Centro de Investigación en Matemáticas (CIMAT).
Unidad Monterrey

Ciencia de Datos Tarea 4

Diego Godinez Bravo

17 de mayo de 2024

1. Problema 1

Calcula lo siguiente:

$$\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}\begin{pmatrix}0 & 1\\ 2 & 3\end{pmatrix} + \begin{pmatrix}7 & 9\end{pmatrix}$$

Usa broadcasting de tal forma que la operación esté bien definida. Antes, averigua y describe qué es broadcasting, en el contexto de numpy.

1.1. Solución

El término '*broadcasting*' se refiere a la manera en cómo NumPy opera con arreglos de diferentes dimensiones durante las manipulacioens aritméticas. Dadas ciertas restricciones, el arreglo de menor dimensión se redimensiona a lo largo del arreglo de mayor dimensión de manera que sus dimensiones sean compatibles y la operación aritmética sea válida.

En este caso, la operación se realiza elemento por elemento $a_{ij} * b_{ij}$

$$\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}\begin{pmatrix}0 & 1\\ 2 & 3\end{pmatrix} = \begin{pmatrix}0 & 2\\ 6 & 12\end{pmatrix}$$

Al sumar el arreglo unidimensional $c_{1\times 2}$, NumPy extiende el arreglo para que se convierta en un arreglo de dimensión 2×2

$$\begin{pmatrix}7 & 9\\ 7 & 9\end{pmatrix}$$

Por lo tanto, NumPy realiza la operación elemento por elemento, de modo que el resultado se define de la siguiente manera

$$\begin{pmatrix}0 & 2\\ 6 & 12\end{pmatrix} + \begin{pmatrix}7 & 9\\ 7 & 9\end{pmatrix} = \begin{pmatrix}7 & 11\\ 13 & 21\end{pmatrix}$$

2. Problema 2

Considera un problema de clasificacón multiclase y una red neuronal densa con una capa oculta, como se muestra en la Figura 1. Consideraremos también el uso de la función sigmoide como activación de las unidades ocultas, la función softmax para las estimaciones en la capa de salida y la entropía cruzada como función de costo.

a) Muestra que softmax es invariante a traslaciones (constantes) del vector de entrada, es decir, para cualquier vector x y cualquier constante c:

$$\text{softmax}(x) = \text{softmax}(x + c),$$

donde la operación $x + c$ se realiza con broadcasting. Recuerda que

$$\text{softmax}(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}}.$$

Lo anterior es útil cuando se escoge $c = -\max(x)$, es decir, quitando el valor mayor en todos los elementos de x, para estabilidad numérica.

b) Para un escalar x, muestra que el gradiente de la función sigmoide es

$$\sigma(x)(1 - \sigma(x))$$

c) Muestra que el gradiente en la capa de salida es

$$\frac{\partial L(y, \hat{y})}{\partial z} = \hat{y} - y,$$

donde $\hat{y} = \text{softmax}(z)$, para algún vector z que proviene de la capa de salida.
La función de costo, como mencionamos al inicio, es la entropía cruzada:

$$L(y, \hat{y}) = -\sum_i y_i \log(\hat{y}_i),$$

donde y es un vector *one-hot* de las clases y $\hat{y}$ es el vector de probabilidades estimadas.

d) Considerando los incisos anteriores, obtén los gradientes respecto a los parámetros del modelo calculando

$$\frac{\partial L(y, \hat{y})}{\partial x},$$

para obtener de esta forma, las ecuaciones de retropropagación de la red. Recuerda que el paso forward calcula las activaciones: $h = \sigma(W_1 x + b_1)$ y $\hat{y} = \text{softmax}(W_2 h + b_2)$. Recuerda también que la función de activación en un vector, se aplica entrada por entrada.

2.1. Solución

Considerando un vector de entrada $x \in \mathbb{R}^n$

$$\text{softmax}(x) = \left(\frac{e^{x_1}}{\sum_j e^{x_j}}, \frac{e^{x_2}}{\sum_j e^{x_j}}, \ldots, \frac{e^{x_n}}{\sum_j e^{x_j}} \right)$$

Al aplicar una traslación constante al vector de entrada

$$\text{softmax}(x + c) = \left(\frac{e^{x_1+c}}{\sum_j e^{x_j+c}}, \frac{e^{x_2+c}}{\sum_j e^{x_j+c}}, \ldots, \frac{e^{x_n+c}}{\sum_j e^{x_j+c}} \right)$$

Manipulando algebraicamente la expresión

$$\text{softmax}(x + c) = \left(\frac{e^{x_1+c}}{\sum_j e^{x_j+c}}, \frac{e^{x_2+c}}{\sum_j e^{x_j+c}}, \ldots, \frac{e^{x_n+c}}{\sum_j e^{x_j+c}} \right)$$

donde

$$\sum_j e^{x_j+c} = e^{x_1+c} + e^{x_2+c} + \ldots e^{x_n+c}$$

$$= e^c \left(e^{x_1} + e^{x_2} + \ldots e^{x_n}\right)$$

$$= (\sum_j e^{x_j}) \cdot e^c$$

Por lo tanto

$$= \left(\frac{e^{x_1} \cdot e^c}{(\sum_j e^{x_j}) \cdot e^c}, \frac{e^{x_2} \cdot e^c}{(\sum_j e^{x_j}) \cdot e^c}, \ldots, \frac{e^{x_n} \cdot e^c}{(\sum_j e^{x_j}) \cdot e^c} \right)$$

$$= \frac{e^c}{e^c} \left(\frac{e^{x_1}}{\sum_j e^{x_j}}, \frac{e^{x_2}}{\sum_j e^{x_j}}, \ldots, \frac{e^{x_n}}{\sum_j e^{x_j}} \right)$$

De manera que

$$\text{softmax}(x) = \left(\frac{e^{x_1}}{\sum_j e^{x_j}}, \frac{e^{x_2}}{\sum_j e^{x_j}}, \ldots, \frac{e^{x_n}}{\sum_j e^{x_j}} \right)$$

Demostrando que softmax se mantiene invariante a traslaciones del vector de entrada para cualquier vector $x \in \mathbb{R}^n$ y cualquier constante c.

2.2. Solución

Sea la función sigmoide definida de la siguiente manera

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Calculamos el gradiente de la función sigmoide respecto a x

$$\frac{d}{dx}\sigma(x) = \frac{d}{dx}\left(\frac{1}{1+e^{-x}}\right)$$

sea $u = (1+e^{-x})^{-1}$

$$= -u^{-2}\frac{d}{dx}u$$

$$= -\frac{1}{(1+e^{-x})^{-2}}(-e^{-x})$$

$$= \frac{e^{-x}}{(1+e^{-x})^{-2}}$$

$$= \frac{1}{1+e^{-x}}\left(\frac{e^{-x}}{1+e^{-x}}\right)$$

donde $\sigma(x) = \frac{1}{1+e^{-x}}$, por lo tanto

$$= \sigma(x)\left(\frac{e^{-x}}{1+e^{-x}}\right)$$

$$= \sigma(x)\left(1 - \frac{1}{(1+e^{-x})}\right)$$

De manera que el gradiente de la función sigmoide se define de la siguiente manera

$$= \sigma(x)\left(1-\sigma(x)\right)$$

2.3. Solución

Sea la función de costo cross-entropy definida de la siguiente manera

$$L(y, \hat{y}) = -\sum_i y_i \log(\hat{y}_i)$$

Calculamos el gradiente en la capa de salida con respecto a z_i

$$\frac{\partial L(y, \hat{y})}{\partial z_j} = \frac{\partial}{\partial z_j}\left(-\sum_i y_i \log(\hat{y}_i)\right)$$

donde $\hat{y}_i = \text{softmax}(z_i)$, para algún vector z que proviene de la capa de salida.

Por lo tanto

$$\begin{aligned}
\frac{\partial L(y, \hat{y})}{\partial z_j} &= \frac{\partial}{\partial z_j}\left(-\sum_i y_i \log(\hat{y}_i)\right) \\
&= \frac{\partial}{\partial \hat{y}_i}\left(-\sum_i y_i \log(\hat{y}_i)\right) \cdot \frac{\partial \hat{y}_i}{\partial z_j} \\
&= -\frac{1}{\hat{y}_i} \cdot \frac{\partial \hat{y}_i}{\partial z_j}
\end{aligned}$$

Calculamos $\frac{\partial \hat{y}_i}{\partial z_j}$ con $\hat{y}_i = \text{softmax}(z_i)$, considerando ambos casos donde $j = i$, y $j \neq i$.

Sea $j = i$

$$\begin{aligned}
\frac{\partial \hat{y}_i}{\partial z_i} &= \frac{\partial}{\partial z_i}\left(\frac{e^{zi}}{\sum_j e^{z_j}}\right) \\
&= \frac{e^{z_i}(\sum_j e^{z_j}) - e^{z_i} e^{z_i}}{\left(\sum_j e^{z_j}\right)^2} \\
&= \left(\frac{e^{z_i}}{\sum_j e^{z_j}}\right)\left(\frac{\sum_j e^{z_j} - e^{z_i}}{\sum_j e^{z_j}}\right) \\
&= \left(\frac{e^{z_i}}{\sum_j e^{z_j}}\right)\left(1 - \frac{e^{z_i}}{\sum_j e^{z_j}}\right)
\end{aligned}$$

Finalmente

$$\frac{\partial \hat{y}_i}{\partial z_i} = (\hat{y}_i)\,(1 - \hat{y}_i)$$

donde $\hat{y}_i = \text{softmax}(z_i)$ con $\text{softmax}(z)_i = \frac{e^{z_i}}{\sum_j e^{z_j}}$

Sea $j \neq i$

$$\begin{aligned}
\frac{\partial \hat{y}_i}{\partial z_j} &= \frac{\partial}{\partial z_i}\left(\frac{e^{zj}}{\sum_j e^{z_j}}\right) \\
&= \frac{-e^{z_i} e^{-z_j}}{\left(\sum_j e^{z_j}\right)^2} \\
&= \left(\frac{e^{z_i}}{\sum_j e^{z_j}}\right)\left(\frac{-e^{z_j}}{\sum_j e^{z_j}}\right)
\end{aligned}$$

Finalmente

$$\frac{\partial \hat{y}_i}{\partial z_j} = (\hat{y}_i)\,(-\hat{y}_i)$$

Es decir, podemos expresar la derivada de la siguiente manera

$$\frac{\partial \hat{y}i}{\partial z_j} = (\hat{y}_i)\,(\delta ij - \hat{y}j)$$

donde δij es una delta de Kronecker definida como

$$\delta_{ij} = \begin{cases} 1 & \text{si } j = i \\ 0 & \text{si } j \neq i \end{cases}$$

Por lo tanto

$$\frac{\partial L(y,\hat{y})}{\partial z_j} = -\sum_i \frac{\partial L(y,\hat{y})}{\partial \hat{y}_i}\frac{\partial \hat{y}_i}{\partial z_j}$$

$$= -\sum_i \left(\frac{y_i}{\hat{y}i}\right) \hat{y}_i(\delta ij - \hat{y}_j)$$

Sea y un vector *one hot*, entonces

$$= -\left(\frac{1}{\hat{y}i}\right) \hat{y}_i(\delta ij - \hat{y}_j)$$

$$= -(1 - \hat{y}_j)$$

$$= \hat{y}_j - 1$$

Finalmente

$$\frac{\partial L(y,\hat{y})}{\partial z_j} = \hat{y}_j - y_j$$

2.4. Solución

Dada la función de costo

$$L(y, \hat{y}) = -\sum_i y_i \log(\hat{y}_i)$$

con

$$\frac{\partial L(y, \hat{y})}{\partial z_j} = \hat{y}_j - y_j$$

donde $\hat{y} = \text{softmax}(W_2 h + b_2)$, de manera que

$$\frac{\partial L(y, \hat{y})}{\partial z_j} = \text{softmax}(W_2 h + b_2) - y_j$$

Calculamos la derivada respecto a W_1.

$$\frac{\partial L(y, \hat{y})}{\partial W_1} = (\text{softmax}(W_2 h + b_2) - y_j) \frac{\partial}{\partial W_1} W_2 h$$

sea $h = \sigma(W_1 x + b_1)$, entonces

$$= (\text{softmax}(W_2 (\sigma(W_1 x + b_1)) + b_2) - y_j) \frac{\partial}{\partial W_1} W_2 (\sigma(W_1 x + b_1))$$

Dado que softmax se mantiene invariante a traslaciones constantes

$$= (\text{softmax}(W_2 \sigma(W_1 x + b_1)) - y_j) \frac{\partial}{\partial W_1} W_2 (\sigma(W_1 x + b_1))$$

Sabemos que dado un escalar x, el gradiente de la función sigmoide es $\sigma(x)(1 - \sigma(x))$, por lo tanto

$$= (\text{softmax}(W_2 \sigma(W_1 x + b_1)) - y_j) W_2 \sigma(W_1 x + b_1) (1 - \sigma(W_1 x + b_1)) \frac{\partial (W_1 x)}{\partial W_1}$$

Finalmente

$$\frac{\partial L(y, \hat{y})}{\partial W_1} = (\text{softmax}(W_2 \sigma(W_1 x + b_1)) - y_j) W_2 \sigma(W_1 x + b_1) (1 - \sigma(W_1 x + b_1)) x$$

Calculamos la derivada respecto a W_2.

$$\frac{\partial L(y, \hat{y})}{\partial W_2} = (\text{softmax}(W_2 h + b_2) - y_j) \frac{\partial}{\partial W_2} W_2 h$$

$$= (\text{softmax}(W_2 h + b_2) - y_j) h$$

Finalmente, dado que softmax se mantiene invariante a traslaciones constantes

$$\frac{\partial L(y, \hat{y})}{\partial W_2} = (\text{softmax}(W_2 h) - y_j) h$$

Calculamos la derivada respecto a b_1.

$$\frac{\partial L(y,\hat{y})}{\partial b_1} = (\text{softmax}(W_2 h + b_2) - y_j)\,\frac{\partial}{\partial b_1} W_2 h$$

sea $h = \sigma(W_1 x + b_1)$, entonces

$$= (\text{softmax}(W_2\,(\sigma(W_1 x + b_1)) + b_2) - y_j)\,\frac{\partial}{\partial b_1} W_2\,(\sigma(W_1 x + b_1))$$

Sabemos que softmax se mantiene invariante a traslaciones constantes, por lo tanto

$$= (\text{softmax}(W_2 \sigma(W_1 x + b_1)) - y_j)\,\frac{\partial}{\partial b_1} W_2\,(\sigma(W_1 x + b_1))$$

Por último, considerando el hecho de que el gradiente de la función sigmoide es $\sigma(x)(1-\sigma(x))$ dado un escalar x, entonces

$$= (\text{softmax}(W_2 \sigma(W_1 x + b_1)) - y_j)\,\sigma(W_1 x + b_1)\big[W_2\,(1 - \sigma(W_1 x + b_1))\,\big]\,\frac{\partial}{\partial b_1} b_1$$

$$\frac{\partial L(y,\hat{y})}{\partial b_1} = (\text{softmax}(W_2 \sigma(W_1 x + b_1)) - y_j)\,\big[W_2\,(1 - \sigma(W_1 x + b_1))\,\big]$$

Calculamos la derivada respecto a b_2.

$$\frac{\partial L(y,\hat{y})}{\partial b_2} = (\text{softmax}(W_2 h + b_2) - y_j)\,\frac{\partial}{\partial b_2} b_2$$

$$= \text{softmax}(W_2 h + b_2) - y_j$$

Finalmente

$$\frac{\partial L(y,\hat{y})}{\partial b_2} = \text{softmax}(b_2) - y_j$$

3. Problema 3

Considera de nuevo los textos de transcripciones de las conferencias matutinas de la presidencia de México que usaste en la tarea 3. En este ejercicio implementarás un método de análisis de tópicos mediante un algoritmo de clustering.

a) Usando un vocabulario obtenido mediante los textos por *semana*, obtén las representaciones de las palabras usando word2vec pre-entrenado en español. En éste *espacio semántico* obtén un modelo de tópicos usando Fuzzy k-means, eligiendo el tamaño adecuado de k. Representa cada tópico mediante un word cloud usando la probabilidad máxima como criterio para elegir las palabras más representativas de cada tópico. ¿Puedes asignar un 'nombre' representativo de cada tópico? ¿Qué diferencias notas respecto a lo que obtuviste con la representación TF-IDF?

b) Como en la tarea anterior, considera cada una de las conferencias del presidente durante los años del estudio como tus 'documentos'. Obtén la representación vectorial respectiva calculando el *promedio* de los embeddings de las palabras que componen cada uno de ellos. Posteriormente, obtén la asignación de cada documento en su tópico correspondiente usando el modelo ajustado en el inciso anterior. Usa visualizaciones de baja dimensión basadas en PCA, Kernel PCA y t-SNE de la asignación de tópicos que obtuviste y reporta los patrones y hallazgos que identifiques.

c) Construye un indicador semanal para cada uno de los k tópicos durante el periodo de estudio y repite el inciso 4e de la tarea anterior.

d) Repite los incisos anteriores usando *fastText*.

e) Realiza un reporte ejecutivo que resuma tus análisis, hallazgos y conclusiones, resaltando las ventajas, desventajas y comparación entre los diferentes métodos que usaste para analizar este conjunto de datos. Incluye sugerencias para mejorar el análisis.

CIMAT

Análisis de Tópicos en las Conferencias Matutinas del Presidente de México Utilizando Embeddings Semánticos y Algoritmos de Clustering

Centro de Investigación en Matemáticas (CIMAT) Unidad Monterrey

Mayo de 2024

Resumen

El presente estudio emplea técnicas de análisis de tópicos utilizando modelos de embeddings preentrenados en español y algoritmos de clustering para analizar las transcripciones de las conferencias matutinas del presidente Andrés Manuel López Obrador. Utilizando *Word2Vec* y *fastText* para obtener representaciones vectoriales de las palabras, se implementó K-Means para identificar tópicos semánticos en los textos. La reducción de dimensionalidad mediante PCA, Kernel PCA y t-SNE permitió visualizar los patrones de asignación de los documentos a sus respectivos tópicos. Los resultados muestran diferencias significativas en comparación con el enfoque TF-IDF, proporcionando una comprensión más profunda de los temas abordados en las conferencias.

Introducción

El análisis de tópicos en textos extensos es una tarea compleja que puede beneficiarse enormemente de técnicas avanzadas de procesamiento de lenguaje natural (NLP) y clustering. En este estudio, se emplean modelos de embeddings preentrenados como *Word2Vec* y *fastText* para obtener representaciones vectoriales de las palabras, capturando relaciones semánticas y contextuales. Posteriormente, se utiliza el algoritmo K-Means para identificar y agrupar tópicos en los textos.

Los word embeddings son representaciones vectoriales de palabras en un espacio de alta dimensión que capturan las relaciones semánticas y contextuales entre ellas. A diferencia de las representaciones tradicionales como el 'bag of words' o TF-IDF, los embeddings proyectan palabras similares en contextos similares a puntos cercanos en el espacio vectorial.

Modelos como *Word2Vec* y *fastText* son ampliamente utilizados para generar estos embeddings. *Word2Vec* aprende a partir de grandes cantidades de texto, creando vectores donde palabras que aparecen en contextos similares están más próximas entre sí. *fastText*, por otro lado, mejora *Word2Vec* al considerar subpalabras o n-gramas, permitiendo una mejor representación de palabras raras y la incorporación de información morfológica.

Estas representaciones permiten realizar tareas avanzadas de procesamiento de lenguaje natural, como la identificación de sinónimos, la detección de temas y la mejora en la precisión de modelos de clasificación de texto, al capturar matices semánticos que las técnicas tradicionales no pueden lograr.

Metodología

Algoritmo K-Means

El algoritmo K-Means es un método de clustering que agrupa objetos en k grupos basándose en sus características. El agrupamiento se realiza minimizando la suma de distancias entre cada objeto y el centroide de su grupo o cluster.

Análisis de Componentes Principales

El análisis de componentes principales (PCA) es una técnica utilizada para reducir el número de variables de un conjunto de datos, generando nuevas variables que expliquen la variabilidad contenida en el conjunto de datos minimizando la pérdida de información. En otras palabras, nos permite simplificar conjuntos de datos complejos para comprender mejor las tendencias y patrones que se ocultan en ellos. Esto se logra por medio de la generación de un conjunto más reducido de variables, conocidas como componentes principales. Es importante recalcar que existe un número óptimo de componentes principales, el cual evita la pérdida de información crucial o la generación de datos excesivamente complicados e innecesarios.

Kernel PCA

Kernel PCA es una extensión de PCA que utiliza métodos de kernel para realizar la reducción de dimensionalidad en un espacio de características no lineal, es decir, permite captar estructuras más complejas en los datos que el PCA lineal no puede.

t-SNE

t-SNE (t-Distributed Stochastic Neighbor Embedding) es una técnica de reducción de dimensionalidad especialmente adecuada para la visualización de datos de alta dimensión. Esta técnica se enfoca en preservar las relaciones de proximidad local, lo que resulta en una visualización que refleja mejor las estructuras internas de los datos.

Resultados

Determinación Óptima del Número de Clusters para el Algoritmo K-Means

Para elegir el número óptimo de tópicos (i.e. número de clusters k) para el análisis de las conferencias matutinas del presidente de México, se utilizaron tres métodos complementarios: el gráfico de codo (elbow plot), el coeficiente de silueta (silhouette coefficient) y el dendrograma.

El elbow plot es una herramienta visual que ayuda a determinar el número óptimo de clusters observando la suma de las distancias cuadradas dentro de los clusters en función del número de clusters. El objetivo es encontrar el punto donde la disminución de la inercia comienza a ralentizarse, formando un 'codo'. Este punto indica que agregar más clusters no proporciona una mejora significativa en la variación explicada por el modelo. En este estudio, se graficó la inercia para diferentes valores de k y se identificó el 'codo' en la gráfica cuando $k = 5$, sugiriendo el número óptimo de clusters (**Figura 1**).

Por otro lado, el coeficiente de silueta mide cuán similares son los puntos dentro de un cluster en comparación con los puntos de otros clusters. Para cada punto, este coeficiente se calcula como la diferencia entre la distancia media al resto de los puntos en su propio cluster y la distancia media al cluster más cercano al que no pertenece. Se calcularon los coeficientes de silueta para diferentes valores de k y se observó el valor que maximiza este coeficiente (**Figura 1**). En este se observa el valor máximo del coeficiente cuando $k = (2, 3)$, sin embargo al tratarse de un análisis de tópicos se optó por implementar otro método que confirmara la elección del valor k.

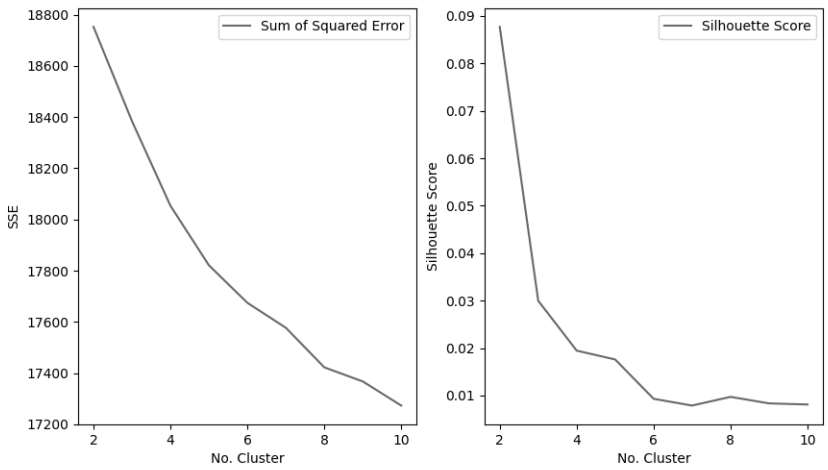

Figura 1. Métodos para seleccionar valor óptimo de clusters k: *Elbow plot* (gráfico izquierdo); *Silhouette Coefficient* (gráfico derecho).

El dendrograma es una representación gráfica utilizada en el clustering jerárquico que muestra cómo se agrupan los puntos a diferentes niveles de similitud. Aunque no se usa directamente para seleccionar el número óptimo de tópicos, proporciona una visión de las relaciones jerárquicas entre los datos. En este análisis, se construyó un dendrograma para explorar las relaciones naturales entre los puntos de datos y corroborar las elecciones de k sugeridas por los otros métodos (**Figura 2**).

Con base a los gráficos generados, se optó por utilizar un valor $k = 5$. Esto gracias a los resultados obtenidos por el gráfico de codo, los cuales fueron coherentes con lo observado en el dendrograma.

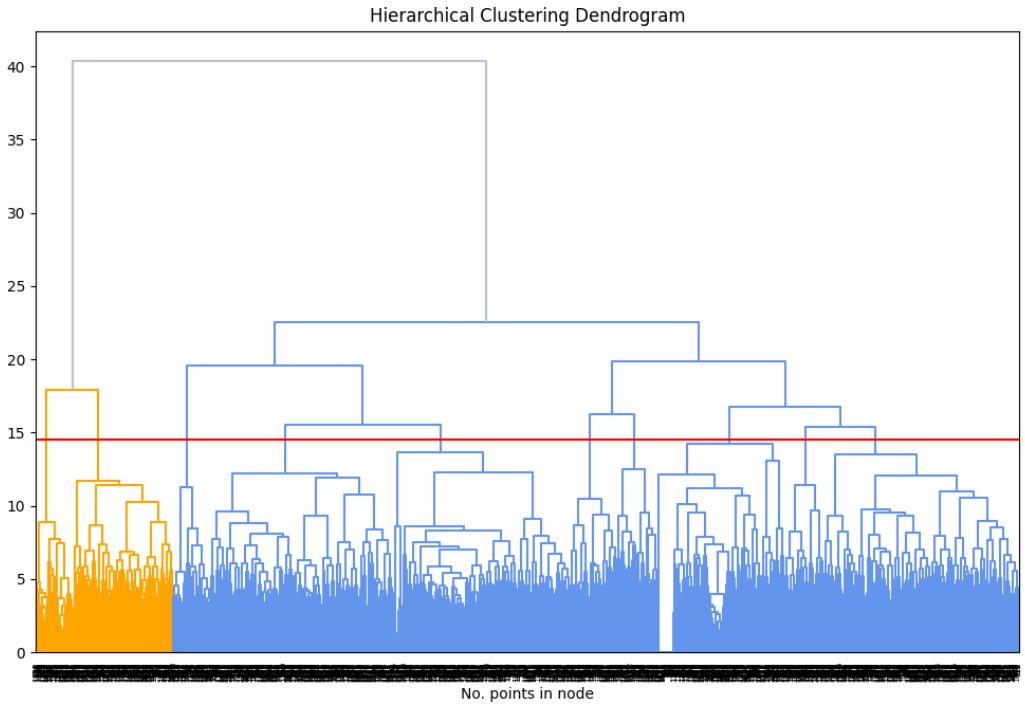

Figura 2. Métodos para seleccionar valor óptimo de clusters k: Dendrograma.

Representaciones obtenidas a partir de un modelo pre-entrenado *Word2Vec*.

Los tópicos obtenidos con *word2vec* mostraron una mayor coherencia semántica en comparación con los obtenidos mediante TF-IDF. Sin embargo, esta mayor coherencia semántica implica que algunas palabras agrupadas tienen más relación en cuanto al contexto y significado de las palabras, y no tanto con un tema específico que las rija, como fue el cado de las representaciones obtenidas mediante TF-IDF (**Figura 3**).

Figura 3. Word clouds generados a partir de las representaciones obtenidas utilizando un modelo pre-entrenado *Word2Vec*.

Se realizaron visualizaciones para identificar patrones en la asignación de tópicos. Tanto PCA como Kernel PCA proporcionaron una separación similar de los tópicos (**Figura 4**, **Figura 5**). En cambio, t-SNE reveló patrones menos significativos, ya que no se observaron agrupaciones definidas en el gráfico (**Figura 6**).

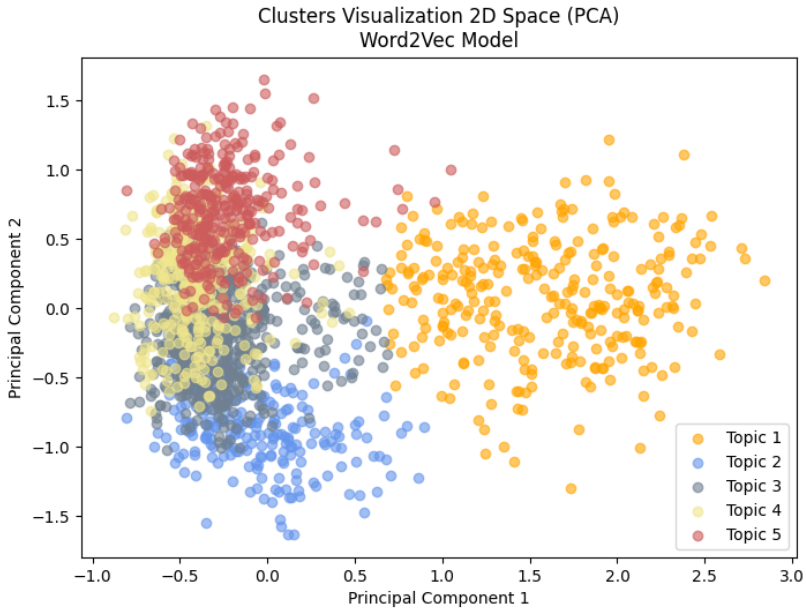

Figura 4. Asignación de documentos a sus tópicos correspondientes a partir de representaciones obtenidas utilizando un modelo pre-entrenado *Word2Vec*. Visualización basada en Análisis de Componentes Principales (PCA).

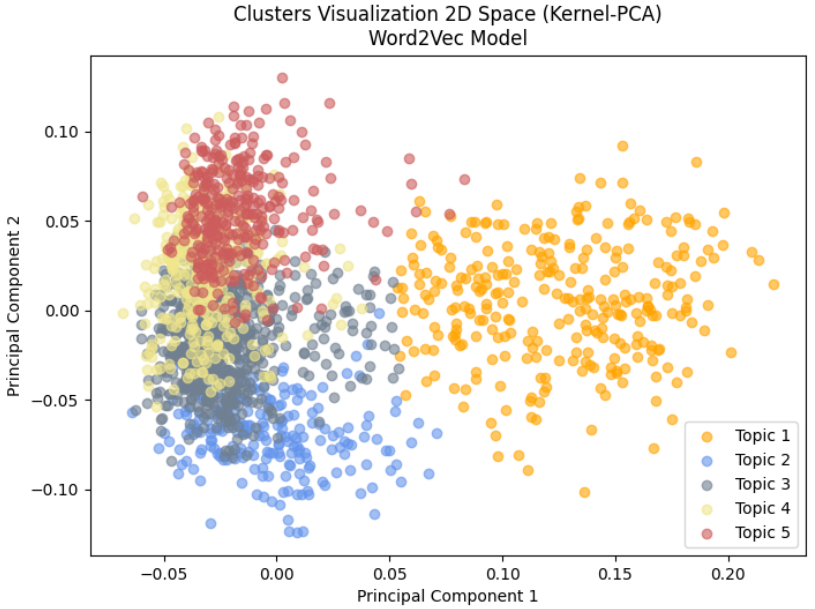

Figura 5. Asignación de documentos a sus tópicos correspondientes a partir de representaciones obtenidas utilizando un modelo pre-entrenado *Word2Vec*. Visualización basada en Kernel PCA.

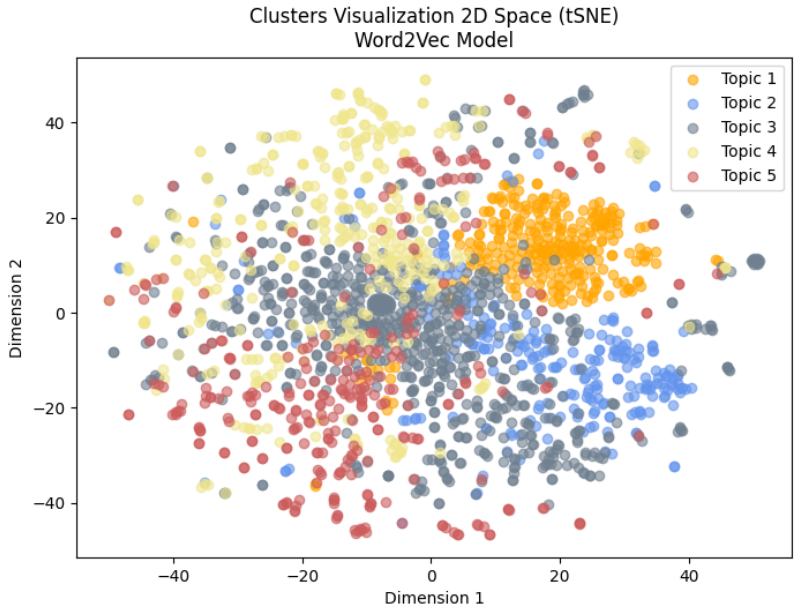

Figura 6. Asignación de documentos a sus tópicos correspondientes a partir de representaciones obtenidas utilizando un modelo pre-entrenado *Word2Vec*. Visualización basada en tSNE (*T-distributed Stochastic Neighbor Embedding*).

Representaciones obtenidas a partir de un modelo pre-entrenado *FastText*.

Se repitió la metodología anterior utilizando *fastText*, otro modelo de embeddings preentrenados en español al igual que *Word2Vec*. Se observaron resultados similares, con diferencias mínimas en cuanto a la distinción de lios tópicos. Al igual que el caso anterior, las palabras agrupadas tienen más relación en cuanto al significado y tipo de palabras, y no tanto por un tema en particular, como es el caso del word cloud para el tópico número 5 el cual agrupa los nombres de los estados de la república mexicana (**Figure 7**).

Figura 7. Word clouds generados a partir de las representaciones obtenidad utilizando un modelo pre-entrenado *FastText*.

De manera similar, se realizaron visualizaciones para identificar patrones en la asignación de tópicos. Se obseran resultados similares que los reportados utilizando el modelo *Word2Vec* Tanto PCA como Kernel PCA proporcionaron una separación clara y similar de los tópicos (**Figura 4**, **Figura 5**). En cambio, t-SNE reveló patrones diferentes, sin embargo en este caso utilizando el modelo *fastText* se observa una distinción precisa de los tópicos, caso contrario al obtener las representaciones utilizando

el modelo *Word2Vec* (**Figura 6**).

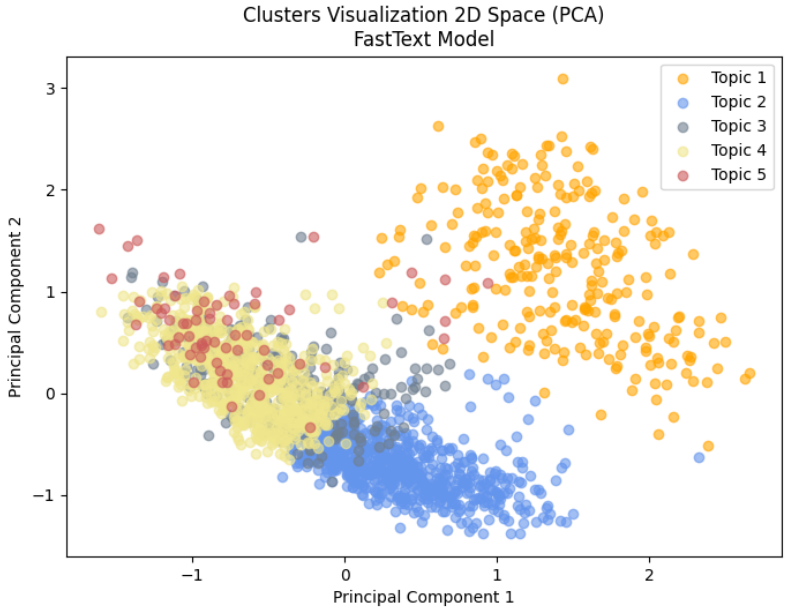

Figura 8. Asignación de documentos a sus tópicos correspondientes a partir de representaciones obtenidas utilizando un modelo pre-entrenado *FastText*. Visualización basada en PCA.

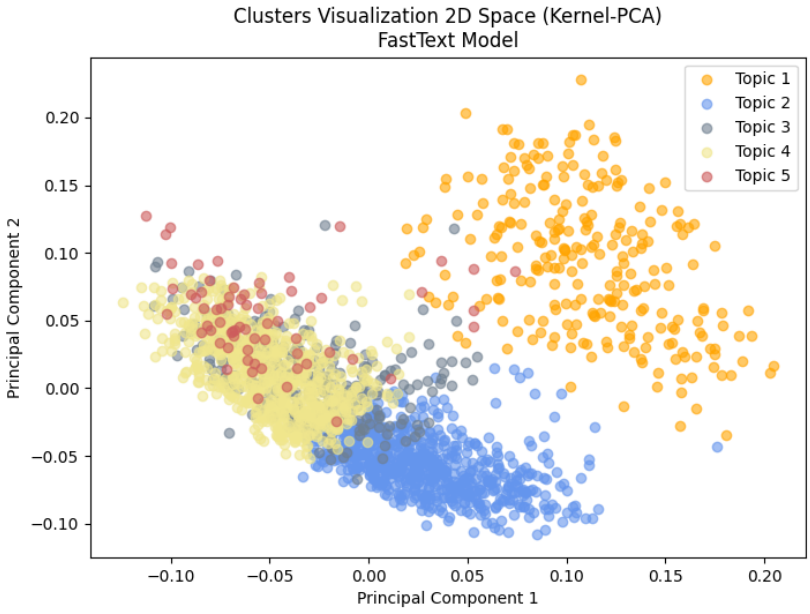

Figura 9. Asignación de documentos a sus tópicos correspondientes a partir de representaciones obtenidas utilizando un modelo pre-entrenado *FastText*. Visualización basada en Kernel PCA.

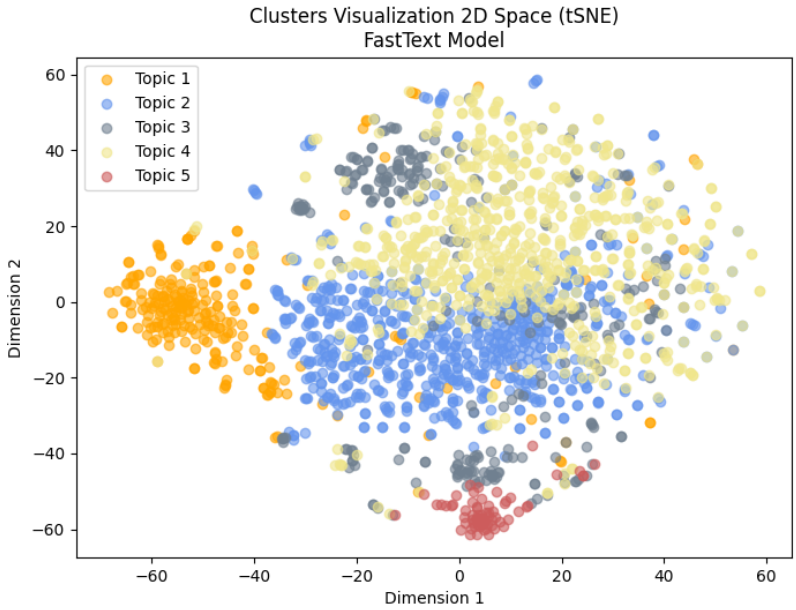

Figura 10. Asignación de documentos a sus tópicos correspondientes a partir de representaciones obtenidas utilizando un modelo pre-entrenado *FastText*. Visualización basada en tSNE (*T-distributed Stochastic Neighbor Embedding*).

Conclusiones y Recomendaciones

El uso de modelos de embeddings pre-entrenados brinda otro enfoque para la identificación de tópicos. En comparación con el enfoque TF-IDF, los word clouds generados parecen agrupar las palabras en base a relaciones semánticas más coherentes. Sin embargo, a diferencia de los resultados obtenidos mediante TF-IDF, las agrupaciones parecen estar ligadas a cuestiones de semántica y sintaxis y no a un tema en particular.

Los modelos de clustering y las técnicas de reducciióón de dimensionalidad proporcionaron una visualización clara de la asignación de los documentos a sus respectivos tópicos. Ambos enfoques asignan de manera clara cada documento con su respectivo tópico, mostrando una clara distinción entre los tópicos. A diferencia de PCA y Kernel PCA, t-SNE muestra patrones que no son fáciles de percibir por medio de gráficos en dos dimensiones.

De manera general, la combinación de un método de clustering, en este caso K-Means con embeddings de palabras permitió captar la superposición y la relación entre los diferentes tópicos.

Se sugiere realizar análisis comparativos adicionales para contrastar distintos métodos de clustering a fin de determinar el algoritmo adecuado para esta tarea. La aplicación de este análisis a otros periodos o incluyendo otros textos relevantes podría ser de relevancia acádemica, pública y privada, esto al tratarse de una metodología que permite la clasificación de documentos en base a los tópicos generados. Por último, utilizar los resultados para informar a los encargados de comunicación y estrategia del gobierno sobre los temas más relevantes y su evolución.