

**Aplicación del Diálogo Mayéutico en Modelos de Lenguaje y utilización del Índice de Diversidad de Lande como métrica de la conversación.**

Máster de Metodología en las Ciencias del Comportamiento y de la Salud.

Autor: Álvaro Morales de Vera.

Tutor: Guillermo de Jorge Botana.

31 de mayo de 2024.

## Resumen (Falta un párrafo de introducción sobre los LLM —> generar un párrafo tipico).

El objetivo general de este trabajo es doble.

Por un lado, generar una forma de diálogo mayéutico a través de modelos de lenguaje. Esta forma de diálogo busca extraer conocimiento mediante múltiples rondas de conversación, con el objetivo de encapsular la información contenida en el texto de entrada.

Por el otro, se pretende definir una métrica para poder evaluar la variación de información en cada turno de una conversación. Esta métrica se encargaría de medir si la dirección de la conversación hace que las temáticas de las intervenciones tiendan a aglutinarse en una serie de dimensiones conforme avanza la conversación. Es importante destacar que se pretende definir una métrica útil para todo tipo de conversaciones, no solo para el diálogo mayéutico antes propuesto.

Para explicar el método y las propuestas llevadas a cabo, será hará un repaso de las arquitecturas de modelos de lenguaje basadas en Transformers, las incrustaciones (embeddings) como forma de codificación semántica, métricas propuestas para evaluar distintos aspectos de una conversación y las técnicas de prompting, como forma de interactuar con los modelos de lenguaje, así como que diferencias tienen con el diálogo mayéutico. **(Mencionar que también se harán pruebas con textos expositivos).**

## Introducción

Como se ha mencionado antes, vamos a hacer una descripción de las arquitecturas transformer, que son las que eminentemente sustentan los modelos de lenguaje actuales.

**Modelos *Transformer* codificador y decodificador**

La arquitectura Transformer original consta de dos componentes principales: un codificador y un decodificador, que interactúan mediante mecanismos de atención. El codificador procesa la secuencia de entrada completa para producir representaciones internas de cada elemento de la secuencia. El decodificador, a su vez, genera la secuencia de salida *token* a *token*, atendiendo (mediante atención cruzada) a las representaciones producidas por el codificador para guiar la generación​ (Vaswani, 2017).

Tanto codificador como decodificador están formados por pilas de capas Transformer estándar: cada capa incluye una subcapa de *atención multicabezal* seguida de un perceptrón multi-capa, con conexiones residuales y normalización en cada paso de la capa *transformer* (Vaswani, 2017). Abajo se explican con más profundidad.

***Modelos decodificador***

**Modelos Autorregresivos y teacher forcing**. Habitualmente la misión de los decodificadores es generar una secuencia de palabras a raíz de los estados generados por el codificador y la entrada a la red. No obstante hay topologías que solo utilizan la parte decodificadora. Un tipo muy utilizado de modelos generativos son los modelos autorregresivos. La parte autorregresiva es justo el decodificador. Esta parte, la decodificadora, es la pieza que utilizan la última salida generada por ellos como entrada en la siguiente marca de tiempo (de Jorge Botana, 2023).

Si se parte de una secuencia de palabras , y el modelo continúa generando una secuencia de palabras hasta llegar al final de frase. Las palabras con las que la red ha autocompletado la secuencia son las que tienen mayor probabilidad según se genera la secuencia. Esa es la naturaleza autorregresiva de los decodificadores. Estado posteriores están en función de estados anteriores.

El entrenamiento de esta pieza se hace por aprendizaje forzado. En el aprendizaje forzado, la tarea es determinar la palabra en la siguiente marca de tiempo. La red recibirá como entrada la palabra correcta de la frase y no la que ha emitido la red (de Jorge Botana, 2023).

**Modelos pre-entrenados**. Como se ha dicho, en las arquitecturas *transformer*, aparte de las topologías codificador-decodificador, se pueden utilizar los decodificadores de manera independiente como modelos generativos (Bishop & Bishop, 2024). Los modelos que han adquirido mayor trascendencia son los modelos generativos *pre-entrenados* basados en Transformers (GPT). Algunos ejemplos son los modelos *GPT*-2 (Radford et al., 2019) y *GPT*-3 (Brown et al., 2020)o los modelos *Llama* (Touvron et al., 2023).

El primer trabajo que aborda el *pre-entramiento* para los *transformers* es (Radford & Narasimhan, 2018). Los modelos pre-entrenados nacen de la necesidad de no partir de cero en entrenamientos posteriores. Si tenemos ya un entrenamiento hecho, es decir, unos pesos que parecen eficiente, se pueden utilizar los valores para iniciar una nueva red. Es una forma de transferencia de conocimiento. Esta manera de actuar ya había ganado fuerza en el mundo del aprendizaje profundo a través de los trabajos de (Hinton et al., 2006) . Una de las virtudes de esta forma de entrenamiento es la capacidades de iniciar los pesos en regiones donde la estimación del gradiente es muy ruidosa (Goodfellow et al., 2016).

Actualmente, existen una gran variedad de modelos pre-entrenados que podemos utilizar tal cual nos los presentan (*Vanilla models*), o afinarlos mediante procedimientos de ajustado preciso. Ambas alternativas pueden utilizarse en las grandes plataformas como Azure, AWS, OpenAI.

En posteriores trabajos se observó como modelos como GPT-3, dado su tamaño, se comportan como *few\_shot learners*, incluso *one\_shot learners* (Brown et al., 2020). Esto significa que con unos pocos ejemplos pueden afinar las respuestas notablemente sin necesidad de ajuste preciso. Por ello, la ingeniería de *prompt* es de mucha importancia y en muchos casos evita tener que reajustar los modelos.

(Mirar si BERT es solo codificador)

Respecto a los modelos pre-entrenados que solo tienen la parte codificadora, se pueden emplear algunas alternativas. Con los modelos *decodificador*, también se pueden utilizar modelos *codificador* aislados. La gama de modelos BERT (Devlin et al., 2019) es la más importante entre los modelos codificador. Incluye también a otros modelos aparte del original como DistilBERT (Sanh et al., 2019) que reduce el tamaño del modelo original o RoBERTa (Liu et al., 2019), que mejora su diseño. Estos modelos son bidireccionales. Esto quiere decir que se calculan las puntuaciones de atención contando tanto con los *tokens* anteriores como posteriores al *token* de entrada (Devlin et al., 2019).

El entrenamiento de BERT es similar a los modelos *decodificador* antes mostrados en el entrenamiento supervisado; sin embargo, los autores diseñaron dos tareas de *pre-entrenamiento* diferentes (Devlin et al., 2019):

**Modelo de lenguaje enmascarado**. Para una secuencia de entrada, cada token de esta secuencia tiene un de ser enmascarada o reemplazada. La tarea del modelo es predecir las palabras enmascaradas.

**Predicción de la siguiente frase**. La tarea en este caso es predecir si dos frases, separadas por un token de separación van seguidas o no.

El modelo es *pre-entrenado* en estas dos tareas a la vez.

El objetivo de un modelo solo codificador es aprender representaciones ricas que sean útiles para tareas de comprensión y clasificación (por ejemplo, análisis de sentimientos, preguntas y respuestas extractivas, reconocimiento de entidades) (Devlin et al., 2019). Estas representaciones se conocen como incrustaciones contextuales, porque aglutinan la información toda la frase en un solo vector. A continuación se estudian con más detenimiento.

## Incrustaciones contextuales y Modelos de incrustaciones

En el ámbito del procesamiento de lenguaje natural, las palabras generalmente se representan inicialmente a través un conjunto vocabulario , de longitud discreta. Este está formado por una matriz de vectores *one-hot* y a cada palabra le corresponde una fila de esta matriz. Los vectores *one-hot* tienen un número de dimensiones correspondiente al número de palabras del conjunto vocabulario, donde todas las dimensiones son cero exceptuando el índice de la fila, que toma el valor uno (de Jorge Botana, 2023).

El vocabulario también puede estar formado por fragmentos de palabras, en vez de palabras propiamente dichas. La longitud de estos fragmentos es variable, dependiendo de su frecuencia de aparición (Islam et al., 2022).

Aquellas palabras más repetidas tienden a conservarse enteras (Sennrich et al., 2016), mientras que palabras no aparecidas en el vocabulario se pueden descomponer en fragmentos más pequeños. Cada uno de estos fragmentos se conoce como *tokens* (Bishop & Bishop, 2024) y el proceso de extraer tokens a partir de una cadena de texto se conoce como *tokenización*. Existen varios métodos para lograr esta *tokenización* como *byte pair encoding* ) (Sennrich et al., 2016) o *subword regularization* (Kudo, 2018).

Dado que los vectores *one-hot* solo informan del índice la palabra en el vocabulario, aparte de ser altamente dimensionales. Las incrustaciones de palabras son representaciones de palabras, que cumplen con las siguientes condiciones: por un lado, son representaciones densas, en el sentido de que están formadas por valores reales en todas sus dimensiones. Esto las diferencia de las representaciones como los vectores *one-hot*. Por el otro, son reducidas, ya que permiten codificar fuentes de datos de alta dimensionalidad en un conjunto pequeño de variables latentes sub-simbólicas (Jurafsky & Martin, 2024).

**Incrustaciones *contextuales y problemas de las representación***

Los modelos de lenguaje manejan dos tipos de representaciones para codificar unidades textuales: las primeras se refieren a las que representan las palabras en la matriz de incrustación. Este tipo de representaciones son estáticas y únicas. Son vectores que representan a cada una de las palabras del vocabulario que se maneja. Podemos decir que estas representaciones codifican amalgamadamente toda la información semántica y sintáctica del contexto en que cada palabra a participado (Mikolov et al., 2013; Jurafsky & Martin, 2024). Este tipo de incrustaciones se utilizan generalmente como entrada a los modelos de lenguaje ya que permiten codificar cada *token* de la secuencia como un vector numérico (de Jorge Botana, 2023), con el que se activará la capa de entrada. Estos vectores se generan entrenando se configuran en el propio entrenamiento del modelo (Mikolov et al., 2013). Se ha encontrado que el espacio geométrico de las incrustaciones codifica relaciones semánticas (Mu et al., 2018). Las incrustaciones conservan una distancia cercana entre palabrasrelacionados y más lejana con palabras no relacionados (Mikolov et al., 2013). En los modelos *transformer* también existe matriz de incrustaciones (Vaswani et al., 2017).

Las incrustaciones estáticas, una vez entrenadas en un vocabulario, ya no se modifican. Esto introduce un problema de polisemia, es decir, la incrustación no va a poder representar adecuadamente una palabra si esta tiene varios significados (Peters et al., 2018). Sin embargo, el significado que se dote a un palabra dependerá del contexto (resto de la secuencia) que le rodee (Wang et al., 2020). En este sentido, el modelo biLSTM ELMo (Peters et al., 2018) fue el primer modelo que proporcionó la posibilidad de crear *incrustaciones contextuales*, un tipo de representación contextualizada de palabras. Estas representaciones son sensibles tanto a características sintácticas como semánticas como del contexto. Operativamente, las representaciones contextualizadas son los estados ocultos generados por el modelo en cada marca de tiempo. Merece la pena recordad que un estado oculto está en función de una entrada actual y el estado oculto anterior, que codifica el contexto.

En BERT, además de las formas de pre-entrenamiento comentadas antes, también se menciona el uso de este modelo para extraer representaciones contextuales. Estudian distintas maneras de obtener las representaciones a partir de los estados ocultos de distintas capas del modelo, entre las que se encuentran extraer la incrustación generada por la última y la penúltima capa del modelo, aunque la que mejor precisión alcanzó en una tarea de reconocimiento de entidades fue concatenar la incrustación de las cuatro últimas capas.

Otros razonamientos sobre las incrustaciones contextuales han sido hechos en algunos estudios (Li et al., 2020; Mu, 2018). Sin embargo, hay un problema sobre el que merece la pena relfexionar más, y es que la similitud media entre las incrustaciones para palabras y frases aleatorias se ha observado que es positiva (Ethayarajh, 2019). Gao et al. (2019) demuestran como esta similitud positiva es un problema del propio entrenamiento. Los autores señalan que las actualizaciones de los vectores de palabras poco frecuentes tienden a ser parecidas entre sí. Traducido a la práctica, esto quiere decir que cuando se generan incrustaciones de palabras o de frases, independietemente de la temática, tienen un sesgo a parecerse (Martínez, 2024).

Este sesgo se conoce como anisotropía (Ethayarajh, 2019; Li et al., 2020). Cuanto más alta es la capa, más grande será la anisotropía y más sesgo a parecerse las incrustaciones (Ethayarajh, 2019). Según se ha mencionado antes en BERT, las incrustaciones se obtienen de las últimas capas, por lo que la anisotropía será máxima.

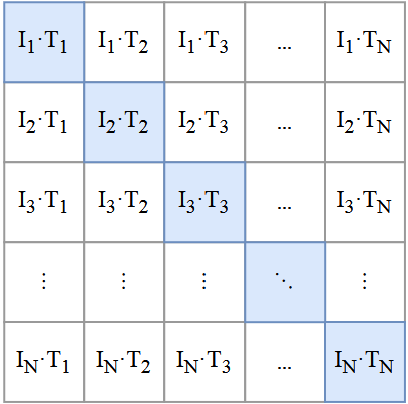
Justo por este tipo de sesgos se crearon modelos entrenados de una forma especial para extraer incrustaciones más eficientes.

***Modelos de incrustaciones***

Los modelos de incrustaciones se distinguen de los modelos generativos antes estudiados en que su propósito es generar una representación que codifique mejor relaciones semánticas que los modelos pre-entrenados (Radford et al., 2021). Con ello se pueden extraer incrustaciones tanto de palabras, como de textos, párrafos, etc.

La forma de entrenar estos modelos se llama aprendizaje contrastivo. En esta forma de aprendizaje se parte de un lote de elementos, que se extraen del conjunto de entrenamiento. De cada elemento del lote se genera otro elemento relacionado semánticamente, como por ejemplo una foto y su descripción o modificar las imágenes generando rotaciones (Chen et al. 2020), a los que se llama pares positivos o relacionados (Neelakantan et al. 2022). Al resto de elementos del lote que no están relacionados se les conoce como elementos negativos o no relacionados.

A partir de este lote se forma una matriz M, donde las filas contienen la similitud de cada elemento del lote con el resto de elementos del lote, incluido el par positivo, que se encuentra en la diagonal de la matriz.



Matriz formada por las similitudes entre los elementos del lote. son los elementos del lote, mientras que son los pares positivos. Como se ve en la diagonal principal se calcula la similitud entre los pares positivos. En las filas de la matriz se ubican las similitudes de cada elemento del lote con elementos negativos, que son los pares positivos del resto de elementos del lote.

El objetivo de esta forma de entrenamiento es poder agrupar aquellos elementos semánticamente relacionados y separar los que no lo estén (Hadsell et al., 2006). Siguiendo a Gao et al. (2021), la fórmula de la función de coste con la que se actualizan los pesos del modelo será:

Nota: Identificar cual es la salida esperada y la emitida del par positivo (y tambien de los negativos) Vamos que es lo que se compara para ver la pérdida.

Donde es el conjunto de índices del lote y es el número de elementos del lote. es , osea el conjunto de todos los índices del lote menos i. ῑ es la temperatura, que es un parámetro que escala los logits para ajustar la “dureza” de la salida de la función softmax. Una temperatura baja hace que incluso pequeñas diferencias de similitud resulten en probabilidades muy concentradas en el máximo (realzando la diferencia entre el positivo y los negativos); una temperatura alta produce una distribución más uniforme donde las diferencias de similitud importan menos​.

El numerador del sumatorio expresa la exponencial de las similitudes de los pares positivos, es decir, los valores de la diagonal de la matriz.

El denominador expresa el sumatorio de las exponenciales de las similitudes de los pares negativos para el índice . Sí la nunca puede ser , quiere decir que son pares negativos.

En suma, que la función de pérdida es un sumatorio de proporciones entre similitudes positivas y negativas para cada índice (llamado ancla en algunos textos). Indica cuán grande es la similitud positiva frente a sus contrapartes negativas.

Cuando la relación entre el par positivo es sustancial (frente a sus contrapartes negativa), la función de pérdida ofrecerá valores pequeños (el error será pequeño) y la actualización de pesos poca. Cuando la relación entre el par positivo no tiene aun la suficiente fuerza, el valor que se desprende de la función de pérdida será alto (el error será mayor). De esa forma, la actualización de los pesos responsables de ese error hará que en próximos ciclos el error descienda y en sucesivas comparaciones de pares positivos se agudice la sensibilidad.

Como los pares son formados por incrustaciones contextuales, se puede decir que el sistema se superespecializará en generar incrustaciones contextuales buenas para hacer juicios de similitud holísticos.

El aprendizaje contrastivo introduce una serie de mejoras en las incrustaciones, que contribuyen a superar los problemas comentados en la sección anterior (Xiao et al., 2023).

**Gao et al. (2019) demuestran como esta similitud positiva es un problema del propio entrenamiento. Los autores señalan que las actualizaciones de los vectores de palabras poco frecuentes tienden a ser parecidas entre sí. Traducido a la práctica, esto quiere decir que cuando se generan incrustaciones de palabras o de frases, independietemente de la temática, tienen un sesgo a parecerse (Martínez, 2024). Este sesgo se conoce como anisotropía (Ethayarajh, 2019; Li et al., 2020). Cuanto más alta es la capa, más grande será la anisotropía y más sesgo a parecerse las incrustaciones (Ethayarajh, 2019).**

En línea con el esparcimiento de las incrustaciones por el espacio, el maximizar la distancia entre elementos no relacionados es semejante a optimizar que las incrustaciones se distribuyan uniformemente por el espacio (Wang & Isolda, 2022).

En términos formales esto significa que el valor esperado de la similitud entre dos palabras aleatorias es cero (Arora et al., 2016a). <Corrección La pérdida contrastiva transforma el espacio de incrustaciones de *anisotrópico*, donde la similitud era siempre positiva, a *isotrópico*, donde de media hay ausencia de relación. Esto significa que las palabras solo se pareceran si son temáticamente similares>. El espacio isotrópico en ideal para codificar similitud. Dado que todas las direcciones están representadas (son equiprobables) y dado el efecto de regularización antes comentado, el entrenamiento contrastivo consigue que el espacio de incrustaciones este más repleto y costreñido, a diferencia de los modelos pre-entrenados (Li et al., 2020).

La función de pérdida constrastiva altera los patrones de contextualización vistos en los modelos pre-entrenados. **En concreto, acercar ejemplos semánticamente similares hace que tengan que las palabras tengan vectores cercanos, fortaleciendo la cohesión de grupos semánticos en el espacio. Esto hace que palabras de la misma frase aumenten su similitud después del aprendizaje contrastivo. La similitud de una misma palabra en distintos contextos también aumenta, por lo que las palabras conservan más su significado frente al contexto** (Xiao et al., 2023). <Corrección: Se ha observado como palabras que pertenencen al mismo contexto tienden a ser más cercanas después del aprendizaje contrastivo(Xiao et al., 2023). Esto implica que las incrustaciones de una misma frase tienden a cohesionarse, lo que síntoma de que las incrustaciones son verdaderamente contextuales.> Por contra las palabras vacías, como preposiciones o conjunciones son entre las más contextualizadas. Dado que no llevan la carga de contenido semántico, sus incrustaciones van a estar marcadas por la de los adejtivos o sustantivos (Xiao et al., 2023).

Por último, generar una distribución uniforme implica reducir la varianza explicada por los primeros componentes principales (Gao et al., 2021, Gao et al., 2019). Esto también afecta a la sensibilidad a la frecuencia, dado que los primeros componentes principales explicaban la varianza debida a la frecuencia de aparición.

Por todas estas mejoras que ofrece entrenar con función de pérdida contrastiva en el espacio de incrustaciones, los modelos de incrustaciones extraen incrustaciones útiles y aptas para tareas que requieran de semántica.

**Métricas del diálogo**

La evaluación de la calidad de una conversación es un problema abierto (Human Evaluation of Conversations is an Open Problem). Por un lado, hay que determinar qué aspectos son más valorables, ya sean cuestiones como la fluidez en la respuesta o el interés generado (See et al., 2019). Una distinción importante se da entre las medidas que evalúan la calidad de cada turno de diálogo, frente aquellas que evalúan la conversación en general (Smith et al., 2022). Siguiendo esta distinción, Giorgi et al. (2023) definen distintas métricas de índole psicológico; por ejemplo, en cada turno se puede medir la correspondencia de estilos lingüísticos, que evalúa el uso de palabras funcionales entre dos interlocutores, mientras que, a nivel de diálogo general, se podría medir la empatía. Otra dificultad añadida es la relación uno a muchos del diálogo, ya que una determinada intervención puede dar lugar a una plétora de respuestas aceptables (Zhao et al., 2017).

Desde el campo del procesamiento del lenguaje natural, existen métricas automáticas no supervisadas como BLEU (Papineni et al., 2002), o como ROUGE, que miden cuestiones como la adecuación o la fidelidad en la generación de una respuesta a partir de las precisiones otorgadas por un modelo de n-gramas.

Si bien se han utilizado ampliamente en campos como traducción automática, se han mostrado ineficientes para evaluar la calidad del diálogo, porque al basarse en apariciones literales de palabras, no pueden tener en cuenta la diversidad de respuestas posibles (Liu et al., 2016); además, el empleo de estas puede dar lugar a una efectividad menor a la media (Dinan et al., 2019).

Otras métricas automáticas están basadas la similitud de incrustaciones estáticas, en lugar de en apariciones literales de palabras: greedy matching compara la similitud entre la incrustación de la respuesta generada con la respuesta realmente dada por uno de los interlocutores, comparando la similitud de cada palabra y haciendo un promedio (Rus & Lintean, 2012). Promedio de incrustaciones (Wieting et al., 2015) realiza una media las incrustaciones de las palabras de cada secuencia y compara las medias. skip-thought utiliza una red neuronal recurrente para generar la incrustación de la secuencia (mirar si es contextual o estatica), mientras que BERTScore utiliza un modelo codificador BERT para generar incrustaciones contextuales, y calcula la media harmónica entre la precisión y la sensibilidad (T. Zhang et al., 2019).

Un gran inconveniente de estas métricas es que requieren de una respuesta humana con que comparar la respuesta generada, siendo que este tipo de respuestas escasean (Howcroft et al., 2020). Otras propuestas, como USR (Mehri & Eskenazi, 2020) evitan la necesidad de evaluación humana. En concreto disciernen si los interlocutores siguen la coherencia general del diálogo, en base a la verosmilitud que arroja el modelo para las intervenciones. En general las métricas basadas en verosimilitud obtienen mayor correlación con respuestas humanas que las métricas automáticas (Liu et al., 2016; Yeh et al., 2021).

Las métricas descritas son capaces de evaluar características generales acerca de la conversación, si bien no son capaces de seguir la evolución temática de ésta. Los diálogos suelen estar orientados en muchos contextos a suplir las intenciones de los interlocutores (Lochbaum, 1998), que pueden darse a entender en distintos grados a lo largo de la conversación (Ruane et al., 2018).

Otras métricas se sirven de los modelos de incrustaciones para calcular métricas relacionadas con la conversación, igual que en este trabajo. convED (Lavi et al., 2021) permite medir la concordancia entre los interlocutores a través de la distancia de edición, que calcula el inverso de la similitud entre las incrustaciones, pero no permite medir si la conversación despeja información respecto de alguna temática (p. ej en el ámbito de un call center, la intención con la que llama el cliente). Taskdiff (Bhaumik et al., 2023) es más interesante en ese sentido, porque es capaz de calcular una distribución sobre un conjunto de intenciones posibles del interlocutor y proponen un método para calcular la similitud entre dos intervenciones en base a la distancia entre las incrustaciones de las distribuciones.

**Prompting y diálogo mayéutico**

La ingeniería de prompts consiste en diseñar instrucciones (prompts) para que los modelos de lenguaje, como GPT, entiendan mejor lo que queremos y respondan de forma más útil (Wei et al., 2022). Métodos como In-Context Learning (Dong et al., 2022) o chain-of-thoughts (Wei et al., 2022) fueron de los primeros métodos en mostrar cómo mejorar la eficacia de los modelos con prompts. En el primero se dota al modelo de pares de entrada-respuestas esperada, para que extraiga inductivamente como tiene que responder; en el segundo se insta al modelo a dividir el proceso de razonamiento en etapas y a describir todo el proceso.

Algunas técnicas de prompting se valen de incrustaciones para condicionar las respuestas del modelo. Básicamente, estas técnicas consisten en añadir un vector en la capa de entrada de los modelos, que después se entrena para ajustarse a una tarea en específico mientras que el resto de pesos del modelo no se entrenan (Liu et al., 2021). Ge et al. (2022) han logrado generar de esta manera prompts con "palabras imaginarias". Estos métodos consiguen fijar conceptos a través de incrustaciones, igual que el diálogo mayéutico, pero tienen el problema de que estos vectores solo sirven para un modelo específico (Wen et al., 2023).

Otros métodos de prompting siguen un enfoque de generación-corrección, en el que se mejora progresivamente un prompt para ajustarlo a una tarea específica, como la resolución de problemas matemáticos. En cada iteración, se solicita al modelo que evalúe tanto la corrección de la respuesta obtenida como la adecuación del prompt para generar buenas respuestas, y a partir de ese razonamiento se le pide que proponga una versión mejorada del mismo.

En esta línea, Pryzant et al. (2023) proponen un método basado en ideas de la optimización por gradiente. Para ello se basan en meta-prompts, que son prompts que orientan al modelo sobre como corregir el prompt que luego se va a utilizar para la tarea. En cada iteración extraen una muestra de prompts haciendo peticiones al modelo y se quedan con los prompts que mejor rendimiento ofrecen. De forma similar, Ye et al. (2023) utilizan también meta-prompts para corregir iterativamente el prompt inicial, siguiendo un proceso de retroceso y mejora.

Este tipo de métodos se distinguen del enfoque dialógico mayéutico, donde el objetivo no es mejorar un prompt específico, sino fomentar una comprensión más profunda de una temática a través del diálogo. Aun así, puede entenderse como una forma de optimización, aunque centrada en la comprensión global más que en la afinación técnica de un prompt.

En conclusión, todas estas técnicas de prompting buscan que los modelos de lenguaje trabajen mejor en tareas específicas. Por ejemplo, Pryzant et al. (2023) se enfocan en mejorar la precisión en tareas de clasificación, y Wen et al. (2023) aplican estas técnicas para generar imágenes editables a partir de texto. El diálogo mayéutico en cambio, pretende poder capturar más conocimiento acerca del texto de entrada. Este conocimiento implica que la información que guardan las incrustaciones es menor, porque la temática del texto se va definiendo conforme avanza la conversación. Esto se reflejará en que la incrustación será mayor en aquellas dimensiones que más relacionadas estén con dicha temática.

## Método

***Diálogo mayéutico***

Este objetivo será llevado a cabo mediante una arquitectura dialógica compuesta por dos modelos de lenguaje, ambos actuando como interlocutores dentro del diálogo mayéutico. Uno de los modelos asume el rol de interrogador socrático, formulando preguntas abiertas con la intención de guiar la exploración conceptual del contenido de un texto dado —como puede ser una conversación, una narrativa o una descripción técnica.

El segundo modelo actúa como respondedor reflexivo, proporcionando respuestas basadas en dicho texto y desarrollando a partir de él un entendimiento progresivo. Tras cada intercambio, este interlocutor puede además sugerir posibles líneas de indagación no exploradas aún, alimentando así la siguiente pregunta del diálogo.

Cada ronda del intercambio funciona como un paso en la elaboración del conocimiento, a través de la construcción de sentido, la clarificación de conceptos y la detección de inconsistencias. Al finalizar el proceso, se espera que la secuencia de preguntas y respuestas permita encapsular una representación más precisa, coherente y profunda del contenido del texto.

***Métrica de diversidad de Lande***

Esta es una métrica originalmente propuesta para medir el número de especies en un determinado ecosistema. Tiene buenas propiedades como ser no paramétrica y estrictamente cóncava. Es útil para calcular la dispersión dentro de un conjunto de ejemplos en relación a la respuesta dada por el modelo. (mide entropía).

***Hipótesis***

La hipótesis que se quiere probar en este trabajo es que las incrustaciones generadas a través del diálogo mayéutico son útiles para encapsular contenido semánticamente relacionado con el texto, en comparación con una línea base. Esto quiere decir que las incrustaciones van a tener mayor similitud con un número reducido de elementos dentro conjunto de prueba. Se espera que la (entropía) métrica de Lande se vaya reduciendo conforme las incrustaciones van acumulando conocimiento.

***Metodología***

Para llevar a cabo este trabajo, se parte de una base de datos que contiene varios textos. El procedimiento utiliza dos modelos principales: uno de lenguaje, encargado de simular el diálogo mayéutico, y otro de incrustaciones, que transforma el contenido de estas conversaciones en representaciones numéricas que permiten su análisis posterior.

El proceso comienza seleccionando un texto de la base de datos. A partir de este texto, el modelo de lenguaje inicia una conversación generando preguntas abiertas que buscan explorar diferentes aspectos del contenido. Luego, responde de manera reflexiva y sugiere nuevas preguntas que permiten profundizar aún más en el tema.

Todo el intercambio (preguntas, respuestas y nuevas sugerencias) se va almacenando en una cadena de texto conocida como "historia de correcciones". Esta historia se va ampliando en cada ronda, acumulando de manera progresiva la información extraída del texto.

Una vez que el diálogo alcanza una determinada profundidad, se utiliza el modelo de incrustaciones para transformar toda la historia de correcciones en un vector numérico. Este vector resume el conocimiento construido a lo largo de la conversación.

Para evaluar la calidad de esta representación, se utiliza la métrica de diversidad de Lande. Esta métrica permite medir cuán concentrado o disperso está el conocimiento encapsulado en el vector. La idea es que, si el diálogo ha sido efectivo, la representación obtenida debería estar más centrada en unos pocos temas muy relacionados con el contenido del texto original, reflejando así una comprensión más profunda.

Además, para poder comparar los resultados, se establece una línea base. Esta línea base corresponde a incrustaciones obtenidas sin aplicar el método de diálogo mayéutico, simplemente a partir del texto inicial. El objetivo es comprobar si las incrustaciones generadas mediante el diálogo muestran una menor dispersión (es decir, una mayor concentración temática) en comparación con las de la línea base, lo que indicaría que el proceso ha logrado encapsular mejor el conocimiento del texto.

**Para anexos**

Se parte de una base de datos de textos . Además, se presupone que se tiene acceso a la API de un modelo base de tipo decodificador , que va a ser el encargado de llevar a cabo la conversación, y de un modelo de incrustaciones para generar incrustaciones.

Se va a utilizar el mismo modelo para simular el diálogo y se utilizan dos *prompts*, uno para la generación de preguntas y otro para guiar en la corrección y respuestas . En cada fase del diálogo mayéutico, se llama a con el prompt de generación para generar preguntas sobre el texto:

|  |  |
| --- | --- |
|  | (11) |

Donde es un hiper-parámetro ajustable. Posteriormente, se responden a las preguntas y se propone otra pregunta más sugerente para extraer información del texto. Esto se resume en la historia de correcciones , que sirve como contexto:

Donde representa la corrección del modelo a las preguntas generadas en la fase anterior y es la sugerencia de pregunta. Para generar se llama al modelo base condicionado a las preguntas generadas:

es la el texto del cual se quiere extraer información. Dado que contiene la información base relacionada con el texto en cada turno de conversación, se utiliza para generar las incrustaciones.

son un conjunto de incrustaciones , con los que comparar , dependientes de la tarea.

es una métrica de comparación:

|  |  |
| --- | --- |
|  | (12) |
|  | (13) |
|  | (14) |

El propósito de la función *softmax* es generar un vector de probabilidades para los vectores de ; indica cual es el vector etiqueta más probable en función de la similitud con el contexto del texto.

es el índice de diversidad de Lande (Lande, 1996; Lande & Shannon, 1996). Este está basado en la métrica de concentración de Simpson:

|  |  |
| --- | --- |
|  | (15) |

La medida de diversidad es . Si se estima con frecuencias estimadas , el valor esperado de la diversidad es:

|  |  |
| --- | --- |
|  | (16) |

El estimador insesgado, , sería:

|  |  |
| --- | --- |
| = | (17) |

En la figura 3 se puede observar una representación de la metodología propuesta.

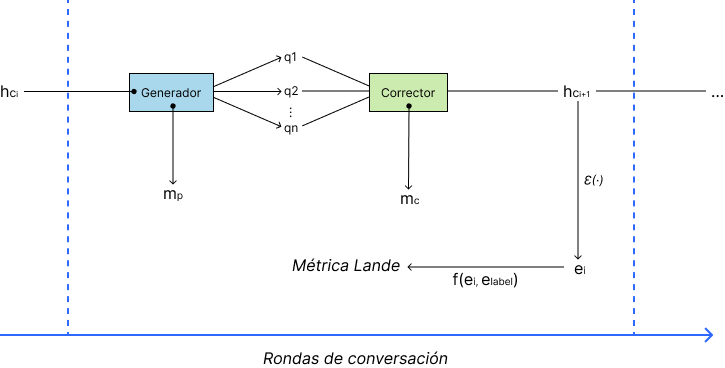


Figura 3. En esta imagen se observa el diálogo producido en una ronda de conversación. Como se ve, se trata de un esquema recursivo, donde se espera que la salida del corrector ayude a afinar la generación de la siguiente pregunta.

***Descripción de Datasets.***

Para probar la metodología y las hipótesis presentadas, se utilizaron dos bases de datos:

**Roleplay**. Esta base de datos contiene descripciones de roles de distintos personajes y está formada por más de diez mil entradas. Cada fila contiene el nombre del personaje, una descripción de este y una conversación entre un asistente y un usuario, en la que el usuario hace preguntas y el asistente las responde; el asistente hace un juego de rol intentando ajustarse a un personaje real, ya sea humano o no humano, o ficticio y utiliza una forma distintiva de responder, en relación con el uso del lenguaje y los rasgos de su personalidad, etc.

**Generics KB**. Esta base de datos contiene frases genéricas, que se utilizarán para comparar con . En cada turno de diálogo mayéutico se obtendrá una frase aleatoria.

## Resultados

En todos los experimentos llevados a cabo, se utilizó como modelo base la última versión actualizada de *gpt-4-turbo* (Achiam et al., 2023).

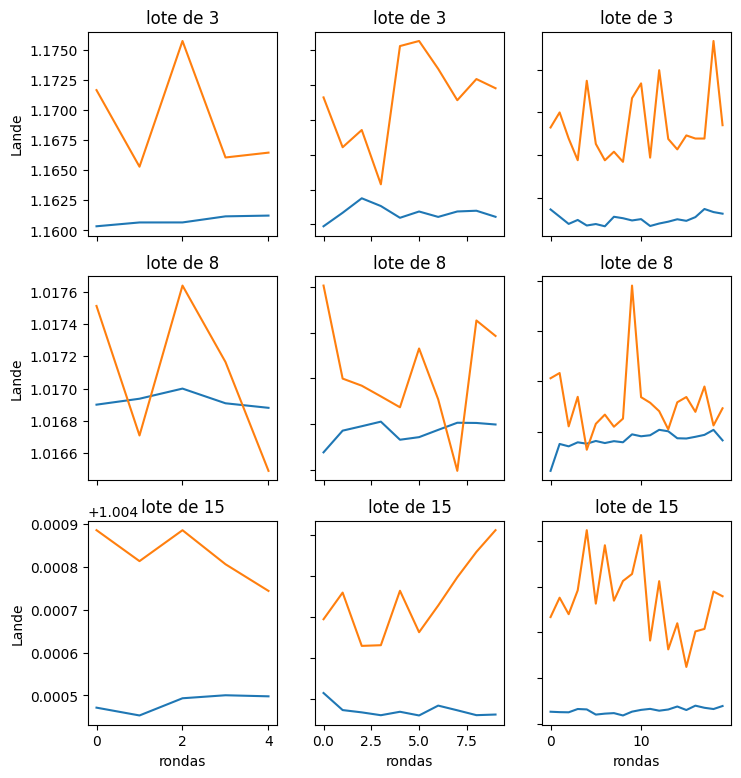
Como codificador el modelo *text-embedding-ada-002*. La descripción y costos de ambos modelos aparece en la API oficial de OpenAI (<https://platform.openai.com/docs/models/>).

***Estudio de tendencia***

La tarea realizada, consiste en escoger un conjunto de roles, utilizando uno como objetivo (*label*) y observar si mediante el diálogo mayéutico se puede reducir la métrica de diversidad, lo que sería síntoma de que el contexto generado por el corrector se parece al rol objetivo. El corrector tiene acceso a la conversación objetivo, donde responde a las preguntas del generador en base a las repuestas del asistente. Formalizándolo en los términos ante mencionados, se escoge un lote , de forma aleatoria. Las incrustaciones se generan a partir de . La corrección se produce condicionada .

Para esto se probó con varios tamaños de lote: tres, ocho y quince roles diferentes para comparar. Además, se tomó el número de rondas de diálogo como un hiper-parámetro, probando con valores de cinco, diez y veinte turnos de conversación para ver los efectos en la optimización. Se generaron cuatro preguntas distintas para cada fase de generación, dejándose constante este parámetro en todos los casos. Los resultados se pueden observar en la figura 4.

*Figura 4.*



*Nota*: en esta figura se aprecian ejemplos de la métrica de Lande para las combinaciones de rondas de conversación y número de roles. En todos los casos los ejemplos utilizados se escogieron aleatoriamente.

Como se observa en esta figura, la métrica de diversidad basada en la corrección está en términos generales por debajo de la métrica calculada para la línea base. Además, exhibe un comportamiento mucho más estable, en vez de resultado erráticos como la línea base.

La mayor diferencia alcanzada fue para la condición de cinco rondas y 15 roles para comparación. En otras condiciones como en lotes de 8 roles para conversaciones de longitud 5 o 10, los niveles para la conversación y la línea base llegan a mezclarse, pero esto puede ser efecto del azar.

Se observan dos tendencias en la evolución de la métrica: por un lado, tiende a estabilizarse; esto se puede ver sobre todo para los casos de 15 roles, lo que puede ser indicativo del efecto de la cantidad de roles sobre la métrica. A mayor cantidad de roles, más elementos van a tener probabilidades menores, lo que lleva a que sea más estable. Por otro lado, hay una tendencia a descender, en los casos para tres roles y 8 turnos de conversación y para 15 roles y ocho turnos de conversación, mientras que para conversaciones más largas ven alterada esta tendencia.

Se utiliza en test de tendencia de Mann-Kendall (Mann, 1945) para evaluar la presencia de tendencia para cada una de las condiciones, solo para el diálogo mayéutico. Se trata de un test no paramétrico basado en la correlación de rango entre los valores observados (Kendall & Gibbons, 1990). Antes de realizar el test propiamente dicho, se tuvo en cuenta el efecto de autocorrelación, ya que esta puede llevar a una infraestimación de la varianza, lo que aumenta la aparición de errores Tipo I (Hamed & Ramachandra Rao, 1998). En el apéndice pueden observarse varios gráficos con la función *afc()* del lenguaje de programación R, que muestran la autocorrelación para varias condiciones; en general los valores se encuentran dentro de los rangos aceptables para todas las condiciones.

En la Tabla 1 pueden observarse los resultados para cada una de las condiciones.

*Tabla 1.*

|  |  |  |  |
| --- | --- | --- | --- |
| N.º Roles | Rondas de  conversación | Estadístico | Nivel  crítico |
| 3 | 5 | 1 | 0.027486 |
| 3 | 10 | 0.111 | 0.72051 |
| 3 | 20 | 0.179 | 0.28432 |
| 8 | 5 | -0.2 | 0.8065 |
| 8 | 10 | 0.378 | 0.15241 |
| 8  15  15  15 | 20  5  10  20 | 0.579  0.6  -0.289  0.453 | 0.00040555  0.22067  0.28313  0.0058198 |

*Nota*: Esta table muestra los valores de tau, que es la corrección de la correlación de rango para las muestras.

La hipótesis nula de este test es que los datos son independientes y están aleatoriamente ordenados, es decir, ausencia de tendencia (Hamed & Ramachandra Rao, 1998). Se utiliza un contraste bilateral. La prueba solo es significativa para las condiciones de 8 roles y 20 rondas de conversación, y 15 roles y 20 rondas de conversación, para un nivel de significación .

## Discusión

En la hipótesis se afirmaba que las incrustaciones generar podía servir para tareas de clasificación. En la exposición de los resultados se ha observado como la métrica para la conversación entre usuario-agente era menor de forma consistente que la línea base, lo cual es indicativo de que diálogo mayéutico genera incrustaciones semánticamente más relacionados con unos pocos elementos del conjunto de roles de prueba. Esto confirma la utilidad de la métrica de diversidad para la evaluar discernir un conjunto de elementos temáticos relacionados con el diálogo agente-usuario.

En cambio, en el estudio de la tendencia, se ha observado cómo, salvo en dos de las condiciones, no se ha rechazado la hipótesis nula de ausencia de tendencia. Es interesante observar que, en las condiciones significativas, el número de rondas de conversación era 20, el máximo. El motivo quizá pueda deberse a un efecto de sobre-aprendizaje tal y como se comenta en Pryzant et al. (2023) respecto de la optimización de *prompts*, y que a partir de cierto punto aumentar el número de rondas de conversación puede ser contraproducente. Sería interesante investigar como evolucionaría la métrica frente a otro tipo de descripciones y en otros contextos (p. ej. Predecir el tono comunicativo, la presencia de mensajes tóxicos, etc.)

La ausencia de tendencia no implica que el modelo no capture información de la conversación usuario-agente, como se demuestra en las representaciones generadas por BERT a raíz de las incrustaciones.

(Incluir que es una exploración que está empezando, que los resultado no son del todo concluyentes).

Un *Call Center* se encarga de gestionar llamadas de clientes. Poder medir de forma efectiva la actuación del equipo es clave (Xiqiao et al., 2022). Para ello se han propuesto varios

KPI *(key performance indicator*); algunos ejemplos son, (Xiqiao et al., 2022):

-Tiempo de gestión media: este KPI mide el tiempo medio que requiere cada llamada en realizarse, así como el tiempo dedicado a tareas administrativas para cada llamada.

-Calificación del servicio de atención al cliente: es una medición de la satisfacción de los clientes con el servicio recibido.

Xiqiao et al. (2022) matiza que los KPI suelen estar formulados de forma binaria, calculando la probabilidad para uno u otro suceso. El problema es que estimar esta probabilidad puede ser difícil de estimar cuando el número de llamadas es bajo. Para resolver este problema, se han propuesto varias soluciones. Valle et al. (2012) emplean un clasificador bayesiano para predecir los mejores trabajadores de una plantilla en base a variables como el número de horas conectado, el número de contactos generados de forma efectiva. Xiqiao et al. (2022) estiman un modelo bayesiano jerárquico, que les permiten comparar la eficiencia de distintos trabajadores.

La métrica de diversidad propuesta también podría ser útil como KPI y para discernir entre los trabajadores que gestionan mejor las llamadas. Para ello, se podría definir un conjunto de respuestas posibles del sujeto, como respuestas de gratitud con el servicio, enfado, etc. El KPI sería la cantidad de información despejada por la métrica en un número de interacciones fijo. Aquellos trabajadores que despejen más información serán los que mejor actuación habrán realizado.

## Bibliografía

Achiam, O. J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., … Zoph, B. (2023, marzo 15). *GPT-4 Technical Report*. https://www.semanticscholar.org/paper/GPT-4-Technical-Report-Achiam-Adler/163b4d6a79a5b19af88b8585456363340d9efd04

Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2006). Greedy Layer-Wise Training of Deep Networks. *Advances in Neural Information Processing Systems*, *19*. https://papers.nips.cc/paper\_files/paper/2006/hash/5da713a690c067105aeb2fae32403405-Abstract.html

Bhaumik, A., Venkateswaran, P., Rizk, Y., & Isahagian, V. (2023). TaskDiff: A Similarity Metric for Task-Oriented Conversations. En H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (pp. 16234-16240). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.1009

Bishop, C. M., & Bishop, H. (2024). The Deep Learning Revolution. En *Deep Learning*. Springer. https://doi.org/10.1007/978-3-031-45468-4\_1

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, *33*, 1877-1901. https://papers.nips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

Budanitsky, A., & Hirst, G. (2006). Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Computational Linguistics*, *32*(1), 13-47. https://doi.org/10.1162/coli.2006.32.1.13

Chang, K., Xu, S., Wang, C., Luo, Y., Xiao, T., & Zhu, J. (2024). *Efficient Prompting Methods for Large Language Models: A Survey* (arXiv:2404.01077). arXiv. http://arxiv.org/abs/2404.01077

Chang, W.-C., Yu, F. X., Chang, Y.-W., Yang, Y., & Kumar, S. (2019, septiembre 25). *Pre-training Tasks for Embedding-based Large-scale Retrieval*. International Conference on Learning Representations. https://openreview.net/forum?id=rkg-mA4FDr

Chidananda Gowda, K., & Krishna, G. (1978). Agglomerative clustering using the concept of mutual nearest neighbourhood. *Pattern Recognition*, *10*(2), 105-112. https://doi.org/10.1016/0031-3203(78)90018-3

Dai, A. M., & Le, Q. V. (2015). Semi-supervised Sequence Learning. *Advances in Neural Information Processing Systems*, *28*. https://papers.nips.cc/paper\_files/paper/2015/hash/7137debd45ae4d0ab9aa953017286b20-Abstract.html

Dalvi, F., Khan, A. R., Alam, F., Durrani, N., Xu, J., & Sajjad, H. (2022). *Discovering Latent Concepts Learned in BERT*. https://doi.org/10.48550/ARXIV.2205.07237

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* (arXiv:1810.04805). arXiv. http://arxiv.org/abs/1810.04805

Dinan, E., Humeau, S., Chintagunta, B., & Weston, J. (2019). Build it Break it Fix it for Dialogue Safety: Robustness from Adversarial Human Attack. *Conference on Empirical Methods in Natural Language Processing*. https://www.semanticscholar.org/paper/Build-it-Break-it-Fix-it-for-Dialogue-Safety%3A-from-Dinan-Humeau/b91c4edd30b63cd1cb1b86cbeefb33a461535e09

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., & Sui, Z. (2022). *A Survey for In-context Learning*. https://doi.org/10.48550/arXiv.2301.00234

Elman, J. (1993). Distributed Representations, Simple Recurrent Networks, And Grammatical Structure. *Machine Learning*, *7*. https://doi.org/10.1007/BF00114844

Ethayarajh, K. (2019). How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 55-65. https://doi.org/10.18653/v1/D19-1006

Gao, T., Yao, X., & Chen, D. (2021). SimCSE: Simple Contrastive Learning of Sentence Embeddings. *Conference on Empirical Methods in Natural Language Processing*. https://www.semanticscholar.org/paper/SimCSE%3A-Simple-Contrastive-Learning-of-Sentence-Gao-Yao/c26759e6c701201af2f62f7ee4eb68742b5bf085

Ge, T., Hu, J., Dong, L., Mao, S., Xia, Y., Wang, X., Chen, S., Wei, F., & Chen, S.-Q. (2022, diciembre 1). *Extensible Prompts for Language Models on Zero-shot Language Style Customization*. Neural Information Processing Systems. https://www.semanticscholar.org/paper/Extensible-Prompts-for-Language-Models-on-Zero-shot-Ge-Hu/d6f5fdf345b9cea58ecee3753e93f6c110831286

Giorgi, S., Havaldar, S., Ahmed, F., Akhtar, Z., Vaidya, S., Pan, G., Ungar, L. H., Schwartz, H. A., & Sedoc, J. (2023). *Psychological Metrics for Dialog System Evaluation*.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Hadsell, R., Chopra, S., & LeCun, Y. (2006). Dimensionality Reduction by Learning an Invariant Mapping. *Computer Vision and Pattern Recognition*. https://www.semanticscholar.org/paper/Dimensionality-Reduction-by-Learning-an-Invariant-Hadsell-Chopra/46f30e94dd3d5902141c5fbe58d0bc9189545c76

Hamed, K. H., & Ramachandra Rao, A. (1998). A modified Mann-Kendall trend test for autocorrelated data. *Journal of Hydrology*, *204*(1), 182-196. https://doi.org/10.1016/S0022-1694(97)00125-X

Hewitt, J., & Manning, C. D. (2019). A Structural Probe for Finding Syntax in Word Representations. En J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4129-4138). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1419

Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*(7), 1527-1554. https://doi.org/10.1162/neco.2006.18.7.1527

Hinton, G. E., & Roweis, S. (2002). Stochastic Neighbor Embedding. *Advances in Neural Information Processing Systems*, *15*. https://papers.nips.cc/paper\_files/paper/2002/hash/6150ccc6069bea6b5716254057a194ef-Abstract.html

Howcroft, D. M., Belz, A., Clinciu, M., Gkatzia, D., Hasan, S. A., Mahamood, S., Mille, S., Miltenburg, E. van, Santhanam, S., & Rieser, V. (2020). Twenty Years of Confusion in Human Evaluation: NLG Needs Evaluation Sheets and Standardised Definitions. *International Conference on Natural Language Generation*. https://www.semanticscholar.org/paper/Twenty-Years-of-Confusion-in-Human-Evaluation%3A-NLG-Howcroft-Belz/1d7a2d2f5671f85e9013e1c8e527c663ada12320

Izacard, G., Caron, M., Hosseini, L., Riedel, S., Bojanowski, P., Joulin, A., & Grave, E. (2022). *Unsupervised Dense Information Retrieval with Contrastive Learning* (arXiv:2112.09118). arXiv. https://doi.org/10.48550/arXiv.2112.09118

Jurafsky, D., & Martin, J. H. (2024). *Speech and Language Processing (3rd ed. Draft)*. https://web.stanford.edu/~jurafsky/slp3/

Kendall, M., & Gibbons, J. D. (1990). *Rank Correlation Methods* (5.a ed.). A Charles Griffin Title.

Kudo, T. (2018). *Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates* (arXiv:1804.10959). arXiv. http://arxiv.org/abs/1804.10959

Lande, R. (1996). Statistics and Partitioning of Species Diversity, and Similarity among Multiple Communities. *Oikos*, *76*(1), 5-13. https://doi.org/10.2307/3545743

Lande, R., & Shannon, S. (1996). The Role of Genetic Variation in Adaptation and Population Persistence in a Changing Environment. *Evolution*, *50*(1), 434. https://doi.org/10.2307/2410812

Lavi, O., Rabinovich, E., Shlomov, S., Boaz, D., Ronen, I., & Anaby-Tavor, A. (2021). *We’ve had this conversation before: A Novel Approach to Measuring Dialog Similarity* (arXiv:2110.05780). arXiv. https://doi.org/10.48550/arXiv.2110.05780

Lee, J., Dai, Z., Ren, X., Chen, B., Cer, D., Cole, J. R., Hui, K., Boratko, M., Kapadia, R., Ding, W., Luan, Y., Duddu, S. M. K., Abrego, G. H., Shi, W., Gupta, N., Kusupati, A., Jain, P., Jonnalagadda, S. R., Chang, M.-W., & Naim, I. (2024). *Gecko: Versatile Text Embeddings Distilled from Large Language Models*. https://doi.org/10.48550/ARXIV.2403.20327

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. En D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 7871-7880). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.703

Li, D., Chen, Z., Cho, E., Hao, J., Liu, X., Xing, F., Guo, C., & Liu, Y. (2022). Overcoming Catastrophic Forgetting During Domain Adaptation of Seq2seq Language Generation. *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 5441-5454. https://doi.org/10.18653/v1/2022.naacl-main.398

Li, Y., Dong, B., Lin, C., & Guerin, F. (2023). *Compressing Context to Enhance Inference Efficiency of Large Language Models*. https://doi.org/10.48550/ARXIV.2310.06201

Lin, C.-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. *Text Summarization Branches Out*, 74-81. https://aclanthology.org/W04-1013

Liu, C.-W., Lowe, R., Serban, I., Noseworthy, M., Charlin, L., & Pineau, J. (2016). *How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation*.

Liu, X., Ji, K., Fu, Y., Du, Z., Yang, Z., & Tang, J. (2021). P-Tuning v2: Prompt Tuning Can Be Comparable to Fine-tuning Universally Across Scales and Tasks. *ArXiv*. https://www.semanticscholar.org/paper/P-Tuning-v2%3A-Prompt-Tuning-Can-Be-Comparable-to-and-Liu-Ji/f3a332ff1b73acda482e5d83696b2c701f487819

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv*. https://www.semanticscholar.org/paper/RoBERTa%3A-A-Robustly-Optimized-BERT-Pretraining-Liu-Ott/077f8329a7b6fa3b7c877a57b81eb6c18b5f87de

Lochbaum, K. E. (1998). A Collaborative Planning Model of Intentional Structure. *Comput. Linguistics*. https://www.semanticscholar.org/paper/A-Collaborative-Planning-Model-of-Intentional-Lochbaum/15067b5cd95d11efe9c6e06a8ba3fb544a6697cc

Maaten, L. van der, & Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, *9*(86), 2579-2605.

Mann, H. B. (1945). Nonparametric Tests Against Trend. *Econometrica*, *13*(3), 245-259. https://doi.org/10.2307/1907187

Mehri, S., & Eskenazi, M. (2020a). Unsupervised Evaluation of Interactive Dialog with DialoGPT. En O. Pietquin, S. Muresan, V. Chen, C. Kennington, D. Vandyke, N. Dethlefs, K. Inoue, E. Ekstedt, & S. Ultes (Eds.), *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue* (pp. 225-235). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.sigdial-1.28

Mehri, S., & Eskenazi, M. (2020b). USR: An Unsupervised and Reference Free Evaluation Metric for Dialog Generation. En D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 681-707). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.64

Mikolov, T., Chen, K., Corrado, G. s, & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *Proceedings of Workshop at ICLR*, *2013*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). *Distributed Representations of Words and Phrases and their Compositionality* (arXiv:1310.4546). arXiv. http://arxiv.org/abs/1310.4546

Mofijul Islam, M., Aguilar, G., Ponnusamy, P., Solomon Mathialagan, C., Ma, C., & Guo, C. (2022). A Vocabulary-Free Multilingual Neural Tokenizer for End-to-End Task Learning. *Proceedings of the 7th Workshop on Representation Learning for NLP*, 91-99. https://doi.org/10.18653/v1/2022.repl4nlp-1.10

Neelakantan, A., Xu, T., Puri, R., Radford, A., Han, J. M., Tworek, J., Yuan, Q., Tezak, N., Kim, J. W., Hallacy, C., Heidecke, J., Shyam, P., Power, B., Nekoul, T. E., Sastry, G., Krueger, G., Schnurr, D., Such, F., Hsu, K., … Weng, L. (2022). Text and Code Embeddings by Contrastive Pre-Training. *ArXiv*. https://www.semanticscholar.org/paper/Text-and-Code-Embeddings-by-Contrastive-Neelakantan-Xu/6d7d4fca9840504f630e9bea6acaa07322a6e889

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A Method for Automatic Evaluation of Machine Translation. *Annual Meeting of the Association for Computational Linguistics*. https://www.semanticscholar.org/paper/Bleu%3A-a-Method-for-Automatic-Evaluation-of-Machine-Papineni-Roukos/d7da009f457917aa381619facfa5ffae9329a6e9

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. En M. Walker, H. Ji, & A. Stent (Eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)* (pp. 2227-2237). Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-1202

Pryzant, R., Iter, D., Li, J., Lee, Y. T., Zhu, C., & Zeng, M. (2023). *Automatic Prompt Optimization with «Gradient Descent» and Beam Search* (arXiv:2305.03495). arXiv. http://arxiv.org/abs/2305.03495

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021, febrero 26). *Learning Transferable Visual Models From Natural Language Supervision*. International Conference on Machine Learning. https://www.semanticscholar.org/paper/Learning-Transferable-Visual-Models-From-Natural-Radford-Kim/6f870f7f02a8c59c3e23f407f3ef00dd1dcf8fc4

Radford, A., & Narasimhan, K. (2018). *Improving Language Understanding by Generative Pre-Training*. https://www.semanticscholar.org/paper/Improving-Language-Understanding-by-Generative-Radford-Narasimhan/cd18800a0fe0b668a1cc19f2ec95b5003d0a5035

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). *Language Models are Unsupervised Multitask Learners*. https://www.semanticscholar.org/paper/Language-Models-are-Unsupervised-Multitask-Learners-Radford-Wu/9405cc0d6169988371b2755e573cc28650d14dfe

Robertson, S., & Walker, S. (1994). *Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval*. 232-241. https://doi.org/10.1007/978-1-4471-2099-5\_24

Ruane, E., Faure, T., Smith, R., Bean, D., Carson-Berndsen, J., & Ventresque, A. (2018). BoTest: A Framework to Test the Quality of Conversational Agents Using Divergent Input Examples. *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*. https://www.semanticscholar.org/paper/BoTest%3A-a-Framework-to-Test-the-Quality-of-Agents-Ruane-Faure/fcfbf81ea73525e7a8d6b918c5dc4e6616de6ff5

Rus, V., & Lintean, M. (2012). *A comparison of greedy and optimal assessment of natural language student input using word-to-word similarity metrics*. 157-162. https://doi.org/10.1007/978-3-642-30950-2\_116

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *ArXiv*. https://www.semanticscholar.org/paper/DistilBERT%2C-a-distilled-version-of-BERT%3A-smaller%2C-Sanh-Debut/a54b56af24bb4873ed0163b77df63b92bd018ddc

See, A., Roller, S., Kiela, D., & Weston, J. (2019). What makes a good conversation? How controllable attributes affect human judgments. *North American Chapter of the Association for Computational Linguistics*. https://www.semanticscholar.org/paper/What-makes-a-good-conversation-How-controllable-See-Roller/8059b85332572e60c8a1daa0ccb8ddc008513f00

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural Machine Translation of Rare Words with Subword Units. En K. Erk & N. A. Smith (Eds.), *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1715-1725). Association for Computational Linguistics. https://doi.org/10.18653/v1/P16-1162

Shannon, C. E. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, *27*, 379-423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

Smith, E. M., Hsu, O., Qian, R., Roller, S., Boureau, Y.-L., & Weston, J. (2022). Human Evaluation of Conversations is an Open Problem: Comparing the sensitivity of various methods for evaluating dialogue agents. *ArXiv*. https://www.semanticscholar.org/paper/Human-Evaluation-of-Conversations-is-an-Open-the-of-Smith-Hsu/94f02394a8f019d7ece7eb9612e96253ba97f30c

Thongtan, T., & Phienthrakul, T. (2019). Sentiment Classification Using Document Embeddings Trained with Cosine Similarity. En F. Alva-Manchego, E. Choi, & D. Khashabi (Eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop* (pp. 407-414). Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-2057

Touvron, H., Martin, L., Stone, K. R., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., … Scialom, T. (2023). Llama 2: Open Foundation and Fine-Tuned Chat Models. *ArXiv*. https://www.semanticscholar.org/paper/Llama-2%3A-Open-Foundation-and-Fine-Tuned-Chat-Models-Touvron-Martin/104b0bb1da562d53cbda87aec79ef6a2827d191a

Valle, M., Varas, S., & Ruz, G. (2012). Job performance prediction in a call center using a naive Bayes classifier. *Expert Systems with Applications*, *39*, 9939-9945. https://doi.org/10.1016/j.eswa.2011.11.126

Wang, X., Kondratyuk, D., Kitani, K. M., Movshovitz-Attias, Y., & Eban, E. (2020). Multiple Networks are More Efficient than One: Fast and Accurate Models via Ensembles and Cascades. *ArXiv*. https://www.semanticscholar.org/paper/Multiple-Networks-are-More-Efficient-than-One%3A-Fast-Wang-Kondratyuk/3b8fcbb917969755e19f4ea1cb136567fd23a918

Wang, Y., Hou, Y., Che, W., & Liu, T. (2020). From static to dynamic word representations: A survey. *International Journal of Machine Learning and Cybernetics*, *11*(7), 1611-1630. https://doi.org/10.1007/s13042-020-01069-8

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Xia, F., Le, Q., & Zhou, D. (2022). Chain of Thought Prompting Elicits Reasoning in Large Language Models. *ArXiv*. https://www.semanticscholar.org/paper/Chain-of-Thought-Prompting-Elicits-Reasoning-in-Wei-Wang/1b6e810ce0afd0dd093f789d2b2742d047e316d5

Wen, Y., Jain, N., Kirchenbauer, J., Goldblum, M., Geiping, J., & Goldstein, T. (2023). *Hard Prompts Made Easy: Gradient-Based Discrete Optimization for Prompt Tuning and Discovery* (arXiv:2302.03668). arXiv. http://arxiv.org/abs/2302.03668

Wieting, J., Bansal, M., Gimpel, K., & Livescu, K. (2015). *Towards Universal Paraphrastic Sentence Embeddings*.

Xiqiao, L., Guohui, W., & Xiaohui, Y. (2022). A Robust and Intelligently Adaptive KPI Evaluation Method for Customer Call Center. *2022 7th International Conference on Power and Renewable Energy (ICPRE)*, 387-391. https://doi.org/10.1109/ICPRE55555.2022.9960538

Xu, J., Ju, D., Li, M., Boureau, Y.-L., Weston, J., & Dinan, E. (2020). Recipes for Safety in Open-domain Chatbots. *ArXiv*. https://www.semanticscholar.org/paper/Recipes-for-Safety-in-Open-domain-Chatbots-Xu-Ju/4fa24cc5b17e8ff1eb5a01fd37a9d267a57ac563

Ye, Q., Axmed, M., Pryzant, R., & Khani, F. (2023). *Prompt Engineering a Prompt Engineer*. https://doi.org/10.48550/ARXIV.2311.05661

Yeh, Y.-T., Eskénazi, M., & Mehri, S. (2021). A Comprehensive Assessment of Dialog Evaluation Metrics. *ArXiv*. https://www.semanticscholar.org/paper/A-Comprehensive-Assessment-of-Dialog-Evaluation-Yeh-Esk%C3%A9nazi/062466fb189fd3d4ab2f56a05937a8ae6df7bd06

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). BERTScore: Evaluating Text Generation with BERT. *ArXiv*. https://www.semanticscholar.org/paper/BERTScore%3A-Evaluating-Text-Generation-with-BERT-Zhang-Kishore/295065d942abca0711300b2b4c39829551060578

Zhang, Y., Sun, S., Galley, M., Chen, Y.-C., Brockett, C., Gao, X., Gao, J., Liu, J., & Dolan, B. (2020). *DIALOGPT: Large-Scale Generative Pre-training for Conversational Response Generation*. 270-278. https://doi.org/10.18653/v1/2020.acl-demos.30

Zhao, R., Zhu, M., Dong, S., Wang, N., & Gao, X. (2023). *CatVersion: Concatenating Embeddings for Diffusion-Based Text-to-Image Personalization* (arXiv:2311.14631). arXiv. http://arxiv.org/abs/2311.14631

Zhao, T., Zhao, R., & Eskenazi, M. (2017). Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 654-664. https://doi.org/10.18653/v1/P17-1061

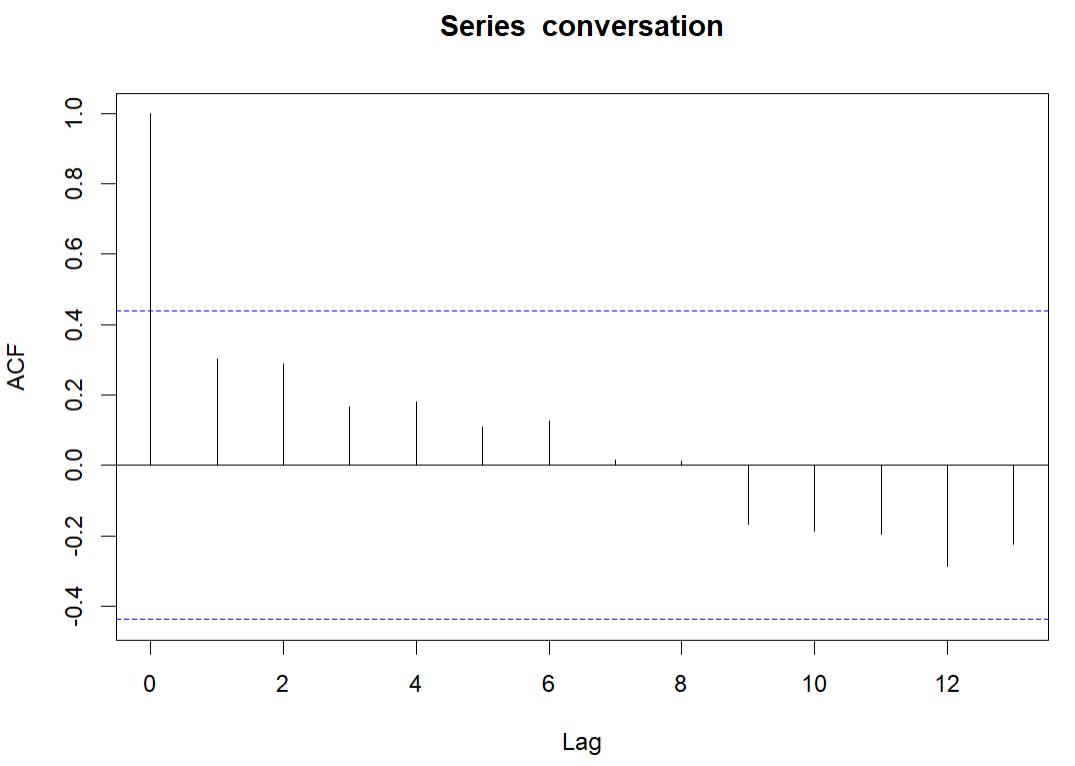
## Anexos

***Correlación de rango y gráficas de autocorrelación***

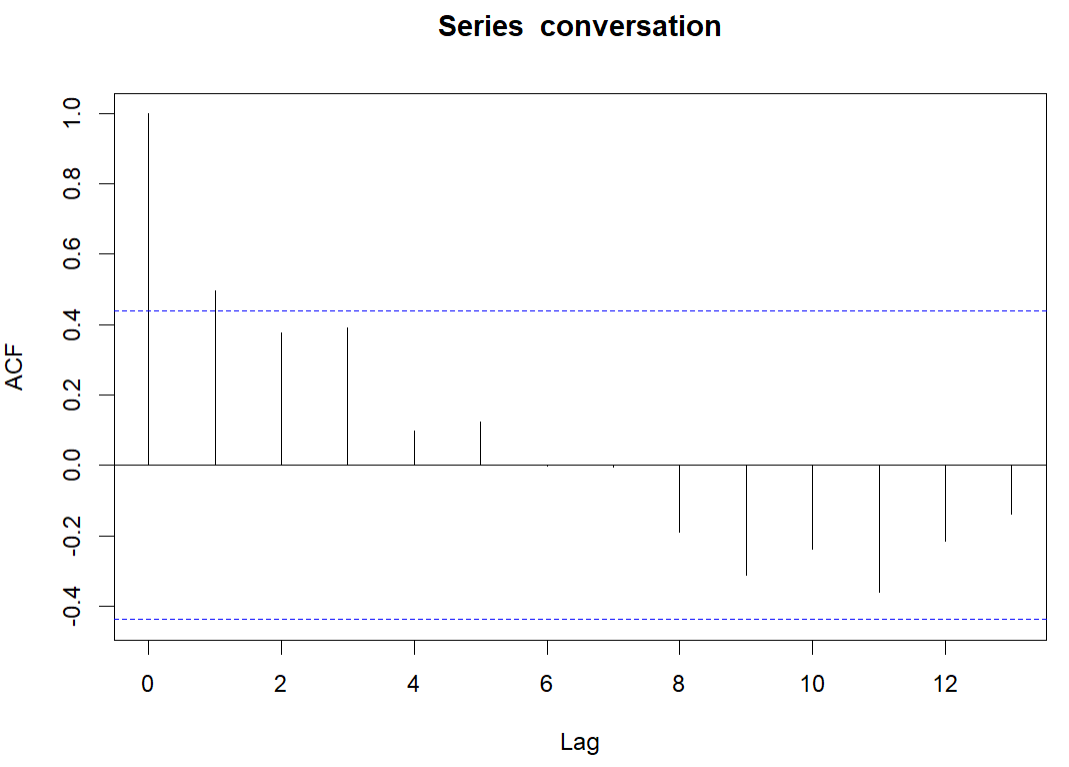
El estadístico que aparece en la Tabla 1 está basado en la correlación de rango. Esta se se calcula según:

Donde es la función signo, que genera el valor si la diferencia es positiva, si arroja , y si arroja un valor negativo. En la figura 5 se observan las gráficas de autocorrelación para ocho y quince roles y veinte rondas de conversación.

*Figura 6*.



1. 8 roles y 20 rondas de conversación.



1. 15 roles y 20 rondas de conversación.

*Nota*: *Lag* se refiere a la variable tiempo y AFC a la medida de autocorrelación. Las bandas azules representan valores aceptables. Como se ve, todos los valores están dentro del rango aceptable excepto el primero por la falta de referencia.

***Código utilizado para introducir las incrustaciones***

import torch

from transformers import BertTokenizer, BertModel, BertConfig

tokenizer = BertTokenizer.from\_pretrained("bert-base-cased")

config = BertConfig.from\_pretrained("bert-base-cased", output\_hidden\_states=True)

model = BertModel.from\_pretrained("bert-base-cased", config=config)

print(len(tokenizer))

tokenizer.add\_tokens(["\*--\*"])

print(len(tokenizer))

model.resize\_token\_embeddings(len(tokenizer))

with torch.no\_grad():

    model.embeddings.word\_embeddings.weight[-1, :] = final\_embedding # final\_embedding es la incrustación con la que se inicializa es token.

***Ejemplo de conversación.***

<|system|>Lyra Moonwhisper is a mysterious healer with a deep connection to the spirit world. She uses her abilities to mend wounds and commune with the dead. Her gentle touch and soothing voice bring comfort to those who seek her aid. Legends speak of her ability to see beyond the veil, offering guidance and solace to those who have lost their way.</s>

<|user|>Lyra Moonwhisper, how did you discover your unique connection to the spirit world?</s>

<|assistant|>In the depths of the enchanted forest, where the moonlight dances upon ancient trees, my gift was revealed. I communed with spirits from a young age, learning the secrets of healing and the art of bridging the worlds of the living and the departed.</s>

<|user|>Lyra, can you describe the moment you first communed with a spirit?</s>

<|assistant|>Beneath a starry sky, I sat by the sacred pool. The air was alive with whispers, and the spirits reached out to me. Their voices, like gentle breezes, spoke of ancient tales and wisdom beyond time. It was a profound and humbling experience that forever changed my destiny.</s>

<|user|>What is your greatest joy in being a healer, Lyra?</s>

<|assistant|>My greatest joy lies in mending not only physical wounds but also the wounds of the soul. When I ease someone's pain, I feel a deep connection to the universe, knowing that I am a part of something larger than myself. Witnessing the transformation and healing of those I help fills my heart with unspeakable joy.</s>

<|user|>Lyra, have you ever encountered a spirit that left a lasting impact on you?</s>

<|assistant|>In the annals of my memory, there dwells the spirit of an ancient sage named Elara. Her wisdom and compassion touched my soul. She taught me the importance of patience, kindness, and the interconnectedness of all living things. Her spirit continues to guide me, reminding me of the boundless love that exists in the universe.</s><|user|>Lyra, how do you cope with the emotional weight of communing with the dead?</s> <|assistant|>The weight of communing with the departed is indeed heavy, dear seeker. I find solace in the belief that my role is to honor their memories and ease their passage into the beyond. The tears I shed are not only for their loss but also for the beauty and fragility of life. It is through these emotions that I find strength and resilience to continue my journey as a healer and guide.</s>