

Entre idiomas: Evaluación semántica con sentencias transformers en ingeniería de requisitos para la obtención de requisitos

Contreras Alan, Albelo Brian, Belnava Gabriela
Profesora tutora: Maria Celia Elizalde

Escuela de Informática, Universidad Nacional del Oeste, Argentina
{Alan.E.G.Contreras; brianrafael32; gabrielabelnava;@gmail.com

Resumen. El veloz avance de las IA's (inteligencias artificiales) como herramientas de uso continuo, abre una brecha que permite visualizarlas como una herramienta potencial en el proceso de la ingeniería de requisitos. Desde esta perspectiva se generan incertidumbres, ¿De qué manera puede ser una pieza importante dentro del proceso de ingeniería de requisitos (IR)? ¿Qué idioma comprende de manera más efectiva? Estas preguntas van surgiendo de manera continua. A partir del uso de sentencias Transformers, se pone en propuesta una distinción entre inglés y español, poniendo a prueba la comprensión semántica de las palabras, con frases que nos ayudan a realizar una comparación efectiva, utilizando técnicas tales como el embedding (incrustación de palabras) y fine-tuning (aprendizaje profundo). Con los embeddings y el método de tokenización es posible medir la proximidad de referencias entre palabras y frases, esto nos permite tener una visualización entre los idiomas de cual tiene el mejor porcentaje de similitud, así podemos ver cual presenta menos ambigüedades. Por otro lado, fine-tuning, permite establecer las reglas de entrenamiento con una sintaxis precargada, y con dataset, de requisitos funcionales (RF) y requisitos no funcionales (RNF). El objetivo es obtener, a partir de estas técnicas, una lista de requisitos preliminares válida.

Palabras Clave: Requisitos funcionales, Requisitos no funcionales, Embedding, fine-tuning, tokenización

1 Introducción

El presente artículo se desarrolla en el marco de la propuesta sobre la aplicación de la Inteligencia Artificial en el proceso de requisitos, de la materia Ingeniería de Software II de la carrera Licenciatura en Informática de la Universidad Nacional del Oeste.

El desarrollo de software incluye un área especializada en la ingeniería de requisitos, la cual se enfoca en comprender a fondo las necesidades de los clientes y en planificar adecuadamente los proyectos de software, siendo esta “una propiedad que es capaz de resolver un problema del mundo real” según Sommerville[1]. Según Pressman, entender los requisitos de un problema es “una de las tareas más difíciles que enfrenta el ingeniero de software”[2]. En los últimos años, esta tarea ha sido optimizada mediante diversos algoritmos de inteligencia artificial (IA). Este

estudio tiene como objetivo explorar los distintos enfoques de IA que se aplican al idioma en español e inglés, para corroborar el nivel de entendimiento que logra la IA.

Para poder realizar este estudio, nos enfocamos en una rama de la IA, un algoritmo automático que imita la percepción humana inspirada en nuestro cerebro, siendo la técnica que más se acerca a la forma de pensar de los humanos debido a que usa una arquitectura de redes neuronales y que es el deep learning (aprendizaje profundo).

Dentro de la disciplina del deep learning, utilizamos fine-tuning que permite tomar modelos previamente entrenados con determinados datos, para resolver tareas específicas. Este proceso aprovecha el conocimiento general ya adquirido por el modelo y lo especializa para mejorar su rendimiento en el nuevo contexto. Es especialmente útil porque permite ahorrar tiempo y recursos, evitando la necesidad de entrenar un modelo desde cero, y generalmente resulta en un mejor desempeño debido al aprovechamiento de patrones y características previamente aprendidos.

Las sentencias Transformers basadas en BERT es una tecnología de procesamiento de lenguaje natural y dentro del Fine tuning, utilizaremos las sentencias transformers que son una versión de la red BERT, pero nos enfocamos en el modelo DistilBERT, que este mismo es un modelo destilado BERT en un 40% y conservando 97% de su comprensión y un 60% más rápido[3].

Por el lado de los embedding utilizaremos sentencias transformers, la sentencia `all-MiniLM-L6-v2`, esta sentencia transformers mapea oraciones y párrafos en un espacio vectorial de 384 dimensiones, que se puede utilizar para realizar agrupaciones, como también búsquedas semánticas.

2 Comparación Semántica

El inglés y el español presentan diferencias importantes en su estructura gramatical y en la forma en que expresan conceptos, lo cual impacta en la precisión y efectividad de los modelos de IA:

Inglés: Tiende a ser más directo y tiene una estructura gramatical relativamente sencilla. La flexibilidad del orden de las palabras en inglés y la menor cantidad de conjugaciones verbales reducen la complejidad en la interpretación de frases.

Español: Tiene una estructura gramatical más compleja, con una mayor variedad de conjugaciones verbales y reglas gramaticales más estrictas. La riqueza y diversidad del vocabulario, junto con la variabilidad en la construcción de oraciones.

En este caso utilizamos el método embedding, que nos permite vectorizar frases para compararlas entre sí, utilizando el coseno de similitud. Los embedding son una técnica que convierte el lenguaje humano en vectores matemáticos, el vector representa la palabra o la frase como dato y permite la manipulación matemática de la misma. El modelo de lenguaje que se utilizó son las sentencias-transformers.[4], que es versión modificada de la red BERT, esta misma utiliza redes siamesas y triplete. Las redes siamesas fueron pensadas para el procesamiento de dos o más entradas[5]. El triplete cumpliría la función de categorizar la frase en positiva, negativa,

pérdida[6]. Con la fórmula del coseno de similitud, comparamos la cercanía que hay entre vectores, cuanto más cerca del 1 este las palabras o las frases, significa que tan igual son a nivel semántico. La sentencia pre entrenada que se utiliza para estas pruebas fue all-MiniLM-L6-v2, esta sentencia- transformers mapea oraciones y párrafos en un espacio vectorial de 384 dimensiones, que se puede utilizar para realizar agrupaciones como también búsquedas semánticas. Utilizando el marco metodológico planteado buscamos comprobar a nivel semántico que idioma comprende mejor. Las pruebas se basan en frases relacionadas a los requisitos funcionales y no funcionales, las mismas son extraídas del siguiente dataset: *FR_NFR_Spanish_requirements_classification*. A estas se le fueron cambiando palabras, por sus sinónimos para ver la distancias semánticas entre ellas, los resultado fueron los siguientes:

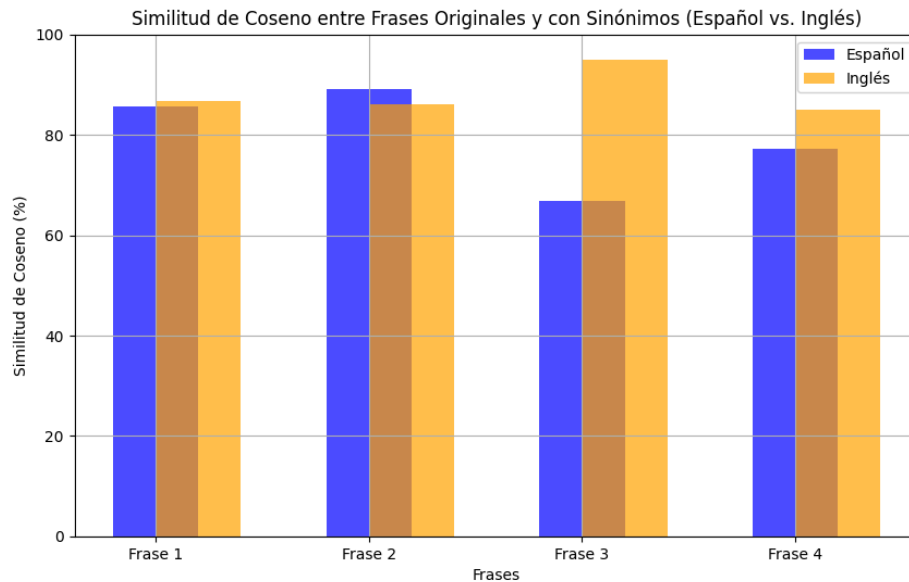


Fig. 1. Similitud de Coseno entre frases originales y con sinónimos (Español vs. Inglés)

Los criterios de comparación fueron usar una frase inicial y a la misma frase se le reemplaza palabras por sinónimos sin perder el significado. Se realizó la tokenización para obtener un valor matemático, ver la distancia semántica que tiene cada una. Y vimos que en el lote de pruebas que presentó mejores resultados con el español.

Frases utilizadas en el análisis:

Frase 1 en Español: "Poder crear un usuario y acceder a través de él a la aplicación"

- Con sinónimo : "Poder generar un usuario y entrar a través de él a la aplicación"

Frase 1 en Inglés: "Be able to create a user and access the application through them"

- Con sinónimo: “Being able to generate a user and enter the application through them”

Frase 2 en español: “Un usuario registrado modifica la imagen de su perfil de usuario, escogiéndola de su ordenador”

- Con sinónimo: “Un usuario validado modifica la imagen de su perfil de usuario, eligiéndola de su ordenador”

Frase 2 en Inglés: “A registered user modifies the image of their user profile, choosing it from their computer”

- Con sinónimo: “A validated user modifies the image of their user profile, choosing it from their computer”

Frase 3 en Español: “Un usuario registrado crea una nueva ruta a través del formulario de creación de rutas”

- Con sinónimo: “Un usuario validado crea una nueva ruta a través del formulario de creación de rutas”

Frase 3 en Inglés: “A registered user creates a new route through the route creation form”

- Con sinónimo: “A validated user creates a new route through the route creation form”

Frase 4 en Español: “Como usuario autenticado quiero crear una fuente de ingreso con un nombre y una descripción”

- Con sinónimo: “Como usuario validado quiero crear una fuente de ingreso con un título y una descripción”

Frase 4 en Inglés: “As an authenticated user I want to create a revenue stream with a name and description”

- Con sinónimo: “As a validated user I want to create an income source with a title and description”

3 Inferencia de requisitos

En el fine-tuning(ajuste fino) un método de entrenamiento que permite adaptar el input y el output, que se utilizó el data set *FR_NFR_Spanish_requirements_classification* [7], para entrenar al modelo, y se usó distilBERT que es una versión destilada del modelo BERT que tiene la particularidad de ser el modelo BERT en un 40% y conservando 97% de su comprensión y un 60% más rápido.

En el proceso de entrenamiento se procede a tokenizar la información, con DistilBertTokenizer[8], modelo el cual fue inicializado con una matriz de incrustación de token de 256.000. El mismo fue entrenado con 100 gigabytes de distintas fuentes. Y con el modelo de DistilBertForSequenceClassification, es el más eficiente para la clasificaciones de texto.

Este conjunto de herramientas fueron los que permitieron obtener una inferencia de requisitos en los cuales distingue entre requisitos funcionales (RN) y requisito no funcionales (RNF). El lote de prueba en español que se utilizó fue el siguiente:

Texto: El sistema permitirá subir a la plataforma, videos, imágenes, texto.

Predicción: Funcional

Respuesta: Correcto

Texto: El Sistema permitirá comentar sobre las publicaciones de otros usuarios.

Predicción: Funcional

Respuesta: Correcto

Texto: El sistema permitirá chatear con otros usuarios e interactuar

Predicción: No funcional

Respuesta: Incorrecto

Texto: El sistema permitirá crear comunidades donde los usuarios publican sobre un tema en específico.

Predicción: No funcional

Respuesta: Correcto

Texto: El sistema proporcionará la funcionalidad de vincular la cuenta del usuario y sus publicaciones a Drive, facilitando la creación de copias de seguridad de manera eficiente.

Predicción: No funcional

Respuesta: Incorrecto

Texto: El sistema permitirá notificar de las nuevas publicaciones de los usuarios que sigas.

Predicción: Funcional

Respuesta: Correcto

Texto: El sistema deberá permitir al usuario iniciar sesión, con un correo electrónico y contraseña.

Predicción: No funcional

Respuesta: Incorrecto

Texto: El sistema garantiza la privacidad por medio de contraseñas y validación por dos pasos.

Predicción: Funcional

Respuesta: Incorrecto

Texto: El sistema contará con un soporte web con un chatbot

Predicción: No funcional

Respuesta: Correcto

Texto: El sistema habilitará una función de filtrado avanzado que permitirá a los usuarios buscar establecimiento de entrega basándose en criterios específicos como ubicación, tipo de cocina, valoraciones de clientes y tiempos de entrega estimados.

Predicción: Funcional

Respuesta: Correcto

Texto: El sistema permitirá que los usuarios puedan leer, escribir y responder comentarios sobre los comercios, así como valorar la utilidad de los mismos.

Predicción: Funcional

Respuesta: Correcto

Texto: El sistema enviará notificaciones en tiempo real sobre el estado del pedido, desde la confirmación hasta la entrega.

Predicción: Funcional

Respuesta: Correcto

Texto: El sistema deberá ofrecer descuentos limitados, por tiempo limitado

Predicción: No funcional

Respuesta: Correcto

Estas fueron las inferencias que realizó en base a un código[9] que se entrena al modelo con un dataset generando un modelo nuevo, solo con los datos expuestos en el dataset. En el código que genera la inferencia[10] entre RF y RNF, mismo toma de punto de partida el modelo generado por dataset. A este punto podemos observar como el modelo entrenado pudo inferir con un

porcentaje de acierto 70%, tomando como criterio si cumplía con dataset de RF y RNF. Con esto pudimos validar una lista de requisitos y poder validarlo con la IA ,gracias a la inferencia que genera.

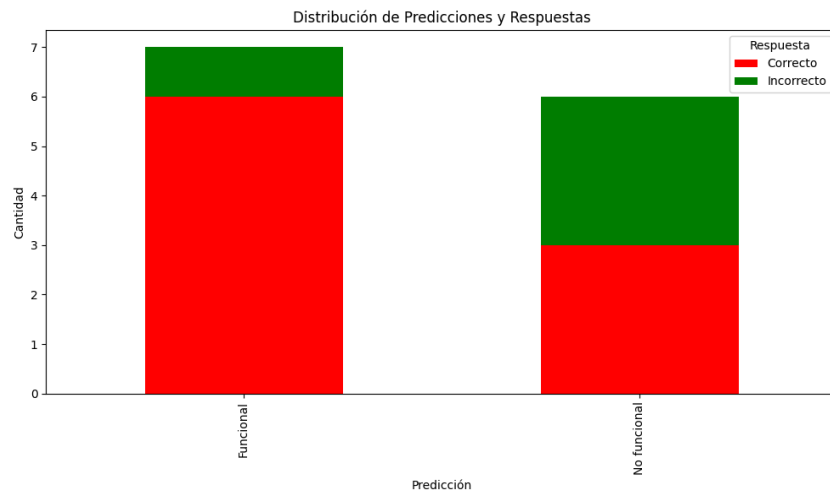


Fig. 2. Distribución de predicción y respuestas

4 Conclusiones

Durante nuestra investigación, alcanzamos una comprensión más profunda al comparar los idiomas inglés y español en cuanto a la precisión de la IA en la identificación de requisitos funcionales y no funcionales. Descubrimos que en español se logra un alto porcentaje de aciertos, específicamente un 70%, en la detección y cumplimiento de estos requisitos. En contraste, el inglés mostró una menor precisión en este aspecto.

Este fenómeno puede explicarse a través del análisis del coseno de similitud, una métrica que nos permite evaluar la proximidad semántica entre palabras y frases en ambos idiomas. En el caso del inglés, observamos que existe una mayor amplitud semántica, lo que significa que las palabras y frases pueden tener una mayor sensibilidad a los cambios, cualquier variación afecta a nivel matemático sus valores vectoriales lo que significa que de una frase inicial haya una distancia mayor a una frase con sinónimos por su escaso léxico. Esta amplitud semántica puede generar ambigüedades y dificultades adicionales en la interpretación precisa de los requisitos, lo que a su vez reduce la tasa de aciertos en comparación con el español.

Gracias a la inferencia de requisitos, podemos generar una lista preliminar de Requisitos Funcionales (RF) y Requisitos No Funcionales (RNF) utilizando herramientas de IA. Aunque obtuvimos conclusiones interesantes, lo más relevante que descubrimos fue la escasez de investigaciones desarrolladas sobre el tema. La información precisa es difícil de encontrar, por lo que sería adecuado continuar trabajando en modelos que permitan obtener mejores resultados en el futuro.

Referencias

1. Ian Sommerville(2011).Ingeniería de Software. 9na Edición
2. Roger S. Pressman(2010).Ingeniería del Software.7ma Edición
3. Victor SANH, Lysandre DÉBUT, Julien CHAUMOND, Thomas WOLF.(01 de marzo de 2020).DistilBERT, a distilled version of BERT: smaller,faster, cheaper and lighter
4. Tomas Arsen.GitHub.<https://github.com/UKPLab/sentence-transformers>
5. Jose Villagran.GitHub.<https://github.com/JoseVillagranE/SiameseNetworks>
6. Nils Reimers and Iryna Gurevych(27 de agosto de 2020).Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks
7. Maria Isabel Limaylla Lunarejo.(15 de julio 2022). HuggingFaceInc.
https://huggingface.co/datasets/MariaIsabel/FR_NFR_Spanish_requirements_classification
8. Nils Reimers.HuggingFaceInc
https://huggingface.co/vocab-transformers/distilbert-tokenizer_256k-MLM_250k
9. Brian Rafael Albelo .GitHub
https://github.com/BrianRafaelAlbelo/EntrenamientoRF_RNF/commit/fc7e35b57c10a3dfc0afa29809fb3cb968912357
10. Brian Rafael Albelo .GitHub
https://github.com/BrianRafaelAlbelo/EntrenamientoRF_RNF/blob/main/InferenciaRF_RNF