Representaciones inspiradas en auto encoders variacionales para procesar texto a imagen

Laura Gómez Bustamante[1], **Miguel Angel Millan Dorado**[1], **and César Zamora Martínez**[1]

[1] Alumnos de Maestría en Ciencias de Datos (ITAM)

This manuscript was compiled on May 28, 2020

La generación de imágenes sintéticas a partir de descripciones de texto es un problema desafiante en el área de inteligencia artificial que tiene muchas aplicaciones prácticas. Entre sus más grandes desafíos está que las representaciones generadas capturen detalles vívidos a partir del significado de las descripciones dadas, que redundan en imágenes con mayor calidad. En este trabajo se presenta una propuesta de modelo inspiradas en auto encoders variacionales para procesar texto a imagen sobre el conjunto de datos Flickr8.

En años recientes, un problema que ha llamado la atención en el contexto de aprendizaje profundo es la posibilidad de generar información sintética de cierto tipo, que no sólo derive de modelos entrenados con datos reales, sino que además tenga una calidad comparable con los datos a partir de los cuales se alimentan tales herramientas, por ejemplo generando rostros de personas que no existen o escritos con contenido sintético. Una variante particular, es la posibilidad de usar el texto de las descripciones de ciertas imágenes para generar nuevas instancias sintéticas de estos datos.

Sobre este último punto, en los últimos años diversos grupos académicos han logrado avances importantes basándose en diferentes enfoques, por ejemplo, empleando redes generativas adversariales (GAN)*, (Reed et al. (2016b), Reed et al. (2016a)). De entre ellos destacan las Redes Adversarias Generativas apiladas *Stackgan* (Zhang et al. (2016)), tales fueron diseñadas para generar imágenes fotos realistas a partir de descripciones de texto. La idea es el diseño de un modelo capaz de generar imágenes de 256×256 condicionadas por descripciones de texto, descomponiendo el problema general en subproblemas más manejables. En concreto, la arquitectura de *Stackgan* considera dos etapas: 1) Stage-I GAN, que se refiere a un aproximación donde la red dibuja la forma primitiva y los colores básicos del objeto en función de la descripción de texto dada, produciendo imágenes de baja resolución (64×64 pixeles), y 2) Stage-II GAN, que parte de los resultados del Stage-I y las descripciones de texto como entradas, y genera imágenes de resolución de baja resolución (256×256 pixeles) con fotos que presentan matices más realistas.

De la revisión de los trabajos más recientes en esta área, el modelo implementado por Zia et al. (2020) considera una arquitectura *encoder*-decoder con mecanismos de atención. Particularmente el encoder es utilizado para mapear el texto de entrada y generar estados ocultos que son utilizados como *input* del *decoder*. La característica fundamental de este modelo se centra en modelar la dependencia entre los mencionados estados ocultos y los pixeles de las imágenes a través del llamado mecanismo de atención; lo cual ocurre en el marco del *decoder*. Este tipo de modelo está inspirado en los éxitos obtenidos a través de mecanismos de atención en tareas como traducción de textos y descripción de imágenes. Exitosamente han obtenido imágenes de (32×32 pixeles) para el conjunto de datos COCO (el cual comprende una gran diversidad de objetos, estilos y fondos), mismas que se aprecian en la figura 1. Sin embargo, los autores destacan: "...las imágenes generadas son pequeñas y en escala de grises, lo cual es un reflejo de las restricciones del modelo debidas a la complejidad computacional y a los requerimientos de menoria del GPU."

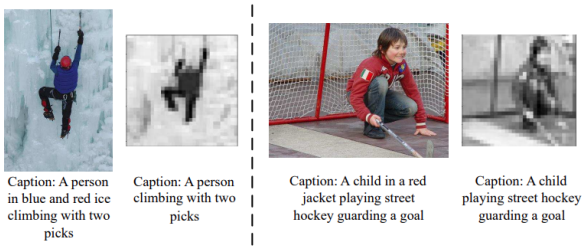

Fig. 1. Dos ejemplos del conjunto de datos COCO. Cada ejemplo contiene una imagen original con su descripción y una imagen obtenida por Zia et al. (2020) con la descripción utilizada en el entrenamiento.

En la literatura, se han encontrado hipótesis que han sido exitosas para otros problemas de aprendizaje profundo para la generación de datos sintéticos. De manera general, tales pueden ser resumidos como sigue: 1) que se teoriza la existencia de una función de encaje hacia una variedad en $\mathbb{R}^n$ cuya topología captura no solo la representación semántica de los textos, sino que permite que se preserven nociones de cercanía entre textos a través del encaje† 2) que la distribución de los datos que codifica la relación semántica entre texto e imagen puede ser capturada en términos de una distribución de probabilidad perteneciente a una familia paramétrica, 3) que los parámetros de dicha distribución de probabilidad pueden ser ajustados a través de un modelo de aprendizaje profundo, minimizando una función de pérdida específica que cuantifique la distancia entre las distribuciones de probabilidad aprendidas en el proceso de entrenamiento‡, 4) la codificación de lo anterior redunda en términos prácticos en un espacio intermedio de los modelos que se conoce como espacio latente, que al ser parametrizado como una distribución de probabilidad, puede

*Las GAN son arquitecturas que usan dos redes neuronales, compitiendo una contra la otra para imitar la distribución de los datos con los que se entrena, con el propósito de ser explotadas para generar nuevas instancias sintéticas de datos que pueden pasar por datos reales. Se utilizan ampliamente en la generación de imágenes, la generación de videos y la generación de voces. Fueron introducidas en Goodfellow et al. (2014)

†Dado que los conjuntos de datos es discreto, la función puede presentar discontinuidades

‡Estos dos incisos equivalen a un problema de calculo variacional, en el sentido en que se busca ajustar datos a partir de primeros principios y suponiendo una relación funcional sobre la que se puede hacer inferencia para determinar que función aproxima esta relación de manera óptima

emplear para simular puntos de dicha distribución y con ello generar imágenes sintéticas.

Particularmente, tales ideas se encuentran presentes en las arquitecturas de tipo auto-enconder variacional, las cuales codifican una imagen a través de un espacio latente que se encuentra asociado a los parámetros de una distribución normal multivariada, en el entendido de que el modelo subyacente aprende la función identidad referida a las imágenes que a la vez son datos de entrada y objetivo.

Al respecto, en el presente trabajo se propone un modelo que retoma las ideas que dan sustento a la arquitecturas de tipo auto-enconder variacional, para combinarlas con otras nacidas en el procesamiento de imágenes, con el objetivo de obtener un medio que permita la generación de imágenes del conjunto de datos Flickr8 (ver detalles más adelante) a partir de sus descripciones de texto (en idioma inglés). Dicha aproximación aborda imágenes de baja resolución (64×64 pixeles) en tres canales.

En este sentido, para facilitar la exposición de trabajo realizado y la literatura consultada, el documento se ha organizado como sigue; en una primera sección se aborda la descripción del problema a resolver, junto con la descripción del conjunto datos Flickr8 así como un desglose de los posibles retos y riesgos identificados para llevar a cabo dicha tarea. Posteriormente, en la segunda segunda sección, se presentan el modelo planteado para enfrentar la generación de imágenes del conjunto de datos a partir de descripciones, junto con las motivaciones teóricas para plantearlo. En una tercera sección se abordan los experimentos numéricos realizados con este modelo y la discusión de los resultados obtenidos. Finalmente, la cuarta sección reune las conclusiones generales de este proyecto, así como las lecciones aprendidas y una recopilación de los puntos que podrían explorarse en un trabajo a futuro.

1. Descripción del problema

A través de la presente sección se abordará el problema que se pretende resolver para el conjunto de datos Flickr8.

A. Conjunto de datos Flickr8. Flickr8[§] es una recopilación de datos que relaciona texto con imágenes (Hodosh and Hockenmaier (2013)). Al respecto, cada imagen está asociada con un conjunto de cinco subtítulos diferentes que describen las entidades y eventos representados en la imagen y que se recopilaron mediante *crowdsourcing*, concretamente haciendo uso del servicio de "Amazon Mechanical Turk"[¶]

Al asociar cada imagen con múltiples oraciones producidas independientemente, se tiene idea de que este conjunto de datos captura parte de la variedad lingüística que se puede usar para describir la misma imagen, en razón de que diferentes detalles en texto de la misma imagen pueden enfocarse en diferentes aspectos de la escena a utilizar.

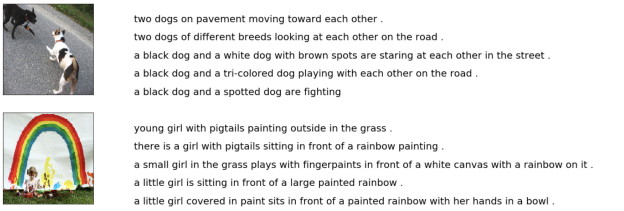

Fig. 2. Ejemplo de imágenes y texto del conjunto de datos Flickr8. Adaptado de https://tinyurl.com/y7zsjtvs

En términos generales, el conjunto de datos en comento consta de 8,091 imágenes de personas realizando actividades, y de 5 descripciones en formato texto para cada imagen. Es decir, la misma salida tendrá hasta 5 posibles entradas, con lo cual, se pueden consolidar 40,455 pares 1-a-1. En este sentido, es relevante decir que 5 personas han dado una descripción en diferentes términos de una imagen.

Cabe destacar que el propósito original para el que fue creado este conjunto de datos fue para realizar la tarea de generación de texto a partir de imágenes, donde el contenido del texto está estrechamente relacionado con éstas[‖].

B. Planteamiento del Problema. El problema a abordar es la creación de modelos basados en aprendizaje profundo que sean capaces de generar imágenes sintéticas a partir del conjunto de datos Flikr8 a partir de entradas en forma de texto que contienen descripciones específicas de tales imágenes.

De manera específica, se pretende consolidar una representación intermedia en un "espacio latente" que se encuentre ligado a los parámetros de una distribución de probabilidad, que puedan ser calibrados en el ajuste del modelo a través de la minimización de una función de pérdida; a partir de los cuales, se puedan generar imágenes sintéticas que tengan relación con el texto que le sirve de entrada.

Como se ha mencionado previamente, este es un problema complejo del área de inteligencia artificial, por lo que a continuación se describen los posibles retos y riesgos para consolidar una solución exitosa.

C. Retos y posibles riesgos para dar solución al problema. De la revisión realizada a la literatura sobre el problema que nos ocupa, así como del estudio de las implementaciones en Python disponibles públicamente para su consulta[**] se desprende que el encontrar una arquitectura suficientemente robusta para capturar la relación semántica entre el texto descriptivo y las imágenes es un problema difícil.

A continuación se exponen algunos de los temas principalmente identificados:

C.1. Generalidad de las imágenes y texto. Los intentos exitosos que se conocen en el ámbito académico abordan conjuntos de datos específicos y focalizados a objetos de los que los modelos pueden abstraer representaciones. Sin embargo derivado del análisis exploratorio realizado a las imágenes de Flickr8, se encontró que tales abarcan una multitud de situaciones que no necesariamente guardan relación entre sí. Para ilustrar este punto basta decir que algunas fotos describen: i) personas (incluyendo usando máscaras, disfraces que cubren parcial o

[§] Las imágenes y las descripciones en texto de este conjunto de datos se encuentran disponibles, respectivamente, para su consulta a través de las siguientes direcciones electrónicas https://github.com/jbrownlee/Datasets/releases/download/Flickr8k/Flickr8k_Dataset.zip y https://github.com/jbrownlee/Datasets/releases/download/Flickr8k/Flickr8k_text.zip .

[¶] De acuerdo a la propia página de Amazon Mechanical Turk, este es "... un mercado de crowdsourcing que facilita a las personas y las empresas externalizar sus procesos y trabajos a una fuerza de trabajo distribuida que puede realizar estas tareas virtualmente. Esto podría incluir cualquier cosa, desde la validación de datos e investigaciones simples hasta tareas más subjetivas como la participación en encuestas, la moderación de contenido y más. MTurk permite a las empresas aprovechar la inteligencia colectiva, las habilidades y los conocimientos de una fuerza laboral global para agilizar los procesos comerciales, aumentar la recopilación y el análisis de datos y acelerar el desarrollo del aprendizaje automático..."

[‖] Véase por ejemplo https://tinyurl.com/y7zsjtvs

[**] De acuerdo a la exploración realizada, descatan algunos repositorios https://github.com/zsdonghao/text-to-image

totalmente su cuerpo) ii) mascotas en diferentes posiciones (perros, de manera individual y en grupo), iii) paisajes, donde tenuemente se distinguen personas haciendo un actividad (por ejemplo, haciendo rapel), iv) frutas (concretamente, sandías que son recogidas del suelo y hasta v) pescados.

Este punto motiva que el modelo desarrollado tenga suficiente complejidad para abarcar situaciones tan generales o al menos se tenga un número de datos necesarios para plasmar en el ajuste de sus parámetros todas las situaciones de Flickr8 dentro de su contexto semántico, tanto a nivel de texto como de imagen.

C.2. Representaciones vectoriales adecuada texto y semántica con imágenes. En el área de procesamiento de lenguaje natural un enfoque que ha tenido mucho éxito es la representación de las palabras a través de vectores numéricos de cierta dimensión d. Ésta se basa en pensar al vocabulario a partir de una abstracción que forma parte de un espacio vectorial de manera que las palabras similares o relacionadas se encuentren representadas por puntos cercanos, por lo que recibe el nombre de *representación vectorial distribuida*, o también un encaje (*embedding*) de palabras. Desde un punto de vista matemático, los encajes son herramientas que permiten representar un objeto dentro de una variedad (usualmente dimensión más baja) de manera densa, preservando la cercanía entre puntos. Este paso funciona como una especie de traducción de lenguaje hacia un espacio de representaciones que pueden ser procesadas usando modelos de aprendizaje de máquina.

Los modelos de aprendizaje de máquina a partir de discurso en texto requieren necesariamente de un procedimiento que permita codificar su contenido en términos numéricos. Típicamente esto se logra a partir de una representación estadística relacionada con el problema que se encuentra conectada a una capa de encaje; la calibración de los pesos de esta última capa pueden ajustarse como parte del proceso de entrenamiento (que puede implicar cómputo intensivo) y también pueden usarse como parte de modelos pre-entrenados.

El reto es por lo tanto encontrar una representación vectorial adecuada que refleje la relación semántica de las imágenes de Flickr8 y sus descripciones.

C.3. Retos técnicos y de implementación. El medio principal de infraestructura para el desarrollo de este proyecto es la herramienta gratuita *Google Colab* de la empresa Google. En sus propios términos tal "... es un producto de Google Research. Colab permite a cualquiera escribir y ejecutar código arbitrario de python a través del navegador, y es especialmente adecuado para el aprendizaje automático, el análisis de datos y la educación. Más técnicamente, Colab es un servicio alojado de portátiles Jupyter que no requiere configuración para su uso, al tiempo que proporciona acceso gratuito a recursos informáticos, incluidas las GPU."[††]. Tal es una herramienta académica sumamente útil pues permite emplear Tensorflow para aprendizaje profundo como backend de la libreria Keras de Python, así como el uso de GPU's sin entrar en detalles de configuración para personas sin tal *expertise* técnico.

Al respecto, el acceso gratuito viene acompañado de una serie de restricciones[‡‡], entre las que destacan: i) no se asegura la disponibilidad de las máquinas virtuales en las que se corren los kernels de Jupyter para programación, ii) tampoco se asegura la disponibilidad de las tarjetas GPU, iii) las conexiones a máquinas virtuales de Google Colab que tienen vidas máximas que pueden ser de hasta 12 horas, aunque en la práctica se observa que ante la falta de actividad los entornos de trabajo se desconectan, iv) como en cualquier equipo, los entornos de trabajo de Google Colab poseen, en su versión gratuita, una RAM limitada (cerca de 13 GB) lo que puede ser insuficiente para el procesamiento simultaneo de imágenes y texto, en el entendido que Colab posee parámetros de seguridad de consumo de la misma, que al rebasarse reinician el entorno de trabajo y hacen que se pierdan los cambios.

C.4. Maldición de la dimensionalidad y algoritmos iterativos de optimización. Los modelos de aprendizaje profundo necesitan ajustar una cantidad considerable de parámetros (en el orden de al menos 10^6), lo que significa que un método de optimización iterativo, como los que se emplean para minimizar versiones empíricas de las funciones de pérdida, deben explorar un espacio dimensionalmente grande, con observaciones de orden sustancialmente bajo.

Esto no es más que otra cara de lo que, desde el punto de vista teórico, se conoce como la maldición de la dimensionalidad y que puede ser enfrentada teniendo acceso a un número mayor de datos; sin embargo, la cantidad de datos presentes en el conjunto Flickr8 es limitada.

Por otro lado, en el contexto optimización, la exploración de una variedad diferenciable a través de un método de *sampling* de tipo finito para buscar minimizar una función objetivo típicamente se enfrenta a problemas propios de la topología de variedades de dimensión alta: presencia de mínimos locales, puntos silla, problemas de convergencia en el caso de métodos basados en derivadas o aproximaciones numéricas de las mismas, dependencia de condiciones iniciales para asegurar la convergencia de un método.

2. Descripción detallada de la solución presentada

A. Procesamiento de la información. Sobre el texto de las descripciones de la imágenes, se realizaron una serie de transformaciones encaminadas a tener una estructura apropiada para trabajar el texto de descripciones e imágenes, las que se resumen como sigue

A.1. Sobre datos faltantes. Se descartaron los 5 registros correspondientes a por ausencia de imagen $2258277193\_586949ec62.jpg$ en el conjunto de datos Flickr8.

A.2. Sobre texto. Se realizaron diferentes acciones para procesar el texto de los comentarios: 1) transformación a minúsculas, 2) eliminar puntuación, 3) cambiar las posibles contracciones de palabras presentes en el idioma inglés, 4) remover *stopwords*. Además, como se verá en la siguiente subsección, el texto se transformó hacía una representación de un modelo pre-entrenado.

A.3. Sobre imágenes. Se realizó la normalización de las imágenes a 256×256 pixeles para posteriormente reducirlas a 64×64 pixeles; manteniéndose en estos casos los tres canales asociados. Además se dividió entre 255 la representación numérica de las imágenes para que la entradas se encontraran en el intervalo $[0, 1]$.

[††] Traducción libre a partir de https://research.google.com/colaboratory/faq.html

[‡‡] Para mayor detalle, véase la página https://research.google.com/colaboratory/faq.htmlresource-limits

B. Representación vectorial distribuida y Glove. Como se ha mencionado antes los encajes permiten traducir representaciones vectoriales del texto hacia espacios vectoriales que son necesarios para modelos de aprendizaje profundo. En la actualidad, existen encajes que organizaciones han entrenado basados en algún corpus y que se ponen a disposición del público.

Entre ellas, destaca la herramienta GloVe de aprendizaje no supervisado(Pennington et al. (2014)), pre-entrenada por la universidad de Stanford. Se basa en un modelo que intenta capturar que la simple observación de que las proporciones de probabilidades de coincidencia palabra-palabra tienen el potencial de codificar alguna forma de significado, para lo cual se usa un modelo log-bilineal que tienen por objetivo minimizar una norma ponderada que refleja dicha idea. Este encaje es de potencial interés para el problema que nos ocupa, pues al ser entrenado con un corpus tan amplio, es probable que pueda capturar contexto y la relación entre texto e imágenes.

De manera particular, en este trabajo se ha explorado la representación pre-entrenada sobre un corpus conjunto de Wikipedia 2014 y Gigaword 5, conocida como glove.6B, la cual integra 6 miles de millones de tokens codificados en 50, 100, 200 y hasta 300 características que definen las dimensiones del encaje.

C. Elementos de infraestructura. Como se ha mencionado previamente, el medio principal de infraestructura para el desarrollo de este proyecto es la herramienta *Google Colab* de la empresa Google.

D. Método propuesto. Como se ha mencionado previamente en la literatura y particularmente en los auto-enconders de tipo variacional, se han encontrado hipótesis que han sido exitosas para otros problemas de aprendizaje profundo para la generación de datos sintéticos: 1) se teoriza la existencia de una función de encaje hacia una variedad en $\mathbb{R}^n$ cuya topología captura no solo la representación semántica de los textos, sino que permite que se preserven nociones de cercanía entre textos a través del encaje[§§] 2) que la distribución de los datos que codifica la relación semántica entre texto e imagen puede ser capturada en términos de una distribución de probabilidad perteneciente a una familia paramétrica, 3) que los parámetros de dicha distribución de probabilidad pueden ser ajustados a través de un modelo de aprendizaje profundo, minimizando una función de pérdida específica que cuantifique la distancia entre las distribuciones de probabilidad aprendizas en el proceso de entrenamiento, 4) la codificación de lo anterior redunda en términos prácticos en un espacio intermedio de los modelos que se conoce como espacio latente, que al ser parametrizado como una distribución de probabilidad, puede emplear para simular puntos de dicha distribución y con ello generar imágenes sintéticas.

Es así que el método propuesto se apoya en los principios de las arquitecturas de tipo de auto-enconder variacional. Concretamente, las cuales codifican una imagen a través de un espacio latente que se encuentra asociado a los parámetros de una distribución normal multivariada, en el entendido de que el modelo subyacente aprende la función identidad referida a las imágenes que a la vez son datos de entrada y objetivo.

En línea con lo anterior, la arquitectura consideró dos módulos: i) Denominado *encoder*, encargado de la parte de procesar la representación vectorizada del texto de los comentarios de las imágenes, para consolidar un espacio latente parametrizado como una distribución normal multivariada, destacándose que los paramáetros de media y varianza se obtiene a partir de capas densas que conectan hacia la representación vectorial distribuida del texto de las descripciones de la imágenes; y ii) llamado *decoder*, que parte de la representación del texto obtenida en la etapa anterior para transformarlo en una imagen de tres canales, con una resolución de 64×64 pixeles.

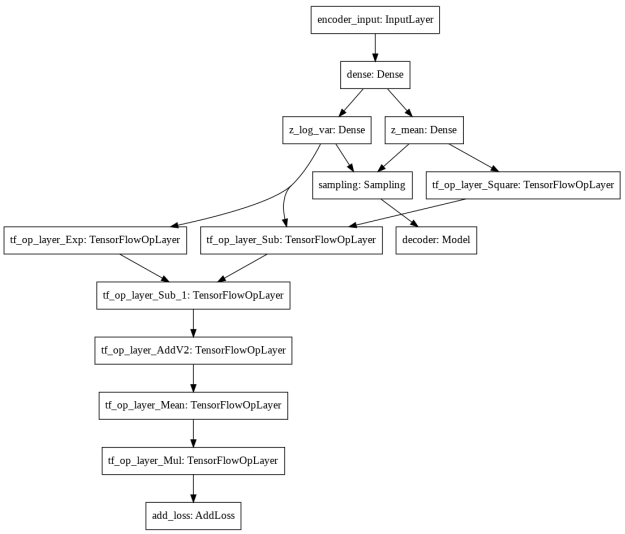

Fig. 3. Diagrama de un "autoencoder" variacion de representación de texto a imagen

3. Experimentos y resultados

A. Metodología seguida. Para probar dicha arquitectura, parar ajustar tal se hicieron las siguientes consideraciones:

- Uso de una función de pérdida que se integra por dos componentes: i) una función de error cuadrático medio para comparar las distancias entre las representaciones matriciales de las imágenes, ii) una representación de la distancia Kullback-Lieber[¶¶] para estimar la distancia entre los parámetros de la distribución normal y una distribución normal estándar. Intuitivamente, ello sugiere el ajuste en términos de los parámetros que codifica la relación entre texto e imagen a través del espacio latente.
- Como método de optimización se empleó el método de RMSprop,
- Se realizó un entrenamiento del modelo considerando lotes (batches) de tamaño 1, a través de hasta 15 épocas;,
- Se evaluó su desempeño contra un split de los datos de entrenamiento (en una relación de 80% y 20%), donde las imágenes y texto de entrenamiento corresponden a los que fueron indexados con el entero 5 dentro del conjunto Flikr8.
- Se registro el valor de la métrica *accuracy* y la pérdida en dichos conjuntos.

[§§] Dado que los conjuntos de datos son discreto, la función puede presentar discontinuidades

[¶¶] En términos matemáticos, este funcional no satisface la definición explícita de una métrica en el espacio de distribuciones, sin embargo se interpreta como una "distancia" al definirse en términos de una medida de probabilidad asociada a una distribución y la derivada Radon–Nikodym de dicha probabilidad base respecto a otra, para el cálculo de la entropía de una en términos de la otra

Por otra parte, se consideraron variantes de dicha arquitectura para realizar diferentes experimentos numéricos y analizar ejemplos de las imágenes generadas. Desafortunadamente, ante la dificultad del problema, se destaca que no se obtuvieron imágenes nítidas o de las que se desprenda una imagen nítida en relación con texto.

El modelo 1 emplea un *learning rate* de 10^{-2}, este une la representación del texto de los mensajes con el GloVe a través de una representación vectorial distribuida hacia los parámetros de una distribución normal empleando capas densas, que definen el encoder. Por otro lado a partir de aquí la capa de decoder emplear una serie de capas densas, cuyas entradas se alinean para reconstruir la imagen de tres canales.

Respecto al modelo 2, este es una modificación del modelo previo donde el *learning rate* se ha hecho más pequeño en un orden de magnitud; es decir, aprende de una forma menos agresiva. Además, se consolidó al modelo 3 como resultado de una modificación del modelo 2, que usa capas Dropout en la parte del decoder, para intentar mejorar la precisión al recobrarse la imagen a partir del espacio latente, pero reduciendo a la mitad los nodos de las capas de enconder.

Cabe destacar que el detalle de las implementaciones de tales modelos se pueden consultar en el repositorio de Github https://github.com/czammar/DL$_{f}$*inalproject*, concretamente a través de su directorio notebooks.

Al respecto, la configuración de los modelos anteriores en términos de sus parámetros se resume en la tabla 1:

Modelo	Parámetros
1	7.64 millones
2	7.64 millones
3	7.58 millones

Table 1. Cantidad de parámetros por modelo

De entre estos, se destaca que el modelo que mejor resultados obtuvo en términos de accuracy fue el **modelo 1**. Además se destacan los siguientes hechos:

- Desafortunadamente, no se puso superar el margen de 50% de accuracy,
- La perdida en entrenamiento fue del orden de 6×10^{-7}
- no se apreciaron mejoras significativas al introducción el droput, aunque sí un desempeño comparable de los modelos 2 y 3, considerando que el segundo tiene un menor número de parámetros.
- En todos los casos, el tiempo para completar una época se encontró cercano a los dos minutos usando la infraestructura de Google Colab, en un entorno habilitado para emplear GPU's.
- Se destaca que en varios de los ejercicios numéricos se obtuvieron re-inicios inesperados de la implementación, dado el exceso de consumo en RAM que por defecto hace que el entorno se desconecte y se pierde el trabajo desarrollado hasta ese punto.

Para ilustrar el desempeño del mejor modelo obtenido, se presentan en complemento los gráficos de la variación de

Table 2. Resultados de ajuste del modelo

Concepto	Modelo 1	Modelo 2	Modelo 3
Training Accuracy*	0.4477	0.4758	0.4781
Training Loss*	0.0684	0.0667	0.0667
Test Accuracy*	0.4961	0.4691	0.4555
Test Loss*	0.0679	0.0667	0.0667
Tiempo por época**	**$\sim$ 2 min**	**$\sim$ 2 min**	**< 2 min**

*valor de última iteración de la última época,

**promedio de todas la época,

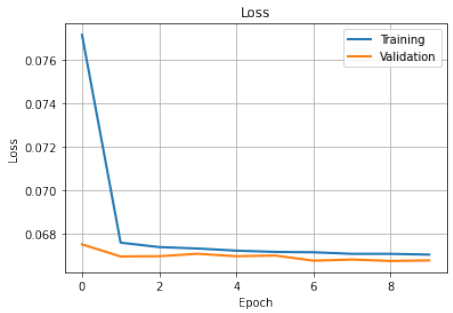

Fig. 4. Curva de pérdida obtenida con el modelo 1. Se aprecia un decrecimiento comparable en los conjuntos de prueba y entrenamiento

la pérdida y del accuracy observados en los conjuntos de entrenamiento y de prueba:

Por otro lado, en la curva para evaluar el accuracy de este modelo se aprecia una especie de oscilación en el valor del accuracy de entrenamiento, esto podría deberse al hecho de que se empleó un learning rate de orden 10^{-2} para aprendizaje "agresivo".

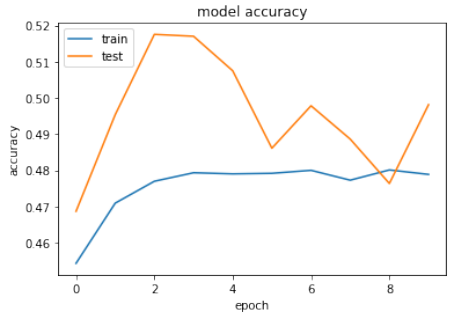

Fig. 5. Curva de accuracy obtenida con el modelo 1.

B. Relación entre textos e imágenes a partir de los modelos.

Dado el desempeño obtenido, es de esperarse que las imágenes generadas por estos modelos no sean necesariamente reflejen la relación entre texto e imágenes.

Para ello se comparó la imagen que los modelos recién ajustados generaron, considerando una selección al azar del conjunto de entrenamiento. El texto se convirtió a la representación vectorial distribuida usando el modelo GloVe preentrenado, descrito en secciones previas.

B.1. Modelo 1. **Mensaje:** "Two helmeted men sit on yellow snowmobiles while another man stands behind watching".

Al respecto, las figuras 5 y 6 indican, respectivamente, la imagen comprimida en resolución de 64×64 pixeles y la imagen generada por el modelo.

Se aprecian algunos cambios de coloración en la zona donde se encuentra la nieve y las personas, aunque obviamente sin definir específicamente la existencia de objetos o personas.

Fig. 6. Imagen del conjunto de datos Flickr8 correspondiente al modelo 1, el mensaje del que se acompaña dice "Two helmeted men sit on yellow snowmobiles while another man stands behind watching"

Fig. 7. Ejemplo de imágenes generadas a partir de texto, modelo 1 entrenado a partir de datos de Flickr8.

B.2. Modelo 2. **Mensaje:** "The small child climbs on a red ropes on a playground ".

Al respecto, las figuras 7 y 8 muestran, respectivamente, la imagen comprimida en resolución de 64×64 pixeles y la imagen generada por el modelo.

Aquí hay evidentemente un deterioro con respecto al modelo 1. Además claramente se aprecian enrojecimiento en la zona donde se localiza el puente y el niño de la imagen, pero sin definir específicamente la existencia de objetos o personas.

B.3. Modelo 3. **Mensaje:** "The brown and white dog with a green collar is biting a stick".

Al respecto, las figuras 9 y 10 exponen, respectivamente, la imagen comprimida en resolución de 64×64 pixeles y la

Fig. 8. Imagen del conjunto Flickr8 correspondiente al modelo 2, que corresponde al mensaje "The small child climbs on a red ropes on a playground ."

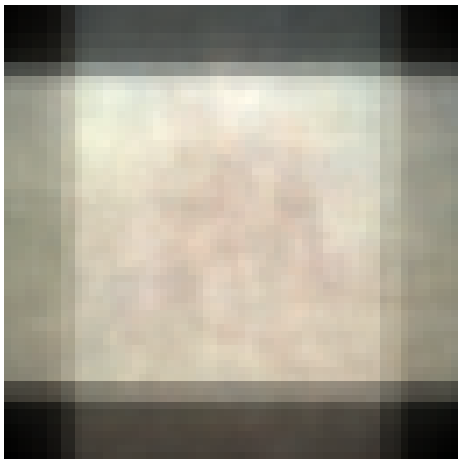

Fig. 9. Ejemplo de imágenes generadas a partir de texto, con el modelo 2.

imagen generada por el modelo.

Aquí el modelo claramente no guarda ninguna relación con la imagen del perro sosteniéndose en dos patas.

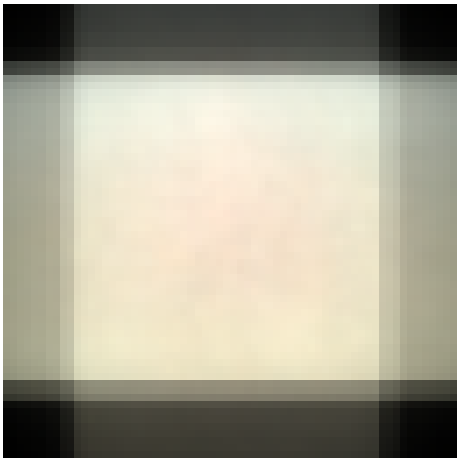

Fig. 10. Ejemplo de imágenes generadas a partir de texto, con el modelo 3.

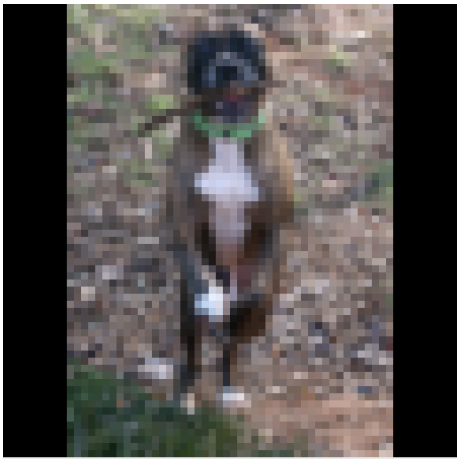

Fig. 11. Imagen correspondiente al modelo 3, que se acompaña del mensaje "The brown and white dog with a green collar is biting a stick ."

4. Conclusiones

A. Generales.

- El problema de generación de imágenes sintética a través de texto es un problema complejo, que se encuentra en la frontera de la investigación de aprendizaje profundo.
- En este documento se presentó una propuesta de modelos basado en aprendizaje profundo para atacar el problema de generación de imágenes sintéticas a partir de texto a partir del conjunto de datos Flikr8.
- Para ello, esencialmente se modificó la arquitectura de modelos de tipo auto-enconder variacional incorporando una representación vectorial distribuida de las descripciones que acompañan a dichas imágenes empleando el modelo pre-entrenado tipo GloVe de Stanford.
- Se comparó el desempeño de tres modelos pertenecientes a la familia propuesta, en términos de las métricas de pérdida y accuracy, sobre conjuntos de entrenamiento y prueba.
- Finalmente, se compararon las imágenes generadas a través de tales modelo, encontrándose desafortunadamente poco ajuste en la generación de imágenes.

B. Lecciones aprendidas.

- Aunque la teoría puede darnos indicios de como resolver un problema, se requiere tiempo y discusión para aterrizar el proceso creativo en código realista. Sin embargo, una buena base teórica y visión para el proceso de programación son claves para el éxito de los proyectos.
- No se puede pensar solo en el diseño de la implementación, sin tener en cuenta la infraestructura necesaria para lograrlo.
- Se requiere aprender estrategias y herramientas de software/hardware para bajar tiempos, mejorar pruebas y demás que aseguren el éxito del proyecto,
- Las herramientas de cómputo en la nube (Google Colab) se pueden aprovechar en nuestro beneficio para resolver problemas complejos, con soluciones que pueden ser compartidas y replicadas fácilmente a través de estos medios.

C. Trabajo a futuro.

- Incorporar otras distribuciones paramétricas, distintas de normales multivariadas, para representar el espacio latente.
- Considerar métodos para aumentación de datos, por ejemplo a partir de procesamiento o filtros de imágenes.
- Alimentar el modelo con otras representaciones de encajes que se encuentran disponibles públicamente, por ejemplo FastText.
- Añadir características que han sido exitosas en otros entorno de generación de imágenes, aumentación de dimensiones,

References

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. (2014). Generative adversarial networks. *ArXiv*, abs/1406.2661.

Hodosh, Micah, P. Y. and Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Reed, S., Akata, Z., Mohan, S., Tenka, S., Schiele, B., and Lee, H. (2016a). Learning what and where to draw.

Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. (2016b). Generative adversarial text to image synthesis.

Zhang, H., Xu, T., Li, H., Zhang, S., Huang, X., Wang, X., and Metaxas, D. N. (2016). Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. *CoRR*, abs/1612.03242.

Zia, T., Arif, S., Murtaza, S., and Ullah, M. A. (2020). Text-to-image generation with attention based recurrent neural networks. *ArXiv*, abs/2001.06658.