procesa, utiliza *POS Tagging* para separar el texto en partes y procesa esas partes para identificar elementos de un modelo conceptual, para luego graficarlo. La visualización del gráfico requiere una herramienta que no está integrada al modelo.

PARTE IV:

EVALUACIÓN

CAPÍTULO 6. VALIDACIÓN DE LA PROPUESTA DE SOLUCIÓN

6.2 Diseño experimental

En esta sección se presentan los elementos principales del experimento a ejecutar para la validación de la propuesta.

- **Tipo de investigación**

  Investigación experimental.

- **Variable independiente**

  Aplicación SI/NO de la técnica en la lectura de los textos

- **Variables dependientes**

  - Comparativo del tiempo que toma procesar, comparando Tiempo aplicando lectura directa con Tiempo aplicando la técnica de NLP. Se mide tomando el tiempo en minutos desde que inicia el proceso hasta que se finaliza para hacer la comparación.

  - Comparativo del nivel de interpretación de la información contenida en los documentos, comparando Nivel de interpretación aplicando lectura directa con nivel de interpretación aplicando la técnica de NLP. Se mide determinando el porcentaje de respuestas correctas en cada documento para la comparación.

    Medir estas dos variables permiten determinar a través de la medición del tiempo si la propuesta permite agilizar el proceso manual, además la medición del número de respuestas correctas permite determinar si se mantiene el nivel de comprensión ejecutando el método propuesto.

- **Tratamiento de grupos**

  - Grupo A, denominado grupo base o grupo de control:

    Ingenieros que toman una serie de documentos escritos en lenguaje natural, leen su contenido y al final contestan un cuestionario para cada documento.
    Se permite tener notas de apoyo y leerlas durante la respuesta al cuestionario.
    Se toma el tiempo que toma la lectura, hasta contestar el cuestionario.

  - Grupo B, denominado grupo con enfoque propuesto:

    Ingenieros que toman una serie de documentos escritos en lenguaje natural, les aplican una o varias herramientas para

generar un modelo de representación, al final contestan un cuestionario para cada documento.

Se permite visualizar el modelo de representación durante la respuesta al cuestionario.
Se toma el tiempo que toma la lectura, hasta contestar el cuestionario.

- **Calificación del modelo generado**

Si bien la ejecución del cuestionario permite asegurar el nivel de comprensión, resulta útil medir otros aspectos que determinen la calidad de los modelos resultantes, es por esto que se tomó el estándar ISO/IEC 9126-3 en el modelo de datos conceptual entidad-relación [41], donde los criterios están definidos en la tabla 10.

Luego de contestar el cuestionario, todos los analistas obtienen el modelo de cada documento y lo califican de 1 a 5 siendo 1 la peor calificación, 5 la mejor calificación, aplicando esta medida a cada uno de los siguientes criterios: Legibilidad, Completitud, Corrección, Minimalidad, Expresividad, Autoexplicación.

Tabla 10. Criterios de calidad en modelos conceptuales

CRITERIO	DESCRIPCIÓN
Legibilidad	Está enfocado a las consideraciones visuales para la lectura y presentación del modelo conceptual (cruces entre relaciones, superposiciones, tipografía, entre otros)
Completitud	El modelo debe incluir totalmente lo que se quiere diseñar, que es aquello que se encuentra plasmado en los requerimientos del sistema por desarrollar. En el caso particular, lo plasmado en los documentos de resolución.
Corrección	Se puede evaluar desde dos perspectivas: - Sintáctica: cuando las distintas partes del modelo están construidas con respecto al lenguaje utilizado. - Semántica: cada elemento se representa haciendo uso de las estructuras adecuadas.
Minimalidad	Un modelo conceptual se considera mínimo si no tiene información redundante o duplicada y , por consiguiente, si se elimina un elemento del esquema se perderá información
Expresividad	El modelo representa la realidad, de manera que con sus elementos esta puede ser comprendida fácilmente. La expresividad intenta medir la capacidad de comunicación del modelo a nivel semántico.
Autoexplicación	En el modelo pueden ser representados todos los requisitos, por consiguiente, la lógica del negocio con respecto a los datos puede ser accedida y entendida por el modelo conceptual.
Extensibilidad	Se refiere a la capacidad de un esquema para poder tolerar cambios y adaptarse a nuevas necesidades de los usuarios.

- **Sujetos**

- Ingenieros de requisitos de una organización.
- Documentos de negocio, específicamente documentos de resoluciones de la CRC.

- **Criterios de satisfacción**

La hipótesis se comprueba si se cumple las siguientes premisas con respecto a las variables dependientes:

1. El tiempo que toma procesar los documentos al grupo B es inferior al tiempo que toma procesar los documentos aplicando lectura directa.
2. El porcentaje de respuestas correctas luego de aplicar la técnica de NLP es igual o superior al porcentaje de respuestas correctas aplicando lectura directa.

Se deben cumplir ambas premisas, si se cumple solo una la hipótesis no puede ser comprobada.

6.3 Especificaciones de la validación

En la validación se considera los siguientes 4 archivos de resoluciones:

1- Resolución 5111 de 2017. Se toma las primeras 8 hojas. Publicado en el siguiente vínculo: https://colombiatic.mintic.gov.co/679/articles-62266_doc_norma.pdf

2- Proyecto de resolución 5397 de 2018. Publicado en el siguiente vínculo: https://www.crcom.gov.co/uploads/images/files/Proy_Res%20Mod%205050%20publicar.pdf

3- Proyecto de resolución 4930 de 2016. Publicado en el siguiente vínculo: https://www.crcom.gov.co/recursos_user/Documentos_CRC_2015/Actividades_regulatorias/CPM_etp2/Proyecto_Resolucion_CPMFase2_vf.pdf

4- Proyecto de resolución 5161 de 2016. Publicado en el siguiente vínculo: https://www.crcom.gov.co/recursos_user/2016/Actividades_regulatorias/ain_ba/30dic/ProyRes_BA_12-2016.pdf

Estos son documentos de regulación de comunicaciones expedidos por la Comisión de Regulación de Comunicaciones de Colombia (CRC), dado que son de dominio público expuestos directamente en su portal web https://www.crcom.gov.co/ o en el portal web del Ministerio de Comunicaciones https://www.mintic.gov.co, están escritos en lenguaje natural en idioma español, y son usados por ingenieros de requisitos para su contextualización, de modo que los productos de software para las empresas que prestan servicios de telecomunicaciones cumplan con la regulación existente.

El contexto de aplicación es apoyar los procesos de análisis de requerimientos para cumplir con la reglamentación legal en una compañía de telecomunicaciones.

Los siguientes son los analistas que participan del experimento. A cada uno se le asigna una sigla con dos letras correspondientes al primer nombre y primer apellido, para su identificación en las tablas de resultados.

Analistas hacen lectura directa de los documentos.

1- Juan Pablo Jaimes Pabón (JJ)
2- Diana Carolina Restrepo (DR)
3- Juan Sebastián Morales Pérez (JM)
4- Juan Esteban Sierra (JS)

Analistas que aplican el método propuesto con las herramientas proporcionadas.

5- Leidy Marcela Zapata (LZ)
6- Eliú Sánchez Madrid (ES)
7- Oscar Fernado Olivo (OO)
8- John Eduar Rincón (JR)

En la tabla 11 se presenta un cuestionario de cada documento antes mencionado. Las preguntas se catalogan en 3 niveles, siendo el nivel 1 el de menor dificultad el de nivel 3 de mayor dificultad, lo que permitirá obtener unas conclusiones más precisas, permitiendo evaluar hasta que nivel permiten los modelos representar el dominio.

Tabla 11. Preguntas de los documentos de resolución

Num	Pregunta	Opciones	Respuesta correcta	Nivel
Documento 1-Fragmento de Resolución 5111 de 2017				
1	¿Cuál es el ámbito de aplicación del régimen dado en la resolución?	- Relación entre usuarios y reguladores - Relaciones entre usuarios y operadores - Relación entre operadores y reguladores	Relaciones entre usuarios y operadores	2
2	¿A favor de quien debe prevalecer los derechos ante una duda en el contrato?	- Usuario - Operador	Usuario	1
3	¿Es el usuario libre de elegir el operador?	- Si - No	Si	1
4	¿Qué obligaciones tiene el usuario? (entre otras)	- Informar con un mes o más de anticipación el deseo de no continuar con el servicio - Cumplir las condiciones del contrato y hacer uso adecuado de los medios de comunicación	Cumplir las condiciones del contrato y hacer uso adecuado de los medios de comunicación	3
5	¿Tiene el usuario derecho a recibir información para decidir las condiciones del servicio que debe prestar el operador?	- Si - No	Si	1

6	¿Qué prohibiciones tiene los contratos de prestación de servicio? (entre otras)	- Renuncia a algún derecho de los usuarios y permanencia mínima en servicios móviles - Permitir a los usuario elegir operador libremente e indemnizar a los usuarios cuando sea pertinente.	Renuncia a algún derecho de los usuarios y permanencia mínima en servicios móviles	3
7	¿Tiene derecho el usuario a terminar el contrato en cualquier momento?	- Si - No	Si	1
8	¿Cuándo se puede hacer modificaciones al contrato?	- Cuando el operador lo considere necesario - Cuando el usuario lo considere necesario - Cuando se pacte con el usuario	Cuando se pacte con el usuario	3
9	El contrato permite eliminar o limitar la responsabilidad de los operadores?	- Si - No	No	2
10	¿Dónde podrá el usuario ver el área de cobertura de su servicio móvil?	- Factura de servicios móviles - Sitio web del operador - Sitios web gubernamentales	Sitio web del operador	2
11	¿Es el usuario o el operador quien elige el plan en el contrato?	- Operador - Usuario	Usuario	1
12	¿En dónde se garantiza las condiciones del contrato?	- Prestación del servicio - Canales de atención al usuario - Facturación mensual	Prestación del servicio	1
13	Son relaciones entre usuario y operador (entre otras)	- El usuario establece plenamente las condiciones del contrato. - El operador informa al usuario y el usuario acepta las condiciones del contrato. - El operador establece las condiciones del contrato y las comunica al usuario.	El operador informa al usuario y el usuario acepta las condiciones del contrato	3
14	¿Dónde se pactan las cláusulas de prestación del servicio?	- En el contrato - En la factura de servicios - En los comunicados emitidos del operador	En el contrato	2
15	¿En qué caso el usuario debe ser compensado?	- Incrementos en los valores aún con previo aviso. - Fallas en el servicio	Fallas en el servicio	2
Documento 2-Proyecto de resolución 5597 de 2018				
1	¿La presente resolución hace referencia al servicio de?	- Televisión - Internet o Datos - Telefonía fija	Internet o Datos	1
2	¿En qué medida se podría decir que el usuario es relevante para la resolución?	- Medianamente relevante - Poco relevante - Muy relevante	Muy relevante	1

3	¿Debe el prestador de servicios suministrar sugerencias de instalación?	- Si - No	Si	1
4	¿Qué dato (además de otros) debe informar el prestador del servicio acerca de los equipos?	- Mac - Marca - Serial	Marca	2
5	¿Los equipos de comunicación son suministrados por?	- Terceros contratados por cualquiera de las partes (operador o usuario) - Operador de servicio - Usuario	Operador de servicio	1
6	¿Qué datos (además de otros) acerca de la velocidad se debe informar ?	- Subida y descarga - Latencia del servicio	Subida y descarga	2
7	¿Puede el usuario conocer el cargo asociado al equipo que está usando?	- Si - No	Si	1
8	¿Qué aspectos debe informar el proveedor acerca de los equipos de comunicación?	- Mac y Serial - Color - Protocolos y bandas de frecuencia	Protocolos y bandas de frecuencia	3
9	¿Qué información deberá tener el prestador para consulta del usuario?	- Costos asociados a la prestación del servicio. - Valor mensual y velocidad contratada - Canales premium	Valor mensual y velocidad contratada	2
10	¿A partir de cuándo rige la presente resolución?	- De su publicación - Enero de 2020 - Diciembre de 2018	De su publicación	1
Documento 3-Proyecto de resolución 4930 de 2016				
1	Tres de los aspectos más importantes para la presente resolución son	- Ente regulador, velocidad de internet y operador - Operador, factura de servicios y precios - Usuario, contrato y servicio	Usuario, contrato y servicio	2
2	¿En dónde son establecidas las cláusulas del permanencia del servicio?	- Factura de servicios - Contrato - Página web del operador	Contrato	2
3	¿Cuál es el tiempo máximo de cláusulas de permanencia mínima?	- Un Año - Un Mes - Dos semanas	Un Año	2
4	¿Pa conexión incluya un costo para conectar al usuario a la red?	- Si - No	Si	1
5	¿Es el usuario o el operador quien elige el plan en el contrato?	- Usuario - Operador	Usuario	1
6	¿En dónde se establecen las condiciones pactadas por acuerdo entre operador y usuario?	- Factura de servicios - Contrato - Página web del operador	Contrato	2

7	¿A qué servicio se refiere la presente resolución?	- Televisión por suscripción - Internet Banda ancha - Telefonía fija	Televisión por suscripción	2
8	¿El cobro de conexión puede ser financiado?	- Si - No	Si	1
9	¿Qué planes son excepciones a este régimen ?	- Planes corporativos o empresariales - Planes para hogares y personas	Planes corporativos o empresariales	3
10	¿Qué costos se asocian al cargo por conexión?	- Costos de nómina del operador - Costos de impresión de factura - Costos asociados a la red de acceso	Costos asociados a la red de acceso	1
11	¿Qué son cláusulas de permanencia mínima?	- Estipulación contractual que obliga al operador a no terminar anticipadamente el contrato. - Estipulación contractual que obliga al usuario a no terminar anticipadamente su contrato. - Estipulación contractual que permite al usuario terminar su contrato cuando considere.	Estipulación contractual que obliga al usuario a no terminar anticipadamente su contrato	3
12	¿Cuáles son las modalidades de contratación del servicio?	- Con cláusula y sin cláusula de permanencia - Con canales premium o sin canales premium - Con servicio de internet o sin servicio de internet	Con cláusula y sin cláusula de permanencia	2
13	¿Es posible prorrogar o renovar las cláusulas de permanencia mínima ya pactadas en un contrato?	- Si - No	No	2
14	¿Qué datos debe dar el operador en la factura de servicios cuando tiene pactado permanencia (entre otros)?	- Velocidad contratada, minutos disponibles - Valor de conexión subsidiado, Fechas de inicio y fin de cláusulas de permanencia, valor por terminación anticipada - Fecha de facturación, cobros adicionales por cambios en el servicio	Valor de conexión subsidiado, Fechas de inicio y fin de cláusulas de permanencia, valor por terminación anticipada	3
Documento 4-Proyecto de resolución 5161 de 2016				
1	¿La presente resolución hace referencia al servicio de?	- Televisión por cable - Internet - Telefonía fija	Internet	1
2	¿La definición de banda ancha está asociada a ?	- Cantidad de megas disponibles para descarga. - La red de acceso al servicio - Velocidad o Capacidad de transmisión	Velocidad o Capacidad de transmisión	2

3	¿Se puede decir que la masificación del servicio de internet está asociado a?	- Contratos suscritos con el gobierno nacional - Prestación del servicio de los operadores - Penetración del servicio	Contratos suscritos con el gobierno nacional	3
4	¿El servicio de ultra banda ancha está limitado por?	- La capacidad de la red de acceso - Velocidad de bajada - La marca del equipo	Velocidad de bajada	2
5	¿Qué servicios provee el servicio de banda ancha?	- Televisión por cable, llamadas IP - Voz, Datos y Video	Voz, Datos y Video	2
6	¿Cuál es la velocidad actual de bajada límite para denominar Ultra Banda Ancha?	- 2.5 mbps - 25 mbps - 5 mbps	25 mbps	3
7	¿Cuál es la fecha límite para cambiar la definición de banda ancha y ultra banda ancha?	- 31 de diciembre de 2019 - 10 de enero de 2021 - 01 de junio de 2020	31 de diciembre de 2019	1
8	¿Qué condiciones de interface WIFI debe informar el proveedor? (entre otras)	- Alcance en metros, posibilidad de caídas - Equipos homologados par el servicio - Máxima velocidad, estándares y cantidad de bandas soportadas	Máxima velocidad, estándares y cantidad de bandas soportadas	3

CAPÍTULO 7. HALLAZGOS Y DIVULGACIÓN

7.2 Resultados

En la prueba, cada analista diligenció un documento donde registró el tiempo que le tomó desde que inició el proceso hasta que contestó las preguntas formuladas, luego se consolidaron los resultados teniendo en cuenta el tiempo y el porcentaje de respuestas correctas por nivel de dificultad, lo que se puede muestra en la tabla 10.

Tabla 12. Resultados de las evaluaciones de los documentos

Documento	Analista	Grupo	Tiempo Minutos	% Correctas Nivel 1	% Correctas Nivel 2	% Correctas Nivel 3	% Correctas Total
1	JJ	A	57	83,33%	100,00%	75,00%	86,67%
1	DR	A	48	83,33%	100,00%	100,00%	93,33%
1	JM	A	73	83,33%	100,00%	100,00%	93,33%
1	JS	A	64	100,00%	100,00%	50,00%	86,67%
2	JJ	A	35	83,33%	100,00%	100,00%	90,00%
2	DR	A	37	100,00%	100,00%	100,00%	100,00%
2	JM	A	51	100,00%	100,00%	100,00%	100,00%
2	JS	A	25	83,33%	100,00%	100,00%	90,00%
3	JJ	A	54	100,00%	100,00%	100,00%	100,00%
3	DR	A	47	100,00%	100,00%	100,00%	100,00%
3	JM	A	62	75,00%	85,71%	100,00%	85,71%
3	JS	A	49	75,00%	85,71%	100,00%	85,71%
4	JJ	A	25	100,00%	33,33%	66,67%	62,50%
4	DR	A	28	100,00%	66,67%	66,67%	75,00%
4	JM	A	47	100,00%	66,67%	100,00%	87,50%
4	JS	A	21	100,00%	66,67%	66,67%	75,00%
1	LZ	B	23	100,00%	100,00%	100,00%	100,00%
1	ES	B	45	83,33%	100,00%	100,00%	93,33%
1	OO	B	40	100,00%	80,00%	75,00%	86,67%
1	JR	B	52	83,33%	80,00%	75,00%	80,00%
2	LZ	B	18	83,33%	100,00%	0,00%	80,00%
2	ES	B	32	100,00%	66,67%	100,00%	90,00%
2	OO	B	25	100,00%	100,00%	100,00%	100,00%
2	JR	B	18	100,00%	100,00%	100,00%	100,00%
3	LZ	B	19	100,00%	71,43%	100,00%	85,71%
3	ES	B	28	100,00%	85,71%	66,67%	85,71%
3	OO	B	30	100,00%	85,71%	66,67%	85,71%
3	JR	B	21	100,00%	85,71%	100,00%	92,86%
4	LZ	B	13	100,00%	66,67%	100,00%	87,50%
4	ES	B	16	50,00%	100,00%	33,33%	62,50%
4	OO	B	15	100,00%	100,00%	100,00%	100,00%
4	JR	B	17	50,00%	100,00%	33,33%	62,50%

Los siguientes son los modelos resultantes de cada documento utilizando la herramienta, que fueron usados directamente por los analistas del grupo B para contestar el cuestionario y por ambos grupos para emitir la calificación, presentados en las figuras 15, 16, 17 y 18.
NOTA: Algunos pueden quedar poco legibles y al pegarlo al documento pierden resolución, lo que ocasiona que no se pueda hacer un acercamiento apropiado dentro del documento.

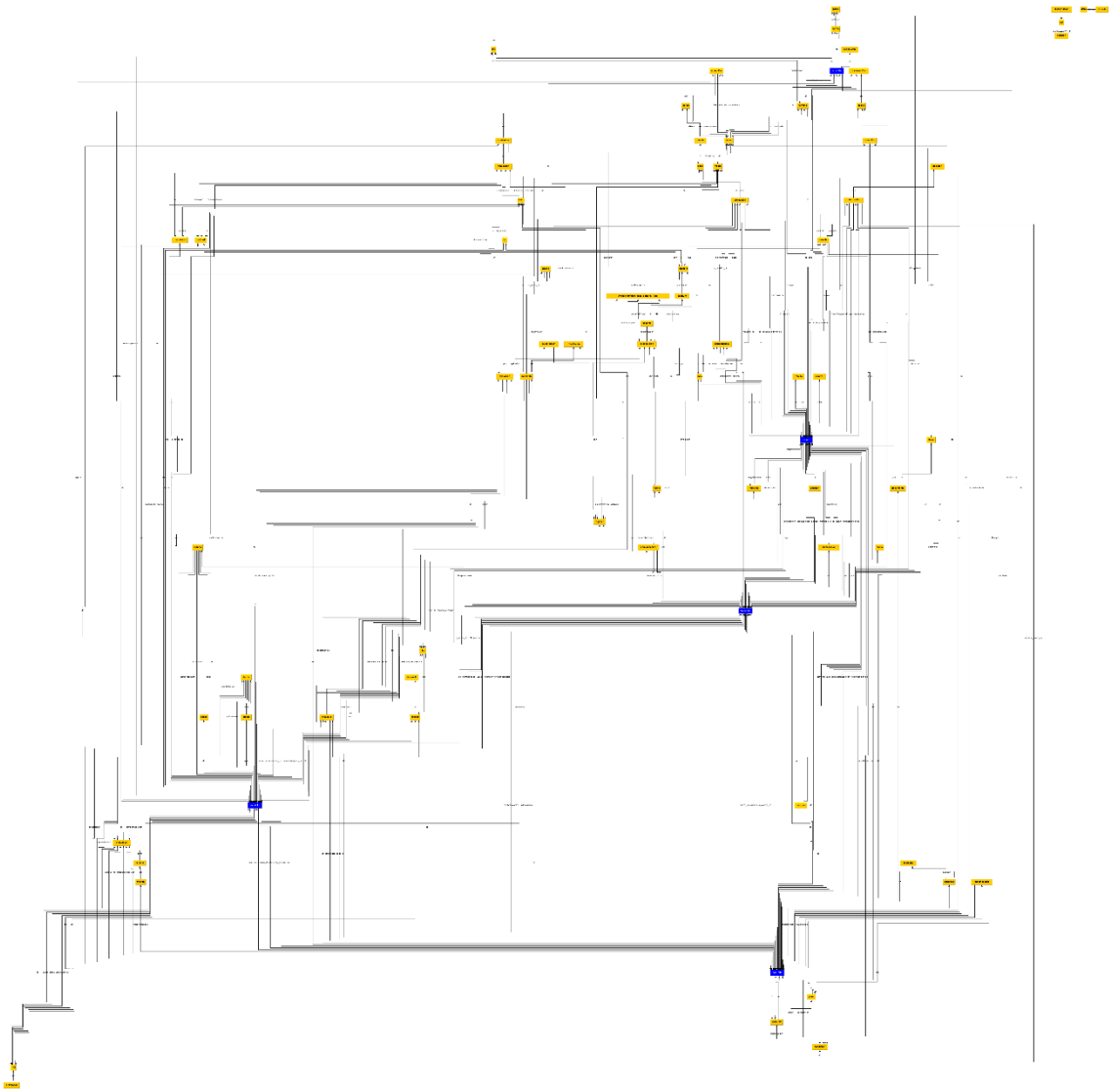


Figura 15. Modelo resultante del documento 1

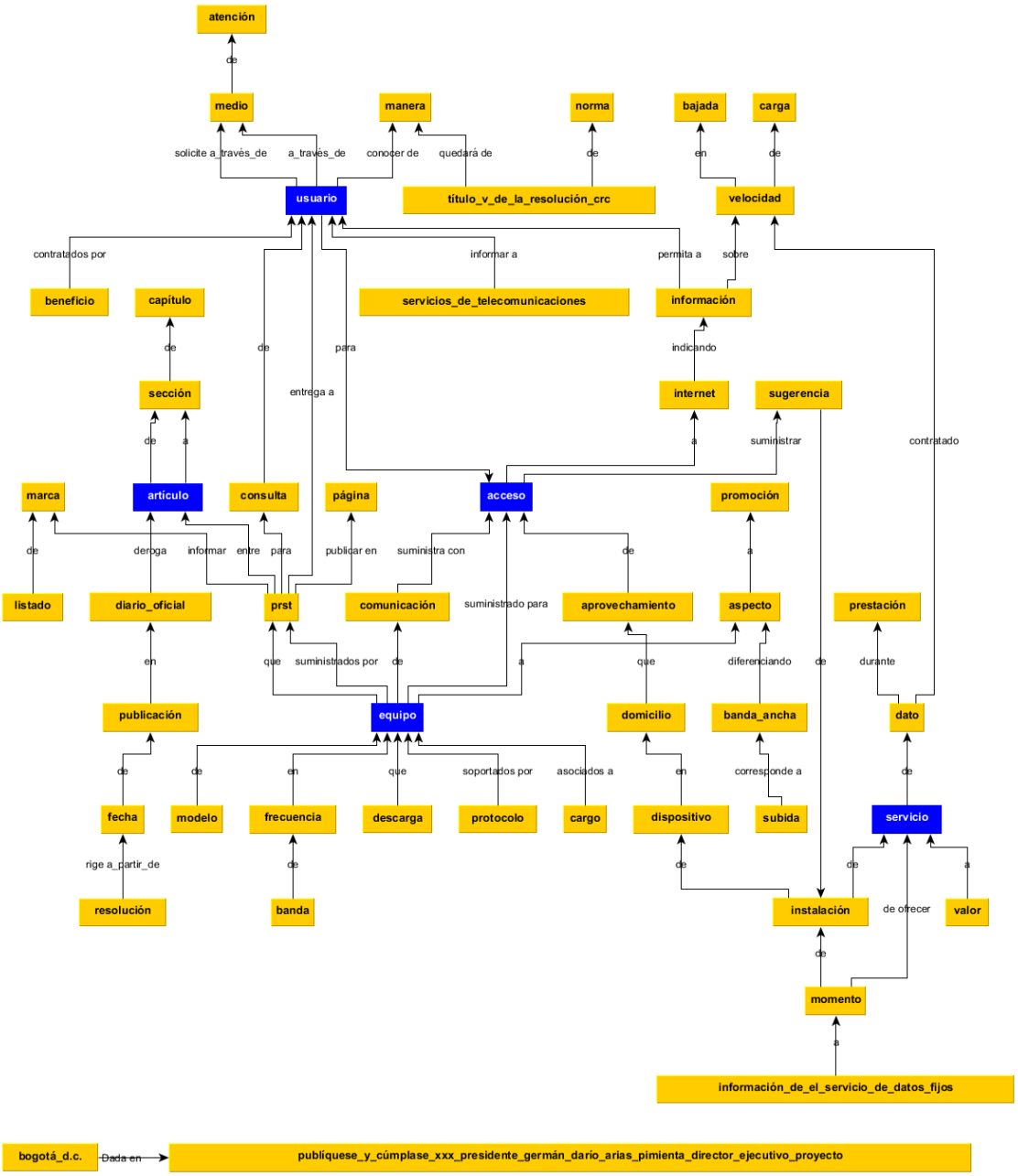


Figura 16. Modelo resultante del documento 2

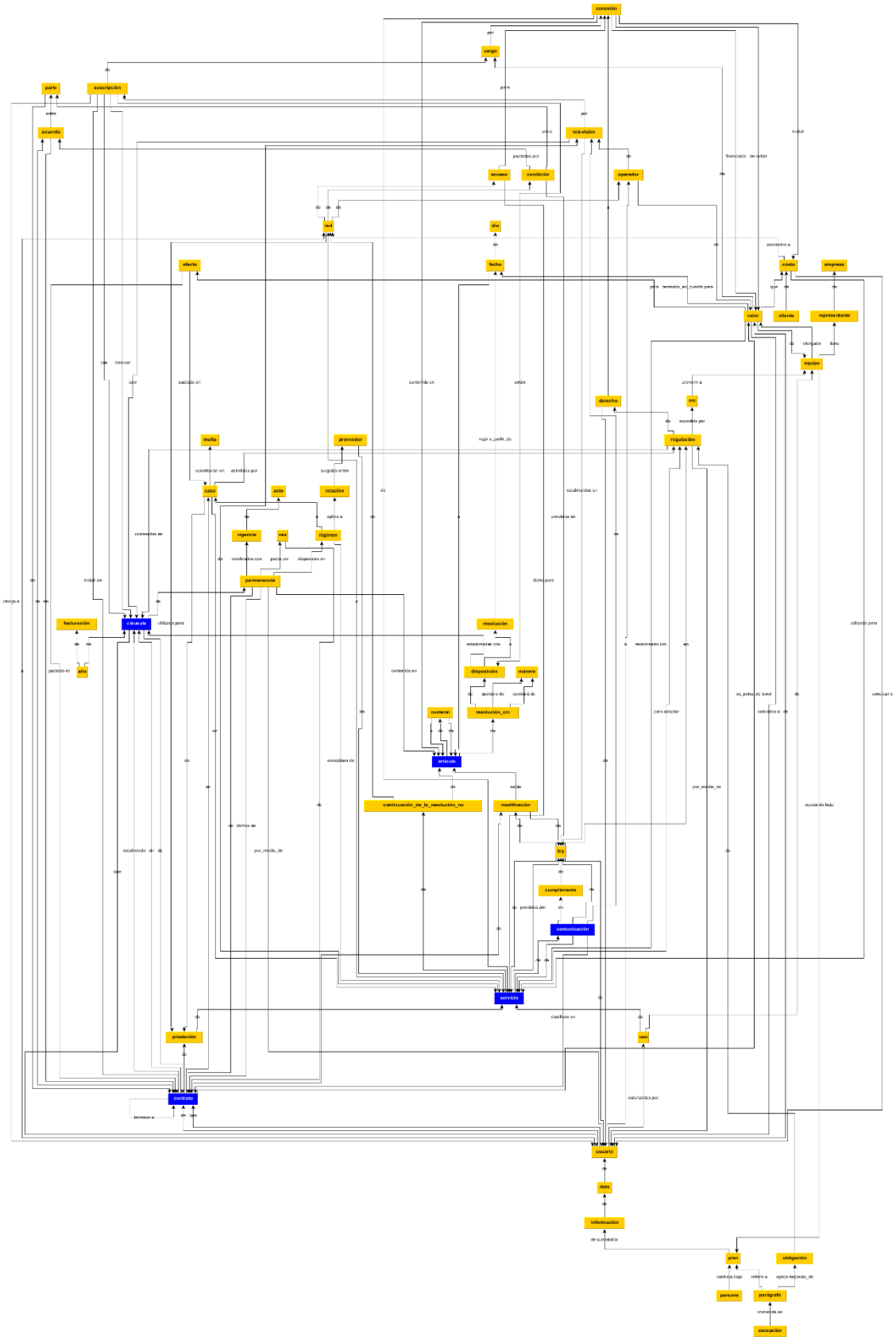


Figura 17. Modelo resultante del documento 3

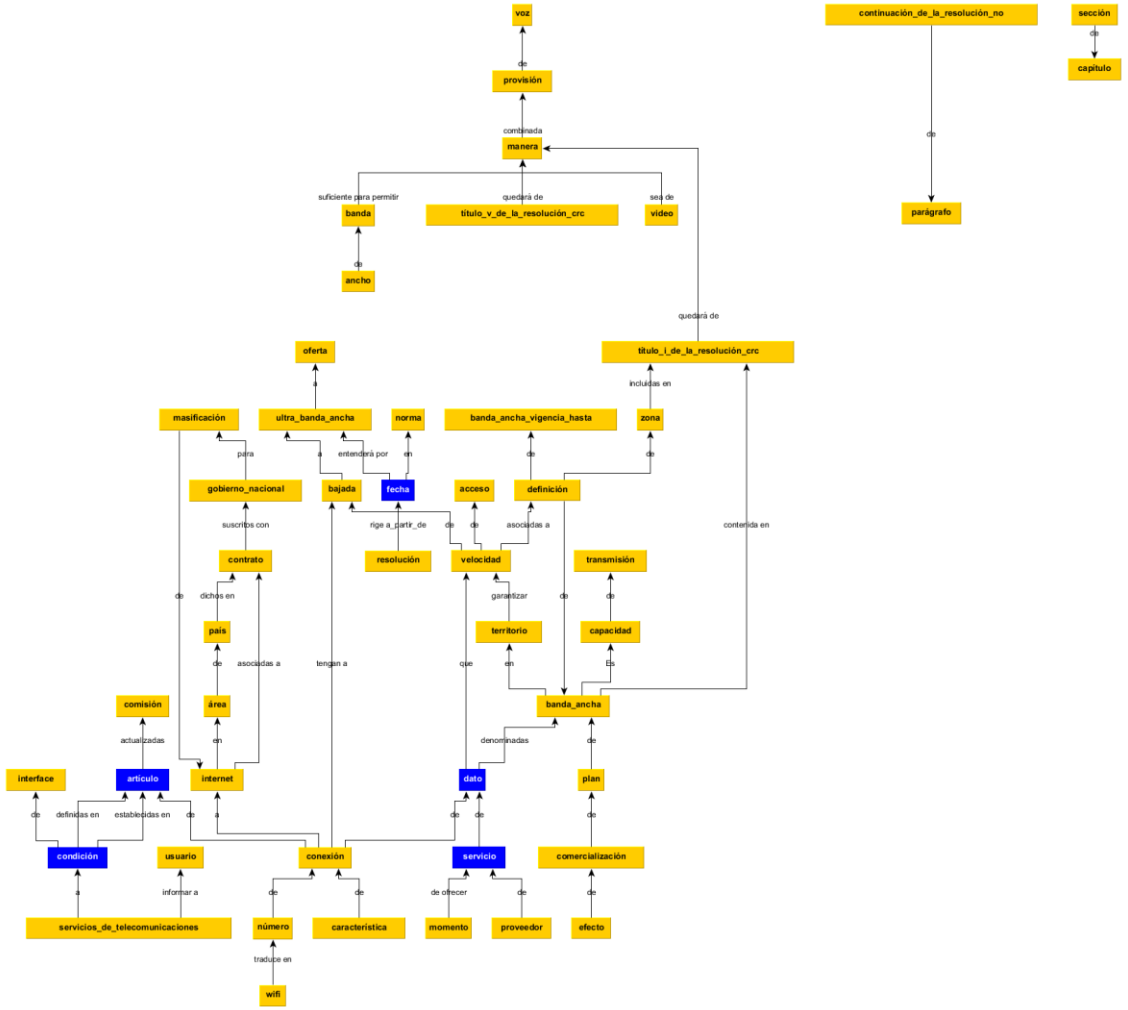


Figura 18. Modelo resultante del documento 4

Para comprobar la hipótesis se ejecuta una prueba de hipótesis así:

- En este caso se están comparando dos poblaciones que contienen 16 datos cada una, al ser un número inferior a 30 se utiliza la distribución t-Student para la prueba.

- Las variables para la definición de hipótesis son las siguientes:
  TU1 = tiempo muestra grupo A o grupo de control
  TU2 = tiempo muestra grupo B o grupo con el método aplicado
  PU1 = porcentaje correctas muestra grupo A o grupo de control
  PU2 = porcentaje correctas muestra grupo B o grupo con el método aplicado

- De acuerdo a la definición dada en el ítem "Criterios de satisfacción", la hipótesis nula (H0) y la hipótesis alternativa (H1) se definen así:

  H0: (TU1 <= TU2) o (PU1 > PU2)
  H1: (TU1 > TU2) y (PU1 <= PU2)

- El nivel de significancia o alfa se define como 0.05. La literatura recomienda un valor entre 0.01 y un 0.05, considerando la naturaleza del experimento y el rigor con el que se abordan la captura de tiempos y el diligenciamiento de los cuestionarios, se opta por el valor 0.05.

- La validez estadística de las conclusiones de las pruebas de hipótesis se garantiza mediante un nivel de confianza del 95% y se buscaron muestras representativas de acuerdo con las posibilidades, intentando evitar aceptar una hipótesis que fuera falsa o rechazar una que fuera verdadera.

- Estas pruebas se realizan mediante el complemento de Análisis de Datos de Microsoft Excel 2016. Para esta se asume que ambas muestras tienen varianzas desiguales, por lo que se ejecuta la función "Prueba t para dos muestras suponiendo varianzas desiguales". En la figura 19 se muestra el cuadro de dialogo para ingresar los parámetros en Excel.
  NOTA: El rango de datos para la variable 1 corresponde a datos del grupo A y el rango de datos para la variable 2 corresponde a datos del grupo B.

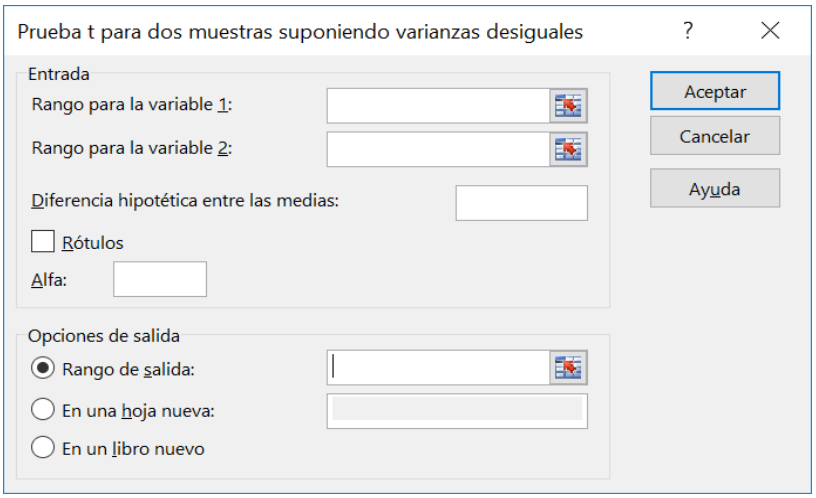


Figura 19. Cuadro de diálogo de prueba t para dos muestras en Excel

- La herramienta solicita la diferencia hipotética entre las medidas, para el caso del ejemplo asumimos que las dos muestras no tienen diferencia hipotética, dado que es precisamente lo que se busca establecer con la prueba, así, el valor para este parámetro es cero (0).

- Dado que la hipótesis usa dos variables de decisión y la herramienta solo permite comparar dos muestras cada vez, se ejecuta dos pruebas t, la primera se usa para comparar las muestras teniendo en cuenta primero el tiempo (TU1 y TU2) y la segunda para comparar el porcentaje de respuestas correctas (PU1 y PU2).

- Los resultados muestran varios datos, entre los cuales se tienen el Valor crítico de t (una cola), Valor crítico de t (dos colas) y el Estadístico t. De acuerdo a los valores y la definición se evalúan resultados con respecto a la hipótesis así:

- En la primera prueba t el valor crítico t de una cola se ubica a la derecha, es decir positivo. Para rechazar la hipótesis nula con respecto al tiempo, el estadístico t debe ser mayor al valor crítico t positivo. En la figura 20 se ejemplifica la evaluación.

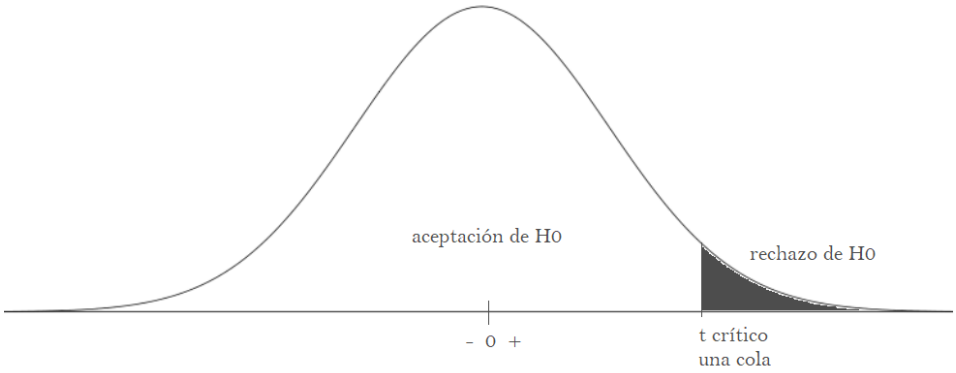


Figura 20. Región de aceptación o rechazo de H0, cola derecha

- En la segunda prueba t el punto crítico t de una cola se ubica a la izquierda, es decir negativo. Para rechazar la hipótesis nula con respecto al porcentaje de respuestas correctas el estadístico t debe ser menor al punto crítico t negativo. En la figura 21 se ejemplifica la evaluación.

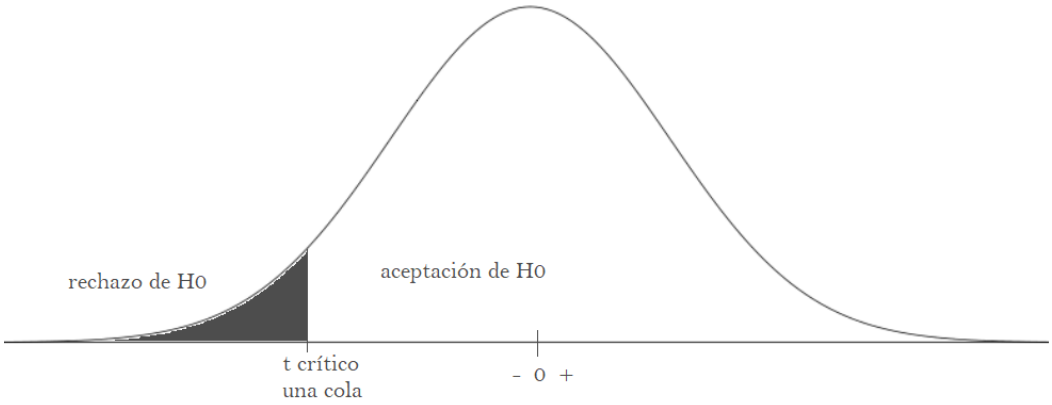


Figura 21. Región de aceptación o rechazo de H0, cola izquierda

- Las tablas resultantes se muestran tal cual las entrega la función de análisis de datos.

Validación para el Tiempo

En la tabla 13 se describen los valores resultantes para la prueba t, para la variable tiempo.

Tabla 13. Resultado prueba t para la variable tiempo

	Variable 1	*Variable 2*
Media	45,1875	25,75
Varianza	237,09583	131,4
Observaciones	16	16
Diferencia hipotética de las medias	0	
Grados de libertad	28	
Estadístico t	4,0502712	
P(T<=t) una cola	0,0001836	
Valor crítico de t (una cola)	1,7011309	
P(T<=t) dos colas	0,0003672	
Valor crítico de t (dos colas)	2,0484071	

Con los valores resultantes en la tabla 11 se pueden evaluar las hipótesis planteadas con respecto al tiempo, así:

- Dado que la prueba corresponde a una cola a la derecha se toma el valor crítico t (una cola) positivo: 1,7011309
- El valor estadístico t es 4,0502712, siendo mayor al valor crítico t.
- Dado lo anterior y teniendo en cuenta únicamente la primera parte de cada hipótesis, se puede afirmar que la hipótesis nula se rechaza.

  H0: (TU1 <= TU2) → Se rechaza.
  H1: (TU1 > TU2) → Se comprueba.

En este sentido, para poder rechazar la hipótesis completa es necesario evaluar la segunda prueba, lo cual se puede ver a continuación.

Validación de porcentaje de respuestas correctas nivel 1

En la tabla 14 se describen los valores resultantes para la prueba t, para la variable porcentaje de respuestas correctas.

Tabla 14. Resultado prueba t para la variable porcentaje de respuestas correctas nivel 1

	Variable 1	*Variable 2*
Media	0,916666667	0,90625
Varianza	0,010185185	0,02951389
Observaciones	16	16
Diferencia hipotética de las medias	0	
Grados de libertad	24	
Estadístico t	0,209121444	
P(T<=t) una cola	0,418059139	
Valor crítico de t (una cola)	1,71088208	
P(T<=t) dos colas	0,836118279	
Valor crítico de t (dos colas)	2,063898562	

Con estos valores se pueden evaluar las hipótesis planteadas con respecto al porcentaje de respuestas correctas de nivel de dificultad uno, así:

- Dado que la prueba corresponde a una cola a la izquierda se toma el valor crítico t (una cola) negativo: -1,71088208
- El valor estadístico t es 0,209121444, siendo mayor al valor crítico t.
- Dado lo anterior y teniendo en cuenta únicamente la segunda parte de cada hipótesis, se puede afirmar que la hipótesis nula no se rechaza, así:

Con respecto al porcentaje

H0: (PU1 > PU2). No se rechaza
H1: (PU1 <= PU2). Se rechaza.

Complementando las hipótesis con la definición dada en el punto 7.2 Resultados, evaluando ambas condiciones (tiempo y porcentaje de respuestas correctas) teniendo en cuenta solo las preguntas de nivel de dificultad 1.

H0: (TU1 <= TU2) o (PU1 > PU2). No se rechaza.
H1: (TU1 > TU2) y (PU1 <= PU2). Se rechaza.

Validación de porcentaje de respuestas correctas nivel 2

Tabla 15. Resultado prueba t para la variable porcentaje de respuestas correctas nivel 2

	Variable 1	*Variable 2*
Media	0,87797619	0,88869048
Varianza	0,03869048	0,01645616
Observaciones	16	16
Diferencia hipotética de las medias	0	
Grados de libertad	26	
Estadístico t	-0,1825003	
P(T<=t) una cola	0,42830282	
Valor crítico de t (una cola)	1,70561792	
P(T<=t) dos colas	0,85660564	
Valor crítico de t (dos colas)	2,05552944	

Con los valores dados en la tabla 15 se pueden evaluar las hipótesis planteadas con respecto al porcentaje de respuestas correctas de nivel de dificultad dos, así:

- Dado que la prueba corresponde a una cola a la izquierda se toma el valor crítico t (una cola) negativo: -1,70561792
- El valor estadístico t es -0,1825003, siendo mayor al valor crítico t.
- Dado lo anterior y teniendo en cuenta únicamente la segunda parte de cada hipótesis se puede afirmar que la hipótesis nula no se rechaza, así:

Con respecto al porcentaje:
H0: (PU1 > PU2). No se rechaza

H1: (PU1 <= PU2). Se rechaza.

Complementando las hipótesis con la definición dada en el punto 7.2 Resultados, evaluando ambas condiciones (tiempo y porcentaje de respuestas correctas) teniendo en cuenta solo las preguntas de nivel de dificultad 2.

H0: (TU1 <= TU2) o (PU1 > PU2). No se rechaza.
H1: (TU1 > TU2) y (PU1 <= PU2). Se rechaza.

Validación de porcentaje de respuestas correctas nivel 3

Tabla 16. Resultado prueba t para la variable porcentaje de respuestas correctas nivel 3

	Variable 1	*Variable 2*
Media	0,890625	0,78125
Varianza	0,03029514	0,09803241
Observaciones	16	16
Diferencia hipotética de las medias	0	
Grados de libertad	23	
Estadístico t	1,22128806	
P(T<=t) una cola	0,11717428	
Valor crítico de t (una cola)	1,71387153	
P(T<=t) dos colas	0,23434855	
Valor crítico de t (dos colas)	2,06865761	

Con los valores dados en la tabla 16 se pueden evaluar las hipótesis planteadas con respecto al porcentaje de respuestas correctas de nivel de dificultad tres, así:

- Dado que la prueba corresponde a una cola a la izquierda se tomas el valor crítico t (una cola) negativo: -1,71387153
- El valor estadístico t es 1,22128806, siendo mayor al valor crítico t.
- Dado lo anterior y teniendo en cuenta únicamente la segunda parte de cada hipótesis se puede afirmar que la hipótesis nula no se rechaza, así:

Con respecto al porcentaje

H0: (PU1 > PU2). No se rechaza
H1: (PU1 <= PU2). Se rechaza.

Complementando las hipótesis con la definición dada en el punto 7.2 Resultados, evaluando ambas condiciones (tiempo y porcentaje de respuestas correctas) teniendo en cuenta solo las preguntas de nivel de dificultad 3.

H0: (TU1 <= TU2) o (PU1 > PU2). No se rechaza.
H1: (TU1 > TU2) y (PU1 <= PU2). Se rechaza.

Validación de porcentaje de respuestas correctas en total

Tabla 17. Resultado prueba t para la variable porcentaje de respuestas correctas total

	Variable 1	*Variable 2*
Media	0,88214286	0,8703125
Varianza	0,01095106	0,01375481
Observaciones	16	16
Diferencia hipotética de las medias	0	
Grados de libertad	30	
Estadístico t	0,30106326	
P(T<=t) una cola	0,38272154	
Valor crítico de t (una cola)	1,69726089	
P(T<=t) dos colas	0,76544309	
Valor crítico de t (dos colas)	2,04227246	

Con los valores dados en la tabla 17 se pueden evaluar las hipótesis planteadas con respecto al porcentaje de respuestas correctas en general, así:

- Dado que la prueba corresponde a una cola a la izquierda se toma el valor crítico t (una cola) negativo: -1,69726089
- El valor estadístico t es 0,30106326, siendo mayor al valor crítico t.
- Dado lo anterior y teniendo en cuenta únicamente la segunda parte de cada hipótesis se puede afirmar que la hipótesis nula no se rechaza, así:

Con respecto al porcentaje

H0: (PU1 > PU2). No se rechaza
H1: (PU1 <= PU2). Se rechaza.

Complementando las hipótesis con la definición dada en el punto 7.2 Resultados, evaluando ambas condiciones (tiempo y porcentaje de respuestas correctas) teniendo en cuenta solo las preguntas en general.

H0: (TU1 <= TU2) o (PU1 > PU2). No se rechaza.
H1: (TU1 > TU2) y (PU1 <= PU2). Se rechaza.

Calificación de los modelos basada en criterios

Además de los datos propios para comprobar la hipótesis, los analistas dieron calificaciones de cada archivo de acuerdo a los criterios definidos en el capítulo anterior, con los siguientes resultados:

Tabla 18. Resultados de calificaciones de los modelos resultantes

	Promedio grupo A ↓	Promedio Grupo B ↓	Promedio total ↓
Legibilidad	3,31	3,56	3,44
Completitud	3,94	3,56	3,75
Corrección	3,69	3,94	3,81
Minimalidad	3,56	3,94	3,75
Expresividad	4,13	3,63	3,88
Autoexplicación	3,75	3,63	3,69
Extensibilidad	3,88	4,06	3,97
Promedio →	**3,75**	**3,76**	**3,75**

Observaciones adicionales durante la ejecución de la prueba

En observaciones adicionales durante la ejecución de la prueba, se evidenció los siguientes aspectos que pudieron influir tanto en el resultado como en la calificación dada a cada criterio en los documentos:

- En los casos en que un solo documento de texto presentaba gran cantidad de texto el modelo resultante derivaba en una gran cantidad de Conceptos, lo que hacía el modelo difícil de leer en principio, sin embargo, los analistas lograban hacerlo.
- En los documentos con poca cantidad de texto la dificultad para interpretar la información contenida es mínima.
- Todos los analistas tenían algún conocimiento o han tenido algo de experiencia en empresas que prestan servicios de tele-comunicaciones por lo que, teniendo en cuenta que estas resoluciones regulan este tipo de servicio, poseen algún conocimiento previo del dominio específico.
- Durante la ejecución de las pruebas, cuando los analistas del grupo A pudieron interpretar más rápidamente el modelo generado para cada documento que los analistas del grupo B.
- Los analistas manifestaron que, en los modelos más grandes y cargados de conceptos, existía conceptos con más de una relación entre sí, lo que generaba dificultades para la lectura del modelo.

7.3 Análisis de hallazgos

La primera variable a tener en cuenta para la hipótesis es el tiempo, en este sentido se evidenció claramente que el tiempo para procesamiento usado por el grupo de control es superior al tiempo tomado por el grupo con la aplicación del método, tal como se comprueba en las pruebas de hipótesis.

Es importante anotar que los analistas previamente interpretaban el modelo antes de proceder a responder las preguntas, proceso que resultaba rápido, sin embargo, al contestar las preguntas tenían que apoyarse en el modelo sin una opción de búsqueda de conceptos por lo que tenían que hacer una búsqueda manual, que en muchos casos resultaba intuitiva. Con la lectura de los documentos las respuestas se podían contestar mucho más rápidamente al poder hacer búsqueda de términos.

Con respeto al porcentaje de respuestas correctas se dividió las preguntas en tres niveles de dificultad de modo que se pudiera evaluar el método propuesto de acuerdo a estos niveles, sin embargo, para todos los casos se observó un comportamiento similar en tanto la prueba de hipótesis tipo t para cada caso no permitió concluir que se acepta la hipótesis planteada en este documento.

Si se tienen en cuenta los promedios en los porcentajes de repuestas correctas, se puede establecer un comparativo entre los resultados del grupo A y el grupo B, encontrando que las diferencias porcentuales para las preguntas de nivel de dificultad uno y dos son cercanas 1%, lo que muestra que el método propuesto permite mantener un nivel de interpretación muy cercano al proceso realizado en forma directa mediante lectura de documentos para cuestiones básicas. Esto no sucede para preguntas de nivel 3 donde la diferencia es más significativa, como se ve en la tabla 19. Con esto, y teniendo en cuenta que el tiempo efectivamente se reduce aplicando el método, este ofrece una buena aproximación a la hipótesis planteada.

Tabla 19. Resultados de calificaciones de los modelos resultantes

Grupo	% Correctas Nivel 1	% Correctas Nivel 2	% Correctas Nivel 3	% Correctas Total
A	91,67%	87,80%	89,06%	88,21%
B	90,63%	88,87%	78,13%	87,03%
Diferencia(A-B)	**1,04%**	**-1,07%**	**10,94%**	**1,18%**

En la calificación dada por los analistas, se observa que en general no se presentan grandes diferencias entre las evaluaciones que se emitieron ambos grupos, anotando que la Legibilidad para ambos es el aspecto con la calificación más baja, lo que muestra que este aspecto es el primero que se debe tener en cuenta para una evolución de la herramienta para implementar el modelo.

En general, las calificaciones son consistentes con los resultados para los porcentajes de respuestas correctas, en tanto no presenta los resultados más altos, sin embargo, se obtienen resultados en niveles que permiten concluir que el modelo es una buena aproximación a solucionar el problema planteado en este trabajo, que requiere de mejoras para establecerse como una solución definitiva.

Los cambios que se pueden implementar para mejorar la ejecución del método son:

- Afinar la identificación de relaciones entre los conceptos. En algunos casos la automatización identifica varias relaciones para dos conceptos, al tomar la más relevante se le resta elementos al modelo sin sacrificar contenido importante, esto aportaría a mejorar la legibilidad del modelo generado.

- Afinar la identificación de conceptos. Si bien el modelo carga conceptos relevantes para el dominio y el contenido mismo de los documentos, en algunos casos se identifican conceptos no tan relevantes, lo que le genera un poco de ruido a los modelos.
- Hacer la identificación automática de atributos. Esto permitiría interpretar mejor la información contenida en los modelos.
- Integrar las herramientas externas en una sola aplicación, lo que aportaría en mejorar los tiempos que toma ejecutar el proceso.

7.4 Listado de divulgación

- Escritura y aprobación de capítulo de libro de investigación "*Analítica de Texto para procesar documentos de requisitos*" del libro "*Tecnologías del lenguaje humano: aplicaciones desde la Lingüística Computacional y de Corpus*", editado por Sello Editorial Universidad de Medellín. El presente trabajo sirvió como caso de estudio para la aplicación del método propuesto en el capítulo.

- Participación en Simposio Regional de Maestría y Doctorado dado en el marco del Seminario Internacional de Ciencias de la Computación (SICC, 2018).

PARTE V:

CONCLUSIONES

CAPÍTULO 8. CONCLUSIONES Y TRABAJO FUTURO

8.2 Conclusiones

- La industria presenta retos para el procesamiento de información de negocio contenida en documentos textuales. Esta información puede ser útil para los ingenieros de requisitos, sin embargo, no siempre la usan para su labor por lo que resulta útil proveer herramientas que faciliten esta tarea.

- Es posible aplicar técnicas de Procesamiento de lenguaje natural que permitan procesar texto y deriven en la generación de modelos de representación del conocimiento, así, la comunidad científica ha propuesto métodos y/o herramientas de software cuyo objetivo es extraer información de texto y generar representación de conocimiento.

- Si bien se evidencian propuestas, existen pocos trabajos que busquen procesar información en idioma español, y que además generen algún modelo de representación de conocimiento, por lo que, en el ámbito regional como países de habla hispana, resulta muy útil presentar un método que atienda este segmento.

- De acuerdo a la experimentación, es útil representar conocimiento en modelos ya conocidos por las comunidades en las que influye el dominio procesado o que sean fácilmente interpretables, como lo son los modelos UML.

- Es de gran ayuda aplicar técnicas o métodos ya propuestos que permitan organizar las actividades a ejecutar, como por ejemplo en el caso particular, el método para proyectos de analítica de texto permitió organizar las actividades para aplicar NLP a documentos de negocio generando representación del conocimiento.

- El método acá propuesto, si bien no presenta una solución definitiva al problema plateado, de acuerdo a los resultados se demuestra que es útil y aplicable para el caso de estudio presentado, y podría ser implementado y probado en otros dominios.

- Si bien la evaluación del método arrojó resultados aceptables, es necesario evolucionarlo de modo que permita mejorar la calidad del conocimiento representado con respecto al contenido de los documentos procesados, tanto para el dominio presentado en el caso de estudio como para otros dominios en los cuales pueda ser aplicado.

8.3 Trabajo futuro

- Afinar el método para que priorice las relaciones identificadas y tome solo la más relevante, evitando poner más de una relación entre dos atributos.

- Adicionar al método la identificación de atributos de los conceptos identificados a partir del documento, sí como su implementación en la herramienta de software para que se identifique y se grafique en el modelo resultante.

- Pre-entrenar *Freeling* para afinar la identificación de conceptos de negocio específicos o que *Freeling* no identifique correctamente.

- Incluir a los analistas que participan de la generación automática de modelos en la identificación de *IC (Information Content)* para los pesos de negocio que sirven de entrada a la herramienta de software.

- Dado que muchos de los documentos son de gran extensión, sería útil hacer una "descomposición funcional", es decir, que el modelador identifique grupos de conceptos en forma automática y genere varios modelos de una vez, de acuerdo a esta agrupación funcional.

- Mejorar gráficamente el modelador de modo que se presente un *Wizard*, lo que brindaría una mejor experiencia de usuario y permitiría acciones adicionales, por ejemplo, seleccionar específicamente los conceptos que desea ver en el modelo

- Integrar una de las herramientas de visualización al modelador mismo, así se evitaría tener que instalar y/o usar otra herramienta de software.

REFERENCIAS BIBLIOGRÁFICAS

[1] P. Velasco-Elizondo, R. Marín-Piña, S. Vazquez-Reyes, A. Mora-Soto, and J. Mejia, "Knowledge representation and information extraction for analysing architectural patterns," *Sci. Comput. Program.*, vol. 121, pp. 176–189, 2016.

[2] C. Burnay, I. J. Jureta, and S. Faulkner, "What stakeholders will or will not say: A theoretical and empirical study of topic importance in Requirements Engineering elicitation interviews," *Inf. Syst.*, vol. 46, pp. 61–81, 2014.

[3] N. Ibrahim, W. Kadir, M. N. Wan, and S. Deris, "Documenting requirements specifications using natural language requirements boilerplates," 2014, pp. 19–24.

[4] V. Arnaoudova, S. Haiduc, A. Marcus, and G. Antoniol, "The Use of Text Retrieval and Natural Language Processing in Software Engineering," in *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, 2015, pp. 949–950.

[5] W. Hsu, C. W. Arnold, and R. K. Taira, "A Neuro-Oncology Workstation for Structuring , Modeling , and Visualizing Patient Records," in *IHI '10 Proceedings of the 1st ACM International Health*, 2010, pp. 837–840.

[6] K. M. Annervaz, V. Kaulgud, S. Sengupta, and M. Savagaonkar, "Natural language requirements quality analysis based on business domain models," in *2013 28th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2013, pp. 676–681.

[7] G. Ninaus *et al.*, "INTELLIREQ: Intelligent Techniques for Software Requirements Engineering," pp. 1161–1166, 2014.

[8] S. M. Jimenez, A. Gelbukh, and G. Sidorov, "Generating summaries by means of synthesis of conceptual graphs," *Rev. Signos Vol 47*, vol. 47, no. 86, pp. 463–485, 2014.

[9] J. F. Allen, "Natural language processing," pp. 1218–1222, 2003.

[10] G. G. Chowdhury, "Natural Language Processing," vol. 37, no. 1, pp. 51–89, 2003.

[11] M. Ozkaya, "What is software architecture to practitioners: A survey," *2016 4th Int. Conf. Model. Eng. Softw. Dev.*, pp. 677–686, 2016.

[12] N. V. Do, "Ontology COKB for Knowledge Representation and Reasoning in Designing Knowledge-Based Systems Ontology COKB for Knowledge Representation," *Commun. Comput. Inf. Sci.*, no. January 2015, pp. 101–118, 2015.

[13] V. Bhala, V. Sagara, and S. Abiramib, "Conceptual modeling of natural language functional requirements," *J. Syst. Softw. Vol. 88*, vol. 88, no. 1, pp. 25–41, 2014.

[14] S. Roychoudhury, N. Bellarykar, and V. Kulkarni, "A NLP Based Framework to Support Document Verification-as-a-Service," in *2016 IEEE 20th International Enterprise Distributed Object Computing Conference (EDOC)*, 2016, pp. 139–148.

[15] K. P. Sawant, S. Roy, D. Parachuri, F. Plesse, and P. Bhattacharya, "Enforcing structure on textual use cases via annotation models," in *ISEC '14 Proceedings of the 7th India Software Engineering Conference*, 2014.

[16] M. A. Naeem and I. S. Bajwa, "Generating OLAP queries from natural language specification," in *ICACCI '12 Proceedings of the International Conference on Advances in Computing, Communications and Informatics*, 2012, pp. 768–773.

[17] Brass M. and Toussaint Y., "Artificial intelligence tools for software engineering: Processing natural language requirements," *Trans. Inf.*

Commun. Technol. vol 2, vol. 11, pp. 275–290, 1995.

[18] B. Javed and S. Sultan M, "Process Support for Requirements Engineering Activities in Global Software Development: A Literature Based Evaluation," *2010 Int. Conf. Comput. Intell. Softw. Eng.*, pp. 1–6, 2010.

[19] X. Xiao, A. Paradkar, and T. Xie, "Automated extraction and validation of security policies from natural-language documents," *Proc. ACM SIGSOFT 20th Int. Symp. Found. Softw. Eng.*, 2012.

[20] O. A. Beltrán G., "Revisiones sistemáticas de la literatura," vol. 20, no. 1, pp. 60–69, 2005.

[21] T. Diamantopoulos, M. Roth, A. Symeonidis, and E. Klein, "Software Requirements as an Application Domain for Natural Language Processing," *Lang. Resour. Eval. Vol. 51*, pp. 495–524, 2017.

[22] U. Iqbal and I. S. Bajwa, "Generating UML activity diagram from SBVR rules," in *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, 2016, pp. 216–219.

[23] D. A. de Araujo, S. J. Rigo, C. Muller, and R. Chishman, "Automatic information extraction from texts with inference and linguistic knowledge acquisition rules," *2013 IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol.*, vol. 3, pp. 151–154, 2013.

[24] I. S. Bajwa, B. Bordbar, and M. Lee, "OCL usability: a major challenge in adopting UML," in *Proceedings of the 3rd International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering - RAISE 2014*, 2014, vol. 92, no. 0, pp. 32–37.

[25] R. Pinquie *et al.*, "Natural Language Processing of Requirements for Model-Based Product Design with ENOVIA / CATIA V6," *PLM 2015 Prod. Lifecycle Manag. Era Internet Things*, pp. 205–215, 2015.

[26] P. Fernandes, L. O. C. Furquim, and L. Lopes, "A supervised method to enhance vocabulary with the creation of domain specific lexica," in *2013 IEEE/WIC/ACM International Conferences on Web Intelligence (WI) and Intelligent Agent Technology (IAT)*, 2013, vol. 3, pp. 139–142.

[27] C. P. Escartín and H. M. Alonso, "Choosing a Spanish Part-of-Speech tagger for a lexically sensitive task," *Proces. del Leng. Nat. Rev. nº 54*, vol. 54, pp. 29–36, 2015.

[28] B. Hnatkowska and T. Gawęda, "Automatic Processing of Dynamic Business Rules Written in a Controlled Natural Language," *Towar. a Synerg. Comb. Res. Pract. Softw. Eng.* , vol. 3, pp. 91–103, 2018.

[29] H. van der Aa, H. Leopold, and H. A. Reijers, "Comparing textual descriptions to process models – The automatic detection of inconsistencies," *Inf. Syst. Vol. 64*, vol. 64, pp. 447–460, 2017.

[30] S. Miranda, F. Orciuoli, and D. G. Sampson, "A SKOS-based framework for Subject Ontologies to improve learning experiences," *Comput. Hum. Behav. Vol. 61*, vol. 61, pp. 609–621, 2016.

[31] D. Movshovitz-Attias and W. W. Cohen, "Natural Language Models for Predicting Programming Comments," in *ACL*, 2013, pp. 35–40.

[32] W. Daelemans, "POS Tagging," in *Encyclopedia of Machine Learning and Data Mining*, T. Zeugmann, Ed. 2017.

[33] B. P. Upadhyaya, *Programming with Scala*. Springer Nature, 2017.

[34] D. W. Embley and B. Thalheim, *Handbook of Conceptual Modeling*. 2011.

[35] C. Larman, *UML y patrones: Una introducción al análisis y diseño orientado a objetos y al proceso unificado*, 2nd ed. Madrid: PERASON, 2003.

[36] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research 1," vol. 28, no. 1, pp. 75–105, 2004.

[37] B. Manrique, J. B. Quintero, D. A. Marin, and J. S. Morales, "Analítica de

Texto para procesar documentos de requisitos," in *Tecnologías del lenguaje humano: aplicaciones desde la Lingüística Computacional y de Corpus*, Sello Editorial Universidad de Medellín, Ed. Medellín, 2018, p. En edición.
[38] John Wiley, *Data Science and Big Data Analytics*. 2015.
[39] A. Olivé, *Conceptual Modeling of Information Systems*. 2007.
[40] D. Rosenberg and M. Stephens, "Domain Modeling," in *Use Case Driven Object Modeling with UML*, 2007, pp. 23–48.
[41] M. F. González Pinzón and J. S. González Sanabria, "Aplicación del estándar ISO/IEC 9126-3 en el modelo de datos conceptual entidad-relación," vol. 22, no. 35, pp. 113–125, 2013.