

# GENERACIÓN DE DESCRIPCIONES DE IMÁGENES Y ADECUACIÓN DE ESTAS A UN TEXTO

## BECA-COLABORACIÓN DEPARTAMENTOS UPM CURSO 2022-23

David Cano Rosillo  
Dr. M<sup>a</sup> del Carmen Suárez de Figueroa Baonza (UPM)

### Tabla de Contenidos

1. Introducción
2. Objetivos
3. Revisión del estado del arte
4. Aportaciones
5. Pruebas realizadas
6. Detalles de la implementación
7. Propuestas de mejora
8. Conclusión

### 1. Introducción

El siguiente proyecto es parte de la línea de investigación de accesibilidad cognitiva. El objetivo de esta línea es utilizar inteligencia artificial para adaptar de forma automática textos escritos para un lector medio en textos que sigan la metodología de lectura fácil. Una de las pautas de dicha metodología dice que en el caso de usar imágenes estas deben estar relacionadas con el texto para una mayor comprensión. Este proyecto pretende desarrollar un software que calcule la adecuación de una imagen al texto relacionado con la imagen considerada. Para ello generaría una descripción de la imagen en lenguaje natural, que se podrá usar como pie de página para facilitar la lectura. A continuación, mediría la distancia semántica entre el texto relacionado con la imagen considerada y la descripción generada. En caso de no ser coherentes, el software sugeriría una sustitución de la imagen por otra más adecuada. Se aplicarían técnicas de aprendizaje automático. Más concretamente, de visión por computador, generación de texto natural y procesamiento de lenguaje natural.

### 2. Objetivos

En concreto, nos hemos centrado en 3 pautas de la metodología de lectura fácil (Muñoz., (2012)). Estas son las siguientes:

- Titular las imágenes.
- Evitar diagramas, gráficos estadísticos y tablas técnicas.
- Utilizar imágenes de apoyo al texto, que hagan referencia al mismo explícitamente y con un vínculo claro.

Para automatizar las respectivas pautas se proponen las siguientes funcionalidades del software implementado:

- Generar descripciones automáticamente en base a una imagen.

- Detectar imágenes con distintos tipos de gráficos para eliminarlas del documento.
- Detectar cuando un texto no referencia explícitamente una imagen.

Si se automatizasen estos objetivos se aceleraría la adaptación de documentos a metodología de lectura fácil. Esto ayudaría a generar más documentos accesibles a todo el mundo.

### 3. Revisión del estado del arte

En la implementación se usan tanto modelos de inteligencia artificial ya entrenados como uno entrenado durante la duración de la beca. Gran parte del tiempo se usó en entender cómo funcionan estos modelos y buscar la mejor solución que puede ofrecer el estado del arte actualmente.

Gran parte de las soluciones se basan en el concepto de *embeddings*. Un *embedding* es una función que transforma datos en un formato cómodo para los modelos de inteligencia artificial, vectores n-dimensionales. Además, tienen la peculiaridad de mantener la distancia semántica en el espacio n-dimensional. En la siguiente imagen podemos apreciar como "newspaper" está cerca de "magazine", y a la vez lejos de "biking".

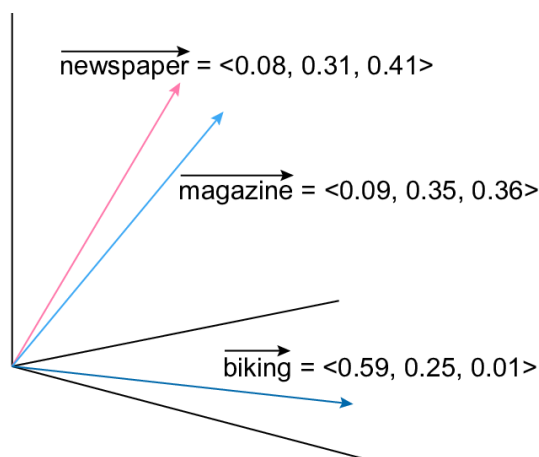


Figura 1: Un embedding 3-dimensional que transforma palabras a vectores.

En nuestra tarea nos concierne una extensión de este concepto, los embeddings multimodales. Estos son embeddings que mapean al mismo espacio objetos de varias modalidades, como por ejemplo descripciones de imágenes e imágenes. La siguiente es una imagen del funcionamiento del modelo CLIP (Alec Radford, 2021).

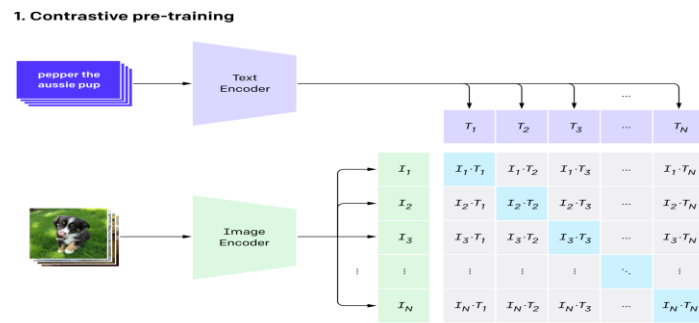


Figura 2: El modelo CLIP mapea descripciones e imágenes al mismo espacio.

La tendencia en los artículos actualmente es centrarse en aprender un buen embedding. Con este embedding se pueden entrenar modelos para distintas tareas. Curiosamente aprender un buen embedding tiene mejores resultados que entrenar directamente para la tarea deseada. Este fenómeno se conoce como *Multi-task learning*.

La última iteración de este concepto es el modelo BLIP2 (Junnan Li, 2023). Sin entrar en más detalle, aprende un embedding optimizando para varias tareas. A continuación, se entrena con el embedding aprendido como base para muchos objetivos. En este proyecto usamos dos modelos entrenados para los siguientes objetivos:

- Generar descripciones automáticas, con una aplicación directa.
- Predecir si una descripción corresponde al contenido de una imagen.

## 4. Aportaciones

Volviendo a nuestras 3 pautas a automatizar, veamos cómo han sido implementadas.

- Titular las imágenes.

Para titular las imágenes usaremos un modelo entrenado en base al embedding de BLIP2. La calidad de estas descripciones es evaluada por la métrica CIDER. El único detalle que podría degradar los resultados es que se traducen las descripciones generadas por el modelo al castellano.

- Detectar imágenes con distintos tipos de gráficos para eliminarlas del documento.

En esta pauta no se encontró un modelo ya entrenado, por lo que se entreno uno nuevo. Para clasificar las imágenes se hizo uso de un dataset de kaggle. El dataset consiste en 15875 imágenes de las siguientes 8 clases.

1. Sin gráfico
2. Gráfico de barras
3. Diagrama
4. Gráfico de flujo
5. Gráfico de ejes

6. *Gráfico de crecimiento*
7. *Gráfico circular*
8. *Gráfico de tabla*

Se eligió en modelo pre-entrenado *ResNet18* en el dataset ImageNet, para ajustarlo a nuestro problema. El único cambio en la arquitectura fue sustituir la última capa, para que en vez de clasificar 1000 clases clasifique entre las 8 de nuestro problema. Se entreno al modelo con un 90% de los datos, mientras que un 10% se reservó para evaluarlo. Se obtuvo un 93% de precisión.

- Utilizar imágenes de apoyo al texto, que hagan referencia al mismo explícitamente y con un vínculo claro.

Anteriormente vimos como existe un modelo basado en BLIP2 que predice si una descripción esta emparejada con una imagen. Es decir, la descripción introducida describe el contenido de la imagen.

En este proyecto se propone usar este modelo para medir la explicitud. La motivación es que, al ser entrenado en descripciones, detecta referencias explícitas a la imagen. También se calculó un umbral en base al cual decidir si una relación es lo suficientemente explícita o no.

Normalmente textos largos no referencian tan explícitamente la imagen como descripciones. Es por esto que establecemos un nuevo umbral en base al cual decidir si un documento sigue esta pauta. En la sección *Pruebas realizadas* se contrastan los distintos métodos probados.

## 5. Pruebas realizadas

Las pruebas se centraron en evaluar la detección de explicitud. Este trabajo fue complicado ya que no se tenía un dataset de relaciones explícitas contra relaciones no explícitas en el que evaluar.

- Utilizar imágenes de apoyo al texto, que hagan referencia al mismo explícitamente y con un vínculo claro.

Es por ello que se limpió un dataset de noticias en internet. De todas las parejas de imágenes-noticias se seleccionaron aquellas que cumplían los siguientes requisitos.

1. La noticia estaba en castellano.
2. La URL correspondiente a la imagen permitía descargarla en menos de 4 segundos.

El resultado fue un archivo json con 2200 noticias en castellano y una carpeta con las 2200 imágenes correspondientes. A continuación, se generó un csv para clasificar entre noticia-imagen emparejada y noticia-imagen no emparejada. Las primeras 2200 filas consisten en los índices de las imágenes con sus respectivas noticias. Las siguientes 2200 filas consisten en imágenes con noticias aleatorias del dataset.

Una buena medida de explicitud debe de ser capaz de diferenciar noticia-imagen emparejada de no emparejada. Esto es debido a que cuando la noticia esta emparejada con la imagen, se referencia explícitamente la imagen. Consideraremos el mejor método de calcular la explicitud al que mejor clasifique si un par noticia-imagen esta emparejado.

Los distintos métodos probados obtienen una distancia entre la imagen y la noticia. Calculamos la precisión de cada método calculando el umbral que mejor clasifica emparejado de no emparejado. En los dos primeros métodos se evalúa la similitud del coseno. Esta tiene rango  $[-1, 1]$ , un resultado de 1 significaría semánticamente iguales, mientras que -1 semánticamente contrarios. El tercer enfoque devuelve porcentajes.

Los enfoques probados fueron los siguientes, todos requieren una traducción previa de las noticias al inglés para ser usados:

- MPNET (Kaitao Song, 2020): Usar el embedding de textos MPNET para calcular la distancia entre el embedding de la descripción generada automáticamente y el embedding MPNET de la noticia.
- CLIP: Usar el embedding CLIP para calcular la distancia entre el embedding de la imagen y el embedding de cada frase de la noticia. De todas las distancias computadas con todas las frases, escoger la menor.
- BLIP2: Usar el modelo basado en BLIP2 que predice si una imagen esta emparejada con una descripción de un texto. Aplicarlo a noticias.

Los mejores umbrales para cada método obtuvieron las siguientes precisiones.

MPNET	CLIP	BLIP2
65%	84%	87%

El mejor resultado es el del enfoque BLIP2. Usando como umbral 8% clasifica noticias-imágenes emparejadas de no emparejadas bien un 87% de las veces.

Una sorpresa de los resultados es que BLIP2 es mejor que CLIP, pero por solo un 3%. El embedding de BLIP2 es mejor en otras tareas que el de CLIP así que cabría esperar una mayor diferencia.

Esto se puede deber a que en el enfoque usado con CLIP se calcula la distancia con todas las frases y se queda con la mejor. Sin embargo, con BLIP2 se calcula la distancia entre todo el texto y la imagen. Sospecho que aplicar la misma técnica con BLIP2 obtendría un mejor resultado, pero las limitaciones computacionales no me han permitido explorarlo. Todas las pruebas fueron hechas en una CPU, lo cual hacía que recorrer todo el dataset tardase aproximadamente 8 horas.

Tanto el código como el dataset usado para evaluar los modelos está disponible en un repositorio en [GitHub](#). Los resultados son replicables con el csv usado también disponible. Animo encarecidamente a probar el método BLIP2 frase a frase si se dispone de una GPU con 6.5GB de memoria para albergar el modelo.


## 6. Detalles de la implementación

La implementación de las 3 pautas de lectura fácil ha sido desarrollada para una API y una web a modo de demo. Actualmente está disponible en un servidor del Ontology Engineering Group. Se está ejecutando sin GPU y con una RAM de 6.5GB para poder almacenar en ella los pesos del modelo más grande. La siguiente imagen muestra el resultado final.

### Implementación pautas E2R con imagenes

Introduzca la imagen a analizar aqui:

wallhaven-e7kw6o.png



#### Descripción automática

Una pintura de un paisaje con montañas y un cuerpo de agua

#### Predicción del tipo de imagen

Un modelo calculara la probabilidad de que la imagen pertenezca a cada una de estas 8 clases.

- Sin gráfico: 99.48261260986328%
- Gráfico de barras: 0.06646721065044403%
- Diagrama: 0.009390786290168762%
- Gráfico de flujo: 0.03940131142735481%
- Gráfico de ejes: 0.11106345057487488%
- Gráfico de crecimiento: 0.025203099474310875%
- Gráfico circular: 0.13278332352638245%
- Gráfico de tabla: 0.13308404386043549%

#### Calculo similitud semántica con texto

El modelo devolvera la probabilidad de que la imagen y el texto esten relacionados. El modelo fue entrenado con descripciones explícitas asi que devuelve probabilidades bajas en muchos textos. Un resultado mayor a 8% predice que un texto esta relacionado con una imagen con una tasa de acierto del 87%.

Introduzca el texto a comparar:

Un paisaje de ciencia ficción.

Porcentaje obtenido 97.16492891311646% Texto adecuado a imagen si. Limite usado para decidirlo 8%.

El código que contiene la API y la página web esta también disponible en otro repositorio [GitHub](#). El repositorio contiene un archivo Dockerfile que configura un contenedor Docker con todo listo para ejecutar. Las principales tecnologías usadas en este proyecto han sido Python, Pytorch, FastAPI, Git, Docker, JavaScript, HTML y CSS.

## 7. Propuestas de mejora

Por último, quería mencionar algunos puntos del proyecto que se podrían mejorar en el futuro. Creo que en futuras becas de colaboración otros compañeros podrían aportar mucho a este proyecto.

- Los subtítulos generados solo se basan en las imágenes. Agregar el texto emparejado como entrada ayudaría a generar descripciones más relevantes.

A menudo las descripciones son correctas, pero poco útiles para ayudar a entender la relación con el texto. Por ejemplo, para una foto del presidente el método actual generaría la descripción "un hombre en traje". Un enfoque que también se fije en el contexto dado en la noticia podría generar la descripción "El presidente en la Moncloa".

- Según nuestros experimentos, la explicitud promedio entre noticias e imágenes es del 8%. ¿Deberían nuestros textos E2R ser más explícitos que las noticias?

Quizás el umbral sobre el cual se decide si la relación es lo suficientemente explícita debería ser mayor. La recopilación de un dataset con ejemplos de relaciones explícitas y no lo suficientemente explícitas podría facilitar establecer un mejor umbral.

- Probar el enfoque BLIP2 frase a frase, cogiendo como distancia general entre texto e imagen la mejor distancia.

Como comento previamente este enfoque se queda en el tintero a falta de una GPU con memoria de 6.5GB. Creo que puede obtener buenos resultados y merece la pena explorarlo.

- Los modelos multimodales han ganado mucha atención desde el lanzamiento de GPT4. Mejores modelos impondrán un nuevo estado del arte pronto.

En el último mes del desarrollo de esta beca se anunció GPT4. Este integra la capacidad de entender textos e imágenes. La atención obtenida se traducirá en mejores modelos, por lo que nos podríamos beneficiar de mejores resultados. Revisar el estado del arte en tan pronto como 2024 puede dar mejoras significativas.

## 8. Conclusión

La aplicación automática de la metodología de lectura fácil es un problema difícil. Sin embargo, los avances recientes en inteligencia artificial nos acercan cada día más a una solución. De tener éxito, se facilitaría el acceso la información a millones de personas con discapacidades intelectuales.

En este proyecto se aplica con éxito el estado del arte a 3 aspectos de la metodología de lectura fácil en relación con las imágenes. Además, establece una nueva métrica para medir la explicitud y se facilita un punto de referencia para evaluar nuevos avances.

## Bibliografía

Artículos citados en este trabajo:

- Muñoz., O. G. ((2012)). Lectura fácil: Métodos de redacción y evaluación.
- Junnan Li, D. L. (2023). BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models.
- Alec Radford, J. W. (2021). Learning Transferable Visual Models From Natural Language Supervision.
- Kaitao Song, X. T.-Y. (2020). MPNet: Masked and Permuted Pre-training for Language Understanding .

Artículos consultados durante la revisión del estado del arte:

- Jacob Devlin, M.-W. C. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding .
- Satanjeev Banerjee, A. L. (2005). METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments.
- Kishore Papineni, S. R.-J. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation.
- Lin, C.-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries.
- Ramakrishna Vedantam, C. L. (2014). CIDEr: Consensus-based Image Description Evaluation .
- You Yang, Y. A. (2022). Tri-RAT: optimizing the attention scores for image captioning .
- Juntao Hu, Y. Y. (2022). Position-guided transformer for image captioning.
- Dense Captioning of Natural Scenes in Spanish Alejandro Gomez-Garay, B. R. (2018). Dense Captioning of Natural Scenes in Spanish.
- Rafael Gallardo-García, B. B.-M. (2020). Evaluación del modelo neuronal de atención visual en la descripción automática de imágenes en español.
- Karan Desai, J. J. (2020). VirTex: Learning Visual Representations from Textual Annotations .
- Luo, R. (2020). A Better Variant of Self-Critical Sequence Training .
- Ron Mokady, A. H. (2021). ClipCap: CLIP Prefix for Image Captioning .
- Lei Ke, W. P.-W. (2019). Reflective Decoding Network for Image Captioning.
- Yekun Chai, S. J. (2021). RefineCap: Concept-Aware Refinement for Image Captioning.
- Marcella Cornia, M. S. (2019). Meshed-Memory Transformer for Image Captioning .
- Jiahui Yu, Z. W. (2022). CoCa: Contrastive Captioners are Image-Text Foundation Models .
- Chia-Wen Kuo, Z. K. (2022). Beyond a Pre-Trained Object Detector: Cross-Modal Textual and Visual Context for Image Captioning .
- Van-Quang Nguyen, M. S. (2022). GRIT: Faster and Better Image captioning Transformer Using Dual Visual Features.
- Xiaowei Hu, Z. G. (2021). Scaling Up Vision-Language Pre-training for Image Captioning .
- Jia Cheng Hu, R. C. (2022). ExpansionNet v2: Block Static Expansion in fast end to end training for Image Captioning.
- Zheng-Jun Zha, D. L. (2019). Context-Aware Visual Policy Network for Fine-Grained Image Captioning .
- Peter Anderson, X. H. (2017). Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering.
- Yu Zhang, X. S. (2021). Image captioning with transformer and knowledge graph.



- Chenliang Li, H. X. (2022). mPLUG: Effective and Efficient Vision-Language Learning by Cross-modal Skip-connections .
- Hossain, M. Z., & Sohel, F. (2021). Text to Image Synthesis for Improved Image Captioning .
- Malali, N., & Keller, Y. (2022). Learning to Embed Semantic Similarity for Joint Image-Text Retrieval .
- Zhongan Wang, S. S. (2022). ArCo: Attention-reinforced transformer with contrastive learning for image captioning.
- Changzhi Wang, X. G. (2023). Learning joint relationship attention network for image captioning .