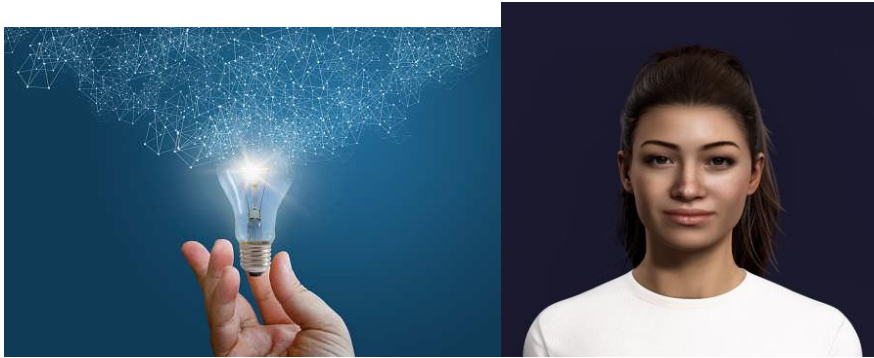


Result Analysis of 'Image to Text Generation' models:

Input images:



Url input image: <https://imgv3.fotor.com/images/side/create-white-and-red-runsters-with-fotor-random-car-generator.jpg>



=====

1. Using the pipelines of transformers.

```
from transformers import pipeline

captioner = pipeline("image-to-text", model="Salesforce/blip-image-captioning-base")
captioner("/content/sample_data/human1.png")
```

Output:

```
[{'generated_text': 'a woman with a white shirt and a black background'}]
```

Output time: 7s.

Url input:

```
captioner("https://imgv3.fotor.com/images/side/create-white-and-red-runsters-with-fotor-random-car-generator.jpg")
```

Output:

```
[{'generated_text': 'two different views of the new sports cars'}]
```

Output time: 5s.

2. Using BlipProcessor (blip-image-captioning-base):

```
import requests
from PIL import Image
```

```

from transformers import BlipProcessor,
BlipForConditionalGeneration

processor = BlipProcessor.from_pretrained("Salesforce/blip-image-captioning-base")
model =
BlipForConditionalGeneration.from_pretrained("Salesforce/blip-image-captioning-base")

```

Input:

```
img_url = 'https://imgv3.fotor.com/images/side/create-white-and-red-runsters-with-fotor-random-car-generator.jpg'
```

Output:

```

a photography of a red and a white sports car
two different views of the new sports cars

```

Output time: 11s

3. Using BlipProcessor (blip-image-captioning-large):

```

import torch
import requests
from PIL import Image
from transformers import BlipProcessor,
BlipForConditionalGeneration

processor = BlipProcessor.from_pretrained("Salesforce/blip-image-captioning-large")
model =
BlipForConditionalGeneration.from_pretrained("Salesforce/blip-image-captioning-large", torch_dtype=torch.float16).to("cuda")

```

Input:

```
img_url = 'https://imgv3.fotor.com/images/side/create-white-and-red-runsters-with-fotor-random-car-generator.jpg'
```

Output:

```

a photography of two sports cars side by side on a road
two pictures of a red and silver sports car and a silver sports car

```

Output time: 1s.

4. Using AutoProcessor(microsoft/kosmos-2-patch14-224):

```

import requests

from PIL import Image
from transformers import AutoProcessor, AutoModelForVision2Seq

```

```
model = AutoModelForVision2Seq.from_pretrained("microsoft/kosmos-2-patch14-224")
processor = AutoProcessor.from_pretrained("microsoft/kosmos-2-patch14-224")

prompt = "<grounding>An image of"

url = "https://imgv3.fotor.com/images/side/create-white-and-red-runsters-with-fotor-random-car-generator.jpg"
```

Output:

```
An image of a red 2019 Chevy Corvette and an image of a silver
2019 Chevy ZR1 convertible parked on a road.
```

Output time: 11s.