

SRI de imágenes usando Machine Learning, modelos de visión y de lenguaje.

La visión y el lenguaje, dos de los métodos más fundamentales para que los humanos perciban el mundo, son también dos piedras angulares clave de la IA. Un objetivo de larga data de la IA ha sido construir agentes inteligentes que puedan entender el mundo a través de entradas de visión y lenguaje, y comunicarse con los humanos a través del lenguaje natural.

Para lograr este objetivo, el preentrenamiento de visión-lenguaje ha surgido como un enfoque efectivo, donde los modelos de redes neuronales profundas se preentrenan en conjuntos de datos de imágenes y textos a gran escala para mejorar el rendimiento en tareas de visión-lenguaje posteriores, como la recuperación de texto-imagen, la generación de subtítulos de imágenes y la respuesta a preguntas visuales.

Como alternativa a esto han surgido modelos de visión y lenguaje, dentro de los cuales, y aplicando estos más a nuestro interés de recuperación de información, se proponen 3 para crear una base para nuestra solución final: BLIP, CLIP y SAM.

- BLIP (Bootstrapping Language-Image Pre-training) es un modelo de preentrenamiento de visión y lenguaje desarrollado por Salesforce Research. Se publicó por primera vez en 2022 [*blip-bootstrapping-lang...*](#). BLIP fue diseñado para entender y generar tareas de visión y lenguaje de manera unificada.
- CLIP (Contrastive Language-Image Pretraining) es un modelo desarrollado por OpenAI. Se publicó en 2021. CLIP fue diseñado para comprender y generar tareas de visión y lenguaje de manera unificada, lo que lo hace relevante en el campo de visión y lenguaje.
- SAM es un modelo desarrollado por Meta AI que se lanzó en 2023 [*blog.roboflow.com*](#). Aunque no se centra específicamente en la intersección de visión y lenguaje, tiene relevancia en el campo de la visión por computadora y la segmentación de imágenes. SAM es un modelo de segmentación de imágenes que produce máscaras de objetos de alta calidad a partir de indicaciones de entrada, como puntos o cajas, y puede utilizarse para generar máscaras para todos los objetos en una imagen.

Utilizando estos 3 modelos se propone crear una base para el desarrollo de un sistema de recuperación de imágenes para consultas detalladas.

Blip-image-captioning

BLIP (Bootstrapping Language-Image Pre-training) es un modelo de entrenamiento desarrollado por Salesforce y alojado en Hugging Face. Este modelo se utiliza para la generación de subtítulos de imágenes, es decir, para convertir imágenes en texto descriptivo.

Usando este modelo se genera, dada una imagen, un texto descriptivo para la misma. Este texto descriptivo puede ser utilizado, potencialmente, para dar uso a técnicas de recuperación de información y formar ranking a la hora de recuperar un documento, en este caso imágenes. Algunos ejemplos de lo que es capaz este modelo son los siguientes:

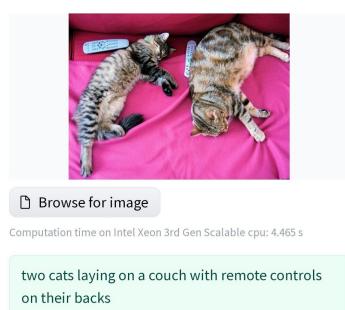


Imagen 1



Imagen 2



Imagen 3

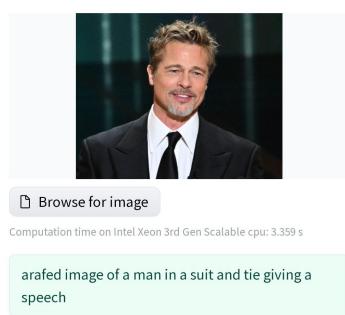


Imagen 4



Imagen 5

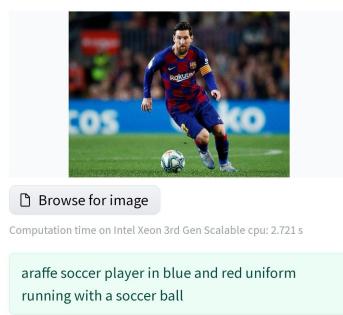


Imagen 6

En estos ejemplos podemos apreciar algunas deficiencias del modelo como son:

- El modelo Blip no está entrenado para reconocer personalidades famosas, no creará texto con nombres de personas.

- Si bien las descripciones son bastante acertadas, nos interesan las descripciones más específicas posibles dado que este es el objetivo final del trabajo (recuperar imágenes de consultas detalladas), en la *Imagen 3* nunca menciona la palabra cuchillo, y este detalle es, en este caso, una parte importante de la imagen, cualquier imagen puede tener un perro y un gato, pero en esta imagen el detalle más importante es, al menos, que hay un cuchillo en la misma.

Se desea recuperar las imágenes usando consultas lo más detalladas posible por los usuarios, y el modelo blip por si solo presenta limitaciones que lo dejarán como un buen modelo para recuperar, taggear, imágenes pero que podemos potenciar. En los próximos temas se explica como se logra potenciar este modelo para mejorar las descripciones de las imágenes.

CLIP

El modelo CLIP (Contrastive Language-Image Pretraining) de OpenAI es una red neuronal que se entrena para entender las relaciones entre texto e imágenes. Este modelo se utiliza para generar incrustaciones (embeddings) que representan tanto imágenes como texto, y luego se puede comparar la similitud entre estos embeddings para determinar la relación entre el texto y la imagen.

Clip es un modelo que ha sido entrenado con 400 millones de imágenes con su texto asociado. Utiliza aprendizaje por contraste que aprende las relaciones entre un texto con su imagen maximizando las relaciones entre los pares correctos y minimizando las relaciones entre los que no se corresponden. Ya no es necesario acertar con la palabra exacta sino con la más parecida. Es un modelo "Zero-shot", puede manejar tareas de clasificación e identificación de objetos en imágenes que nunca vió durante su entrenamiento. Esta capacidad lo hace extremadamente útil en una amplia gama de aplicaciones y en nuestro caso particular para la clasificación de imágenes.

Uno de sus usos particulares es, dado una descripción y un conjunto de imágenes, darle a cada una de estas imágenes cierto porcentaje de posibilidad de pertenecer a la descripción dada, y viceversa (conjunto de textos para una imagen). A diferencia de BLIP, si bien no está entrenado específicamente para reconocer personalidades, este modelo de cierto modo las reconoce, dado que ha sido entrenado con muchos datos incluyendo páginas de internet, además de ser un modelo más potente que BLIP en cuanto a detalles de las imágenes ya que no tiene que especializarse en generar texto, sino que debe hallar similitud entre texto e imágenes.

En el caso de CLIP, este modelo identifica el cuchillo que BLIP no taggueó en la *Imagen 3*. Veamos algunos ejemplos de lo que puede hacer CLIP.

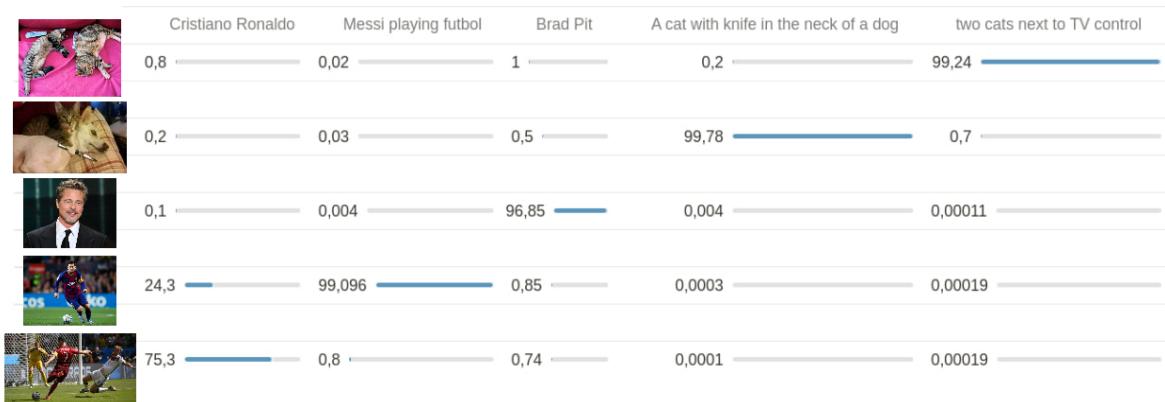


Imagen 7: Resultados del modelo clip para cada consulta como entrada

Aunque el modelo CLIP, como todo modelo de aprendizaje automático, no es del todo preciso, ejemplos de resultados no esperados podemos encontrar en su [paper oficial](#):

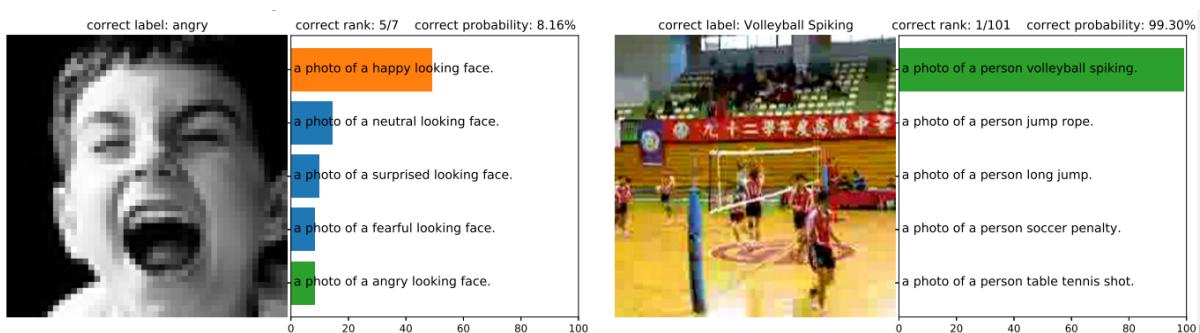


Imagen 8: Captura del paper oficial de CLIP openAI

Si bien CLIP está especializado en hallar similitud entre texto e imágenes, este modelo no está entrenado para generar texto, es decir no genera texto desde una imagen de entrada como lo hace el modelo BLIP. Entonces, en una primera observación, se podría decir que este modelo es el ideal para, una vez tengamos las imágenes ordenadas por un ranking, compararlas con la consulta de entrada, pero hasta ahora no estamos mejorando la generación de descripciones para nuestras imágenes. Para lograr esto se aplicaran técnicas que se explican en el proximo modelo SAM.

SAM

SAM (Segment Anything Model) es un modelo de aprendizaje automático desarrollado por Meta. Este modelo innovador tiene la capacidad de segmentar objetos con precisión en imágenes, incluso en fondos complejos. Se basa en el aprendizaje profundo (Deep Learning). La arquitectura única de SAM permite segmentar imágenes con alta precisión.

Al igual que los modelos vistos anteriormente SAM es también, un modelo de "Zero-shot", puede segmentar objetos en imágenes sin haber visto ejemplos exactos de esos objetos durante su entrenamiento. Esto se debe a que SAM fue entrenado en un conjunto de datos de 11 millones de imágenes y 1.1 billones de "máscaras de segmentación", que le permiten generalizar a nuevos tipos de objetos más allá de lo que vió durante su entrenamiento.

Hasta ahora se han presentado dos modelos (BLIP y CLIP), ambos especializados en la clasificación de imágenes y texto. Usando BLIP se obtiene una descripción en forma de texto para una imagen de entrada, mientras que usando CLIP obtenemos un conjunto de probabilidades que describen la similitud que existe entre imágenes y texto.

En algunos casos se presenta el problema de que el modelo BLIP no logra hacer una descripción perfectamente detallada de la imagen que se le pasa, o al menos obvia detalles de la misma que son imprescindibles para una correcta y detallada descripción. Como propuesta para darle solución a este problema es utilizar SAM para segmentar la imagen, una vez procesada esta, pasar cada segmentación de la misma al modelo BLIP para generar una descripción de esta segmentación. Con cada una de las descripciones, crear una descripción general de la imagen de entrada. Por ejemplo en la siguiente imagen:



[Browse for image](#)

Computation time on Intel Xeon 3rd Gen Scalable cpu: 3.970 s

there is a cat and a dog laying on a bed together

Se observa que la descripción generada por BLIP no hace referencia a un cuchillo, el cual está en primer plano de la imagen y es, de hecho, la palabra más importante para la descripción; y nuestro objetivo es, de ser posible, recuperar información lo más precisa posible, lo más cercana posible a la descripción de la consulta en cuestión.

Imagen 9

Una opción es entonces pasarle la imagen al modelo de segmentación de meta, y usar cada una de las capas de salida de este modelo. Veamos que pasaría en este caso particular:



Imagen 10

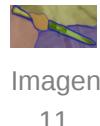


Imagen
11

Este sería el resultado de las capas de segmentación de SAM, se puede apreciar que una de las capas es el cuchillo, lo cual es el interés del uso de este modelo. Pero a su vez SAM presenta el siguiente problema:

Al segmentar toda la imagen y pasar una por una de estas imágenes resultantes al modelo BLIP surgen varios problemas, el primero es que BLIP generara descripciones no deseadas cuando se le pasen capas difíciles de comprender, como es la capa de fondo(amarilla) del sofa, o la capa del ojo del perro.

Dado este problema se propone como solución: no usar todas las descripciones generadas por BLIP, las descripciones que genera, la mayoría no tienen nada de similitud con la imagen, sin embargo nos interesan descripciones de algunas de esas segmentaciones.

a bird that is sitting on a bed with a blanket

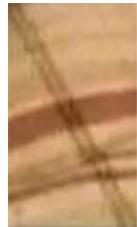


Imagen 12: No nos interesa.

a banana that is sitting on a table



Imagen 13: No nos interesa.

a close up of a knife on a cutting board with a knife in it



Imagen 14: Necesitamos esta descripción para agregar knife a nuestras palabras clave.

Una vez obtengamos las descripciones de BLIP se las pasamos al modelo CLIP para hallar la similitud con la imagen original, veamos un ejemplo basico de esto (solo 4 descripciones).



Imagen 15



Imagen 16



Imagen 17

Se tienen estas 3 descripciones generadas por BLIP de capas que devuelve SAM. Luego cada una de estas descripciones agregadas a la descripción principal se le pasa al modelo CLIP para que este último se encargue de evaluar cada texto, y como resultado se obtiene:



Imagen 18

La técnica parece bastante útil pero aún presenta como dificultad que SAM segmenta a niveles muy bajos de píxeles y esto no genera una imagen clara para pasársela al modelo BLIP. Y al usar cada segmentación de SAM se generan demasiadas imágenes que no nos interesan y que aumentan el costo computacional a la vez que aumenta los inputs del modelo CLIP, lo cual se observó, y CLIP pierde precisión por esta razón (mayor cantidad de entradas).

a small bird sitting on a branch in the air

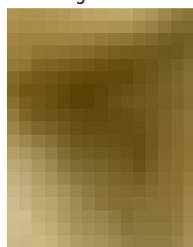


Imagen 19

blurry image of a blurry image of a person holding a cell phone

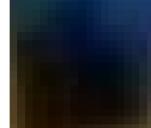


Imagen 20

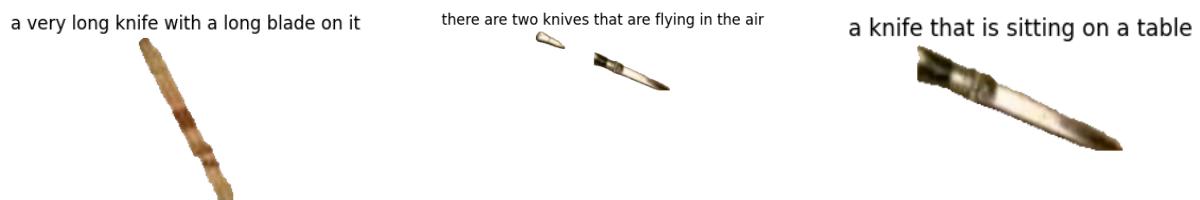
blurry image of a piano in a living room with a couch



Imagen 21

Como solución a esto se decide definir un límite de tamaño mínimo para un cuadro de segmentación, de esta forma se logró reducir la cantidad de imágenes a describir, se evitan imágenes muy poco claras. Hay que tener en cuenta que el modelo SAM no está pensado para este tipo de tareas de generar descripciones, por lo cual su uso debe ser tratado manualmente para no generar imágenes no deseadas.

Otra opción que se probó fue utilizar las segmentaciones exactas para pasarlas al modelo BLIP, es decir en vez de usar el cuadro de pixeles en el cual se encuentra ubicada la segmentación, se utilizarían solo los pixeles que comprendían la misma, pero esto tuvo un resultado negativo en cuanto al modelo BLIP.



Propuesta y Observaciones

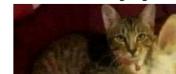
El objetivo que se persigue es recuperar información, imágenes como caso particular; la propuesta inicial a desarrollar es entonces:

- Usando el modelo BLIP, generar una descripción para una imagen de entrada; como se ha planteado, esta descripción no es lo suficientemente detallada para nuestro interés.
- La imagen original la procesamos con el modelo de segmentación SAM para segmentar la misma y usar estas imágenes segmentadas para crear nuevas descripciones.
- Cada una de esas descripciones nuevas sumarlas a la descripción de la imagen principal.
- Usar CLIP para hallar la similitud entre las descripciones generadas y la imagen original.
- Eliminar palabras de la descripción final comparando similitud contra la descripción actual y la descripción sin la palabra dada.

Veamos algunas observaciones de la propuesta panteada:



there are two cats that are sitting together in a red chair



a close up of a knife on a cutting board with a knife in it



someone is cutting a dog's hair with a pair of scissors



Imagen 22 Input

```

original_caption: a cat and a dog laying on a bed together a close up of a knife on a cutting board with a knife in it
reduced_caption: a cat and a dog laying a bed together a close up a knife on a with a knife it

ranking:
1. 83.61%: a cat and a dog laying a bed together a close up a knife on a with a knife it
2. 14.29%: a cat and a dog laying on a bed together a close up of a knife on a cutting board with a knife in it
3. 1.28%: a cat and a dog laying on a bed together someone is cutting a dog's hair with a pair of scissors
4. 0.40%: a cat and a dog laying on a bed together araffe and a cat laying on a couch with a pair of scissors
5. 0.24%: a cat and a dog laying on a bed together a close up of a cat with a blurry look on its face
6. 0.08%: a cat and a dog laying on a bed together blurry image of a cat with a long tail and a long tail
7. 0.02%: a cat and a dog laying on a bed together a close up of a plate of food with a sandwich on it
8. 0.01%: a cat and a dog laying on a bed together
9. 0.01%: a cat and a dog laying on a bed together someone is holding a blue ball in their hand
10. 0.01%: a cat and a dog laying on a bed together a bird that is sitting on a bed with a blanket
.
.
23. 0.00%: a cat and a dog laying on a bed together a cat that is sitting on a couch with a blue ball

```

Imagen 23: Output

En la salida vemos la descripción final que nos proporciona el uso de los 3 modelos como fue planteada y la descripción final reducida que consiste en usar CLIP para comparar similitudes entre la descripción final y la descripción final en cuestión con palabras eliminadas. En este caso particular se elimina los términos ‘cutting board’ (tabla de cortar), lo cual no tiene sentido en la imagen original, y en este caso nos conviene eliminar. No obstante este método de eliminar palabras no siempre es del todo efectivo por lo cual debe perfeccionarse y decidir si usar el mismo o no según las futuras decisiones y técnicas que se decidan utilizar. Por ejemplo en la siguiente imagen esta opción de eliminar palabras de la descripción nos da un resultado no deseado:

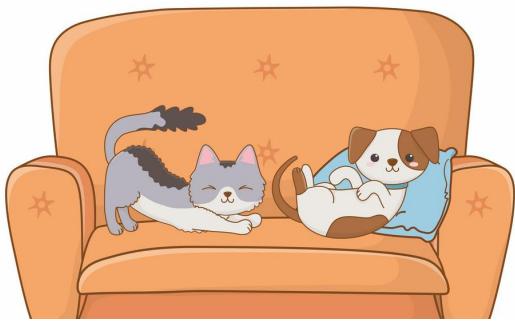


Imagen 24: Input

Con esta imagen como entrada se genera una descripción detallada, pero a nivel de recuperación de información, para formar un ranking y utilizar palabras clave la descripción reducida podría ser más útil. Pero hay un detalle muy importante que CLIP no comprende como imprescindible, y es que la palabra ‘cartoon’ disminuye la similitud con la imagen, y en este caso es de las palabras más importantes.

```

original_caption: cartoon illustration of a cat and dog lying on a couch cartoon dog with a blue collar and a brown nose
reduced_caption: illustration of a cat and dog on a couch dog with a blue and a brown

ranking:
1. 54.61%: illustration of a cat and dog on a couch dog with a blue and a brown
2. 13.55%: cartoon illustration of a cat and dog lying on a couch cartoon dog with a blue collar and a brown nose
3. 5.76%: cartoon illustration of a cat and dog lying on a couch cartoon cat with a tail curled up and eyes closed
4. 5.40%: cartoon illustration of a cat and dog lying on a couch cartoon cat with eyes closed and paws crossed
5. 3.36%: cartoon illustration of a cat and dog lying on a couch cartoon dog laying on a pillow with a pillow cover
6. 3.25%: cartoon illustration of a cat and dog lying on a couch
7. 3.20%: cartoon illustration of a cat and dog laying on a couch illustration of a cat and dog laying on a couch
8. 2.92%: cartoon illustration of a cat and dog lying on a couch cartoon dog laying on a pillow with a cat on it
9. 2.22%: cartoon illustration of a cat and dog lying on a couch illustration of a dog and cat sleeping on a couch together
10. 2.03%: cartoon illustration of a cat and dog lying on a couch cartoon illustration of a cat sleeping on a couch with a pillow
.
.
.
21. 0.05%: cartoon illustration of a cat and dog lying on a couch cartoon illustration of a man in a top hat and a suit

```

Imagen 25: Output

Una propuesta para tratar estos temas sería utilizar palabras claves compuestas, como parejas de sustantivo y adjetivos, de esta forma no podria eliminarse un token(palabra) de una descripcion sin que se elimine consigo la palabra que esta asociada, por ejemplo en este ultimo caso se tendría como tokens de la descripción original: `['cartoon-illustration', 'of', 'a', 'cat', 'and', 'dog-lying', 'on', 'a', 'couch', 'cartoon-dog', 'with', 'a', 'blue-collar', 'and', 'a', 'brown-nose']`. De esta forma se logra una mejor experiencia a la hora de rankear consultas mas precisas, claro en estos casos las palabras claves serían compuestas y separadas, de esta forma no dependes de poner exactamente ambas palabras como “cartoon illustration”, en este caso las consultas detalladas se tratan de manera orientada a ello.

Futuras implementaciones y propuestas

Una propuesta interesante para el proyecto sería implementar modelos para el procesamiento del lenguaje natural de las consultas, por ejemplo:

- Crear un modelo y dataset que tenga como objetivo extraer las palabras que no se desean recuperar, se crearía un dataset básico para el entrenamiento de este modelo, el cual estará creado sobre la base de las consultas más comunes por un usuario a la hora de buscar imágenes. Por ejemplo en la siguiente imagen:



Imagen 26

Se puede apreciar que aparece la palabra *cartoon*, una posible consulta de usuario que no desea recuperar una imagen *cartoon* podría ser: "*image of a dog and a cat, but no cartoon*". Este caso CLIP no es capaz de entender al usuario en la frase "*no cartoon*" por lo cual una opción para evitar recuperar imágenes no deseadas es dar ranking a estas teniendo en cuenta que no queremos que la palabra *cartoon* aparezca en la descripción de la misma. La dataset para este problema sería generada manualmente con ayuda de modelos de lenguaje como GPT-3.

Detalles

Los modelos y códigos utilizados son computacionalmente costosos que no se pueden correr en procesadores sin GPU, usando GPU el proceso de generar la descripción tarda entre 20-40 segundos para imágenes de más de 1000px*1000px con un promedio de 20-30 segmentaciones para un área mínima de partición asignada manualmente (imagen partida entre 64). Para crear un software automático que utilice esto se necesita hostearlo en un hardware potente para ello. A continuación se especifican algunos detalles de interés:

- Repositorio de github donde se está desarrollando [https://github.com/rb58853/images_RIS-ML-Conv-NLP].
- Enlace a Colab para probar el modelo [[notebook](#)].

Bibliografía

1. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. *Junnan Li Dongxu Li Caiming Xiong Steven Hoi Salesforce Research.* [[paper](#)]

2. Hugginface BLIP [[huggingface-blip-image-captioning-large](#)]
3. Repositorio de github oficial del modelo BLIP
[<https://github.com/salesforce/BLIP>]
4. Blog oficial de BLIP [[blog](#)]
5. Primera publicacion sobre el modelo BLIP en internet [[deepai-blip](#)]
6. Sitio oficial de CLIP en OpenAI [[clip-openai.com](#)].
7. CLIP: Learning Transferable Visual Models From Natural Language Supervision.
Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, Ilya Sutskever [[paper](#)]
8. Repositorio de github oficial del modelo BLIP [<https://github.com/openai/CLIP>]
9. Hugginface CLIP [[hugginface-clip](#)]
10. Segment Anything Model (SAM) [[web oficial](#)].
11. Segment Anything. *Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao...* [[paper](#)]
12. Introducing Segment Anything: Working toward the first foundation model for image segmentation [[blog](#)]
13. Repositorio oficial de github Segment-Anything
[<https://github.com/facebookresearch/segment-anything>]
14. Repositorio de github BLIP-2 [[githug-LAVIS/BLIP-2](#)]
15. Hugginface BLIP-2 [[huggingface-BLIP2](#)]
16. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. *Junnan Li Dongxu Li Silvio Savarese Steven Hoi*[[paper](#)]