INSTITUTO NACIONAL DE ESTADÍSTICA E INFORMÁTICA

PROGRAMA DE EXTENSIÓN ACADÉMICA – CIENCIA DE DATOS

PROYECTO INTEGRADOR

Clasificación y codificación de respuestas a preguntas abiertas del módulo gobernabilidad de la ENAHO para el periodo 2019-2023 mediante técnicas de procesamiento de lenguaje natural

AUTORES

Carlos Amador Cahuana Torres

Diego Manuel Jesús Vásquez Lévano

Mauricio Rodolfo Condori Maldonado

Probo Camayo Crisóstomo

ASESOR

Alexander Wilder Quispe Rojas

Lima, 9 de diciembre de 2024

Clasificación y codificación de respuestas a preguntas abiertas del módulo gobernabilidad de la ENAHO durante el periodo 2019-2023 mediante técnicas de procesamiento de lenguaje natural

RESUMEN

De los múltiples procesos productivos llevados a cabo en las oficinas nacionales de estadística, uno de los más importantes es la clasificación y codificación. Este proceso, que asigna los registros a un conjunto de categorías predefinidas, permite agrupar los datos bajo una misma descripción, lo que facilita su manejo y análisis. Sin embargo, analizar estos registros lleva mucho más tiempo, especialmente cuando hay un gran número de respuestas. Por tal motivo, instituciones como el INEI realizan estas tareas con la ayuda de procesos dependen en mayor medida de la asistencia de expertos humanos. La clasificación y codificación manual de las respuestas abiertas no solo requiere mucho tiempo, sino que también corre el riesgo de introducir errores humanos o inconsistencias en las respuestas. El trabajo que a continuación se presentará tiene como objetivo evaluar el uso e incorporación de técnicas de procesamiento de lenguaje natural (NLP) como solución computacional a este problema para poder llevar las respuestas abiertas a través de diversas técnicas a texto con etiquetas cercanas a la categóricas, usando tantos modelos clásicos de ML como de clusterización semántica acompañados de modelos de lenguaje como los LLM. Para ello, tomamos la pregunta 2 acerca de la opinión de los principales problemas del Perú del módulo de Gobernabilidad de la Encuesta Nacional de Hogares (ENAHO) 2019-2013. Compararemos diferentes modelos, presentaremos los resultados del análisis y discutiremos la viabilidad de reemplazar el proceso con asistencia humana por un proceso de codificación automatizada utilizando algoritmos de NLP.

Palabras clave: Clasificación; codificación; automatización; procesamiento de lenguaje natural.

INTRODUCCIÓN

En todo proyecto de generación de información estadística por lo general existen preguntas abiertas que dan como resultado un conjunto de descripciones (usualmente en forma de texto) sobre la temática que se está levantando. Debido a que los encuestados pueden ofrecer ideas sobre procesos o razones, o simplemente pueden expresar opiniones e ideas que no se ajustan a las opciones proporcionadas en las preguntas de respuesta cerrada, las preguntas abiertas pueden ayudar a los investigadores a ampliar el alcance de sus conocimientos sobre el tema de estudio. Al proceso de asignarle una clave alfanumérica a estas con fines de explotación de la información se le llama codificación (Bradburn et al., 2004).

En los censos de población y vivienda, así como en las encuestas en hogares, hay diferentes variables que entran al proceso de clasificación y codificación, las más conocidas son la ocupación y actividad económica. Sobre estas variables ya se han desarrollado diversos estudios como Ruiz, Perez y Lopez (2020) y (Instituto Nacional de Estadísticas, 2019).

Actualmente, el área del Instituto Nacional de Estadística e Informática (INEI) a cargo de la codificación de encuestas en hogares, como la Encuesta Nacional de Hogares (ENAHO), ofrece otras variables que cuentan con un alto índice de respuesta. Un ejemplo de esto son las opiniones políticas de las personas acerca de gobernabilidad. El módulo 85 de la ENAHO captura esto en su pregunta 2 como la opinión de los principales problemas del país. La cual tiene 17 categorías ya definidas y categoría 16 (“Otros”) es donde se registran las repuestas abiertas a esta pregunta, esta es la que se usará principalmente en este proyecto.

Generalmente el proceso que realizan las Instituciones Nacionales de Estadísticas o Instituciones Públicas que manejan datos con preguntas abiertas, tratan a estas variables de forma manual. Empiezan identificando los temas recurrentes a partir de las palabras más frecuentes y se asignan códigos o categorías a las respuestas según los temas detectados. Por ejemplo, si la palabra "corrupción" aparece con alta frecuencia, se clasifica bajo la categoría "Corrupción". De esta manera, cada respuesta abierta se agrupa de acuerdo con los códigos previamente definidos. Posteriormente, un equipo de expertos humanos revisa y valida las clasificaciones para asegurarse de que las respuestas se han asignado correctamente a las categorías correspondientes, garantizando la precisión y consistencia en todo el proceso de clasificación (Candela y Cañari, 2022).

1. PROBLEMA

Un costo asociado a este proceso de análisis de respuestas abiertas bajo manipulación humana es que ha sido tradicionalmente mucho más lento y complejo en comparación con el análisis de respuestas cerradas (Kunz et al., 2020; Reja et al., 2003). Considerando que las respuestas abiertas

son datos textuales o cualitativos, estas deben etiquetarse en categorías o códigos (Miles y Huberman, 1994) para facilitar su comparación y, en última instancia, alcanzar entendimientos significativos (Holsti, 1969). Dependiendo del número de respuestas, la complejidad de las preguntas y algunas variables, el proceso de codificación sigue representando un reto en términos de tiempo, recursos humanos y calidad. Autores como Bradburn et al. (2004) señalan que este proceso de categorización puede generar errores durante la codificación de cada respuesta, debido a que se trata de un procedimiento realizado manualmente.

Se ha podido identificar que el INEI ha seguido desarrollando sus procesos de codificación para distintas variables de forma manual, incluyendo la nuestra pregunta de interés, lo cual ha resultado en una gran cantidad de información etiquetada por personal especialmente capacitado para tales tareas. Por la relevancia de este problema, este proyecto busca abordar la siguiente pregunta de investigación:

¿Cómo optimizar el proceso de la codificación y clasificación de respuestas a preguntas abiertas? Específicamente para este proyecto, ¿cómo optimizar el proceso de codificación y clasificación de la variable asociada a la opinión sobre los principales problemas del país, empleando datos del módulo de gobernabilidad de la ENAHO durante el 2019-2023?

Para dar respuesta a la interrogante planteada, este trabajo propone un cambio en el enfoque tradicional de la codificación manual de respuestas abiertas, mejorándola por un proceso automatizado basado en un modelo óptimo desarrollado para clasificar textos. Esto tendrá como fin mejorar la eficiencia y la calidad de la clasificación.

Con el fin de optimizar el proceso, se incorporarán avances recientes en el campo del procesamiento de lenguaje natural (NLP), que permiten tratar grandes volúmenes de texto de manera automatizada. Estas metodologías permiten superar las limitaciones de los enfoques tradicionales, los cuales dependen en gran medida de la intervención humana, y ofrecen una forma más precisa y eficiente de clasificar las respuestas.

El procesamiento de lenguaje natural (NLP) hace referencia a un conjunto de técnicas y algoritmos diseñados para el tratamiento automático del lenguaje, lo que permite que las máquinas comprendan, procesen y clasifiquen textos de manera similar a cómo lo haría un ser humano (Jurafsky y Martin, 2008). En este trabajo, el NLP se aplicará específicamente al procesamiento, codificación y clasificación de las respuestas contenidas en la base de datos de la Encuesta Nacional de Hogares (ENAHO) durante los años 2019-2023, con un enfoque particular en la pregunta abierta 2 del módulo de Gobernabilidad.

2. OBJETIVOS

OBJETIVO GENERAL

- Evaluar la viabilidad de los métodos de procesamiento de lenguaje natural (NLP) para la codificación y etiquetado de respuestas en preguntas abiertas. Empleando como datos la pregunta 2 del módulo de Gobernabilidad de la ENAHO para los años 2019-2023.

OBJETIVOS ESPECIFICOS

- Explorar y comparar los diferentes modelos de cauterización Semántica basado en reducción de dimensionalidad y escalamiento, utilizando las métricas de Silhouette Score, Davies-Bouldin Index y Calinski-Harabasz Index.
- Evaluar diferentes modelos de lenguaje (LLM) para la generación de etiquetas o tópicos representativos, realizando un análisis cualitativo para determinar el modelo más adecuado según las limitaciones y objetivos del proyecto.
- Codificar y etiquetar las respuestas abiertas en nuevas categorías identificadas mediante análisis semántico.
- Diseñar e implementar un pipeline de procesamiento de datos que automatice la codificación y etiquetado de respuestas abiertas, integrando los resultados de los procesos de clusterización semántica y generación de tópicos.
- Generar embeddings de las respuestas abiertas usando Bert y aplicar KMeans para una clusterización no supervisado.
- Desarrollar e implementar un modelo supervisado basado en SVM para clasificar las respuestas abiertas en categorías predefinidas, utilizando los clusters generados como etiquetas y evaluarlo su desempeño usando precisión, recall y f1-score.
-

3. JUSTIFICACIÓN

La justificación de esta investigación radica en la necesidad de mejorar los procesos de clasificación y codificación, haciendo uso de las herramientas tecnológicas actuales para ofrecer resultados más rápidos y de mayor calidad. Al explorar la viabilidad de integrar NLP en las fases de codificación y clasificación, este estudio contribuirá a la mejora de la eficiencia y efectividad en la gestión de grandes volúmenes de datos cualitativos. Además, de mejorar el tiempo de los procesos, reducir los costos por recursos humanos y mejorar en calidad de los resultados, lo cual tiene implicaciones directas para la generación de estadísticas nacionales precisas.

4. MARCO TEÓRICO

El Procesamiento de Lenguaje Natural (NLP) es un campo interdisciplinario dentro de la inteligencia artificial que se centra en la interacción entre las computadoras y el lenguaje humano. Su objetivo es permitir que las máquinas entiendan, interpreten y generen texto o habla de una manera que sea útil. Esto involucra tareas como la traducción automática, la clasificación y codificación de texto, el análisis de sentimientos y la extracción de información (Khurana et al., 2023).

En un proceso de clasificación y codificación de preguntas abiertas (open-ended answer), el NLP se enfoca en transformar el lenguaje natural en representaciones que las máquinas puedan procesar y analizar, con el fin de asignar categorías o etiquetas a las preguntas.

4.1. Embeddings

Los embeddings se refiere al proceso de convertir datos no estructurados, como el texto, en representaciones numéricas o vectoriales que puedan ser procesadas por un computador, manteniendo sus relaciones, similitud o cercanía entre diferentes textos u oraciones, es decir, que preserven sus relaciones locales y globales luego de la transformación. En el contexto del NLP, los embeddings son vectores densos que representan el significado semántico del texto, es decir, el significado contextual de las palabras o frases en función de su uso en el corpus de texto. Modelos como FastText, MiniLM o BERT son ejemplos de cómo podemos generar representaciones eficientes de texto, preservando la información semántica y contextual.

FastText es un modelo de clasificación de texto lineal que opera en base a n-gramas, similar a word2vec, con una base sólida para muchas tareas de clasificación de texto y de rendimiento comparable a otros métodos, al tratar cada palabra no solo como una unidad atómica, sino como una secuencia de n-gramas (subcomponentes de palabras). Lo que le permite que, incluso si una palabra no está en el vocabulario, FastText puede generar una representación semántica basada en los n-gramas que la componen. Esto mejora el manejo de palabras raras o fuera del vocabulario, especialmente en lenguajes con morfología compleja (Joulin, et al., 2017).

MiniLM es un modelo de lenguaje preentrenado y destilado, lo que lo hace más eficiente que los modelos tradicionales como BERT. Utiliza una representación compacta que mantiene un rendimiento competitivo en tareas NLP con un menor uso de recursos computacionales. Aunque BERT es conocido por su capacidad de generar representaciones ricas y contextuales de palabras, MiniLM es una versión con menor complejidad, lo que lo hace más adecuado para implementaciones rápidas y escalables (Wang y Wei, 2020).

4.2. Reducción de Dimensionalidad

Cuando se trabaja con embeddings de texto, a menudo estos tienen una alta dimensionalidad debido a lo complejo que puede ser representar el lenguaje, lo que significa que contienen un gran número de características que pueden dificultar el análisis y/o la visualización de los datos. Para abordar este problema, se utilizan técnicas de reducción de dimensionalidad que tiene como objetivo reducir el número de dimensiones en los datos mientras se intenta preservar su estructura local y global, así como sus propiedades más relevantes, como PCA (Análisis de Componentes Principales) y UMAP (Proyección y Aproximación de Manifold Uniforme).

PCA es una técnica matemática que transforma los datos a un nuevo sistema de coordenadas, donde las nuevas dimensiones (llamadas componentes principales) son lineales y están ordenadas según la cantidad de varianza que explican en los datos. El objetivo principal de PCA es encontrar una forma eficiente de representar los datos en un espacio de menor dimensión, manteniendo la mayor parte de la información posible. Esta técnica es útil cuando los datos tienen una alta correlación y se quiere reducir la redundancia (Taloba, Eisa, e Ismail, 2018).

UMAP es una técnica de reducción de dimensionalidad no lineal que busca preservar tanto la estructura global como local de los datos. Es especialmente eficaz para la visualización de datos de alta dimensión y se ha vuelto popular en el campo del NLP debido a su capacidad para representar datos complejos de manera más interpretable. A diferencia de PCA, que es una técnica lineal, UMAP

maneja relaciones más complejas entre las variables, lo que lo hace más adecuado para datos de texto (McInnes, Healy, y Melville, 2020).

4.3. Semantic Clustering

El clustering semántico es una técnica que se utiliza para agrupar documentos, oraciones o palabras que son semánticamente similares entre sí. A diferencia de otros enfoques de clustering, como el clustering basado en características exactas, el clustering semántico se basa en la similitud de significado, lo que significa que los textos que comparten un significado similar se agruparán incluso si usan palabras diferentes. En este contexto, los algoritmos de clustering como DBSCAN (Density-Based Spatial Clustering of Applications with Noise), HDBSCAN (Hierarchical DBSCAN) y OPTICS (Ordering Points to Identify the Clustering Structure) que se han encontrado en la literatura son los principales modelos de Maching Learning para cubrir esta necesidad.

DBSCAN es un algoritmo de clustering basado en densidad que identifica regiones densas en los datos y las agrupa como clústeres. Además. no requiere que los grupos tengan una forma geométrica específica, sino que identifica regiones de alta densidad separándolas de las regiones de baja densidad para formar los clusters. Los puntos que no pertenecen a ninguna región densa se etiquetan como "ruido". Esta propiedad es útil para identificar clústeres de formas arbitrarias y con cierto ruido (Creţulescu et al., 2019).

HDBSCAN es una extensión de DBSCAN diseñada para manejar datos con densidad variable. Utiliza una jerarquía de clústeres y permite que el algoritmo se ajuste a diferentes niveles de densidad a través de la construcción de una jerarquía de clústeres que permite explorar múltiples niveles de granularidad. HDBSCAN es eficaz para descubrir estructuras más complejas en los datos sin que sea necesario depender de la especificación del número de clústeres de antemano como en los algoritmos tradicionales (Stewart y Al-Khassaweneh, 2022).

OPTICS es otro algoritmo basado en densidad, pero a diferencia de DBSCAN, no requiere que se defina el número de clústeres ni un umbral de densidad fijo. OPTICS genera una ordenación jerárquica de clústeres y puede identificar clústeres de diferentes densidades y formas. Es especialmente útil cuando los datos tienen una estructura compleja o no uniforme o alta dimensionalidad (Ankerst, et al., 1999).

KMeans es un algoritmo de clustering no supervisado que agrupa datos en un número predefinido de clusters basándose en la similitud de sus características. El objetivo es minimizar la distancia dentro de cada grupo y maximizar la distancia entre grupos, asignando a cada punto de datos el centroide más cercano.

4.4. Evaluación de métricas

Después de aplicar los algoritmos de clustering, es fundamental evaluar la calidad del clustering utilizando métricas adecuadas (Ashari et al., 2023). Las siguientes métricas son comúnmente utilizadas en la evaluación de clustering:

S**ilhouette Score**: Mide qué tan bien se asignan los puntos a los clusters. Se calcula tomando en cuenta tanto la cohesión dentro del mismo cluster, qué tan cerca están los puntos entre sí dentro del

mismo grupo, como la separación entre diferentes clusters, qué tan alejados están los puntos de otros clusters (Mamat et al., 2018).

Davies-Bouldin Index: Mide la calidad del clustering evaluando la relación entre la separación y la cohesión de los clusters. Un valor bajo indica que los clústeres son compactos y bien separados. Es útil para evaluar clusters de diferentes tamaños y formas (Lima y Cruz ,2020).

Calinski-Harabasz Index: También conocido como el índice de varianza entre los clusters, mide la separación entre los clusters con respecto a la cohesión interna dentro de los clusters. Un valor más alto de este índice sugiere un mejor agrupamiento, ya que indica que los clústeres están bien separados y los puntos dentro de un clúster están muy cerca unos de otros (Murpratiwi et al., 2021).

4.5. Clasificación y Codificación

Modelos de Lenguaje natural

Una vez que se han procesado y agrupado los datos, los modelos de lenguaje de gran escala (LLMs) como T5 (Text-to-Text Transfer Transformer) y LLaMA son utilizados para tareas de clasificación y análisis profundo debido a los mecanismos de atención. Estos modelos, basados en arquitecturas Transformer, son capaces de realizar una variedad de tareas complejas, como la clasificación de texto, la generación de texto o la respuesta a preguntas, utilizando las representaciones semánticas previamente generadas y los clústeres identificados.

T5 es otro modelo basado en la arquitectura Transformer que trata todas las tareas de NLP como problemas de transformación de texto a texto. Por ejemplo, la tarea de clasificación de texto se podría considerar como un problema de "transformar" el texto en una etiqueta. T5 es muy flexible y puede realizar una variedad de tareas, como traducción, resumen, clasificación, entre otras, de manera unificada (Raffel et al., 2019).

LLaMA es un modelo de lenguaje desarrollado por Meta que se enfoca en generar texto de alta calidad y es competitivo con otros modelos grandes como GPT-3. LLaMA se destaca por su capacidad para generar respuestas coherentes y contextuales a partir de prompts complejos, lo que lo hace útil para tareas que requieren generación de texto o análisis en profundidad (Touvron et al.,2023).

Modelos de ML clásicos

SVM Las Máquinas de Vectores de Soporte (SVM, por sus siglas en inglés) son un algoritmo de aprendizaje supervisado ampliamente utilizado en tareas de clasificación. Su principal objetivo es encontrar un hiperplano que separe las clases de datos de manera óptima, maximizando la distancia entre las fronteras de cada clase (Cortes y Vapnik, 1995). En el contexto del procesamiento de lenguaje natural (NLP), SVM puede ser usado para tareas como análisis de sentimientos, clasificación de documentos y codificación de respuestas abiertas.

5. METODOLOGÍA DE INVESTIGACIÓN

5.1. Fuente de datos

Se usará el módulo 85 de Gobernabilidad de la Encuesta Nacional de Hogares de los años 2019 al 2023, se tomará como año de referencia el 2019 para facilitar el procesamiento y cantidad la cantidad de registros de respuestas abiertas. La pregunta de interés es la número 2 ("t1ocu"), planteada como: "En su opinión, actualmente, ¿cuáles son los principales problemas del país?", esta pregunta puede ser respondida en 17 categorías: 1) la corrupción, 2) la falta de credibilidad y transparencia del gobierno, 3) la falta de empleo, 4) falta de seguridad ciudadana, 5) violencia en los hogares, 6) falta de cobertura / mala atención en salud pública, 7) falta de cobertura del sistema de seguridad social, 8) mala calidad de la educación estatal, 9) violación de derechos humanos, 10) bajos sueldos / aumento de precios, 11) pobreza, 12) falta de vivienda, 13) falta de apoyo a la agricultura, 14) mal funcionamiento de la democracia, 15) delincuencia, 16) otro 17) ninguno. De estas categorías, la número 16 es la que representa las respuestas abiertas a la pregunta de interés y estas repuestas están registradas en la variable "t1txt". Se tomará como datos a evaluar los textos registrados en esta variable.

5.2. Herramientas

Se usarán los modelos y técnicas mencionadas en el marco teórico:

1. Para convertir datos no estructurados en representaciones numéricas o vectoriales a través de embeddings se usarán modelos como FastText, MiniLM y BERT.
2. Para reducir la dimensionalidad de los embeddings se usará técnicas como PCA y UMAP.
3. Para agrupar palabras semánticamente similares entre si usaremos técnicas de Semantic Clustering como DBSCAN, HDBSCAN y OPTICS.
4. Para evaluar la calidad del clustering se utilizará métricas como Silhouette Score, Davies-Bouldin Index y Calinski-Harabasz Index.
5. Para clasificar y codificar se usaron modelos como T5, LLaMA y SVM.

5.3. Estrategia metodológica

El desarrollo de nuestro proyecto se estructuró en tres etapas principales, claramente diferenciadas, que abordan problemas complementarios: (1) clusterización semántica, (2) generación de tópicos o categorías y (3) implementación de un pipeline de datos para la codificación y etiquetado automático. Estas etapas están diseñadas para resolver de manera integral la problemática de analizar y organizar grandes volúmenes de datos textuales a través de un procesamiento secuencial de la información.

Etapa Previa: Limpieza de datos

Para esta etapa inicial, se realizó una limpieza de los datos a través de expresiones regulares, para luego realizar una lematización y el pase a minúsculas para la uniformización de la información, así como la eliminación de conectores no necesarias, con el fin de uniformizar la información.

Primera etapa: Clusterización Semántica Basada en Reducción de Características y Embeddings

El objetivo de esta etapa fue identificar grupos de textos semánticamente similares, capturando relaciones tanto locales como globales en el corpus. Esto se logró mediante un enfoque metodológico que integra embeddings de texto, técnicas de reducción de dimensionalidad y algoritmos de clusterización. El proceso se desarrolló en dos fases clave

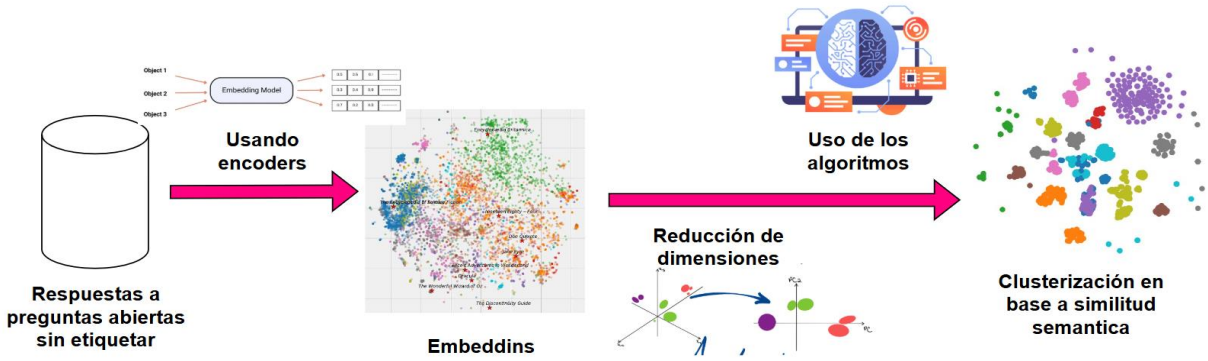


Gráfico sobre el proceso de Clusterización semántica basada en embeddings

Comparación de las mejores configuraciones

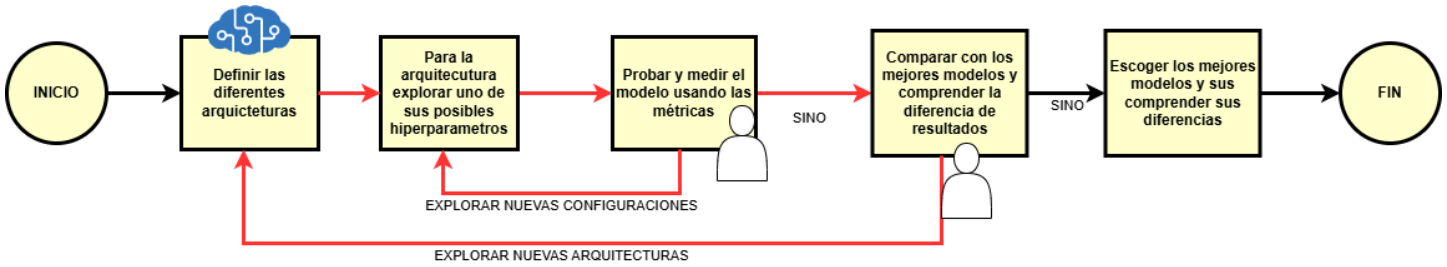
Para encontrar la mejor arquitectura de clusterización semántica, se evaluaron distintas configuraciones basadas en

- **Embeddings:** Comparación entre FastText y MiniLM, dos enfoques distintos para representar semánticamente los textos.
- **Reducción de Dimensionalidad:** Uso de PCA (Análisis de Componentes Principales) y UMAP (Mapeo Uniforme Aproximado y Proyectado) para reducir la dimensionalidad de los embeddings y preservar las relaciones semánticas.
- **Algoritmos de Clusterización:** Evaluación de los modelos DBSCAN, HDBSCAN y OPTICS, seleccionados por su capacidad para manejar estructuras complejas y datos ruidosos.

A su vez, el rendimiento de cada combinación se evaluó de forma cualitativa mediante las métricas de **Silhouette Score**, **Davies-Bouldin Index** y **Calinski-Harabasz Index**, que permitieron medir la cohesión y separación de los clústeres generados en este problema no supervisado.

Aplicación de una heurística iterativa para Manejo de Outliers

Debido a que, por construcción de los modelos utilizados, ciertos puntos en los datos se categorizan como outliers o ruido, aunque no necesariamente lo sean, debido a las limitaciones inherentes de los algoritmos, se diseñó una heurística iterativa para volver a procesar estos datos y reducir la pérdida de información. Este enfoque permitió reetiquetar datos inicialmente descartados como outliers y mejorar la consistencia de los clústeres. La heurística se aplicó a las arquitecturas más prometedoras obtenidas de la parte anterior, ajustando algunos parámetros, como el número de iteraciones, para asegurar un etiquetado más completo y representativo.



Representación del flujo de trabajo de la etapa 1: Proceso que representa como se exploraron las diferentes configuraciones, arquitecturas y heurísticas para encontrar las propuestas óptimas a nuestro problema

Segunda etapa: Generación de Tópicos para los Clústeres

Una vez formados los clústeres de texto semánticamente similares, el siguiente paso fue generar una descripción clara y concisa que representará el contenido de cada clúster. Esta tarea implicó la aplicación de **grandes modelos de lenguaje (LLMs)** para la generación automática de etiquetas o categorías que sintetizaran los temas abordados en cada grupo. Para esto, se buscó abordar el problema a través de los siguientes modelos Bert, T5 y Llama3

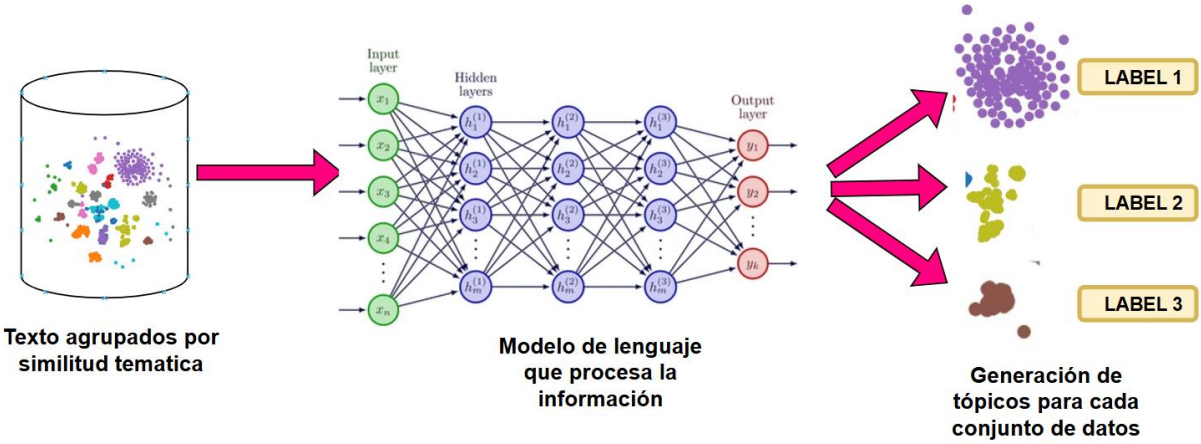


Gráfico de la representación de la generación de tópicos para grupo de textos

Tercera etapa (adicional): Desarrollo de un Pipeline para la Codificación y Etiquetado Automático Para las respuestas abiertas del INEI del año 2019 al 2023

Con los resultados de las etapas anteriores, se diseñó un **pipeline de datos** para automatizar la codificación y etiquetado de respuestas abiertas recopiladas por el INEI entre 2019 y 2023 con el fin de codificar de forma rápida y clara las respuestas a preguntas abiertas sin codificar y etiquetar. Este pipeline integra como resultado de las etapas previas:

- La configuración óptima de clusterización semántica para agrupar textos similares.
- Adicionalmente se desarrolló clusters con el método (K-Means) dentro del SVM para la clasificación supervisada
- El modelo de generación de etiquetas seleccionado para asignar categorías significativas a cada clúster.

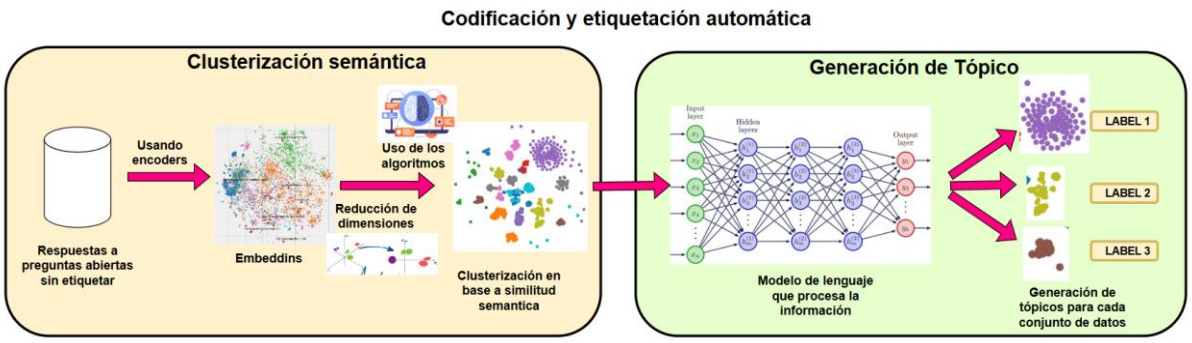


Gráfico que representa flujo de trabajo para obtener un automatic encoding a través de la clusterización semántica y la generación de tópicos

6. RESULTADOS

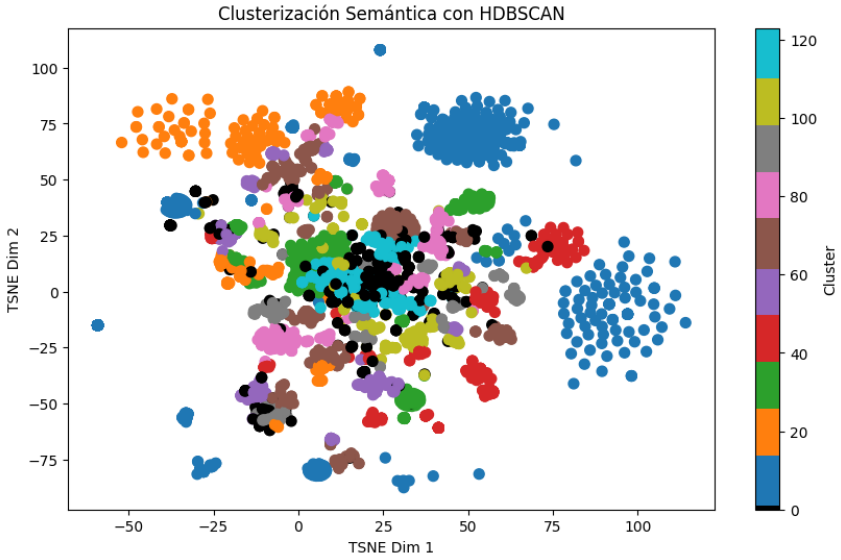
Dado los diferentes modelos propuestos para la clusterización semánticas se han obtenido los siguientes resultados, tomando como referencia la pregunta de interés del 2019, con un número de datos de n=5000.

Para los modelos sin aplicar la heurística de iteración:

Modelo	outliers	Silhouette Score	Davies Bouldin score	Calinski harabasz score	Time (s)
MiniLM + Scaler + DBSCAN (cosine)	452	0.2173	1.5024	48.0604	3.6572
MiniLM + Scaler + HDBSCAN (euclidean)	1606	0.3230	0.6735	914.0697	16.8431
MiniLM + Scaler + PCA (80) + HDBSCAN (euclidean)	1646	0.0922	1.1790	849.6905	5.2555
MiniLM + Scaler + UMAP (80, cosine) + HDBSCAN (euclidean)	358	0.9388	0.2948	27329.4622	34.0424
MiniLM + Scaler + UMAP (80, cosine) + OPTICS (cosine)	1598	0.8234	0.4960	99538.0719	77.1770
MiniLM + UMAP (80, cosine) + OPTICS (cosine)	1444	0.8160	0.5148	105394.8025	70.1973
FastText + HDBSCAN (euclidean)	1660	0.9223	0.6288	1832.0920	12.6565
FastText + OPTICS (cosine)	1512	0.9170	0.7107	1152.7707	119.2204
FastText + Scaler + PCA + OPTICS (cosine)	1561	0.1524	1.2714	1001.8432	50.9309
FastText + Scaler + UMAP (80, cosine) + OPTIC (cosine)	1367	0.8245	0.5119	71905.9911	127.4634
BERT + Kmeans	894	0.4511	2.3723	525.17	

Del cuadro podemos ver que el cuarto modelo nos da los resultados más balanceados y óptimos, donde es el mejor con Silhouette Score y Davies Bouldin score, y teniendo también un Calinski Harabasz score bastante bueno, a su vez que maneja el número de outliers bastante bien, teniendo

el menor número de outliers, lo que podría indicar que está manejando mejor los outliers, categorizándolos de mejor forma. Lo que se puede ver mejor en los gráficos.



Gráfica de representación usando TSNE del modelo MiniLM + Scaler + UMAP (80, cosine) + HDBSCAN (euclidean), con una reducción a dos dimensiones para la representación geométrica y espacial de la clusterización.

Por otra parte, vemos que el quinto modelo da resultados más rápidos y de calidad medianamente aceptables, dado que tiene un Silhouette Score bastante alto (cercano al anterior que es el óptimo) y un Davies Bouldin score ligeramente alto, así como un Calinski harabasz score no tan bajo, pero tampoco alto, pero que se ha realizado en tiempos de ejecución bastante bajos.

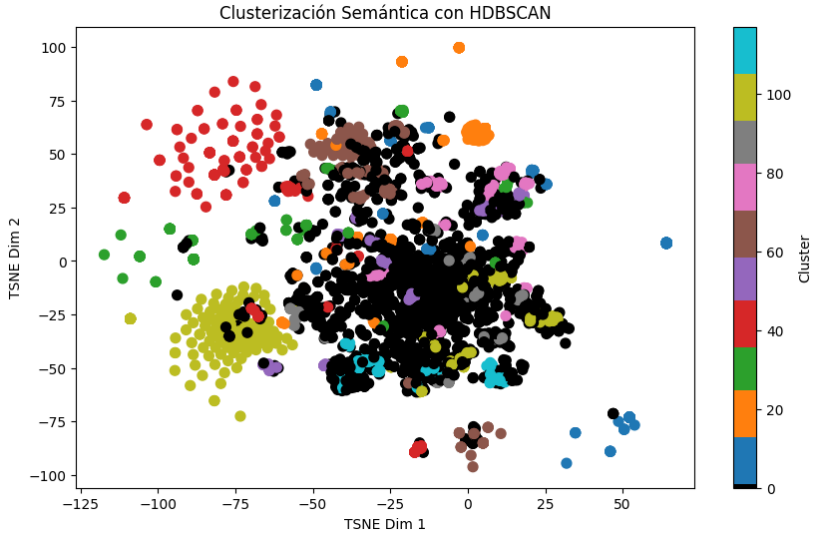


Gráfico de representación usando TSNE del modelo FastText + HDBSCAN (euclidean) con una reducción a dos dimensiones para la representación geométrica y espacial de la clusterización.

A su vez, realizando algunas pruebas utilizando la heurística iterativa. Sin embargo, no se obtuvieron los resultados esperados, viendo que las métricas decaían conforme aumentaban las iteraciones. Dentro de uno de los casos explorados es el modelo óptimo MiniLM + Scaler + UMAP (80, cosine) + HDBSCAN (euclidean) obtenido, pero con la modificación iterativa.

Outliers	Silhouette Score	Davies Bouldin score	Calinski harabasz score
386	0.9387	0.2826	28326.0191
49	0.2572	3.2285	523.9103

Por otro lado, al usar SVM las métricas reflejan que el modelo clasifica de manera confiable las respuestas en las categorías definidas previamente. Este nivel de rendimiento resalta la eficacia del enfoque basado en SVM para la automatización de procesos tradicionalmente manuales, como la clasificación de respuestas abiertas.

Los resultados obtenidos con el modelo SVM demuestran su capacidad para integrar eficientemente los clusters generados en una solución supervisada robusta. Esto optimiza el proceso de clasificación, reduciendo errores humanos y el tiempo requerido para la codificación manual. Dado su alto rendimiento, el modelo es adecuado para su implementación en producción y para clasificar respuestas abiertas de otros años o módulos similares, sin embargo, la evaluación de los cluster aun no es del todo clara y puede ser cuestionada.

Categoría (Cluster)	Precisión	Recall	F1-Score	Soporte (Cantidad de datos)
0 (Discriminación)	0.97	1.00	0.98	83
1 (No sabe)	1.00	1.00	1.00	1331
2 (Pandemia)	1.00	1.00	1.00	681
3 (Servicios básicos)	0.99	0.99	0.99	973
4 (Crisis y corrupción)	1.00	0.99	1.00	3126
5 (Coronavirus)	1.00	1.00	1.00	692

Por otra parte, luego de obtener los resultados anteriores, procedemos a realizar el siguiente pipeline de codificación y etiquetado automatizado, a través de la utilización de modelos de LLM, debido a los alcances del tiempo y la limitación de data no etiquetada, se realizó una comparación más cuantitativa del desempeño del etiquetado para los diferentes clusters. Obteniéndose los siguientes resultados:

Modelo	Resultados obtenidos
Llama3	Genera etiquetas acorde al contenido al texto del cluster. Sin embargo, requiere un buen prompt y medianamente específico para poder realizar las tareas de prompt enginer
T5	No se desempeña muy bien en su forma base, por lo que se requiere un fine Turing del modelo para que pueda funcionar correctamente para la tarea requerida

Finalmente, como parte de la propuesta de valor adicional, se realizó el siguiente pipeline para la codificación y etiquetado automático de los datos del INEI, teniéndose que se siguiendo el procesamiento del gráfico, se logró cubrir un 92% de las respuestas abiertas del 2019 al 2023, lo que

da un total de 34428 respuestas abiertas a ser etiquetadas para la pregunta 2 y en un tiempo menor a 10 minutos. Por otra parte, también se podría aplicar a las diferentes preguntas abiertas tanto del módulo como de otros módulos que puedan ser de interés.

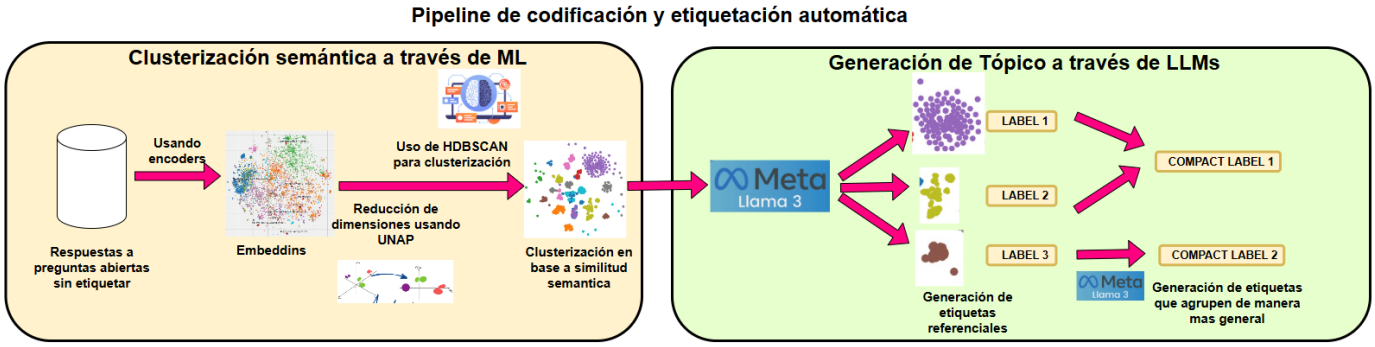


Gráfico del Pipeline de procesamiento de datos para la generación automatizada de codificaciones y etiquetas para las respuestas abiertas del INEI 2019-2023

7. CONCLUSIONES

El presente proyecto ha desarrollado y evaluado modelos para clasificar y codificar las respuestas a preguntas abiertas. Se han utilizado técnicas y modelos de procesamiento de lenguaje natural. Al evaluar las diferentes variaciones de los modelos propuestos se encontró que al no usar la iteración heurística el modelo óptimo es el MiniLM + Scaler + UMAP (80, cosine) + HDBSCAN (euclidean) que nos brinda estadísticos balanceados para las métricas Silhouette Score y Davies Bouldin score y Calinski Harabasz. También se encontró que al usar la iteración heurística no se obtienen resultados esperados.

Por otra parte, se buscó complementar con el uso de modelos de lenguaje natural adaptados al español, sin embargo, debido a las limitaciones de tiempo y recursos, se hizo una comparación cualitativa entre T5 y Llama, dando que Llama funciono mejor en su forma base a comparación de T5, el cual requiere de un fine-tuning para poder trabajar esta tarea.

El modelo SVM mostró gran precisión y recall en la clasificación de respuestas abiertas, evidenciando su eficacia. Sin embargo, su desempeño depende de la calidad de los clusters generados previamente. Problemas como datos ambiguos o categorías solapadas pueden afectar la asignación de etiquetas. Es importante continuar refinando los métodos de clusterización inicial y supervisar el pipeline para mantener la precisión en futuros análisis.

Luego, el desarrollo del pipeline de procesamiento de datos para poder procesar los datos del INEI a fin de generar categorías se puede realizar de forma automática, debido a que como se mostró en los resultados del proyecto, funciona en un periodo menor a 10 minutos, lo cual es bastante rápido y muestra resultados bastantes aceptables.

Por otro lado, el objetivo general de este trabajo es evaluar la factibilidad de incorporación de técnicas de procesamiento de lenguaje natural dentro de los actuales procesos productivos que se siguen para la codificación de variables de encuestas en hogares realizadas por el INEI. Los resultados

encontrados muestran que es posible mejorar el proceso de clasificación y codificación mediante la automatización usando modelos de NLP. Sin embargo, esto recién es un planteamiento de mejora al proceso tradicional.

8. LIMITACIONES Y EXTENSIONES DEL TRABAJO

Limitaciones

- Calidad de los datos: Los textos deben estar bien preprocesados, evitando la mala ortografía ya que el modelo puede agruparlos de manera incorrecta.
- Falta de información etiquetada.
- Falta de recursos computacionales como GPU/TPU: Los modelos transformers hacen uso de altos requisitos de cómputo y memoria. La falta de estos hace que el entrenamiento del modelo sea lento, el tiempo de respuesta se vuelve mayor y te limita a usar modelos más complejos.
- Problemas de atención a nuestros requerimientos por parte del INEI, retraso excesivo en él envió de la información solicitada para nuestro proyecto, falta de atención y confusión del personal.

Extensiones del trabajo

- Actualmente, en los modelos considerados para la generación de etiquetas, se recomienda explorar modelos más orientados a esa tarea especifica.
- También se recomienda revisar como estrategia el uso de técnicas de generación de data sintética para poder explorar modelos supervisados, lo cual podría ayudar a obtener mejores modelos y pipelines de procesamiento de datos.

9. ANEXOS

El código fue subido a la carpeta drive llamada “Códigos”.

10. REFERENCIAS

1. Ankerst, M., Breunig, M. M., Kriegel, H.-P., & Sander, J. (1999). OPTICS: Ordering points to identify the clustering structure. *ACM SIGMOD Record, 29*(2), 49-60. https://doi.org/10.1145/345025.345028

2. Ashari, I. F., Nugroho, E. D., Baraku, R., Yanda, I. N., & Liwardana, R. (2023). Analysis of Elbow, Silhouette, Davies-Bouldin, Calinski-Harabasz, and Rand-Index evaluation on K-Means algorithm for classifying flood-affected areas in Jakarta. *Journal of Applied Informatics and Computing (JAIC), 7*(1), 89-97. http://jurnal.polibatam.ac.id/index.php/JAIC

3. Bradburn, N. M., Sudman, S., & Wansink, B. (2004). *Asking questions: The definitive guide to questionnaire design -- For market research, political polls, and social and health questionnaires* (2nd ed.). Jossey-Bass.

4. Cretulescu, R. G., Morariu, D. I., Breazu, M., & Volovici, D. (2019). DBSCAN algorithm for document clustering. *International Journal of Advanced Statistics and IT&C for Economics and Life Sciences, 9*(1), 58-66. https://doi.org/10.2478/ijasitels-2019-0007

5. Candela Rojas, E. C., & Cañari Huerta, E. G. (2022). *Machine learning para la categorización de respuestas de preguntas abiertas*. Oficina de Seguimiento y Evaluación Estratégica (OSEE), Ministerio de Educación.

6. Holsti, O. R. (1969). *Content analysis for the social sciences and humanities*. Addison-Wesley.

7. Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2017). Bag of tricks for efficient text classification. *arXiv*. https://arxiv.org/abs/1607.01759

8. Jurafsky, D., & Martin, J. H. (2008). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition* (2nd ed.). Prentice Hall.

9. Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023). Natural language processing: State of the art, current trends and challenges. *Multimedia Tools and Applications, 82*, 3713-3744. https://doi.org/10.1007/s11042-022-14279-3

10. Kunz, T., Quoß, F., & Gummer, T. (2020). Using placeholder text in narrative open-ended questions in web surveys. *Journal of Survey Statistics and Methodology*. https://doi.org/10.1093/jssam/smz027

11. Lima, S. P., & Cruz, M. D. (2020). A genetic algorithm using Calinski-Harabasz index for automatic clustering problem. *Revista Brasileira de Computação Aplicada, 12*(3), 97–106. https://doi.org/10.5335/rbca.v12i3.11117

12. Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook*. Sage.

13. Mamat, A. R., Mohamed, F. S., Mohamed, M. A., Rawi, N. M., & Awang, M. I. (2018). Silhouette index for determining optimal k-means clustering on images in different color models. *International Journal of Engineering Technology, 7*(2), 105–109. https://doi.org/10.14419/ijet.v7i2.14.11464

14. McInnes, L., Healy, J., & Melville, J. (2020). UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv*. https://doi.org/10.48550/arXiv.1802.03426

15. Murpratiwi, S. I., Agung Indrawan, I. G., & Aranta, A. (2021). Analisis Pemilihan Cluster Optimal Dalam Segmentasi Pelanggan Toko Retail. Jurnal Pendidikan Teknologi Dan Kejuruan, 18(2), 152. https://doi.org/10.23887/jptk-undiksha.v18i2.37426

16. Reja, U., Manfreda, K. L., Hlebec, V., & Vehovar, V. (2003). Open-ended vs. close-ended questions in web questionnaires. *Developments in Applied Statistics, 19*(1), 159-177.

17. Raffel, C., Shazeer, N. M., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2019). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research, 21*, 140:1-140:67. https://api.semanticscholar.org/CorpusID:204838007

18. Ruiz Sánchez, J. A., Pérez Sánchez, J., & Pastor López Monroy, A. (2022). Evaluación de técnicas de procesamiento de lenguaje natural y Machine Learning para los procesos de codificación de encuestas en hogares. *Realidad, Datos y Espacio: Revista Internacional de Estadística y Geografía, 13*(2).

19. Stewart, G., & Alkhassaweneh, M. (2022). An implementation of the HDBSCAN* clustering algorithm. *Applied Sciences, 12*(5), 2405. https://doi.org/10.3390/app12052405

20. Taloba, A. I., Eisa, D. A., & Ismail, S. S. I. (2018). A comparative study on using principal component analysis with different text classifiers. *International Journal of Computer Applications, 180*(31), 1-6.

21. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., & Lample, G. (2023). LLaMA: Open and efficient foundation language models. *arXiv*. https://doi.org/10.48550/arXiv.2302.13971

22. Instituto Nacional de Estadísticas. (2019). *Sistema de clasificación y codificación automática en la encuesta nacional de empleo*. Instituto Nacional de Estadísticas.

23. Wang, X., & Wei, F. (2020). MiniLM: Deep self-attention distillation for transformer models. *arXiv preprint arXiv:2002.10957*. https://doi.org/10.48550/arXiv.2002.10957