TECNIACUSTICA ELCHE · 2022

53º CONGRESO ESPAÑOL DE ACÚSTICA
XII CONGRESO IBÉRICO DE ACÚSTICA

ACERCANDO LOS AUTOCODIFICADORES VARIACIONALES AL GRAN PÚBLICO

PACS: Acoustical Neural Networks, Artificial intelligence, Musical sounds, analysis, synthesis, and processing

Cámara Largo, Mateo José
Blanco Murillo, José Luis

Information Processing and Telecommunication Center (IPTC)
ETSI de Telecomunicación (ETSIT), Universidad Politécnica de Madrid (UPM)
Avda. Complutense 30, 28040 Madrid, 910672388
{mateo.camara, jl.blanco}@upm.es

Palabras Clave: autocodificadores, audio, espacio latente, interacción, TensorFlow.

ABSTRACT. Sound is a fundamental asset in audiovisual creation that requires meticulous treatment to match consumers' expectations. Artists have at their disposal numerous databases and records. However, in practice it is impossible to store all this variability.
Variational Autoencoders (VAEs) are generative systems that condense information into a small vector of latent variables. Modifying this vector provides new observations (sounds) that resemble references without the need to incorporate them into databases and display differences that match desired effects. We present a web platform that connects via API to Freesound and is hosted entirely in the browser (without any backend), which allows work and exploration of the latent space of any VAE oriented to generative audio.
This contribution reports on the design and evaluation of a solution for the exploration of the latent space starting from the two-dimensional plane in which users interact. We address expanded representations for placing new sounds within a compact area of the latent space that ensures high-quality generation. The results have been included in our online demonstrator.

RESUMEN. El sonido es un activo fundamental en la creación audiovisual que requiere de un tratamiento minucioso para ajustarse a las expectativas de los consumidores. Los artistas tienen a su disposición numerosas bases de datos y gran número de registros. Sin embargo, en la práctica resulta imposible manejar toda esta variabilidad.
Los Autocodificadores Variacionales (VAEs) son sistemas generativos que condensan la información en un pequeño vector de variables latentes. Modificar este vector da lugar a nuevas observaciones (sonidos) que se asemejan a las referencias e incorporan rasgos propios de los efectos deseados, sin necesidad de agregarlas a las bases de datos y con claras diferencias.
Presentamos una plataforma web que conecta mediante la API de Freesound y se aloja íntegramente en el navegador (sin ningún *backend*), que permite trabajar y explorar el espacio latente de cualquier VAE orientado al audio generativo.
La contribución se centra en el diseño y evaluación de una solución para la exploración del espacio latente partiendo de la interactúan los usuarios. Se evalúan representaciones que permiten situarnos en una zona compacta del espacio latente que asegure buenos resultados en generación. Los resultados se incluyen en un demostrador online.

1. INTRODUCCIÓN

A lo largo de los años las tecnologías de representación, análisis y síntesis en acústica y audio han ido ofreciendo nuevas aplicaciones aprovechando las capacidades de últimas tecnologías. Su complejidad ha ido en aumento y cada vez se centra más en el tratamiento matemático de

los datos. Esto las aleja de sus usuarios habituales. En las últimas décadas, los esquemas de aprendizaje profundo han revolucionado la Inteligencia Artificial. Hoy reconocemos el riesgo que supone que sus usuarios se sientan incapaces de aproximarse a las tecnologías, y trabajamos en su *divulgación*, *interpretabilidad* y *explicabilidad*.

Este trabajo pretende acercar al público a los entresijos de una de las herramientas más utilizadas para representar la variabilidad de un conjunto de datos y sintetizar nuevas muestras de forma generativa: los autocodificadores variacionales (VAEs, *Variational AutoEncoders*).

A muy alto nivel, los autocodificadores (Aes) codifican los registros de audio en base a unas variables que definen un nuevo espacio de representación: el espacio latente, e interpretan estas nuevas representaciones, reconstruyendo los registros a partir de aquellas. En el caso de los esquemas variacionales, los VAEs logran, además, generar nuevos registros consistentes con los datos de entrenamiento a partir del muestreo de nuevos puntos en el espacio latente. Esta singular capacidad generativa los convierte en herramientas extremadamente potentes para gestionar y complementar la enorme cantidad de datos disponibles en las grandes bases de datos de audio, apoyándose únicamente en la información acústica de los registros.

Los VAEs se cuentan hoy entre las herramientas que en estos últimos años han revolucionado los sistemas representación, análisis y síntesis. Se han publicado innumerables trabajos y revisiones que se sirven de estas estructuras para la síntesis de sonidos [1], de efectos [2], música [3], incluso para la síntesis de voz [4], la anonimización de hablantes [5], la transformación de voces [6] o la detección de emociones [7], por mencionar algunas. Todos estos trabajos subrayan su extraordinaria capacidad generativa. Pero más allá de que su uso individual sea cada día más extendido [8], es habitual encontrarlos dentro de esquemas más complejos aprovechando su extraordinaria capacidad para representar la variabilidad subyacente y generalizar resultados centrándose únicamente en los datos de partida [9].

Contrariamente a lo que sucede en cuanto a la representación y generación, la calidad de sus resultados en audio está todavía muy lejos de los reportados en el procesado de imagen [12]. La calidad parece depender de la representación de entrada [10], de la estructura interna del autocodificador y de sus hiperparámetros [11]. La reconstrucción y generación de registros de calidad plantea retos importantes, derivados del mayor grado de exigencia que impone el oído frente al de la vista. Resulta desarrollar herramientas que faciliten el proceso de evaluación de los registros trabajando indistintamente en el espacio de las señales y en el espacio latente.

Toda vez que los VAEs han alcanzado un nivel de calidad, comparable al de otras técnicas, podemos centrarnos en el espacio latente. Por un lado, la calidad del modelo de síntesis de aquellos VAEs que reconstruyen registros sonoros puede cuantificarse automáticamente para comparar registros reconstruidos con sus referencias -por ejemplo, PEAQ [13], POLQA [14] o ViSQOL [15]. Por otro, la estructura del VAE facilita la interacción directa con su espacio latente, lo que facilita su estudio. Aprovechando la estructura del espacio latente del VAE, podemos emplear estos esquemas de evaluación y otras métricas como las basadas en la similaridad espectral [16] para cuantificar las diferencias cuando el VAE genera nuevos sonidos y navegar alrededor de distintas referencias sonoras a través del espacio latente.

En este trabajo nos centramos en facilitar la interacción con el espacio lantente a través de una plataforma diseñada y desarrollada por los autores que permite trabajar con VAEs programados con TensorFlow. La plataforma opera íntegramente desde el navegador *web* y sin necesidad de ningún tipo de backend [17]. En este trabajo buscamos hacerla útil para el gran público, facilitando una interacción sencilla con los elementos más profundos del sistema y unaa exploración guiada de las representaciones. Los usuarios pueden trabajar con modelos preentrenados o sus propios modelos, navegar por el espacio latente y evaluar los resultados.

El resto del trabajo se ha estructurado como sigue. La sección 2 presenta formalmente el autocodificador variacional, hace un repaso al estado del arte, los retos que el espacio latente, y su exploración. En la sección 3 se describen los materiales empleados y en la 4 los experimentos sobre la evaluación de los audios generados y en la exploración del espacio latente. Los resultados incluyen en la sección 5 y las conclusiones en la sección 6.

2. ESTADO DEL ARTE

2.1 VAEs y espacio latente

Formalmente, el VAE es una arquitectura de aprendizaje profundo que transforma los datos de entrada en puntos de un nuevo espacio de variables latentes, y de allí de nuevo en las señales de entrada. En audio y acústica estas observaciones de entrada y de salida, $x \in R^E$, son, en general, espectrogramas o representaciones similares basadas en el análisis frecuencial de las secuencias a lo largo del tiempo. Los vectores codificados del espacio latente, $z \in R^D$, son de una dimensionalidad mucho menor, $D \ll E$, lo que convierte al espacio latente en el verdadero cuello de botella de esta arquitectura y nodo de concentración de la información.

El VAE se construye en una única fase siguiendo los dos mismos pasos que todos los autocodificadores: la codificación, $p_\theta(z|x)$, y la decodificación, $p_\theta(x,z) = p_\theta(x|z) \cdot p_\theta(z)$. Ambas emplean el mismo juego de parámetros, θ, que concentra todas las variables del espacio latente. El modelo del decodificador genera nueva información condicionada por el espacio latente, mientras que la estructura compacta del codificador se obtiene por la regularización sobre una distribución *a priori*. Esta última suele hacerse coincidir con una distribución normal estándar multivariada, $N(z; 0, I_D)$, con I_D la matriz identidad de dimensiones $D \times D$. Empleando esta distribución los coeficientes del espacio latente tienden de manera natural a números similares, mientras que los vectores son ortogonales. En este nuevo espacio, aquellas posiciones que están más próximas (en distancia, típicamente euclídea) corresponden a señales que se parecen entre sí. De forma natural el autocodificador variacional pueda representar variabilidad y generarla sin necesidad de marcadores semánticos. A pesar de esto, la capacidad de reflejar diferencias semánticas a partir de las distancias en el espacio latente ofrece grandes posibilidades y ha motivado el uso del VAE en múltiples aplicaciones.

En la decodificación, se considera una distribución *a posteriori* con una media no restringida y una varianza prefijada por una distribución normal multivariada subyacente. A pesar de las referencias a la distribución gaussiana, no se ha comprobado que ésta sea la más apropiada. Además, resulta imposible de tratar en forma completa, sirviéndonos de aproximaciones a un modelo paramétrico, $q_\varphi(z|x) = N(z; \tilde{\mu}_\varphi(x), \tilde{\sigma}_\varphi(x))$, donde $\tilde{\mu}_\varphi(x) \in R^D$ y $\tilde{\sigma}_\varphi(x)$ son salidas controladas por φ. El entrenamiento de la red se centra en evaluar los valores de esta media y estas varianzas, respectivamente. Esto se logra al adaptar la mínima cota de $\log p_\theta(x)$, calculada sobre el conjunto de entrenamiento *x*. La literatura ofrece prueba de que la distribución marginal exacta para un único vector de entrada no puede ser calculada. Sin embargo, la menor cota variacional (*Variational Lower Bound*, *VLB*) puede maximizarse mediante el método de propagación y la optimización basada en el gradiente:

$$L(\varphi, \theta, x) = E_{q_\varphi(z|x)}[\log p_\theta(x|z)] - \beta \cdot d_{KL}\left(q_\varphi(z|x)|p_\theta(z)\right) \quad (1)$$

d_{KL} representa la diferencia de Kullback-Leibler, y β es un parámetro que controla el aprendizaje. El primero de los términos se centra en la precisión en la reconstrucción de la media. El segundo término caracteriza el regularizador que lleva a la distribución a aproximarse continuamente al *a priori* $p_\theta(z)$. En la forma básica de un VAE tenemos que $\beta = 1$; mientras que su generalización a través del β-VAE permite adaptar con la tasa de aprendizaje y se ha generalizado en los últimos años. Empleamos los métodos de Montecarlo ante la imposibilidad de dar un tratamiento analítico al término de reconstrucción [19, 11]..

En este trabajo empleamos implementaciones estándar del β-VAE. Nuestro objetivo es acercar estas arquitecturas al público y facilitar la interacción directa con el espacio latente. Por tanto, no cabe innovar en cuanto a las estructuras. Nos centraremos en los resultados que ofrece el VAE, tanto a los vectores de variables latentes como a su decodificación, interpretándolo como un esquema de codificación -decodificación dado por una transformación *f* tal que podemos:

Codificar: $z = f(x)$, con x la representación de entrada del registro codificado en z.
Decodificar: $\hat{x} = f^{-1}(z)$, con $\hat{x}$ la representación reconstruida a partir del vector z.

En general, sabemos que $\hat{x} \approx x$, pero no igual; y que dos entradas diferentes, $x_1 \neq x_2$ deberían arrojar codificaciones distintas, $z_1 \neq z_2$. Podemos evaluar estas diferencias tanto en el espacio de partida como en el espacio latente. En primera instancia consideraremos el entrenamiento y la calidad de sus resultados monitorizando la reconstrucción (MAE, *Mean absolute* Error):

$$\|\varepsilon_x\|^2 = \|x - \hat{x}\|^2 \text{ cuando } \|\varepsilon_z\|^2 = \|z - \hat{z}\|^2 = 0 \quad (2)$$

El entrenamiento minimiza este error cuya métrica está ligada al tamaño del espacio latente. Uno grande es más lento en entrenamiento y ejecución, requiere más datos.

Podemos evaluar el impacto del error en la calidad percibida. Se comprueba que ésta aparece condicionada por la de entrenamiento, pero no igual. En estos trabajos empleamos métricas de calidad (*Quality Metric, QM*) basadas en referencia, obteniendo una valoración (o *score*):

$$s = QM(x, \hat{x}) = QM\left(x, f^{-1}(z)\right) \quad (3)$$

En segunda instancia, podemos utilizar estas mismas métricas para evaluar las diferencias entre los audios de referencia anteriores y las variaciones generadas por el VAE en el entorno de aquellos. Finalmente, esta misma métrica, u otras que sigan la misma lógica, basadas, por ejemplo, en la similaridad (*Similarity Metric*, *SM,* [16]), pueden emplearse para cuantificar las diferencias entre las distintas referencias y puntos arbitrarios del espacio latente.

Resulta particularmente ilustrativo circunscribirnos al entorno de los datos. La Figura 1 ilustra esta idea. Se observa una serie de registros (puntos) sobre un plano proyectado cualquiera del espacio latente de un VAE. Incluimos distintas clases (en distinto color) para evidenciar cómo se organiza el espacio. Identificamos zonas de alta densidad, donde recaen muchas de las observaciones, y zonas de baja densidad, donde disponemos de pocos datos. En las primeras, la capacidad de representación y generalización está garantizada. La ausencia de información en las segundas ofrecerá peores prestaciones. Estamos interesados en navegar las primeras.

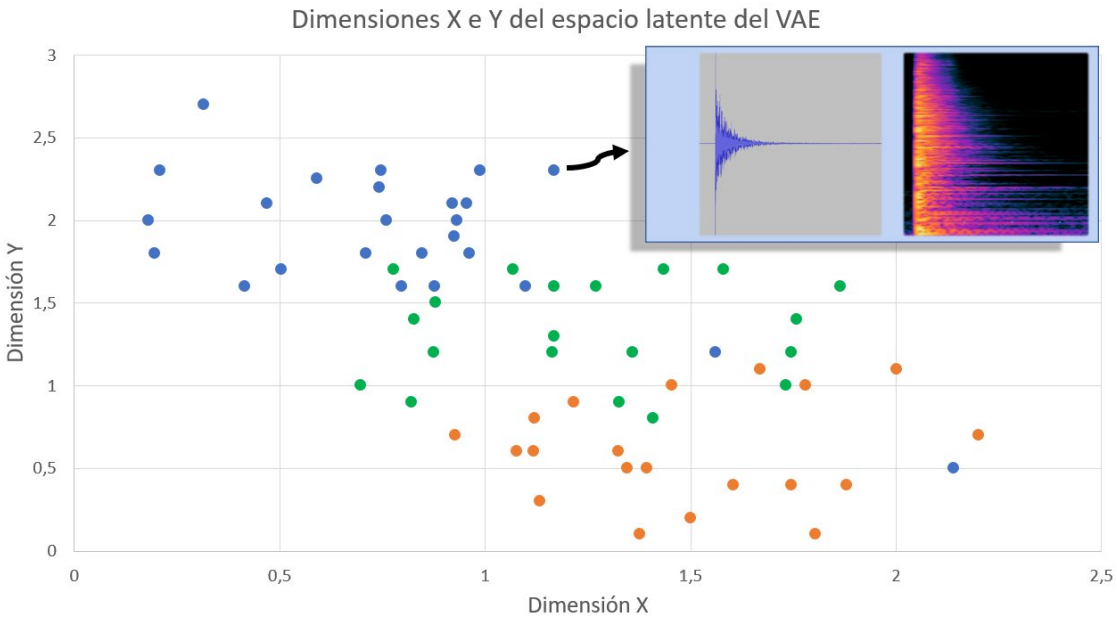

Figura 1 – Representación del espacio latente en dos dimensiones arbitrarias. Los datos tienden a concentrarse en virtud del grado de similaridad entre las representaciones.

En suma, las métricas nos permiten analizar diferencias a medida que exploramos el espacio latente en: (a) reconstrucción (entrenamiento) o generación (exploración), $\varepsilon_z = 0$, como en (b) el entorno de un registro, $\|\varepsilon_z\| \leq e$, y (c) al alejarnos de éste, $\varepsilon_z = z_1 - z_0$ (esto es, $z_0' = z_1$).

$$z_0 = f(x_0) \rightarrow x_0' = f^{-1}(z_0'),\ z_0' = z_0 + \varepsilon_z \quad (4)$$

2.2. Audio Intellimixer

La herramienta web *Audio Intellimixer* permite interactuar directamente con el espacio latente. Extiende un proyecto anterior de Audio Commons puramente demostrativo "*Timbral Explorer*". Aquel ofrece una interfaz que representa, en un espacio de dos variables, la posición de registros descargados de Freesound a través de su API [18]. La herramienta Intellimixer generaliza esta interacción. Permite a los usuarios conectar su propio VAE (o uno por defecto), seleccionar las dimensiones, codificar y representar los registros de Freesound y generar nuevos en las regiones intermedias. Esto equivale a muestrear el espacio latente. Todo esto además de visualizar los espectrogramas, escucharlos y descargarlos.

La Figura 2 muestra una captura de la interfaz de usuario. La arquitectura aparece descrita en la Figura 3. La herramienta se conecta automáticamente con Freesound a través de su API, descarga registros de prueba y un modelo VAE. Codifica los registros en el espacio latente y señala su posición sobre el plano analizado. Seguidamente, permite a los usuarios generar audios correspondientes a puntos del espacio latente. Ya sea recuperando audios originales (sin reconstrucción) o sintetizando (decodificando) nuevos. Es interesante ver las similitudes entre las Figuras 1 y 2 en cuanto al posicionamiento de los puntos en el espacio latente.

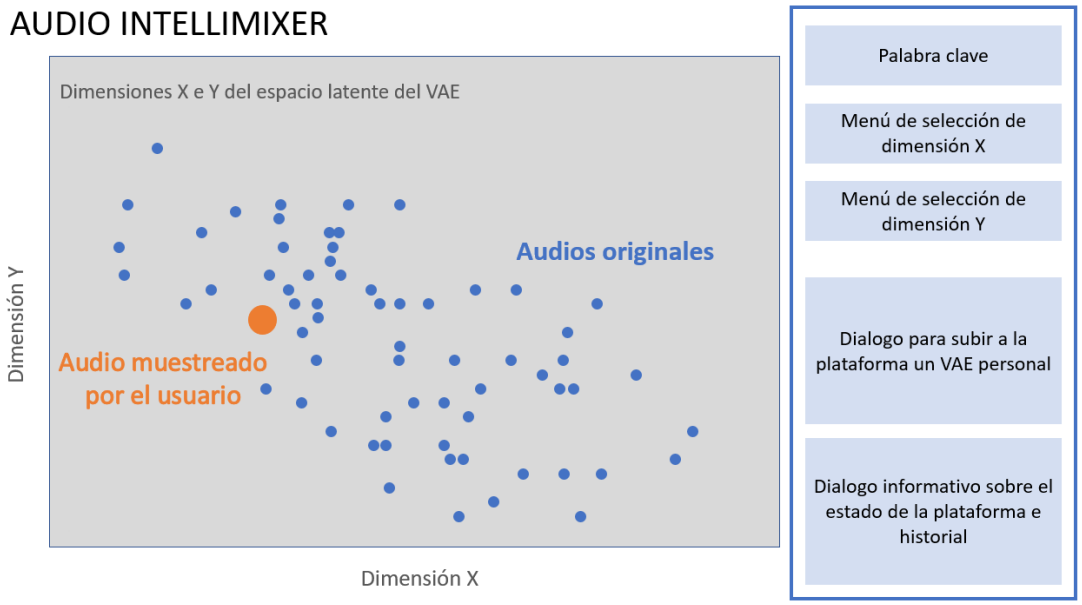

Figura 2 – Captura del interfaz de usuario (UI) de la herramienta *Intellimixer*. Integra los desarrollos descritos en este trabajo para la interacción con el espacio latente y su exploración.

En la Figura 3se identifican los módulos de descarga y cálculo, para preparar las entradas del VAE, la síntesis de la salida, así como la conexión con Freesound y la interfaz gráfica.

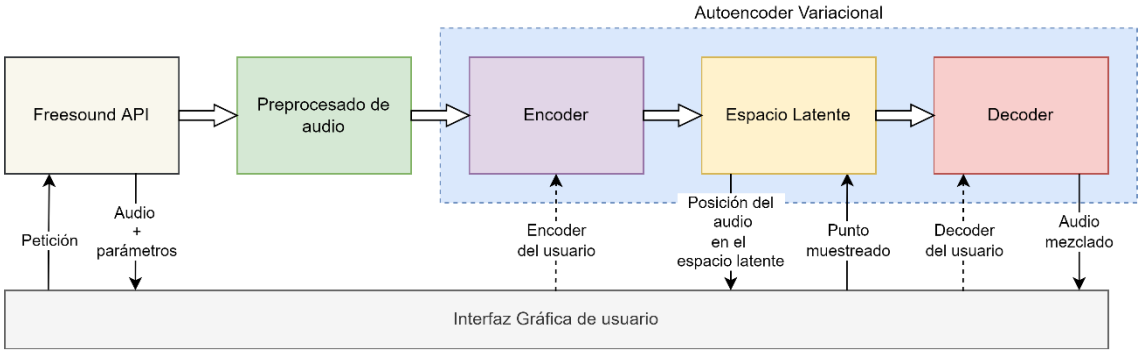

Figura 3 – Arquitectura *software* de la herramienta *Intellimixer*. La implementación evita cualquier tipo de *backend*, ejecutándose íntegramente en el navegador del usuario.

El reto radica en facilitar la interacción de los usuarios en el espacio latente. Seleccionado un punto del plano bidimensional mostrado en la herramienta, se generará un registro consistente acústicamente con esta posición. Esto requiere definir un expansor que obtenga posiciones para las dimensiones restantes consistentes con las observaciones (registros) originales. Así, para dos dimensiones, $(z[i], z[j])$, buscamos una función vectorial $g = (g_1(\cdot), g_2(\cdot), \cdots)$ tal que:

$$(z[i], z[j]) \rightarrow z = \left(g_1\left(z[i], z[j] | \{z_k\}_{k=1,\ldots,L}\right), g_2(z[i], z[j] | \cdots), \cdots, z_i, \cdots, z_j, \cdots, g_D(z[i], z[j] | \cdots)\right) \quad (5)$$

Existen multitud de formas expansoras. La implementación original de la aplicación trabaja con dos casos: (i) la asignación del vecino más próximo (NN), y (ii) la réplica de la media muestral.

NN: $g_d\left(z[i], z[j] | \{\bar{z}_k\}_{k=1,\ldots,L}\right) = z_k[d]$, d minimizando la distancia en (i, j) (6)

Media: $g_d\left(z[i], z[j] | \{\bar{z}_k\}_{k=1,\ldots,L}\right) = \frac{1}{L}\sum_{k=1}^{L} z_k[d]$, $1 \leq d \leq D$, $d \neq i$ y $d \neq j$ (7)

Estas aseguran que el vector completo, que pasará por el decodificador antes de la síntesis, se sitúa en posiciones convenientes. En el primer caso, en la posición de un registro de audio. En el segundo, las dimensiones se centran sobre la región de interés. Sus coordenadas serán consistentes, en media, con los audios originales. Por contra, no lo serán las distancias entre registros. Si seleccionamos en el espacio latente un punto próximo a otro dado, pero no igual, estaríamos asignando coordenadas potencialmente muy alejadas a las referencias. Todo depende de los datos; y a la vista del cuello de botella que establece el VAE, $D \ll E$, podría ser fatal, conllevando un mal resultado en la síntesis y una mala experiencia para el usuario. A continuación, trataremos de identificar una forma expansora consistente en distancia.

2.3. Métricas y expansores de alta dimensionalidad

La literatura recoge distintos estudios sobre métricas relativas al espacio latente. Entre otras, la distancia Euclídea [19] y su generalización a través de la matriz de Mahalanobis [20], así como la del coseno [21], más interesada en la dirección de las diferencias que en su magnitud. Las primeras están particularmente indicadas para regiones locales (poca distancia) entorno a una referencia, o para transitar de un sonido a otro (distancia más elevada). Las segundas para analizar el sentido de los cambios, con independencia de su magnitud.

Los expansores (de alta dimensionalidad) son estructuras que elevan la dimensionalidad de un vector tomando en cuenta distintas relaciones entre dimensiones. Así, el expansor más simple sencillamente replica un valor a lo largo de todas las dimensiones. Esta puede ser una opción válida si las dimensiones son independientes. Sin embargo, es habitual que los valores de los vectores estén condicionados, cuando no correlados. En ese caso, es razonable que tratemos de preservar las distancias a las referencias. En otras palabras, buscamos coordenadas que minimizan la diferencia entre las distancias en el espacio de menor dimensionalidad (el/los plano(s) de interacción con el usuario) y el espacio latente completo (de muestreo).

Supongamos conocida una serie de $D_1 \geq 2$ coordenadas de z, $\bar{z}$, y las restantes $D_2 = D - D_1$ desconocidas, $\tilde{z}$. Escribimos $z = (\bar{z}, \tilde{z})$, preservando el orden en las dimensiones del vector al realizar la concatenación. En este trabajo consideramos que la bondad de la expansión viene dada por la suavidad en las regiones donde existen observaciones [22]. Para ello evaluamos las tendencias en la calidad al variar linealmente la representación latente. Vamos a trabajar con dos representaciones complementarias. La primera basada en el centroide de dispersión entorno a las referencias, para analizar la reconstrucción de las referencias y las vecindades entorno a aquellas. La segunda centrada la preservación de la distancia.

Retícula de dispersión

Dados *L* registros de audio por sus correspondientes representaciones en un espacio latente de dimensión *D*, $\{z_i\}_{i=1,\cdots,L} \in R^D$, definimos una **retícula** centrada en un ancla $z_k \in R^D$ como:

$$z = z_k + \sum_{i=1,\ i\neq k}^{L} \alpha_{ki} \cdot (z_i - z_k) \quad (8)$$

donde $z \in R^D$ representa un punto cualquiera del espacio latente y $\alpha = (\alpha_{k1}, \alpha_{k2}, , \alpha_{kL}) \in R^L$ es una representación de este punto en las coordenadas de la retícula. En este caso, $\alpha \to \bar{0}$ cuando $z \approx z_k$, y se igualarán cuando $\alpha = \bar{0}$. Variando el ancla y ajustando los coeficientes podemos analizar las vecindades ($0 \leq |\alpha| \ll \|z_k\|^2$), o explorar el espacio (cuando $|\alpha| \approx 1$).

Operando en el espacio latente con distancia Euclídea para un punto $z \in R^D$ y un ancla $z_l \in R^D$:

$$dist^2(z, z_l) = \|z - z_l\|^2 = \sum_{i=1}^{D}(z[i] - z_k[i])^2 = (\sum_{i=1}^{D} z^2[i] + z_k^2[i]) - 2\sum_{i=1}^{D} z_k[i] \cdot z[i] \quad (9)$$

Y cuando además lo hagamos en el entorno de $z_k \in R^D$, esperamos que con $z_l \neq z_k$:

$$dist^2(z, z_k) = \left\| \sum_{\substack{i=1 \\ i \neq k}}^{L} \alpha_{ki} \cdot (z_i - z_k) \right\|^2 \approx \gamma^2 \ll dist^2(z, z_l) \quad (10)$$

donde γ representa un ruido aleatorio de media nula y potencia conocida, σ_γ^2, lo que establece un centroide de dispersión entorno al ancla. Alternativamente, podemos pensar en términos de la distancia Mahalanobis, que se generaliza con una rotación y un escalado de los vectores.

Expansor basado en multilateralización

Lamentablemente, la aproximación anterior deja de tener sentido cuando salimos de la región alrededor del ancla. En otras localizaciones afrontaremos una representación basada en las distancias a las anclas que debe asegurar, al igual que la anterior, que para un punto ancla la reconstrucción sea acorde a la calidad alcanzada durante el entrenamiento del VAE.

Las ecuaciones de multi-lateralización permiten la localización a partir de una multitud de referencias empleando ecuaciones lineales a partir de las diferencias en las distancias:

$$dist^2(z, z_k) - dist^2(z, z_l) = [\|z_k\|^2 - \|z_l\|^2] - 2\sum_{i=1}^{D}(z_k[i] - z_l[i]) \cdot z[i] = a_{kl} - b_{kl} \cdot z \quad (11)$$

donde a_{kl} es una constante y b_{kl} es un vector, ambos dependientes de las coordenadas de las referencias. Dado que querremos preservar las distancias a las referencias, buscando coordenadas que satisfagan un sistema de ecuaciones que en cada entrada incluye:

$$K \cdot [dist^2(\bar{z}, \bar{z}_k) - dist^2(\bar{z}, \bar{z}_l)] \approx [\|z_k\|^2 - \|z_l\|^2] - 2\sum_{i=1}^{D}(\tilde{z}_k[i] - \tilde{z}_l[i]) \cdot \tilde{z}[i] \quad (12)$$

Formamos $W \cdot \tilde{z} = u$, $W_{i\cdot} = -2(\tilde{z}_k - \tilde{z}_l)^T$,$u_i = [\|\bar{z} - \bar{z}_k\|^2 - \|\bar{z} - \bar{z}_l\|^2] - [\|z_k\|^2 - \|z_l\|^2]$, con $z = W^\dagger \cdot u$ con $W^\dagger = (W^{\mathrm{T}} \cdot W)^{-1} \cdot W^{\mathrm{T}}$. En general, el sistema resultante es compatible, pero su determinación dependerá de $L \cdot (L-1) \lessgtr 2 \cdot D_2$, con L anclas disponibles. Para $D = 50$ y $D_1 = 2$, bastan $L \leq 11$ anclas. La solución a las ecuaciones se incluye en la aplicación Intellimixer. La subrutina se ejecuta cuando el usuario muestrea un nuevo audio y las anclas se han codificado.

3. MATERIALES

En este trabajo nos vamos a centrar en el modelo VAE por defecto que emplea la herramienta web *Audio Intellimixer*. Entrenamos el modelo fuera de la herramienta, empleando Python 3.8, el entorno Keras y TensorFlow en su versión 2.7.0 con aceleración por hardware mediante una tarjeta GPU NVIDIA TITAN X. Los datos empleados fueron registros de audio con sonidos de pasos de un segundo de duración [23]. La base de datos está publicada y disponible. Los registros fueron preprocesados para validar su contenido. Dividimos el conjunto de datos para entrenamiento (60%), validación (20%) y ensayo (20%). Empleamos validación cruzada para no lastrar el entrenamiento, tomando 500 épocas, con paciencia de 30 para la parada temprana del proceso, junto con el optimizador Adam [24] y una tasa de aprendizaje de 10^{-4}.

Para poder utilizar el VAE desarrollado dentro de la herramienta es necesario separar los modelos de codificación y decodificación en dos bloques para ajustarse a la arquitectura de la Figura 3. Además, no es necesario incorporar la capa Lambda para el muestreo del espacio latente, puesto que serán los usuarios utilizando la herramienta quienes realicen esta tarea.

4. EXPERIMENTOS

En este trabajo analizamos la calidad de los registros de audio sintetizados por el VAE para distintas posiciones del espacio de muestreo, generadas a partir de la herramienta Audio Intellimixer y distintos expansores. Estas posiciones se sitúan en distintas zonas del espacio latente en cada uno de nuestros experimentos. En primer lugar (subsección 4.1), revisamos el proceso de entrenamiento del VAE, y evaluamos la calidad obtenida al reconstruir los audios del conjunto de ensayo. En segundo lugar (subsección 4.2), analizamos las zonas próximas a las posiciones de las referencias. Finalmente (subsección 4.3) analizamos toda la región de interés, obviando zonas exteriores contenidas en el espacio de muestreo donde las prestaciones del VAE serán más bajas debido a la falta de datos en el entrenamiento.

Todos los experimentos emplean la retícula de la ecuación (7) y se evaluaron los tres expansores descritos anteriormente: vecino más próximo, réplica de la media, y multilateral.

4.1 Evaluación de la reconstrucción de observaciones (ε = 0)

Analizamos el MAE de reconstrucción comparándolo con los valores MOS de PEAQ y ViSQOL tanto en la media como en su dispersión, a través de la desviación típica. Estos valores nos dan una idea de la capacidad de representación del modelo entrenado y del error esperable.

4.2 Evaluación de vecindades (ε → 0)

En este experimento evaluamos las diferencias introducidas al incorporar un término de ruido aditivo a la representación latente y evaluar el resultado de su reconstrucción. De forma arbitraria se ha elegido un ruido gaussiano, de media nula y desviación típica σ_γ^2, siguiendo el esquema de la ecuación (9). Se generaron $K = 1000$ observaciones aleatorias por registro de validación, De nuevo, se calcularon el MAE y los valores MOS de PEAQ y ViSQOL.

4.3 Exploración del espacio latente

Finalmente, analizamos distintas posiciones dentro de la región de interés, desplazándonos entre pares de puntos anclas (origen, z_k, y destino, z_l), correspondientes a distintos registros del conjunto de validación. Se tomaron aleatoriamente $P = 300$ pares y se evaluaron $Q = 5$ posiciones equiespaciadas a lo largo de la correspondiente dirección del espacio latente.

$$z[p] = z_k + \frac{q}{\mathrm{Q}-1} \cdot u_{kl},\ u_{kl} = (z_l - z_k),\ 0 \le q \le (\mathrm{Q}-1) \qquad (13)$$

5. RESULTADOS

Los resultados mostrados en esta sección corresponden al VAE por defecto de la herramienta Intellimixer. Consta de 4 dimensiones en el espacio latente. Se han tomado errores sobre el subconjunto de test que se empleó en el entrenamiento del VAE. Asimismo, se exploraron los mismos experimentos en espacios latentes de mayor dimensionalidad.

La Tabla 1 resume los valores obtenidos para el MAE y niveles MOS según PEAQ y ViSQOL en la reconstrucción de los registros del conjunto de validación a partir del VAE entrenado (primera columna), así como sus valores cuando perturbamos los vectores con ruido aditivo de potencia $\sigma^2 = \|z\|^2/SNR$. Los valores son equivalentes para las distintas dimensiones, por tanto, no se incluyen. Esto sugiere que el VAE está correctamente dimensionado para el problema.

Tabla 1 – Valores obtenidos en la reconstrucción de los audios.

Métrica	Valor (media y desv. Tip.)					
	$SNR \to \infty$	$SNR = 10^2$	$SNR = 10^0$	$SNR = 10^{-1}$	$SNR = 10^{-2}$	$SNR = 10^{-3}$
MAE	9,3 ± 1,3	9,3 ± 1,3	9,6 ± 1,8	11,6 ± 2,8	15,8 ± 5,4	22,1 ± 6,4
ViSQOL	4,3 ± 0,1	4,3 ± 0,1	4,3 ± 0,1	4,2 ± 0,2	4,0 ± 0,2	3,9 ± 0,2
PEAQ	1,4 ± 0,3	1,4 ± 0,3	1,4 ± 0,4	1,4 ± 0,5	1,2 ± 0,5	1,0 ± 0,5

El error de reconstrucción aumenta en todas las métricas a medida que disminuye la relación señal a ruido (SNR). Esto es, aumenta con el ruido incorporado. Los resultados indican que pequeñas perturbaciones dan lugar a audios muy similares. Únicamente con un ruido elevado, a partir de SNR = 10^{-1}, podemos diferencias relevantes. Además, los valores obtenidos en la reconstrucción son consistentes con lo que describe la literatura para los VAEs [23].

La Tabla 2 resume los valores para los tres expansores con las mismas métricas anteriores según (13). Cuando nos acercamos a los extremos (q=0 para ancla 1, q=4 para el ancla 2) los resultados son análogos a los de la Tabla 1. En el caso del NN, los extremos son exactamente iguales a los originales (marcados con *). Por tanto, no se observa desviación alguna.

Tabla 2 – Valores obtenidos en la exploración del espacio latente.

Ancla de referencia	Métrica	Expansor	Valor (media y desv. Tip.)				
			$q = 0$	$q = 1$	$q = 2$	$q = 3$	$q = 4$
Ancla 1 (origen)	MAE	NN	0*	4,1 ± 1,3	4,5 ± 1,9	4,0 ± 1,3	0*
		Media	7,4 ± 1,7	6,8 ± 2,0	6,8 ± 2,0	6,8 ± 2,1	7,1 ± 2,1
		Multilat.	10 ± 2,0	9,5 ± 2,3	9,3 ± 2,5	9,5 ± 2,3	9,9 ± 2,0
	PEAQ	NN	5,0*	4,0 ± 1,2	4,3 ± 1,2	4,5 ± 0,9	5,0*
		Media	3,5 ± 0,5	3,3 ± 1,7	3,9 ± 1,6	4,1 ± 1,5	4,3 ± 1,4
		Multilat.	3,9 ± 0,1	3,1 ± 1,9	3,7 ± 1,8	4,0 ± 1,7	4,2 ± 1,5
	ViSQOL	NN	4,7*	4,5 ± 0,2	4,5 ± 0,2	4,5 ± 0,2	4,7*
		Media	4,2 ± 0,3	4,3 ± 0,3	4,3 ± 0,2	4,3 ± 0,3	4,2 ± 0,3
		Multilat.	4,0 ± 0,3	4,0 ± 0,3	4,1 ± 0,3	4,0 ± 0,3	4,0 ± 0,3

Todo ello es consistente con el modelo en (13) para generar los vectores. Se observa como el expansor propuesto para la exploración del espacio latente mantiene una tendencia suave en todas las métricas consideradas. Esto es consistente con el hecho de que el vecino más próximo tiende a sesgar hacia un extremos, y que el conjunto de datos analizados presenta una distribución multimodal, por lo que la media no ofrece resultados convenientes.

Los lectores interesados pueden dirigirse a la web de *Audio Intellimixer* para escuchar algunos de los audios generados e interactuar con el VAE. Los oyentes experimentarán grandes diferencias en función del método empleado. El vecino más próximo dará un sonido claro, y sin embargo idéntico al que encontramos en nuestra base de datos. La media genera siempre sonidos muy parecidos entre sí. La multilateralización genera un sonido novedoso parecido al ancla más cercana. Suele ser un sonido de calidad subjetiva inferior a la encontrada en la base de datos, pero ofrece un interesante punto de partida hacia la exploración del espacio latente.

6. CONCLUSIONES

Los VAEs son modelos profundos que logran excelentes resultados en representación de datos y síntesis generativa. Recientes trabajos en audio reportan resultados similares a los de técnicas consolidadas. Sin embargo, los VAEs se mantienen alejados del gran público. La aplicación web *Audio Intellimixer* se diseñó para interactuar directamente con los VAEs y su espacio latente, controlando el proceso de muestreo y facilitando la exploración.

Durante la reconstrucción de los registros de validación perturbados comprobamos que el VAE propuesto logra capturar de mantera eficiente la variabilidad presente en los registros de audio. Durante la exploración del espacio latente comprobamos que la métrica condiciona la interacción con la herramienta y la experiencia de los usuarios en cuanto a la calidad. Nuestros experimentos subrayan las diferencias en la calidad trabajando con distintas métricas, comprobando que existe relación entre aquellas y la calidad objetiva. En ocasiones, los audios resultantes se alejan bastante de los almacenados en la base de datos, sin que se disminuya su calidad. Se trata de una demostración de que el espacio latente conserva en sus dimensiones métricas subyacentes desconocidas. La capacidad de explicar e interpretar esta información subyacente es una línea de investigación activa. Nuestra aplicación pretende apoyarla y profundizar en ella aportando una herramienta más a los investigadores.

Hemos analizado tres expansores de dimensión diferentes como propuesta para explorar adecuadamente un espacio latente de un VAE (dadas dos dimensiones conocidas, en el contexto de Intellimixer). La estrategia de vecino más próximo es muy dependiente del muestreo del espacio latente. Cuando se conocen multitud de vectores ancla, la probabilidad de que un audio se esté próximo a un lugar arbitrario es elevada. No obstante, aunque es una forma correcta de analizar un espacio latente, fractura por completo la posibilidad de explorar áreas desconocidas de este. Esta situación aplica idénticamente al caso de la media, que si bien puede ser un valor desconocido en el espacio latente (ningún ancla se encuentra en ese punto), es fijo para todas las posibles exploraciones.

La estrategia que presentamos para la exploración del espacio latente es la solución a las ecuaciones de multilateralización conocidas dos de las dimensiones del espacio latente. Los resultados de la Tabla 2 apuntan a que se ha logrado identificar un método capaz de explorar eficientemente el espacio latente, de error equivalente al método más simple, y con capacidad de investigar cualquier zona conocida o desconocida a priori. No obstante, el método del vecino más próximo ha arrojado errores menores. Ello es debido a que se ha utilizado una gran base de datos, con una gran cantidad de muestras que se distribuyen a lo largo de su rango en el espacio latente. En estas circunstancias favorables, el vecino más próximo tiende a lograr un menor error. Sin embargo, no es extrapolable a toda clase de problemas y, sobre todo, no logra completar los espacios que quedaron sin observan en el espacio latente.

Actualmente, la aplicación está orientada a la investigación. Próximamente permitirá recoger información sobre el trabajo de los usuarios. Esto será fundamental cuando estos usuarios sean productores o generadores de contenidos, que, guiados por sus particulares necesidades, exploren el espacio latente buscando sonidos con características de interés. En el futuro, la aplicación debe permitir a los productores y generadores de contenidos explorar de forma nativa la variabilidad presente en el espacio latente, trabajando con representaciones eficientes, centradas en las características de su interés. La aplicación aporta un paso más hacia una transferencia eficiente y transparente al usuario de información acústica desde repositorios públicos, así como para la generación novedosa e inteligente de sonidos.

AGRADECIMIENTOS

Este trabajo ha sido financiado conjuntamente por el Ministerio de Economía y Competitividad del Gobierno de España dentro del proyecto PID2021-128469OB-I00, el Programa de Investigación e Innovación de la Unión Europea Horizon 2020 dentro del "*Grant Agreement* No. 101003750", y el Programa propio de Investigación de la Universidad Politécnica de Madrid.

REFERENCIAS

[1] Moffat, D. AI Music Mixing Systems. *Handbook of Artificial Intelligence for Music*, 2021, pp. 345–375.

[2] Roche, F.; Hueber, T.; Garnier, M.; Limier, S.; Irin, L. Make That Sound More Metallic: Towards a Perceptually Relevant Control of the Timbre of Synthesizer Sounds Using a Variational Autoencoder. *Transactions of the International Society for Music Information Retrieval (TISMIR)*, vol. 4, 2021, pp. 52–66.

[3] Roche, F.; Hueber, T.; Limier, S.; Girin, L. Autoencoders for music sound modeling: A comparison of linear, shallow, deep, recurrent and variational Models. *arXiv preprint arXiv:1806.04096,* 2018.

[4] Saito, Y.; Ijima, Y.; Nishida, K.; Takamichi, S. Non-Parallel Voice Conversion Using Variational Autoencoders Conditioned by Phonetic Posteriorgrams and D-Vectors. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5274–5278.

[5] Espinoza-Cuadros, F. M.; Perero-Codosero, J. M.; Antón-Martín, J.; Hernández-Gómez, L. A. Speaker De-identification System using Autoencoders and Adversarial Training. *arXiv preprint arXiv:2011.04696,* 2020.

[6] Lee, Y. K.; Kim, H. W.; Park, J. G. Many-to-many unsupervised speech conversion from nonparallel corpora. *IEEE Access, vol. 9*, 2021, pp. 27278–27286.

[7] Latif, S.; Rana, R.; Qadir, J.; Epps, J. Variational autoencoders for learning latent representations of speech emotion: A preliminary study. *arXiv preprint arXiv:1712.08708,* 2017.

[8] Natsiou, A.; O'Leary, S. Audio representations for deep learning in sound synthesis: A review. *2021 IEEE/ACS 18th International Conference on Computer Systems and Applications (AICCSA)*, 2021, pp. 1-8.

[9] Le Vaillant, G.; Dutoit, T.; Dekeyser, S. Improving Synthesizer Programming From Variational Autoencoders Latent Space. *Proceedings of the 24-th Int. Conf. on Digital Audio Effects (DAFx20in21)*, vol. 2, 2021, pp. 276–283.

[10] Esling, P.; Bitton, A. et al. Generative timbre spaces: regularizing variational auto-encoders with perceptual metrics. *arXiv preprint arXiv:1805.08501,* 2018.

[11] Girin, L.; Roche, F.; Hueber, T.; Leglaive, S. Notes on the use of variational autoencoders for speech and audio spectrogram modeling. *DAFx 2019-22nd international Conference on Digital Audio Effects*, 2019, pp. 1–8.

[12] Paliwal, K. K.; Alsteris, L. Usefulness of phase spectrum in human speech perception. *Eighth European Conference on Speech Communication and Technology*, 2003.

[13] Int. Telecomm. Union. ITU-R Rec. BS.1387: Method for objective measurements of perceived audio quality. 2001.

[14] Beerends, J. G. et al. Perceptual objective listening quality assessment (polqa), the third generation itu-t standard for end-to-end speech quality measurement part i—temporal alignment. *Journal of the Audio Engineering Society 61.6*, 2013.

[15] Chinen, M.; Lim, F. S. C.; Skoglund, J.; Gureev, N.; O'Gorman, F.; Hines, A. ViSQOL v3: An Open Source Production Ready Objective Speech and Audio Metric, 2020.

[16] Seyerlehner, K.; Widmer, G.; Knees, P. Frame level audio similarity-a codebook approach. *Proc. of the 11th Int. Conf. on Digital Audio Effects (DAFx-08)*, 2008.

[17] Cámara, M.; Blanco, J. L. Expanding the Frontiers of Web Audio with Autoencoders and JavaScript. *Aceptado para publicación en Journal of Audio Engineering Society,* 2022.

[18] Freesound Technical Demo. *ACM International Conference on Multimedia (MM'13)*, 2013, pp. 411–412.

[19] Saez-Mingorance, B. et al. Object Positioning Algorithm Based on Multidimensional Scaling and Optimization for Synthetic Gesture Data Generation. *Sensors 21*, no. 17, 2021.

[20] Verma, N. Distance preserving embeddings for general n-dimensional manifolds. *Conference on Learning Theory, pp. 32-1. JMLR Workshop and Conference Proceedings*, 2012.

[21] Tillquist, R. C. Low-dimensional embeddings for symbolic data science. *PhD diss.*, University of Colorado at Boulder, 2020.

[22] Bengio, Y.; Courville, A.; Vincent, P. Representation Learning: A review and New Perspectives. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 35(8), 2013, 1798-1828.

[23] Cámara, M.; Blanco, J. L. Phase-Aware Transformations in Variational Autoencoders for Audio Effects. *Aceptado para publicación en Journal of Audio Engineering Society,* 2022.

[24] Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980, 2014.*