Multimodal LLMs

CS XXX: Introduction to Large Language Models

Contents

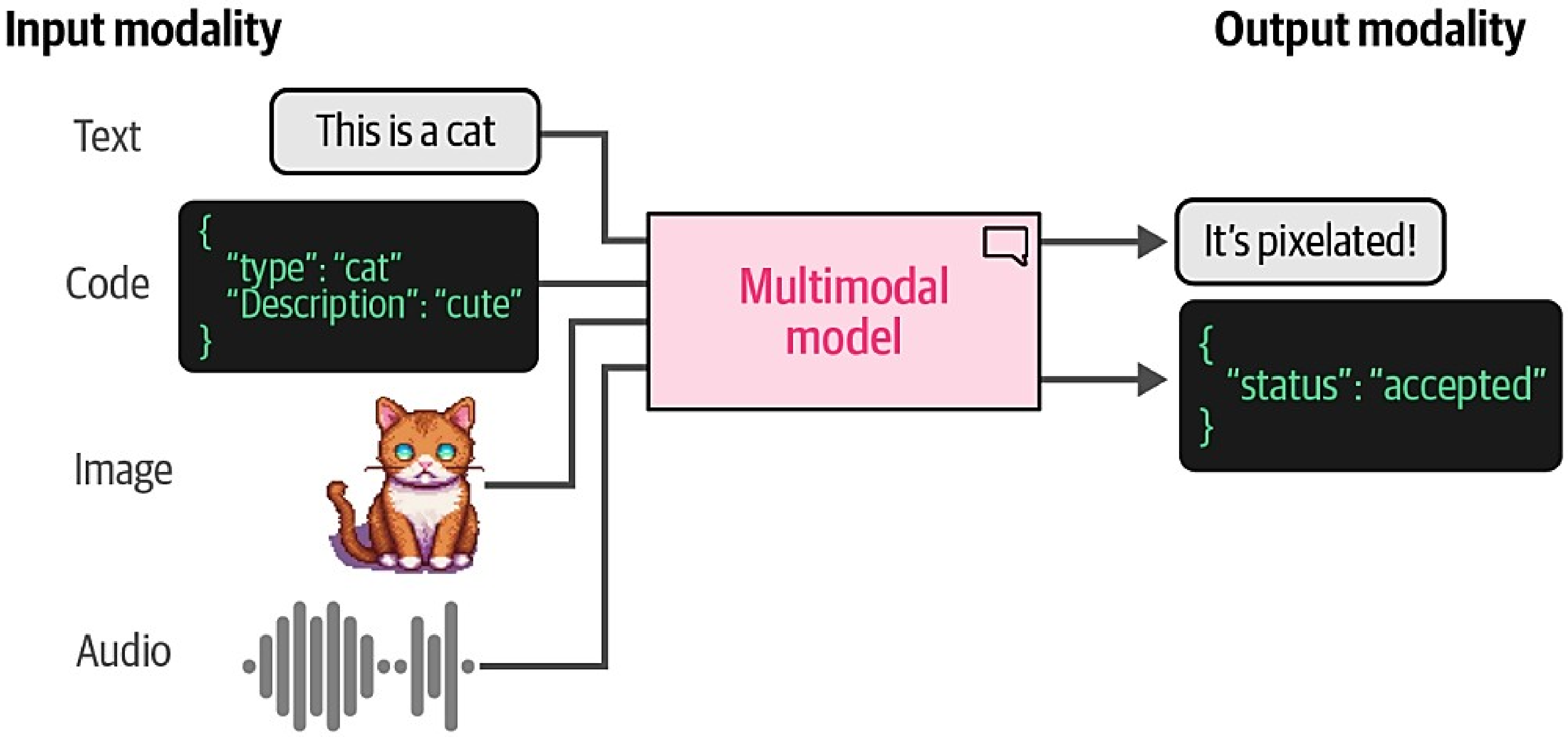
- Multimodality
- Transformers for Vision
- Multimodal Embedding Models
- CLIP
- Making Text Generation Models Multimodal – BLIP2
- BLIP-2 Use Cases

Multimodality

- Language models can be much more useful if they’re able to handle types of data other than text. It’s very useful, for example, if a language model is able to glance at a picture and answer questions about it.

Multimodality

- Models that are able to deal with different types (or modalities) of data, such as images, audio, video, or sensors, are said to be multimodal. It’s possible for a model to accept a modality as input yet not be able to generate in that modality.



```
{
  “type”: “cat”
  “Description”: “cute”
}
```

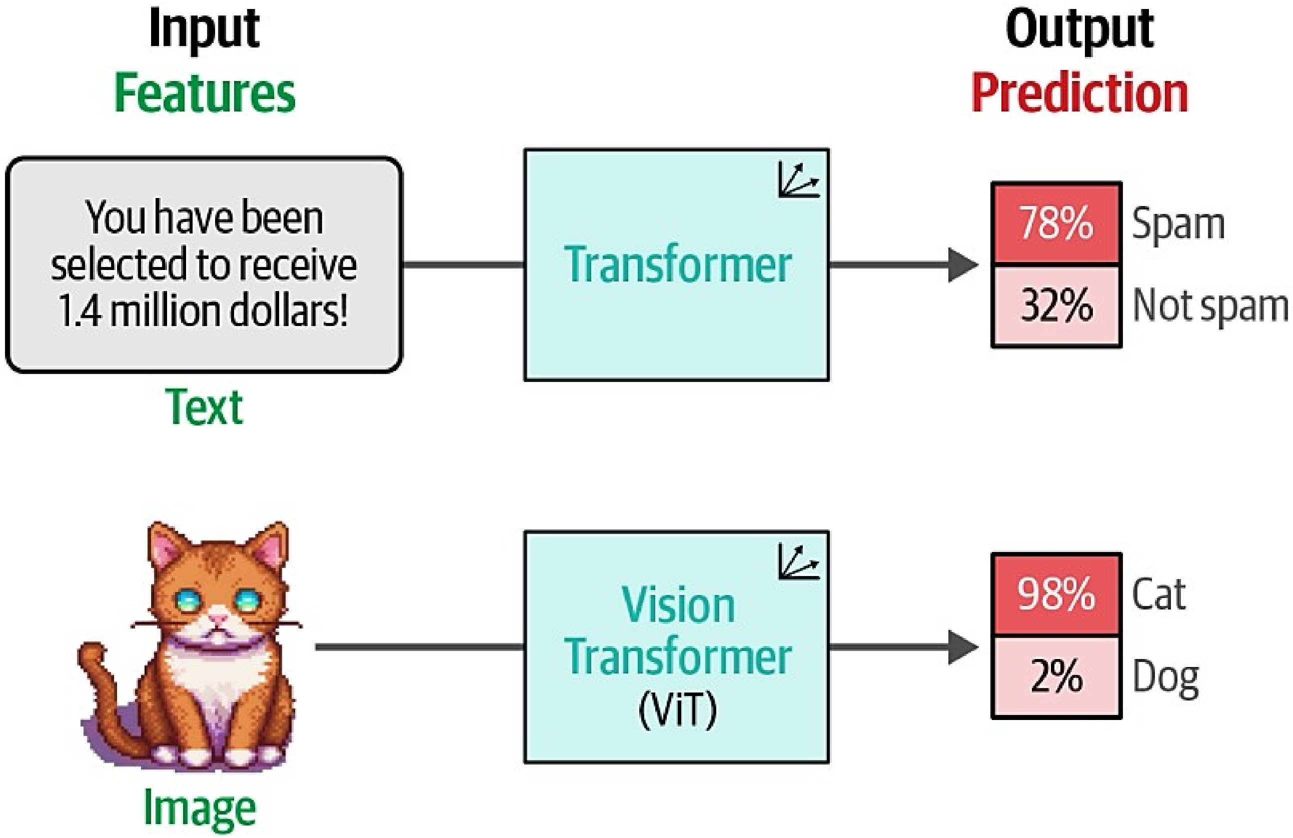
```
{
  “status”: “accepted”
}
```

Transformers for Vision

- we have seen the success of using Transformer based models for a variety of language modeling tasks, from classification and clustering to search and generative modeling. So it might not be surprising that researchers have been looking at a way to generalize some of the Transformer’s success to the field of computer vision.
- The method they came up with is called the Vision Transformer (ViT), which has been shown to do tremendously well on image recognition tasks.

Transformers for Vision

- Like the original Transformer, ViT is used to transform unstructured data, an image, into representations that can be used for a variety of tasks, like classification.



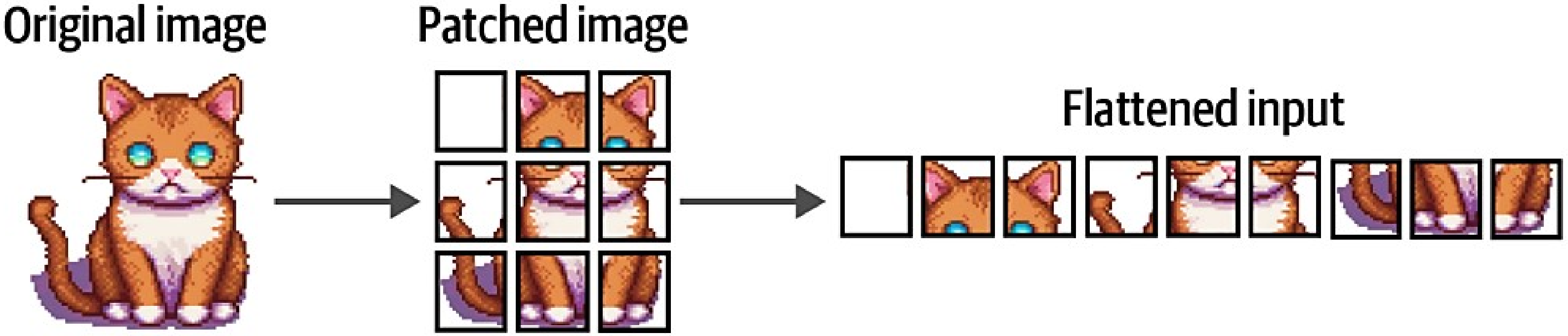
Transformers for Vision

- Since an image does not consist of words the tokenization process cannot be used for visual data. Instead, the authors of ViT came up with a method for tokenizing images into “words,” which allowed them to use the original encoder structure.

- Imagine that you have an image of a cat. This image is represented by a number of pixels, let’s say 512 × 512 pixels. Each individual pixel does not convey much information but when you combine patches of pixels, you slowly start to see more information.

Transformers for Vision

- A ViT converts the original image into patches of images (sub-images). In other words, it cuts the image into a number of pieces horizontally and vertically. The image in the example is patched into 3 × 3 patches.



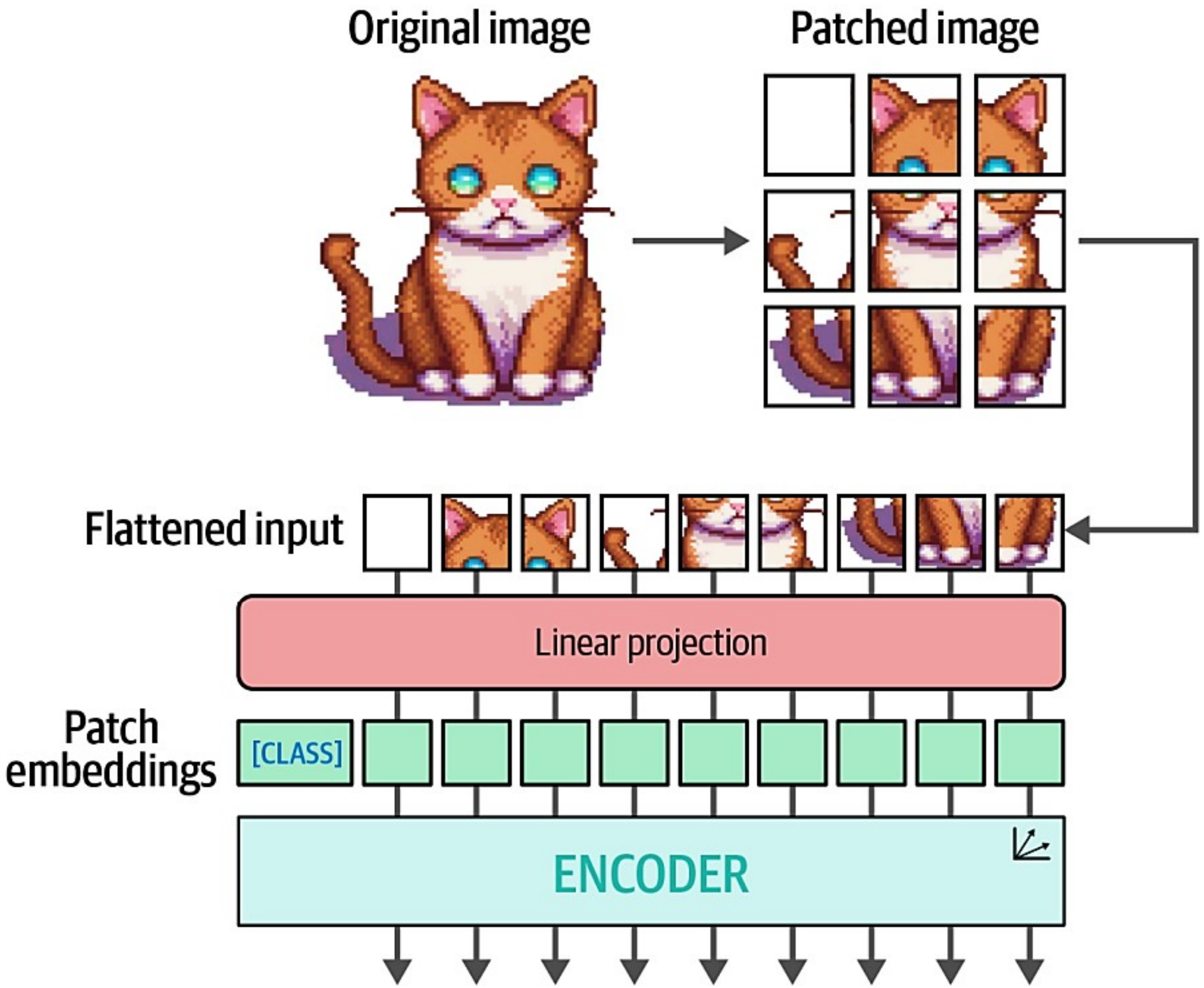
- The flattened input of image patches can be thought of as the tokens in a piece of text.

Transformers for Vision

- However, unlike tokens, we cannot just assign each patch with an ID since these patches will rarely be found in other images, unlike the vocabulary of a text.
- Instead, the patches are linearly embedded to create numerical representations, namely embeddings. These can then be used as the input of a Transformer model.

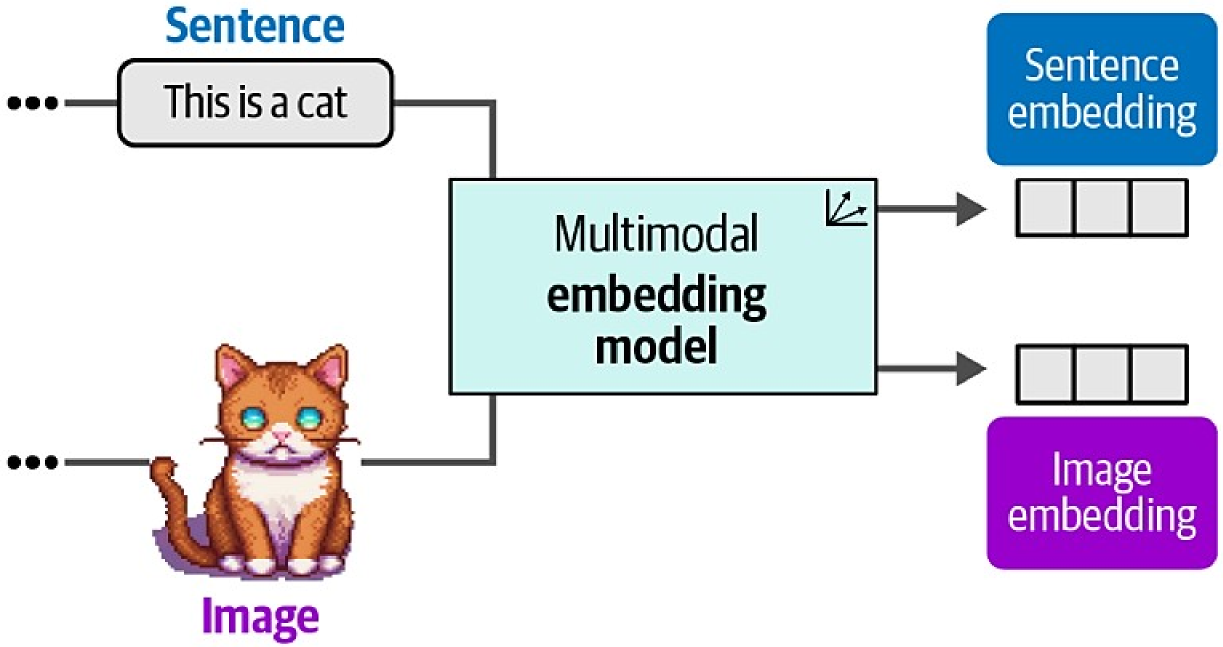
Transformers for Vision

- After patching the images and linearly projecting them, the patch embeddings are passed to the encoder and treated as if they were textual tokens.



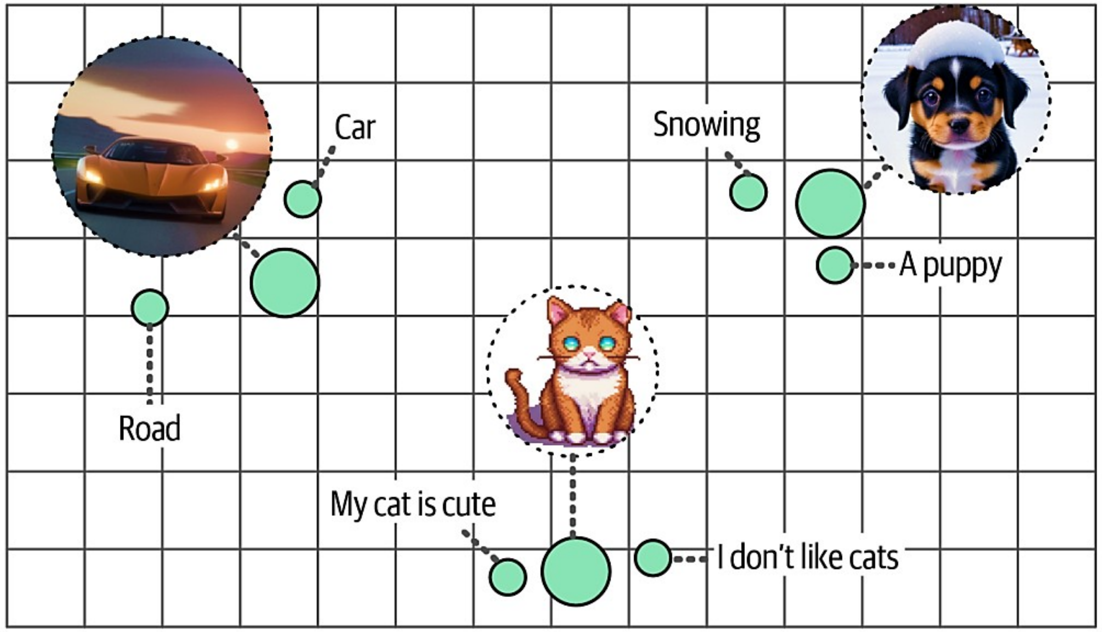
Multimodal Embedding Models

- We have looked at text-only embedding models thus far, which focus on generating embeddings for textual representations. Although embedding models exist for solely embedding imagery, we will look at embedding models that can capture both textual as well as visual representations.
- Multimodal embedding models can create embeddings for multiple modalities in the same vector space.



Multimodal Embedding Models

- Using a multimodal embedding model, we can find images based on input text. What images would we find if we search for images similar to “pictures of a puppy”? Vice versa would also be possible.



- Despite having coming from different modalities, embeddings with similar meaning will be close to each other in vector space.

CLIP

- There are a number of multimodal embedding models, but the most well-known and currently most-used model is Contrastive Language-Image Pre-training (CLIP).

CLIP

- CLIP is an embedding model that can compute embeddings of both images and texts. The resulting embeddings lie in the same vector space, which means that the embeddings of images can be compared with the embeddings of text. This comparison capability makes CLIP, and similar models, usable for tasks such as:
  - **Zero-shot classification:** We can compare the embedding of an image with that of the description of its possible classes to find which class is most similar.
  - **Clustering:** Cluster both images and a collection of keywords to find which keywords belong to which sets of images.
  - **Search:** Across billions of texts or images, we can quickly find what relates to an input text or image.
  - **Generation:** Use multimodal embeddings to drive the generation of images (e.g., stable diffusion4).

CLIP

- Imagine that you have a dataset with millions of images alongside captions

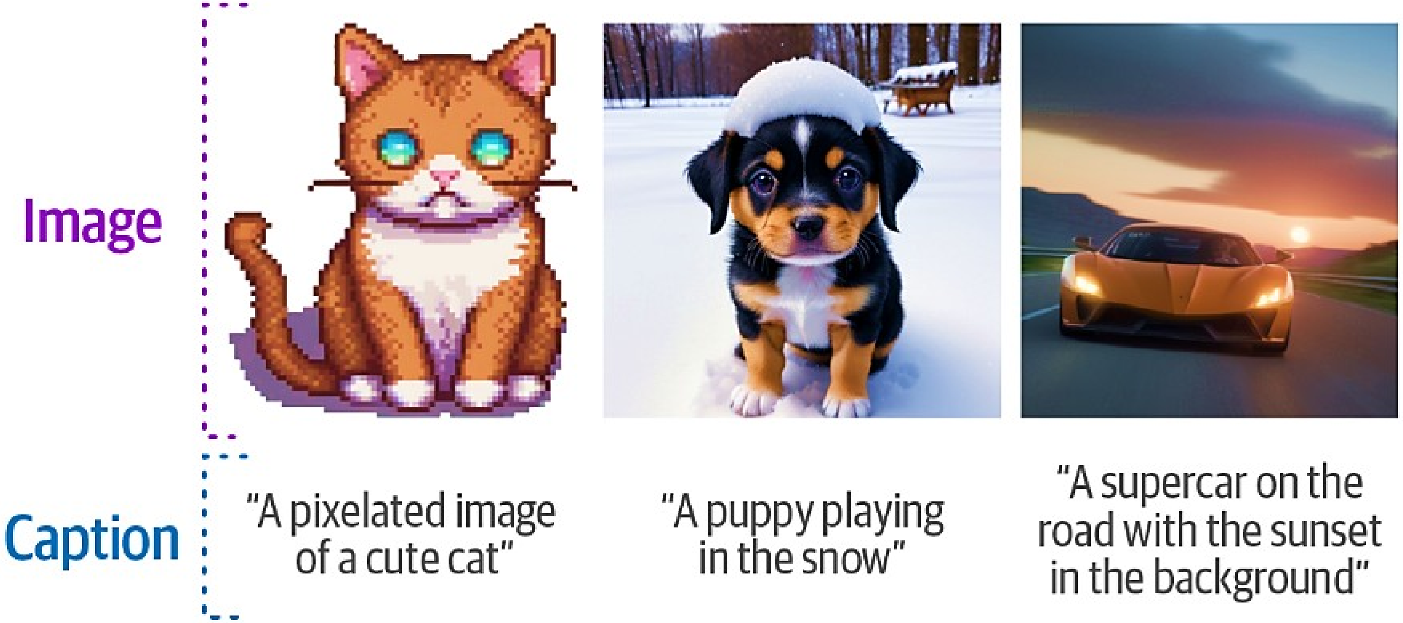
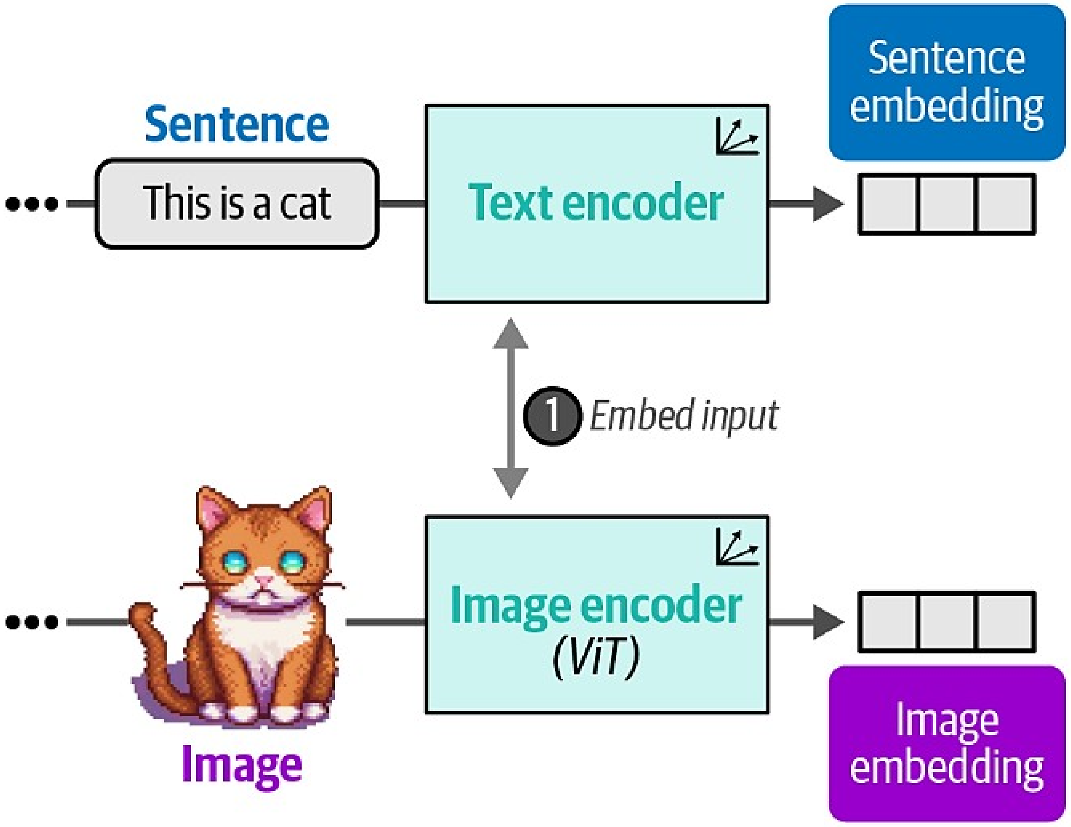


Image
Caption
"A pixelated image of a cute cat"
"A puppy playing in the snow"
"A supercar on the road with the sunset in the background"

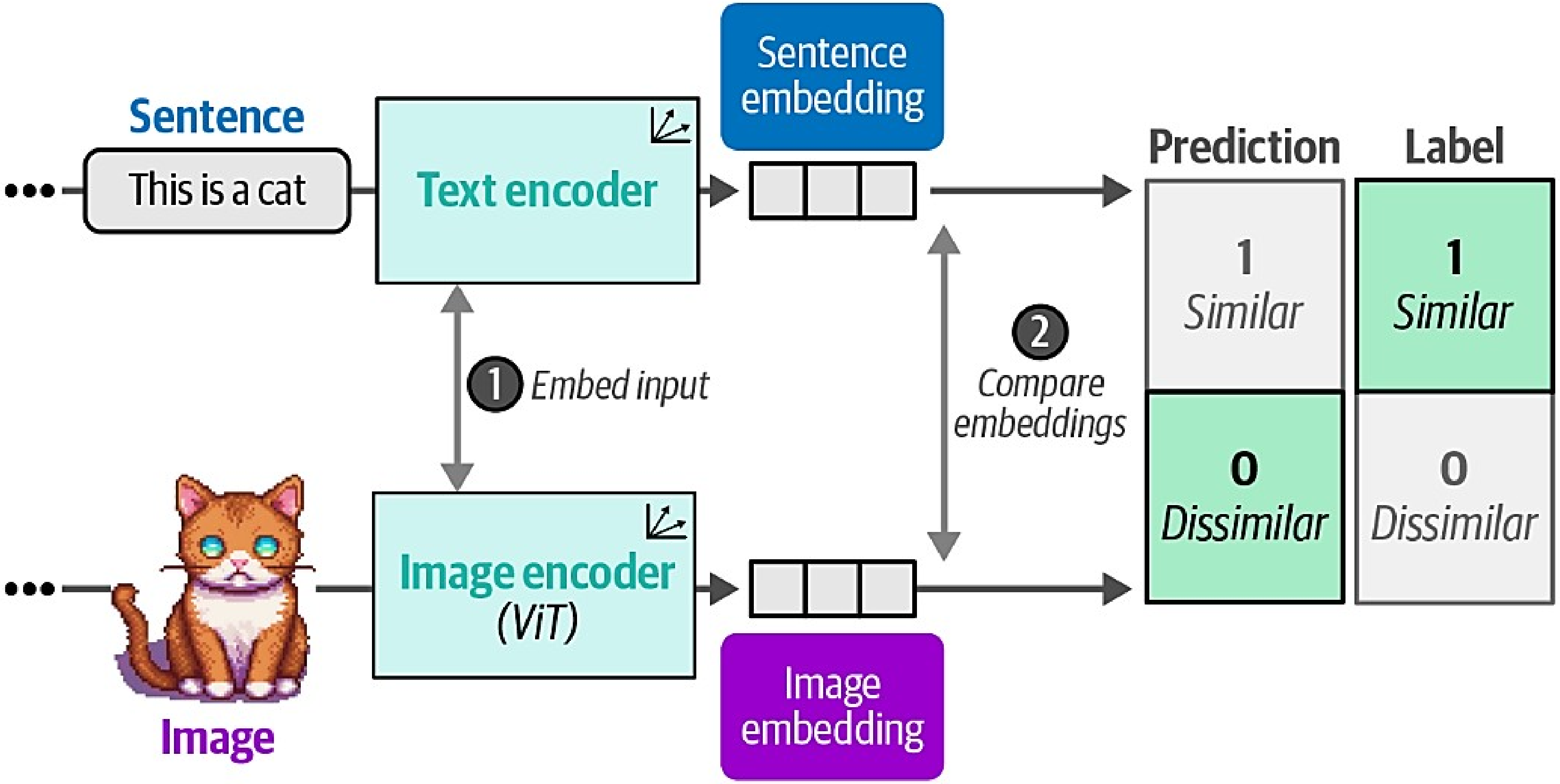
CLIP

- In the first step of training CLIP, the dataset is used to create two representations for each pair, the image and its caption. To do so, CLIP uses a text encoder to embed text and an image encoder to embed images.



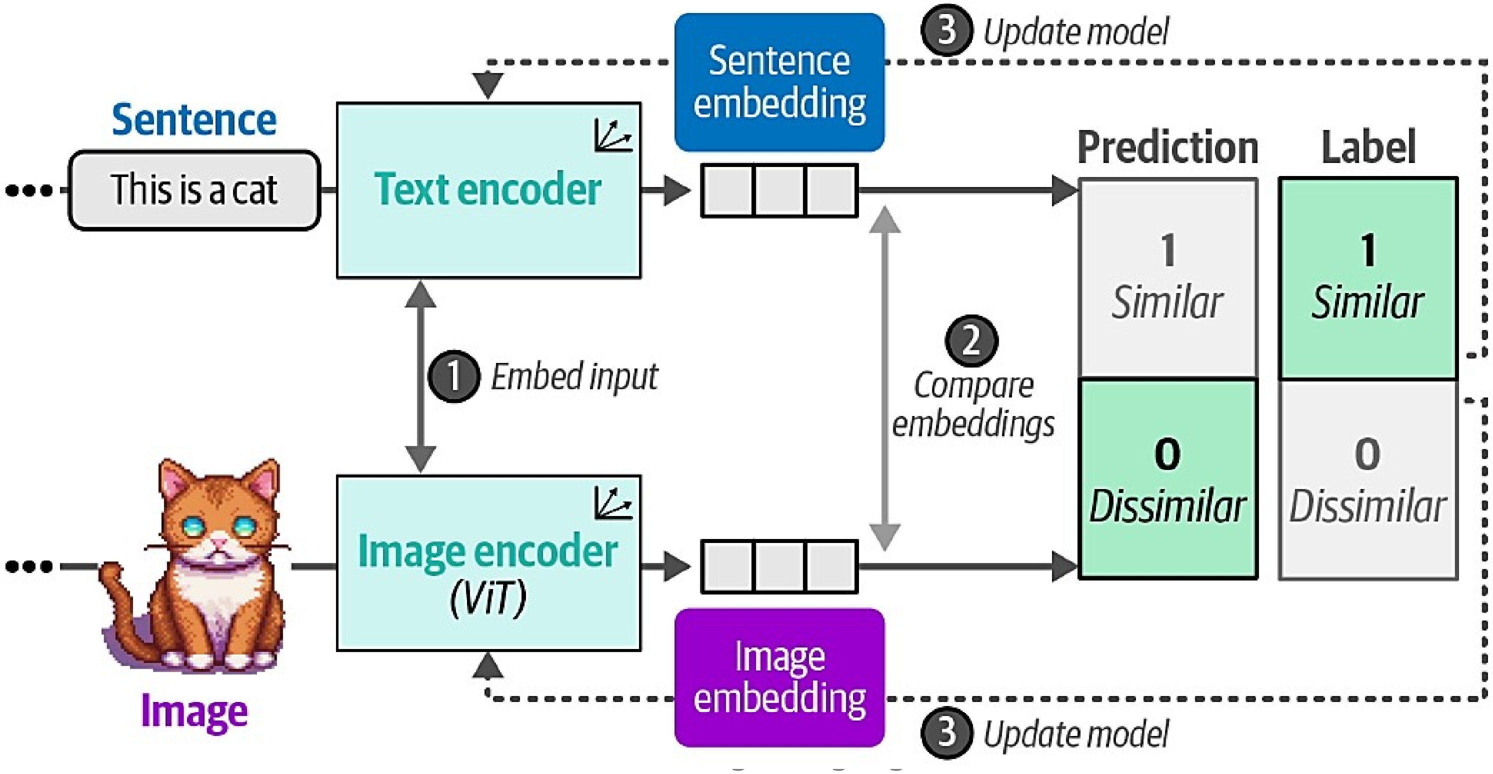
CLIP

- In the second step of training CLIP, the similarity between the sentence and image embedding is calculated using cosine similarity.



CLIP

- In the third step of training CLIP, the text and image encoders are updated to match what the intended similarity should be. This updates the embeddings such that they are closer in vector space if the inputs are similar. Eventually, we expect the embedding of an image of a cat would be similar to the embedding of the phrase “a picture of a cat.”

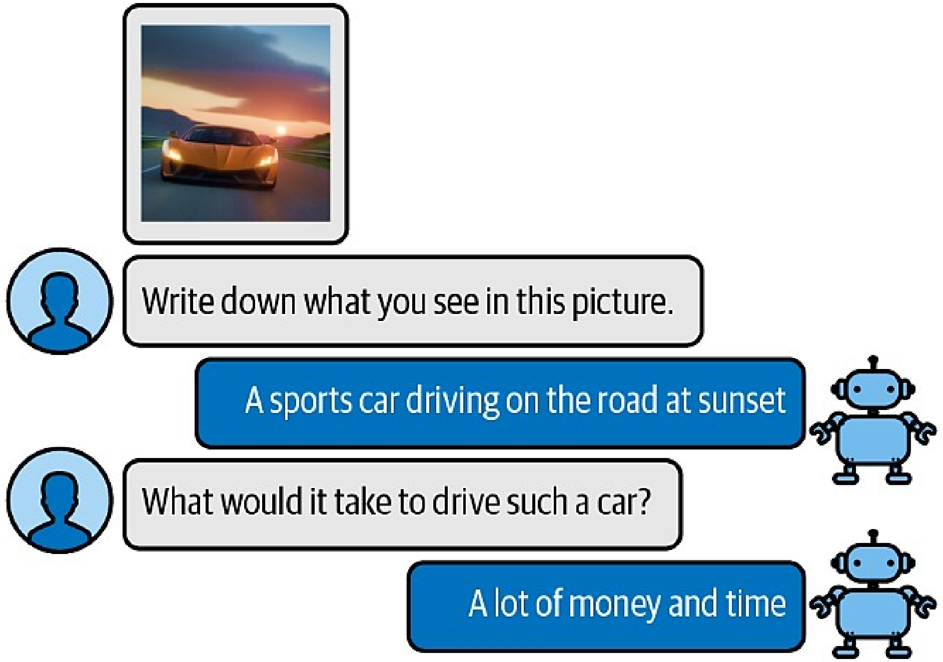


Making Text Generation Models Multimodal – BLIP2

- Traditionally, text generation models have been, limited to the modality they were trained in, namely text. As we have seen before with multimodal embedding models, the addition of vision can enhance the capabilities of a model.
- In the case of text generation models, we would like it to reason about certain input images. For example, we could give it an image of a pizza and ask it what ingredients it contains.

Making Text Generation Models Multimodal – BLIP2

- To bridge the gap between these two domains, attempts have been made to introduce a form of multimodality to existing models. One such method is called BLIP-2.
- BLIP-2 is an easy-to-use and modular technique that allows for introducing vision capabilities to existing language models. An example of reasoning about input images:



Write down what you see in this picture.

A sports car driving on the road at sunset

What would it take to drive such a car?

A lot of money and time

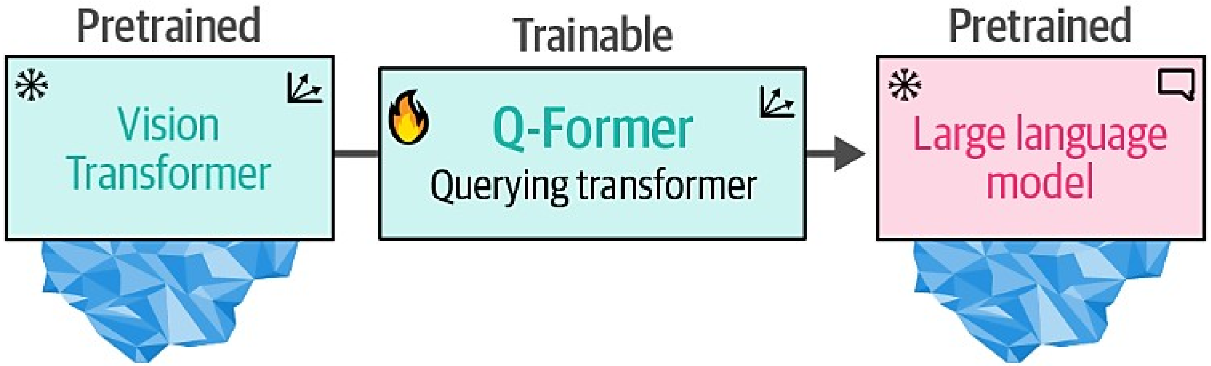
Making Text Generation Models Multimodal – BLIP2

- Creating a multimodal language model from scratch requires significant computing power and data. We would have to use billions of images, text, and image-text pairs to create such a model. As you can imagine, this is not easily feasible!
- Instead of building the architecture from scratch, BLIP-2 bridges the vision-language gap by building a bridge, named the Querying Transformer (Q-Former), that connects a pretrained image encoder and a pretrained LLM.

Making Text Generation Models Multimodal – BLIP2

- By leveraging pretrained models, BLIP-2 only needs to train the bridge without needing to train the image encoder and LLM from scratch.

- The Querying Transformer is the bridge between vision (ViT) and text (LLM) that is the only trainable component of the pipeline.



Making Text Generation Models Multimodal – BLIP2

- To connect the two pretrained models, the Q-Former mimics their architectures. It has two modules that share their attention layers:
  - An Image Transformer to interact with the frozen Vision Transformer for feature extraction
  - A Text Transformer that can interact with the LLM
- The Q-Former is trained in two stages, one for each modality.
  - In step 1, representation learning is applied to learn representations for vision and language simultaneously.
  - In step 2, these representations are converted to soft visual prompts to feed the LLM.

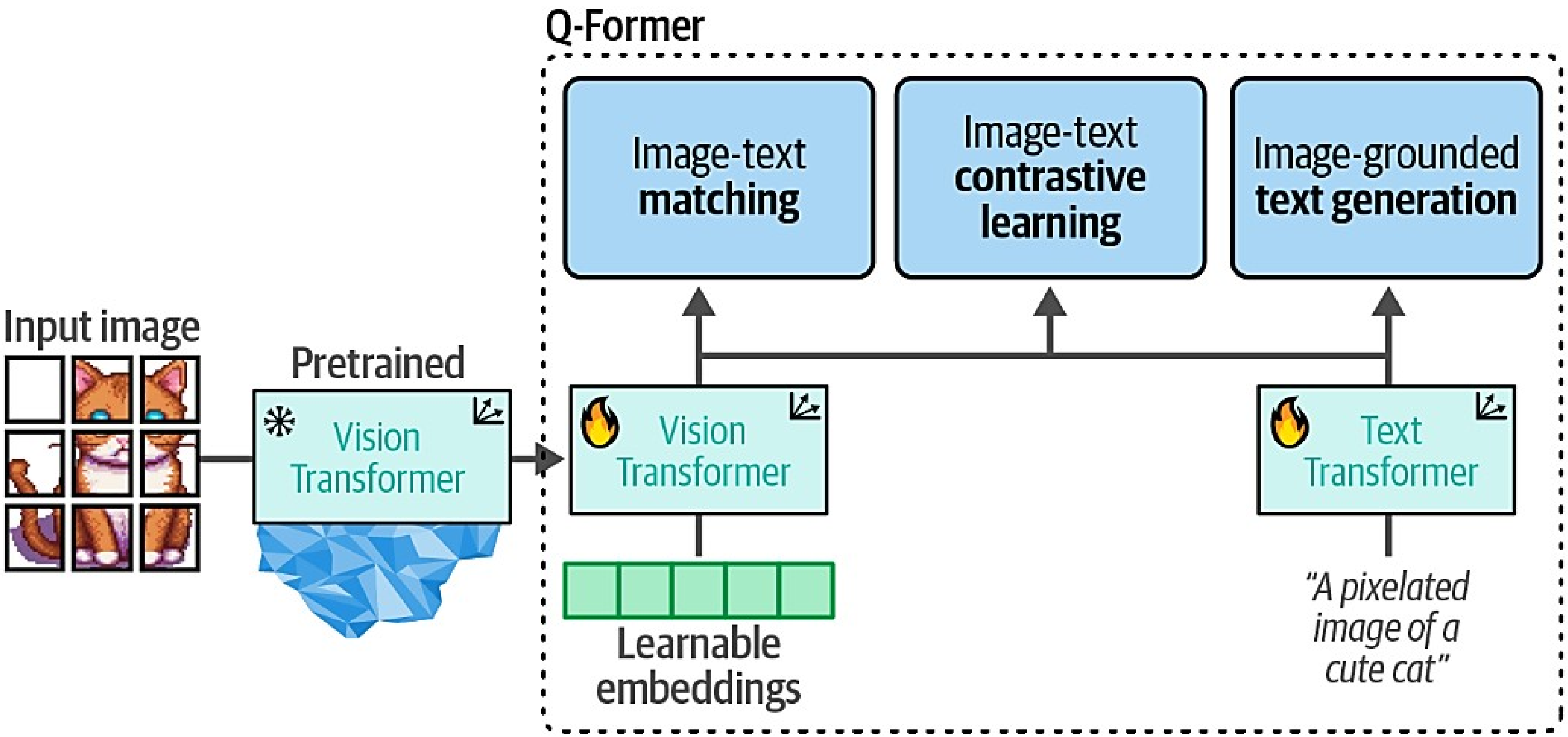
Making Text Generation Models Multimodal – BLIP2

- Training Q-Former Step 1:
  - image-document pairs are used to train the Q-Former to represent both images and text. These pairs are generally captions of images.
  - The images are fed to the frozen ViT to extract vision embeddings. These embeddings are used as the input of Q-Former’s ViT. The captions are used as the input of Q-Former’s Text Transformer
  - With these inputs, the Q-Former is then trained on three tasks:
    - Image-text contrastive learning: This task attempts to align pairs of image and text embeddings such that they maximize their mutual information.
    - Image-text matching: A classification task to predict whether an image and text pair is positive (matched) or negative (unmatched).
    - Image-grounded text generation: Trains the model to generate text based on information extracted from the input image.

    These three objectives are jointly optimized to improve the visual representations that are extracted from the frozen ViT.

Making Text Generation Models Multimodal – BLIP2

- Training Q-Former Step 1:
  - In step 1, the output of the frozen ViT is used together with its caption and trained on three contrastive-like tasks to learn visual-text representations.

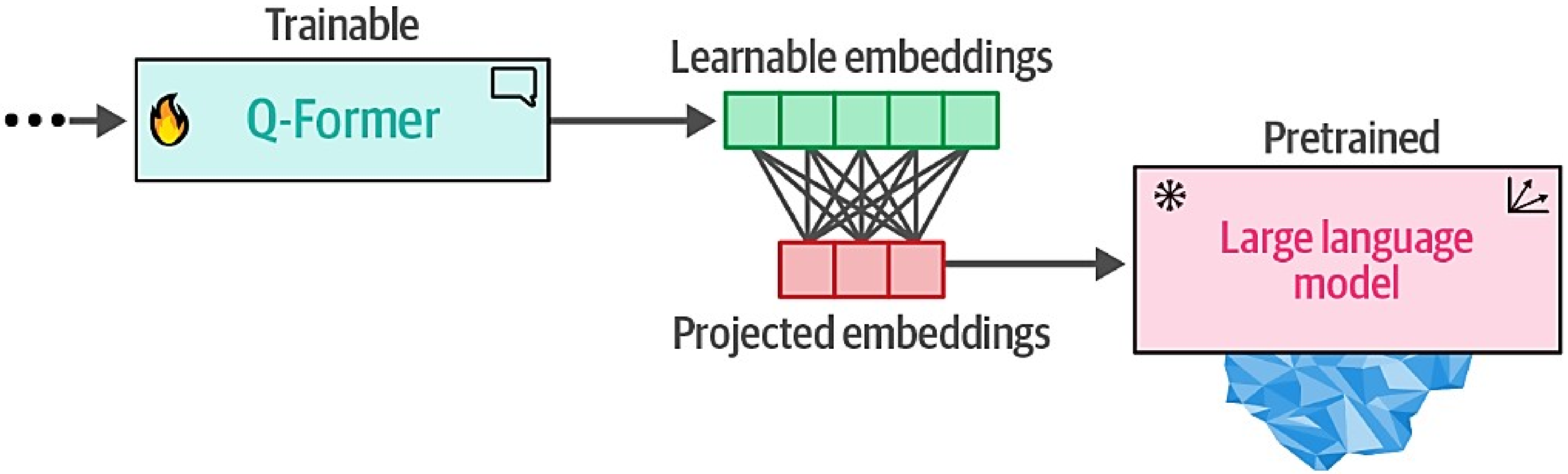


Making Text Generation Models Multimodal – BLIP2

- Training Q-Former Step 2:
  - In step 2, the learnable embeddings derived from step 1 now contain visual information in the same dimensional space as the corresponding textual information. The learnable embeddings are then passed to the LLM. In a way, these embeddings serve as soft visual prompts that condition the LLM on the visual representations that were extracted by the Q-Former.
  - There is also a fully connected linear layer in between them to make sure that the learnable embeddings have the same shape as the LLM expects.

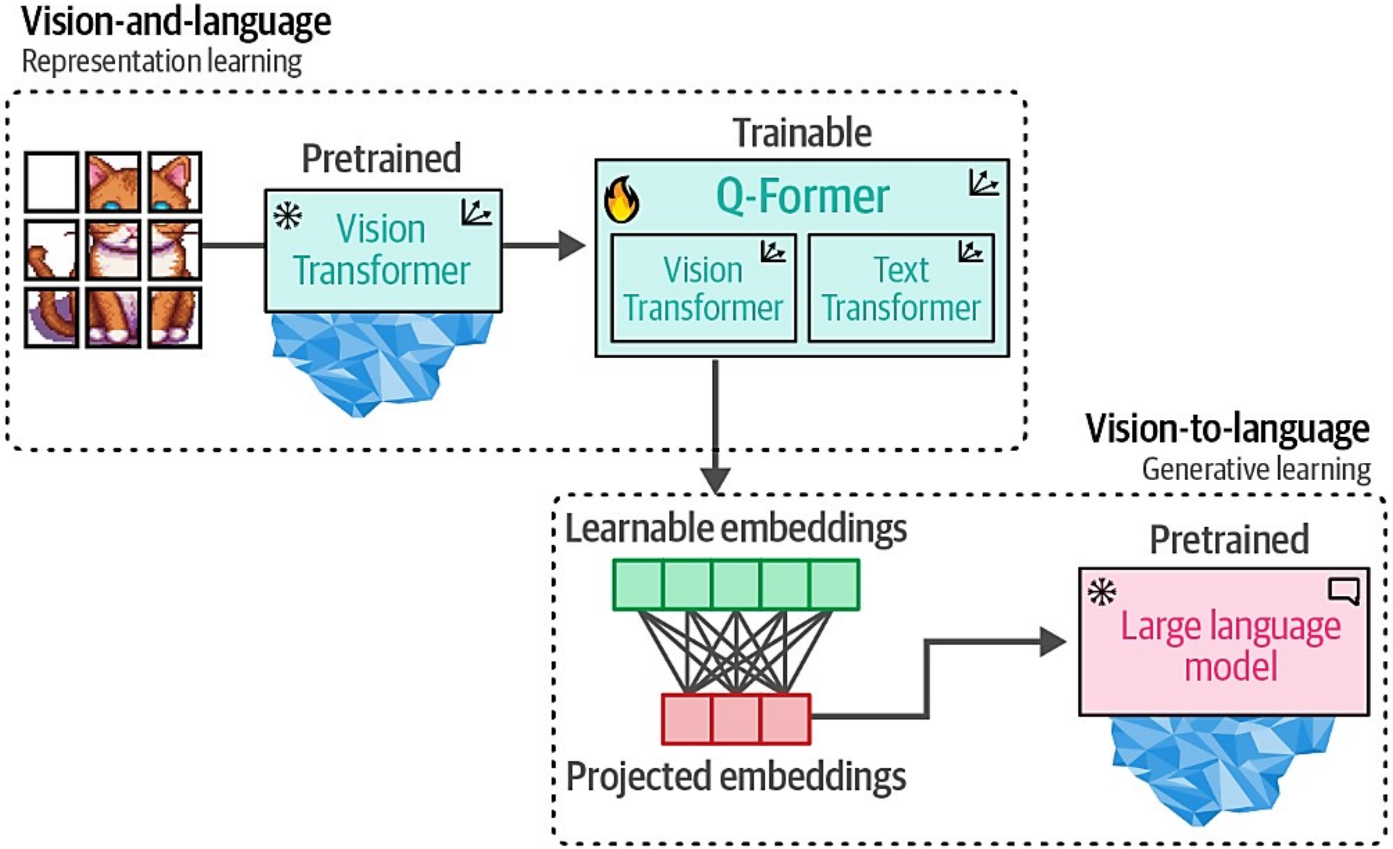
Making Text Generation Models Multimodal – BLIP2

- Training Q-Former Step 2:
  - In step 2, the learned embeddings from the Q-Former are passed to the LLM through a projection layer. The projected embeddings serve as a soft visual prompt



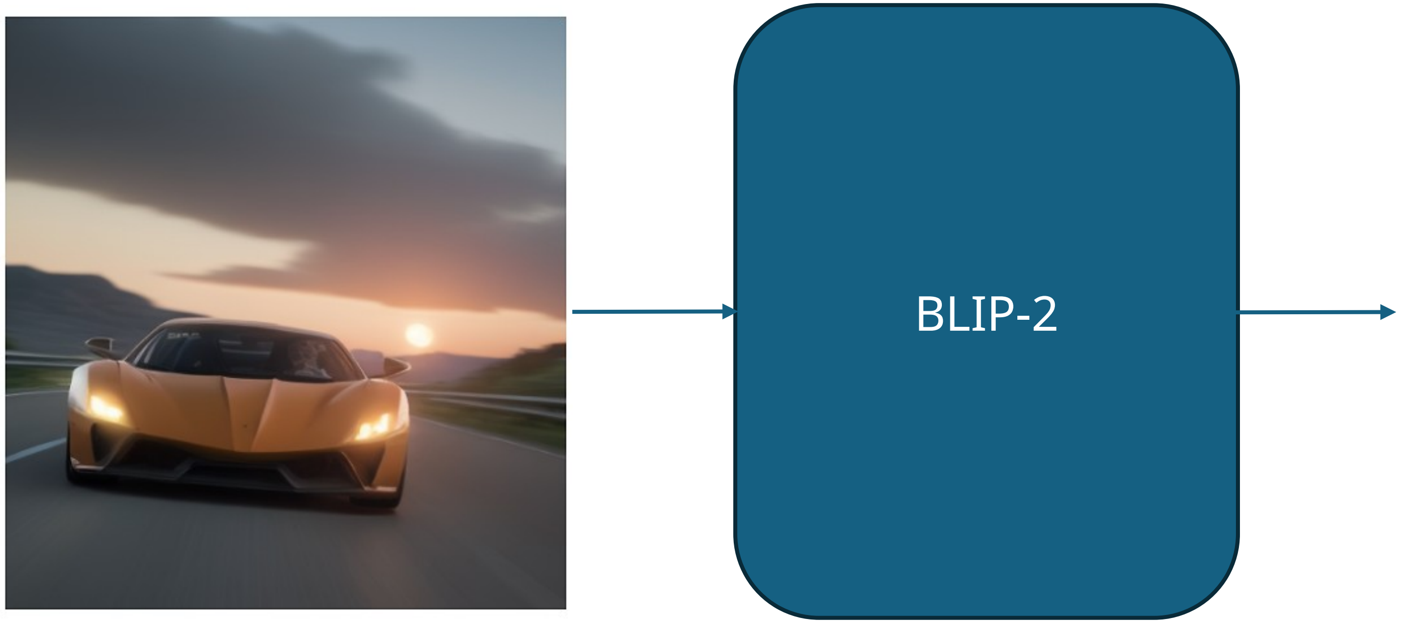
Making Text Generation Models Multimodal – BLIP2

- When we put these steps together, they make it possible for the Q-Former to learn visual and textual representations in the same dimensional space, which can be used as a soft prompt to the LLM. As a result, the LLM will be given information about the image in a similar manner to the context you would provide an LLM when prompting.



BLIP-2 Use Cases

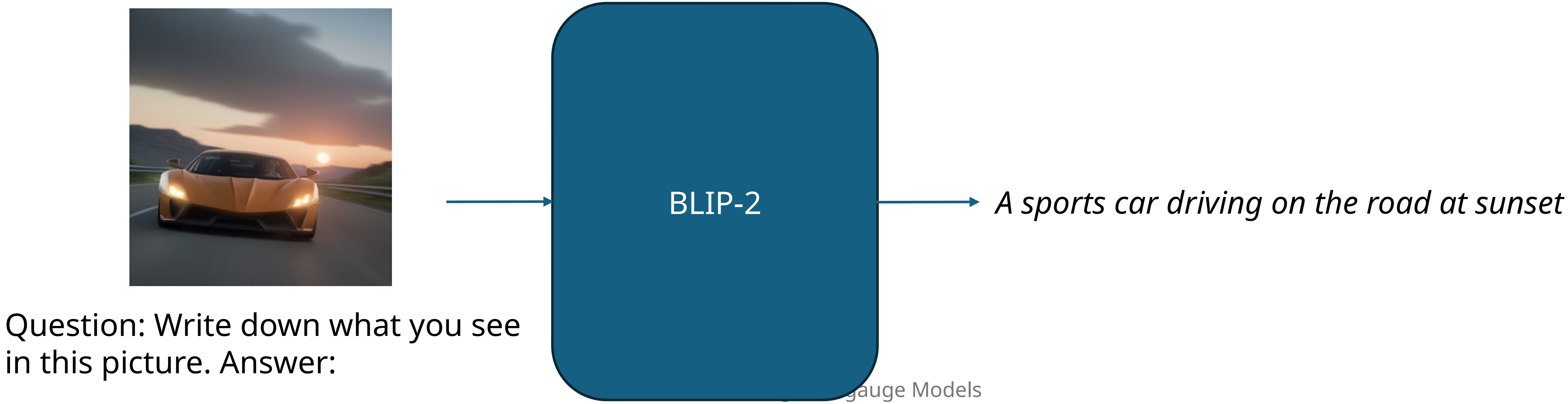
- Now that we know how BLIP-2 is created, there are a number of interesting use cases for such a model. For example
- Use Case 1: Image Captioning
  - The most straightforward usage of a model like BLIP-2 is to create captions of images that you have in your data.



An orange supercar driving on the road at sunset

BLIP-2 Use Cases

- Now that we know how BLIP-2 is created, there are a number of interesting use cases for such a model. For example
- Use Case 2: Multimodal Chat-Based Prompting
  - In this particular use case, we give the model an image along with a question about that specific image for it to answer. The model needs to process both the image as well as the question at once. We present both modalities simultaneously by performing what is called visual question answering.



Question: Write down what you see in this picture. Answer:

A sports car driving on the road at sunset

References

- Alammar, J., & Grootendorst, M. Hands-On Large Language Models: Language Understanding and Generation. O'Reilly Media.