Generación automática de instancias del modelo cognitivo-afectivo (COGAF) a partir de textos usando Procesamiento de Lenguaje Natural (PLN)

Estudiante:

Alejandro Velásquez Arango

avelas56@eafit.edu.co

Director:

José Lisandro Aguilar Castro

Área de Computación y Analítica

jlaguilarc@eafit.edu.co

UNIVERSIDAD EAFIT

MEDELLÍN

2024

Resumen

Los juegos serios son una herramienta muy útil para incentivar la educación y el entrenamiento en diversas áreas del conocimiento. Crear juegos serios que mejoren de manera eficaz las funciones cognitivas y las emociones que influencia una capacidad determinada puede ser un gran desafío. Para esto el modelo Cognitivo-Afectivo (COGAF) presenta una conceptualización de los componentes necesarios para el diseño de juegos serios. Sin embargo, la instanciación de estos componentes actualmente se hace de manera manual a partir de la descripción textual de un caso de estudio que se pretenda simular a través de un juego serio. Este trabajo tiene como objetivo generar automáticamente instancias del modelo COGAF siendo definido como una ontología, aplicando un sistema de población automática de ontología y utilizando técnicas de Procesamiento de Lenguaje Natural (PLN) a partir de textos de casos de estudios para facilitar y acelerar el proceso de diseño de juegos serios.

1. Planteamiento del problema

Los juegos serios son aquellos que han sido diseñados con un objetivo principal diferente al de la mera diversión. Normalmente estos juegos tienen el objetivo de entrenar o enseñar al usuario sobre algún tema en particular y resolver problemas del mundo real. Los juegos serios han ido ganando popularidad en distintos sectores debido a la efectividad que tienen para mejorar la experiencia del aprendizaje. Esto se logra con la integración de simulaciones, mecánicas de juego y estrategias pedagógicas que mejoran la experiencia del usuario y mitiga posibles problemas de memoria, atención y concentración.

La Ingeniería Dirigida por Modelos (MDE de sus siglas en inglés) es un acercamiento para el desarrollo de software que se centra en la creación y utilización de modelos conceptuales de un dominio del conocimiento que puedan facilitar y acelerar el proceso de desarrollo. Estos modelos pueden incluso ser utilizados para generación automática de código y otros artefactos. Dentro de este ámbito, Llanez et al. [1] definieron el metamodelo Cognitivo-Afectivo (COGAF) que describe las dimensiones cognitivas y afectivas involucradas en los juegos serios, ya que no se encontraba un mecanismo para el entrenamiento cognitivo-afectivo a través de juegos serios.

El modelo COGAF permite conceptualizar el diseño de los juegos serios para el entrenamiento cognitivo-afectivo, identificando diferentes componentes y las relaciones entre la cognición, la emoción y las mecánicas del juego, y puede ser instanciado para representar caso particulares. Este modelo permite que el desarrollo de juegos serios que entrenen capacidades cognitivas y afectivas pueda llevarse a cabo dentro el marco de la metodología MDE, resultando en mejor productividad, calidad y mantenimiento del software desarrollado gracias a la posibilidad que brinda el modelo de generar código de forma automática.

Actualmente, la instanciación de los componentes del modelo cognitivo-afecto se hace manualmente a partir de textos que detallen los casos de estudios que pueden ser modelados. Un ejemplo en particular es el caso de un accidente minero por explosión redactado por la Agencia Nacional de Minería (ANM) de Colombia, descrito en un documento textual en donde se detalla el evento ocurrido, las causas que generaron el accidente, las consecuencias y las lecciones aprendidas para evitar que ocurran de nuevo siniestros similares. Del texto se pueden determinar instancias de los componentes del

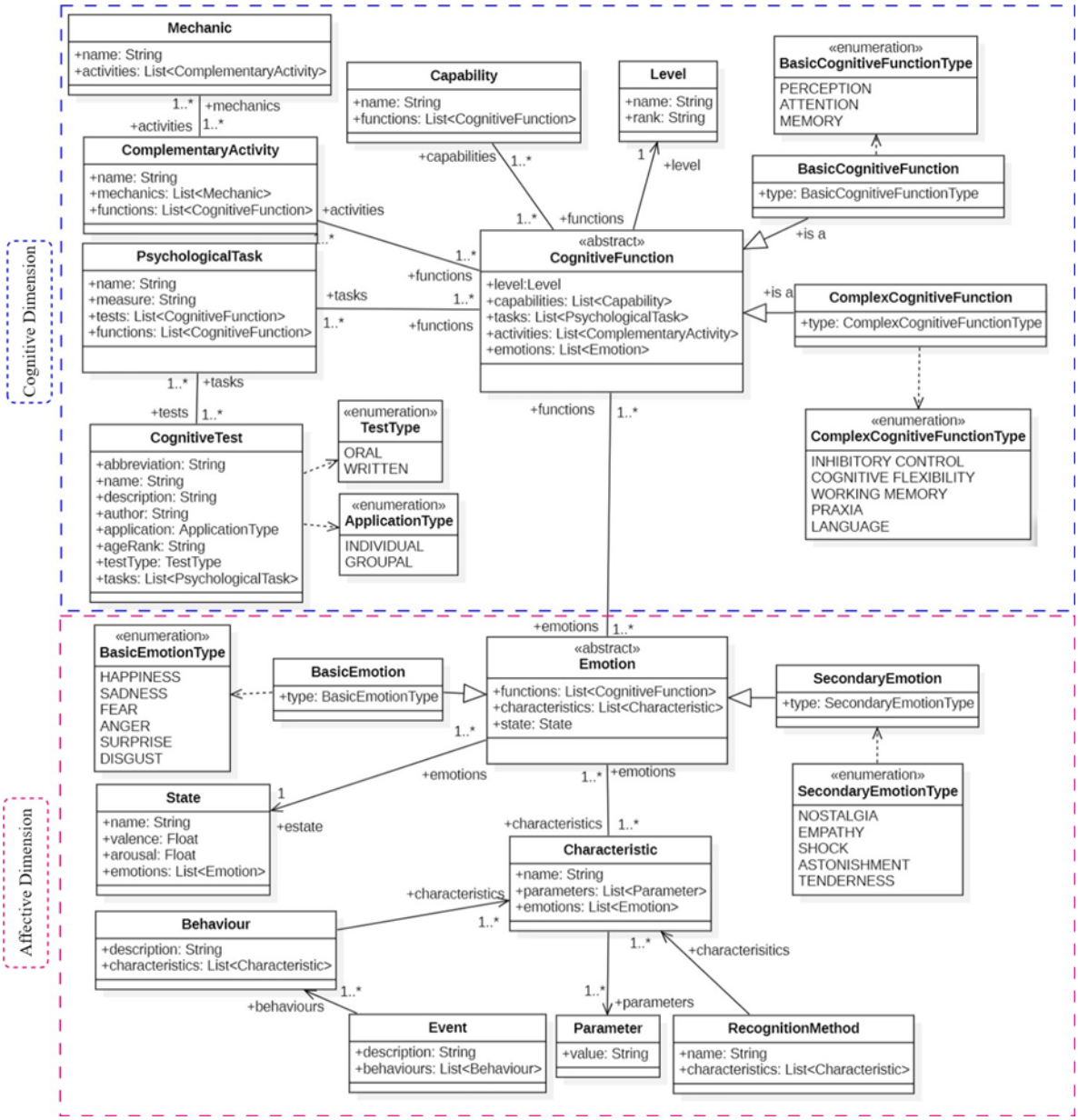

Figura 1: Modelo Cognitivo-Afectivo (COGAF)

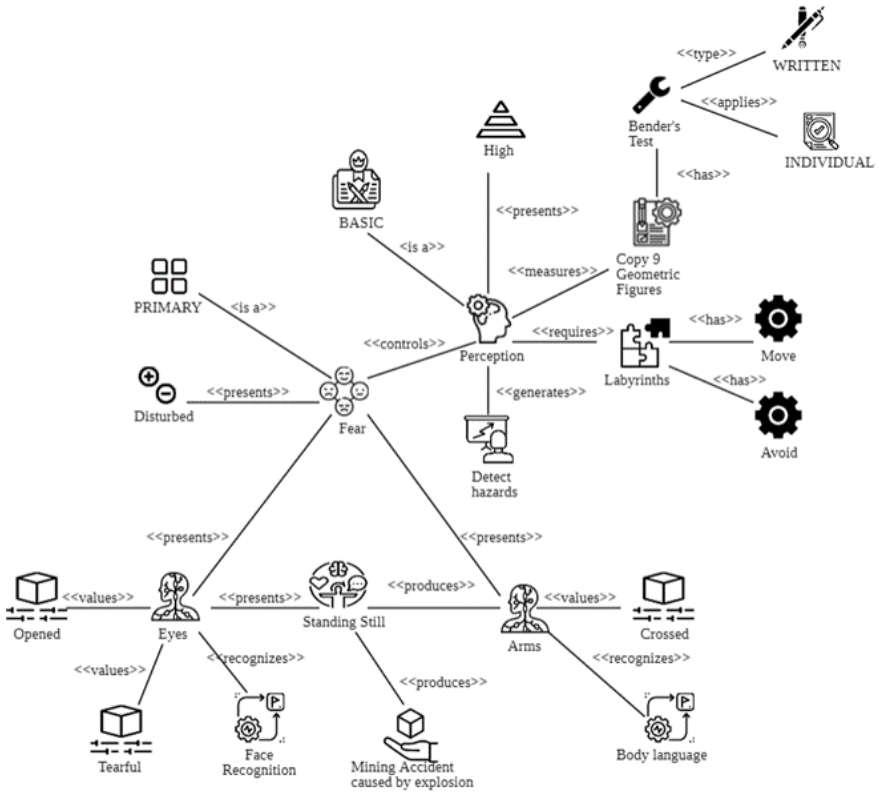

Figura 2: Ejemplo de instanciación del Modelo COGAF

modelo, como se ve en la figura 2

La determinación de las instancias del modelo COGAF a partir de un caso de estudio particular toma bastante tiempo y requiere conocimiento experto. De esto se deriva la necesidad de poder realizar este proceso de instanciación de manera automática para acelerar este proceso. A partir de técnicas de Procesamiento de Lenguaje Natural (PLN) es posible desarrollar un modelo de Inteligencia Artificial que sea capaz de identificar las instancias a partir de un texto similar al caso de estudio mencionado anteriormente.

Adicionalmente, el modelo COGAF puede ser representado formalmente como una ontología, una representación formal del conocimiento que de un dominio en particular. Los conceptos, atributos y relaciones de los componentes del modelo COGAF pueden ser modelados dentro de una ontología formalmente definida. Esto permite que el modelo sea legible por computadora y que sea posible generar instancias de los conceptos de manera que cumplan la estructura definida por la ontología. Así, el problema de la generación automática de instancias del modelo COGAF puede ser definido como un

problema de población de ontología, esto es, una vez definidos los conceptos abstractos y sus relaciones entre ellos, se busca instancias específicas de estos conceptos (ejemplos reales) y que cumplan las relaciones definidas en la ontología.

2. Justificación

Este proyecto sería de gran apoyo para el desarrollo sistemático de juegos serios para diferentes tipos de industrias. Empresas de los sectores como el de la minería, la construcción y la agricultura serían beneficiados de la implementación de los juegos serios como parte del entrenamiento de los trabajadores. Los juegos serios son una poderosa herramienta para el mejoramiento de las funciones cognitivas y emociones implicadas en las labores y potenciales accidentes que puedan enfrentarse. La posibilidad instanciar automáticamente el modelo COGAF aceleraría considerablemente el desarrollo de estos juegos serios, abaratando costos y acortando tiempos, permitiendo que los juegos serios puedan ser producidos en mayor escala.

3. Objetivos

3.1. Objetivo General

Generar automáticamente instancias del modelo cognitivo-afectivo partiendo de textos y aplicando Procesamiento del Lenguaje Natural (PLN).

3.2. Objetivos Específicos

- Definir el modelo COGAF como una ontología legible por computadora
- Obtener textos de lecciones aprendidas de accidentes laborales de distintos sectores de la industria
- Desarrollar un modelo de Procesamiento de Lenguaje Natural que genere instancias del modelo COGAF a partir de la ontología y los textos

4. Marco Teórico y Estado del Arte

4.1. Ontologías

En ciencias de la computación, una ontología es una representación formal y explícita de la conceptualización del conocimiento de un dominio de discurso en particular, definiendo clases, propiedades, relaciones, restricciones y axiomas. [2]. Una ontología provee una forma estructurada y explícita de representar el conocimiento, lo cual permite que este pueda ser leído y procesado por máquina. En otras palabras, una ontología es un conjunto altamente organizado que abarca conceptos, instancias, propiedades y las relaciones entre esos conceptos de un dominio específico, al igual que incluye definiciones formales y axiomas que acota el entendimiento del vocabulario [3]. Estas definiciones hacen que las ontologías constituyan una técnica de representación de conocimiento que encuentra diversas aplicaciones reales.

Por ejemplo, en 2022, Abbasi y otros desarrollaron un modelo ontológico para representar el conocimiento del sistema de acuaponía dentro del contexto de la industria 4.0[4] para unificar el conocimiento de este campo de distintas fuentes de conocimiento. En el mismo año, González y otros desarrollaron la ontología de la Pandemia del COVID-19 que permite la integración de otras ontologías para cubrir todos los aspectos de la enfermedad viral [5].

Una ontología esta compuesta por los siguientes elementos [6]:

- **Clases**: Objetos que definen categorías y sirven para representar conceptos.
- **Individuos**: Son instanciaciones de las clases. Son ejemplos reales de los conceptos representados por las clases.
- **Atributos**: Son propiedades asociadas con las clases y se les pueden definir un tipo de dato en particular.
- **Relaciones**: Son enlaces que vinculan las clases entre ellas. Pueden ser taxonómicas, que implican algún tipo de jerarquía, o no taxonómicas para otro tipo de relaciones que no definen jerarquías.

- **Axiomas**: Representan definiciones formales de la ontología. Sirven para definir condiciones y restricciones que deben cumplir los elementos de la ontología.

Las ontologías pueden ser especificadas a través de lenguajes formales para que puedan ser legibles por computadora. Entre los diversos lenguajes que existen para este fin [7, 8], el más utilizado es OWL (Web Ontology Language) [9]. Adicionalmente, existe software que facilita la creación de ontologías como Protégé [10], una plataforma libre de código abierto que provee una conjunto de herramientas para el desarrollo de ontologías y de aplicaciones integradas con estas.

4.2. Aprendizaje de ontologías

Las ontologías pueden ser construidas manualmente a partir de conocimiento experto y de fuentes de información, pero este proceso es altamente costoso y propenso a errores. El aprendizaje de ontologías busca desarrollar sistemas que sean capaces de construir ontologías de forma automática o semi-automática. Estos sistemas pueden consistir de un solo proceso que genere todos los componentes de la ontología o de múltiples etapas y herramientas para cada componente en particular. Los componentes obligatorios de una ontología son las clases y las relaciones, y cuando es enriquecida a través de axiomas que la estructuran y formalizan, se dice que la ontología es expresiva. Las instancias de una ontología son componentes adicionales y el proceso que genera estas instancias se conoce como población de ontologías, mientras que la creación de nuevos conceptos y relaciones que modifican la estructura se conoce como enriquecimiento.

Los sistemas de aprendizaje de ontologías pueden utilizar diferentes técnicas que se pueden agrupar en tres categorías [11]. Las técnicas léxico-sintácticas utilizan reglas y patrones del idioma para identificar los conceptos y relaciones de la ontología. Por ejemplo, para hallar relaciones taxonómicas, es frecuente el uso de patrones de Hearst [12] que permiten identificar relaciones de hiponimia. Las técnicas estadísticas se utilizan para agrupar los términos que aparecen con cierta frecuencia y evaluar su relevancia en el texto. El TF-IDF es un ejemplo de una técnica estadística.

Por último, las técnicas de machine learning se pueden utilizar junto con los métodos mencionados anteriormente para mejorar el rendimiento de los sistemas de aprendizaje de ontologías [13]. Por ejemplo, algoritmos de clasificación como SVMs y k-NN se

utilizan para la clasificación de instancias dentro de las clases definidas por una ontología ya estructurada. También, el uso de Deep Learning es ahora más frecuente en estos sistemas gracias a su capacidad de capturar información de naturaleza no lineal, a los avances en el poder de cómputo y a la abundancia de datos de entrenamiento disponibles.

A continuación se mencionan algunos trabajos relacionados al aprendizaje de ontologías, particularmente con población de ontologías debido a que es el caso que se presenta en esta propuesta.

En 2023, Chasseray et al. [14] presentaron un enfoque independiente del dominio basado en reglas léxico-sintácticas para la población automática de ontologías en un contexto no supervisado, mencionando que en la mayoría de casos de población de ontologías no se tienen datos anotados disponibles que permitan un desarrollo de un sistema supervisado, por lo que es importante que los sistemas de población puedan operar sin conocimiento previo de las entidades a ser extraídas. También presentaron un nuevo método de evaluación de rendimiento para estos sistemas no supervisados, con el método basándose en la explotación de datos de referencia pero que no están específicamente relacionados a aquellos utilizados para la extracción de conocimiento. En el mismo año, Sambandam et al. [15] implementaron el modelo de deep learning SECNN (*Spiking Equilibrium Convolutional Neural Network*) junto con técnicas de PLN para la identificación y extracción de términos textuales claves y asignarlos a los componentes predefinidos de una ontología existente en el dominio del análisis urbano. El modelo fue evaluado con métricas de precisión, exhaustividad y F1-Score.

En 2019, Reyes-Ortiz [16] presentó un modelo de enriquecimiento y población de una ontología sobre eventos criminales, utilizando patrones lingüísticos para extraer sintagmas nominales y verbales y, consecuentemente, capturar los eventos y causas mencionados en noticias escritas en español. El modelo fue evaluado comparando los resultados con textos etiquetados manualmente con las categorías de los eventos específicos. En el mismo año, Ayadi et al. [17] presentaron un sistema de población basado en deep learning y PLN para la Ontología de la Red Biomolecular, desarrollada por los mismos autores [18]. Para este año, las técnicas de deep-learning, aunque ya bastante avanzadas, no eran aún comúnmente aplicadas en sistemas ontológicos, por lo que los autores aprovecharon la capacidad de estas técnicas de extraer información de textos junto con PLN para desarrollar un sistema de población de la ontología mencionada a partir

de la gran cantidad de documentos textuales que existen sobre redes biomoleculares complejas.

En 2014, Faria et al. [19] propusieron un proceso genérico para la población automática de ontologías. Este proceso utiliza una ontología existente para generar las reglas que permiten la extracción de instancias en el texto y clasificarlas según las clases de la ontología. Estas reglas pueden ser generadas de ontologías de cualquier dominio, haciendo el proceso propuesto independiente del dominio y, por tanto, permitiendo la instanciación de ontologías con mayor rapidez y a menor costo.

5. Metodología

Para poder cumplir con los objetivos planteados, la metodología propuesta para el desarrollo del proyecto es CRISP-DM, la cual es extensivamente utilizada en la industria para la realización de proyectos relacionados con ciencia de datos e inteligencia artificial. La metodología CRISP-DM se divide en 6 etapas: Entendimiento del negocio, comprensión de los datos, preparación de datos, modelado, evaluación y despliegue. El flujo de estas etapas se ilustra en la figura 3.

- **Entendimiento del negocio:** En esta etapa se define el problema a resolver, los objetivos que se quieren alcanzar (presentados anteriormente) y se realiza una revisión de literatura sobre los trabajos ya realizados que guardan relación con el problema. Como se presentó en secciones anteriores, el problema es instanciar automáticamente los componentes del modelo COGAF, el cual puede ser visto como un problema de población de ontología. El marco teórico y algunos trabajos relacionados están presentados en la sección anterior. Adicionalmente, en esta fase se define un criterio de rendimiento que se utilizará en la etapa de evaluación.

- **Comprensión de los datos:** Esta etapa involucra la identificación de fuentes de datos al igual que su explotación para llevar a cabo el proyecto. Aquí se incluyen dos actividades: una siendo la definición del modelo COGAF como una ontología con el software Protégé, y la otra, la extracción de documentos de lecciones aprendidas que estén disponibles en la web. Para esto, se utilizará web-scraping para la obtención de estos textos. Las tecnologías a utilizar aun están por definir.

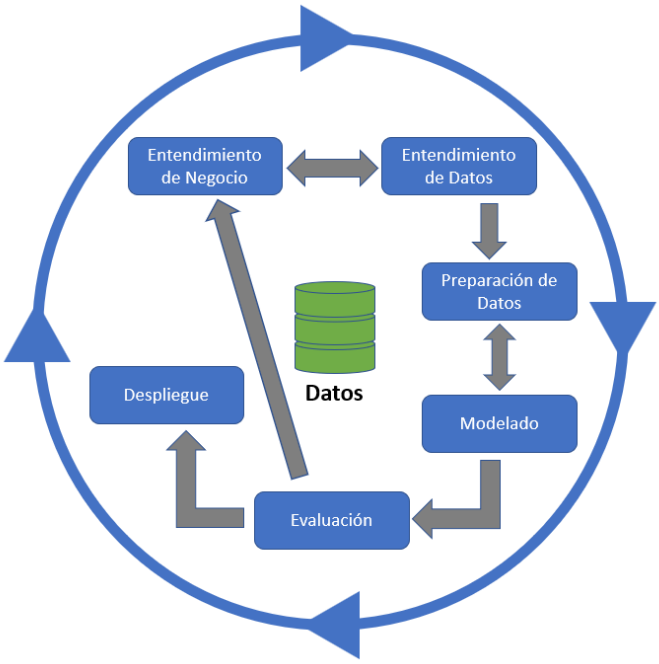

Figura 3: Esquema ilustrativo de las etapas de la metodología CRISP-DM

- **Preparación de los datos:** Ya con la ontología definida y los datos obtenidos, esta información debe ser preparada para ser alimentada al sistema de población de ontología. El procesamiento de los textos involucra el uso de técnicas de PLN para la transformación de datos no-estructurados a estructurados y que puedan ser procesados. Por otro lado, de la ontología se obtienen las reglas que relacionan los conceptos y definen las restricciones de las clases que también son inyectadas al sistema para proveer mayor información.

- **Modelado:** Para esta etapa, las reglas extraídas de la ontología y los datos de los textos están listos para alimentar los modelos de población de ontología. Este proceso es iterativo, ya que varios modelos son entrenados y sintonizados para obtener los mejores resultados de cada modelo. Incluso, de ser necesario, se puede revertir a la etapa anterior en caso de que se necesite mayor manipulación de los datos para entrenar algún modelo.

- **Evaluación:** Aquí, los modelos ya fueron entrenados y se evalúan su rendimiento según el criterio de rendimiento definido en la etapa de entendimiento. Aquí se escoge el modelo final que mejor rendimiento presente y se estudia los resultados que arroje para extraer conclusiones.

- **Despliegue:** La etapa final consiste en poner en funcionamiento el modelo escogido anteriormente y presentar un artículo que detalle el desarrollo y los resultados del proyecto. En caso de ser puesto en funcionamiento, esta etapa también involucra el monitoreo del modelo en producción.

6. Productos Esperados

Los productos finales que se entregarán en la finalización del proyecto consisten de un artículo científico sobre la elaboración del sistema y los resultados obtenidos de este, al igual que un repositorio que contenga el sistema automático de población de la ontología y el código escrito para entrenar el modelo.

7. Plan de Gestión de Datos

Para la elaboración del proyecto se utilizarán textos abiertos, es decir, aquellos que se puedan encontrar y extraer libremente de la web sin la necesidad de ningún tipo de acceso. Estos datos serán recolectados con técnicas de web scraping, evitando en la medida de lo posible violar los términos de servicio de los sitios web de donde se extraerán los datos. Los textos recopilados serán procesados, transformados y compartidos exclusivamente para el desarrollo del proyecto y no serán distribuidos a personas o entidades que no estén involucrados con este. Se utilizarán textos que no contengan información personal o confidencial de individuos o entidades.

8. Aspectos Éticos

El desarrollo de este proyecto facilitará y acelerará el desarrollo de los videojuegos serios que estimulen las funciones cognitivas de los jugadores. Los desarrolladores tendrán, además del modelo COGAF, una gran cantidad de instancias de este, lo cual potencia los beneficios que la metodología MDE trae a la creación de juegos serios.

Como se mencionó, los textos que se obtengan a partir del web-scraping serán, en principio, de carácter abierto y estarán dispnibles a todo publico en la web, por lo que no se requiere consentimiento para la extracción de estos textos.

Los textos utilizados no contienen información personal o confidencial de individuos o entidades, por que lo no es requerido la anonimización de los datos. En caso de que algunos de los textos utilizados haga mención de alguna persona en particular, se dispondrá del texto y no será utilizado para la elaboración del proyecto.

Referencias

[1] C. Y. G. Llanez, P. Vallejo, J. Aguilar, A generic metamodel for cognitive-affective training of users using serious games, IEEE, 2023, pp. 1–9. doi:10.1109/CLEI60451.2023.10346132.
URL https://ieeexplore.ieee.org/document/10346132/

[2] C. Faria, R. Girardi, P. Novais, Analysing the problem and main approaches for ontology population, IEEE, 2013, pp. 613–618. doi:10.1109/ITNG.2013.94.
URL http://ieeexplore.ieee.org/document/6614374/

[3] S. Dimassi, F. Demoly, H. Belkebir, C. Cruz, K.-Y. Kim, S. Gomes, H. J. Qi, J.-C. André, A knowledge recommendation approach in design for multi-material 4d printing based on semantic similarity vector space model and case-based reasoning, Computers in Industry 145 (2023) 103824. doi:10.1016/j.compind.2022.103824.
URL https://linkinghub.elsevier.com/retrieve/pii/S0166361522002202

[4] R. Abbasi, P. Martinez, R. Ahmad, An ontology model to represent aquaponics 4.0 system's knowledge, Information Processing in Agriculture 9 (2022) 514–532. doi:10.1016/j.inpa.2021.12.001.
URL https://linkinghub.elsevier.com/retrieve/pii/S2214317321000937

[5] A. González-Eras, R. D. Santos, J. Aguilar, A. Lopez, Ontological engineering for the definition of a covid-19 pandemic ontology, Informatics in Medicine Unlocked 28 (2022) 100816. doi:10.1016/j.imu.2021.100816.
URL https://linkinghub.elsevier.com/retrieve/pii/S2352914821002811

[6] S. Grimm, A. Abecker, J. Völker, R. Studer, Ontologies and the Semantic Web, Springer Berlin Heidelberg, 2011, pp. 507–579. doi:10.1007/978-3-540-92913-0_13.
URL http://link.springer.com/10.1007/978-3-540-92913-0_13

[7] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, F. Yergeau, Extensible markup language (xml) 1.0 (fifth edition), W3C Recommendation, available at http://www.w3.org/TR/REC-xml/ (2008).

[8] O. Lassila, R. R. Swick, Resource Description Framework (RDF) Model and Syntax Specification (1999).
URL http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/

[9] D. McGuinness, F. van Harmelen, Owl web ontology language overview, W3c recommendation, World Wide Web Consortium (February 2004).
URL http://www.w3.org/TR/2004/REC-owl-features-20040210/

[10] M. A. Musen, The protégé project: a look back and a look forward, AI Matters 1 (4) (2015) 4–12. doi:10.1145/2757001.2757003.
URL https://doi.org/10.1145/2757001.2757003

[11] M. Lubani, S. A. M. Noah, R. Mahmud, Ontology population: Approaches and design aspects, Journal of Information Science 45 (2019) 502–515. doi:10.1177/0165551518801819.
URL http://journals.sagepub.com/doi/10.1177/0165551518801819

[12] M. A. Hearst, Automatic acquisition of hyponyms from large text corpora, Vol. 2, Association for Computational Linguistics, 1992, p. 539. doi:10.3115/992133.992154.
URL http://portal.acm.org/citation.cfm?doid=992133.992154

[13] A. C. Khadir, H. Aliane, A. Guessoum, Ontology learning: Grand tour and challenges, Computer Science Review 39 (2021) 100339. doi:10.1016/j.cosrev.2020.100339.
URL https://linkinghub.elsevier.com/retrieve/pii/S1574013720304391

[14] Y. Chasseray, A.-M. Barthe-Delanoë, S. Négny, J.-M. L. Lann, Knowledge extraction from textual data and performance evaluation in an unsupervised context,

Information Sciences 629 (2023) 324–343, cited by: 0. doi:10.1016/j.ins.2023.01.150.
URL https://linkinghub.elsevier.com/retrieve/pii/S0020025523001640

[15] P. Sambandam, D. Yuvaraj, P. Padmakumari, S. Swaminathan, Spiking equilibrium convolutional neural network for spatial urban ontology, Neural Processing Letters 55 (2023) 7583 – 7602, cited by: 0. doi:10.1007/s11063-023-11275-4.
URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-85159365623&doi=10.1007%2fs11063-023-11275-4&partnerID=40&md5=1b99773b599c346fb40f2f67f52ab427

[16] J. A. Reyes-Ortiz, Criminal event ontology population and enrichment using patterns recognition from text, International Journal of Pattern Recognition and Artificial Intelligence 33, cited by: 7 (2019). doi:10.1142/S0218001419400147.
URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-85062590101&doi=10.1142%2fS0218001419400147&partnerID=40&md5=6ea79b2de5ea838e152ff915e1275c98

[17] A. Ayadi, A. Samet, F. de Bertrand de Beuvron, C. Zanni-Merk, Ontology population with deep learning-based nlp: a case study on the biomolecular network ontology, Procedia Computer Science 159 (2019) 572–581, sistema NLP de población de ontologías para poblar la Ontología de Red Biomolecular a partir de textos no estructurados. Utiliza deep learning, el cual, según el documento, no es usado comúnmente en población de ontología. Los resultados preliminares son. doi:10.1016/j.procs.2019.09.212.
URL https://linkinghub.elsevier.com/retrieve/pii/S1877050919313961

[18] A. Ayadi, C. Zanni-Merk, F. de Bertrand de Beuvron, J. Thompson, S. Krichen, Bno—an ontology for understanding the transittability of complex biomolecular networks, Journal of Web Semantics 57 (2019) 100495. doi:10.1016/j.websem.2019.01.002.
URL https://linkinghub.elsevier.com/retrieve/pii/S1570826819300022

[19] C. Faria, I. Serra, R. Girardi, A domain-independent process for automatic ontology population from text, Science of Computer Programming 95 (2014) 26–43, special Issue on Systems Development by Means of Semantic Technologies.

doi:10.1016/j.scico.2013.12.005.
URL https://linkinghub.elsevier.com/retrieve/pii/S0167642313003419