ANL488
Business Analytics
Applied Project

Image to Audio Generator

Via the Transformer Model Network

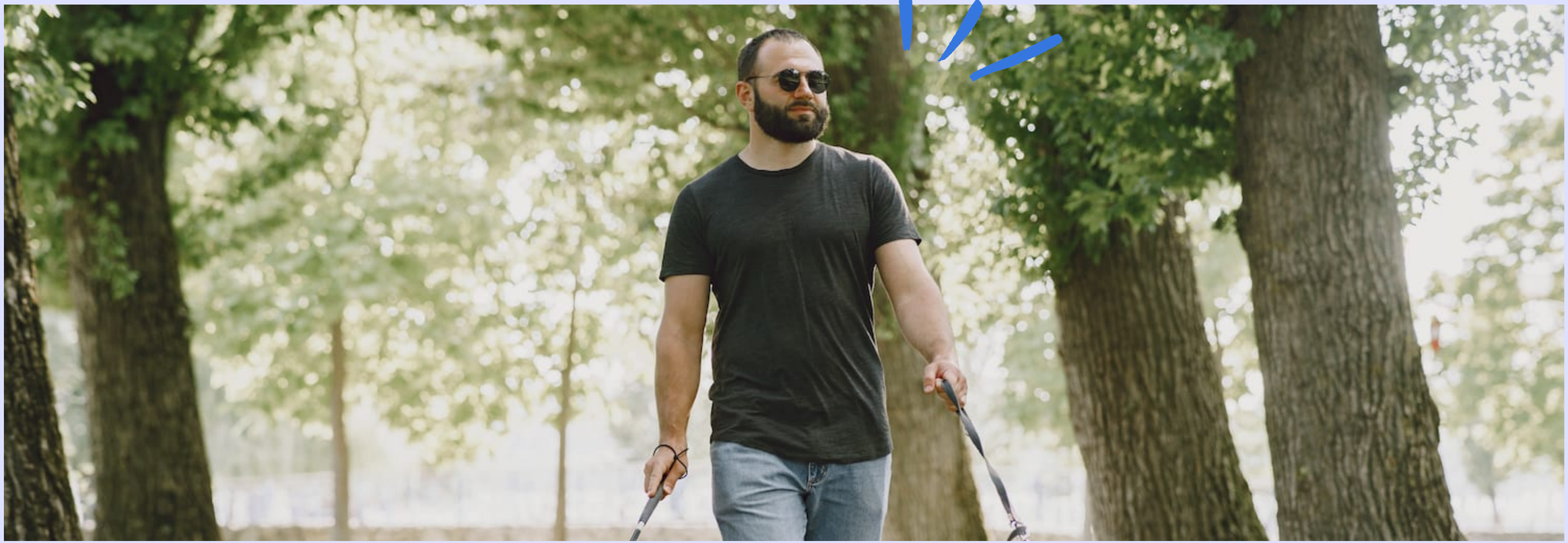
Presented by:
Koh Wei Jie Edmund

Agenda

1. Introduction
2. Objective
3. Literature Review
4. Model Exploration
5. Model Evaluation
6. Proposed Modelling
7. Tool Optimisation
8. Recommendation

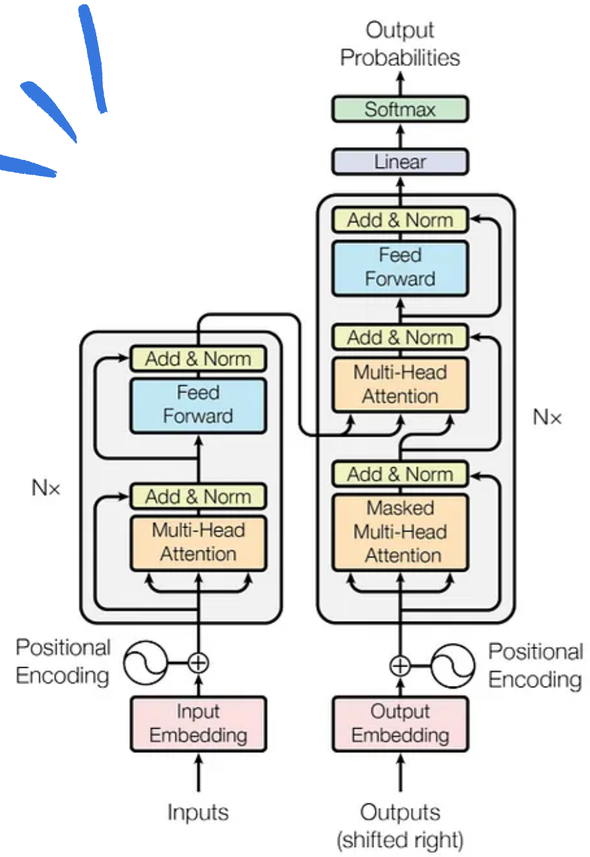
Motivation

Image-to-Audio Generator Via the Transformer Model Network



Business Problem	Analytics Objective
• The global cost of productivity losses due to vision impairment is projected to be US$ 411 billion.	• Develop a state-of-the-art image captioning model that would accurately depict the details of an image

What is Transformer?



- “Attention Is All You Need,” published in 2017
- No Labels, More Performance
- State-of-the-art technique in the field of NLP

Literature Review (Related Work)

Dosovitskiy et al. (2020)	ViT is found to be the best performer when the computational cost of pre-training the model is taken into account reaching state-of-the-art on the majority of recognition benchmarks at a lower pre-training cost.
Vaswani et al. (2017)	Transformer model used has achieved a BLEU score of 28.4 which outperform the best-performing model including the ensembles model by more than 2.0 BLEU while using only a fraction of the training cost of others.
Ghandi et al. (2023)	The future of image captioning would revolve around Vision-language pre-training (VLP) methods and Transformers as they are likely to be inseparable components of the models.

Model Selection

Tasks 1 | Libraries | Datasets | Languages | Licenses | Other

Filter Tasks by name | Reset Tasks

Multimodal

- Feature Extraction
- Text-to-Image
- Image-to-Text
- Text-to-Video
- Visual Question Answering
- Document Question Answering
- Graph Machine Learning

Computer Vision

- Depth Estimation
- Image Classification
- Object Detection
- Image Segmentation
- Image-to-Image
- Unconditional Image Generation
- Video Classification
- Zero-Shot Image Classification

Natural Language Processing

- Text Classification
- Token Classification
- Table Question Answering
- Question Answering
- Zero-Shot Classification
- Translation
- Summarization
- Conversational
- Text Generation
- Text2Text Generation
- Fill-Mask
- Sentence Similarity

Models 266 | Filter by name | new Full-text search | ↑↓ Sort: Most Downloads

- nlpconnect/vit-gpt2-image-captioning — Image-to-Text • Updated Feb 27 • 1.92M • 548
- Salesforce/blip-image-captioning-large — Image-to-Text • Updated Aug 1 • 1.17M • 359
- Salesforce/blip-image-captioning-base — Image-to-Text • Updated Aug 1 • 506k • 224
- microsoft/trocr-base-handwritten — Image-to-Text • Updated Jan 26 • 333k • 93
- microsoft/git-large-coco — Image-to-Text • Updated Jun 27 • 146k • 65
- Salesforce/blip2-opt-2.7b — Image-to-Text • Updated 14 days ago • 136k • 138
- microsoft/trocr-large-printed — Image-to-Text • Updated Jan 25 • 123k • 31
- Salesforce/blip2-flan-t5-xl — Image-to-Text • Updated 12 days ago • 118k • 29
- microsoft/trocr-large-stage1 — Image-to-Text • Updated Apr 1 • 116k • 9
- fxmarty/pix2struct-tiny-random — Image-to-Text • Updated Jun 1 • 91.3k • 1
- laion/mscoco_finetuned_CoCa-ViT-L-14-laion2B-s13B-b… — Image-to-Text • Updated Jun 11 • 84.9k • 14
- Salesforce/instructblip-vicuna-7b — Image-to-Text • Updated Jul 17 • 76.6k • 36
- microsoft/trocr-small-handwritten — Image-to-Text • Updated Jan 25 • 49.9k • 15
- kha-white/manga-ocr-base — Image-to-Text • Updated Jun 22, 2022 • 46.9k • 47
- bipin/image-caption-generator — Image-to-Text • Updated Jul 5 • 41.3k • 8
- microsoft/git-base — Image-to-Text • Updated Apr 24 • 38.7k • 23

How to load a Transformer

How to use from the 🤗/transformers library

```python
# Use a pipeline as a high-level helper
from transformers import pipeline

pipe = pipeline("image-to-text", model="Salesforce/blip-image-captioning-large")
```

Copy

```python
# Load model directly
from transformers import AutoProcessor, AutoModelForSeq2SeqLM

processor = AutoProcessor.from_pretrained("Salesforce/blip-image-captioning-large")
model = AutoModelForSeq2SeqLM.from_pretrained("Salesforce/blip-image-captioning-large")
```

Copy

Model Selection

```
vit_gpt2 = pipeline("image-to-text", model="nlpconnect/vit-gpt2-image-captioning")
```

```
blip_base = pipeline("image-to-text", model="Salesforce/blip-image-captioning-base")
```

```
blip_2 = pipeline("image-to-text", model="Salesforce/blip2-opt-2.7b")
```

Comparison of Models

```
vit_gpt2(URL1, max_new_tokens=100)[0]['generated_text']
```

```
'a refrigerator freezer sitting in a kitchen next to a counter '
```

```
blip_base(URL1, max_new_tokens=100)[0]['generated_text']
```

```
'a white refrigerator sitting in a kitchen next to a sink'
```

```
blip_2(URL1, max_new_tokens=100)[0]['generated_text']
```

```
'a refrigerator sitting in a kitchen with cabinets\n'
```

Comparison of Models

```
vit_gpt2(URL2, max_new_tokens=100)[0]['generated_text']
```

```
'a bench on a mountain with a view of the mountains '
```

```
blip_base(URL2, max_new_tokens=100)[0]['generated_text']
```

```
'a bench on a rocky mountain with a view of a mountain'
```

```
blip_2(URL2, max_new_tokens=100)[0]['generated_text']
```

```
'a bench sitting on a hill\n'
```

Comparison of Models

```
vit_gpt2(URL3, max_new_tokens=100)[0]['generated_text']
```

```
'a large jetliner flying through a cloudy sky '
```

```
blip_base(URL3, max_new_tokens=100)[0]['generated_text']
```

```
'a plane flying in the sky'
```

```
blip_2(URL3, max_new_tokens=100)[0]['generated_text']
```

```
'a large jetliner flying through the air\n'
```

Model Evaluation (BLEU)

```
# Calculating BLEU Score using the frige example
reference = [
             ['this', 'is', 'the', 'back', 'of', 'a', 'freezer', 'in', 'the', 'middle', 'of', 'a', 'kitchen', 'floor'],
             ['the', 'appliance', 'in', 'the', 'kitchen', 'may', 'be', 'broken', '.'],
             ['the', 'back', 'of', 'a', 'refrigerator', 'in', 'a', 'kitchen'],
             ['a', 'refrigerator', 'sitting', 'in', 'the', 'middle', 'of', 'a', 'kitchen', 'floor', '.'],
             ['the', 'back', 'of', 'a', 'refrigerator', 'standing', 'in', 'the', 'middle', 'of', 'a', 'kitchen', '.'],
            ]
```

```
vit_candidate = ['a', 'refrigerator', 'freezer', 'sitting', 'in', 'a', 'kitchen', 'next', 'to', 'a', 'counter ']
```

```
vit_score = sentence_bleu(reference, vit_candidate)
print(vit_score)
```

Model Evaluation (BLEU)

```
vit_candidate = ['a', 'refrigerator', 'freezer', 'sitting', 'in', 'a', 'kitchen', 'next', 'to', 'a', 'counter ']
```

```
vit_score = sentence_bleu(reference, vit_candidate)
print(vit_score)
```

```
5.008605395783359e-78
```

```
print('Cumulative 1-gram: %f' % sentence_bleu(reference, vit_candidate, weights=(1, 0, 0, 0)))
print('Cumulative 2-gram: %f' % sentence_bleu(reference, vit_candidate, weights=(0.5, 0.5, 0, 0)))
print('Cumulative 3-gram: %f' % sentence_bleu(reference, vit_candidate, weights=(0.33, 0.33, 0.33, 0)))
print('Cumulative 4-gram: %f' % sentence_bleu(reference, vit_candidate, weights=(0.25, 0.25, 0.25, 0.25)))
```

```
Cumulative 1-gram: 0.636364
Cumulative 2-gram: 0.504525
Cumulative 3-gram: 0.308321
Cumulative 4-gram: 0.000000
```

Model Evaluation (BLEU)

```
blip_base_candidate = ['a', 'white', 'refrigerator', 'sitting', 'in', 'a', 'kitchen', 'next', 'to', 'a', 'sink']
```

```
blip_base_score = sentence_bleu(reference, blip_base_candidate)
print(blip_base_score)
```

```
5.731095145962094e-78
```

```
print('Cumulative 1-gram: %f' % sentence_bleu(reference, blip_base_candidate, weights=(1, 0, 0, 0)))
print('Cumulative 2-gram: %f' % sentence_bleu(reference, blip_base_candidate, weights=(0.5, 0.5, 0, 0)))
print('Cumulative 3-gram: %f' % sentence_bleu(reference, blip_base_candidate, weights=(0.33, 0.33, 0.33, 0)))
print('Cumulative 4-gram: %f' % sentence_bleu(reference, blip_base_candidate, weights=(0.25, 0.25, 0.25, 0.25)))
```

```
Cumulative 1-gram: 0.545455
Cumulative 2-gram: 0.467099
Cumulative 3-gram: 0.368341
Cumulative 4-gram: 0.000000
```

Model Evaluation (BLEU)

```
blip_2_candidate = ['a', 'refrigerator', 'sitting', 'in', 'a', 'kitchen', 'with', 'cabinets\n']
```

```
blip_2_score = sentence_bleu(reference, blip_2_candidate)
print(blip_2_score)
```

```
0.4810977290978808
```

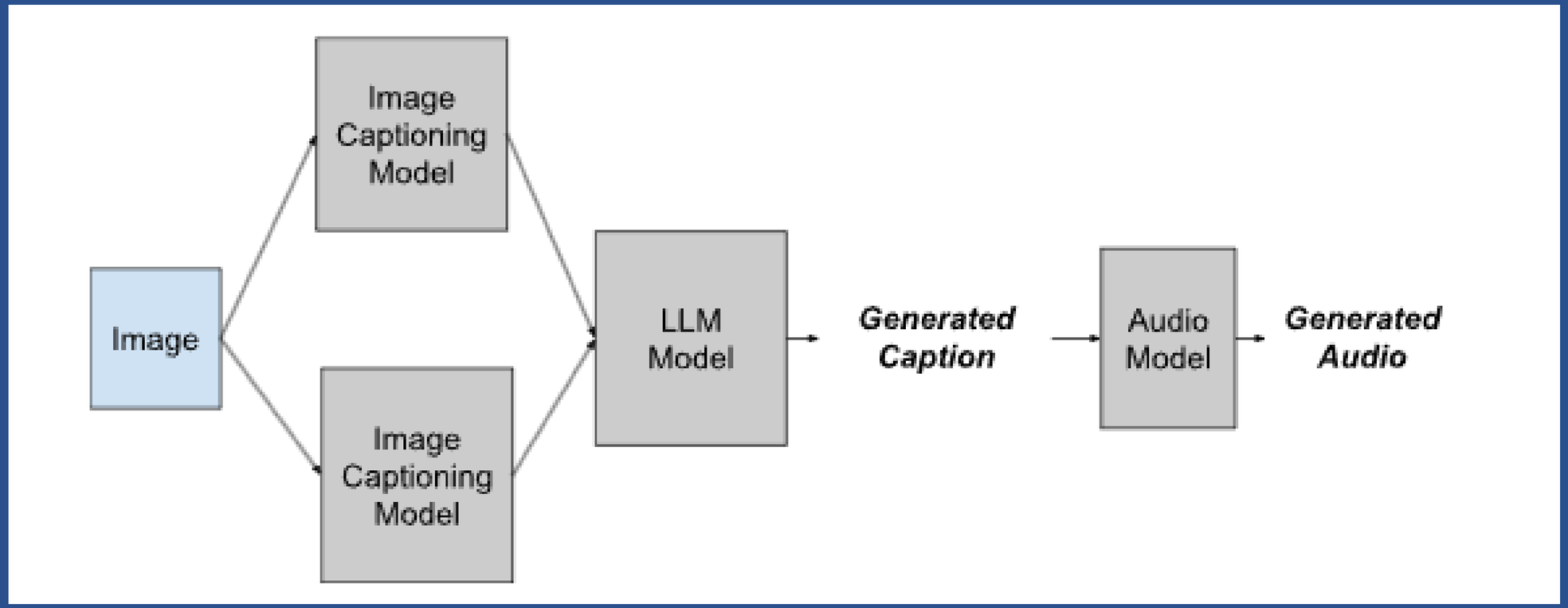
```
print('Cumulative 1-gram: %f' % sentence_bleu(reference, blip_2_candidate, weights=(1, 0, 0, 0)))
print('Cumulative 2-gram: %f' % sentence_bleu(reference, blip_2_candidate, weights=(0.5, 0.5, 0, 0)))
print('Cumulative 3-gram: %f' % sentence_bleu(reference, blip_2_candidate, weights=(0.33, 0.33, 0.33, 0)))
print('Cumulative 4-gram: %f' % sentence_bleu(reference, blip_2_candidate, weights=(0.25, 0.25, 0.25, 0.25)))
```

```
Cumulative 1-gram: 0.750000
Cumulative 2-gram: 0.731925
Cumulative 3-gram: 0.647453
Cumulative 4-gram: 0.481098
```

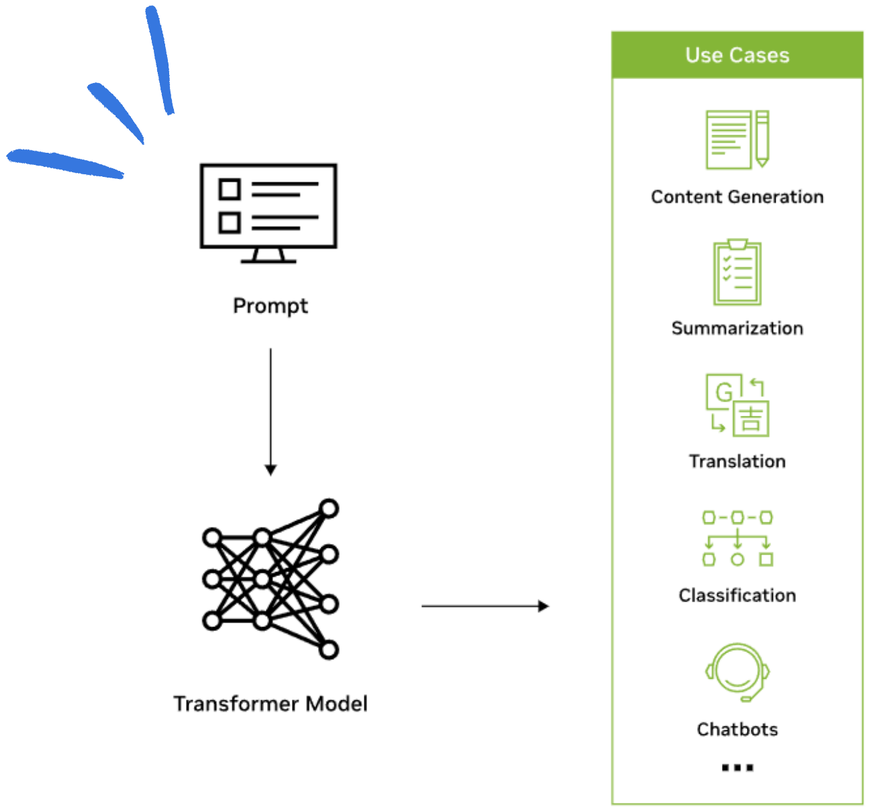
Human Evaluation

	VIT_GPT2	BLIP_BASE	BLIP_2
	a refrigerator freezer sitting in a kitchen next to a counter	a white refrigerator sitting in a kitchen next to a sink	a refrigerator sitting in a kitchen with cabinets\n
	a bench on a mountain with a view of the mountains	a bench on a rocky mountain with a view of a mountain	a bench sitting on a hill\n
	a large jetliner flying through a cloudy sky	a plane flying in the sky	a large jetliner flying through the air\n

Proposed Pipeline

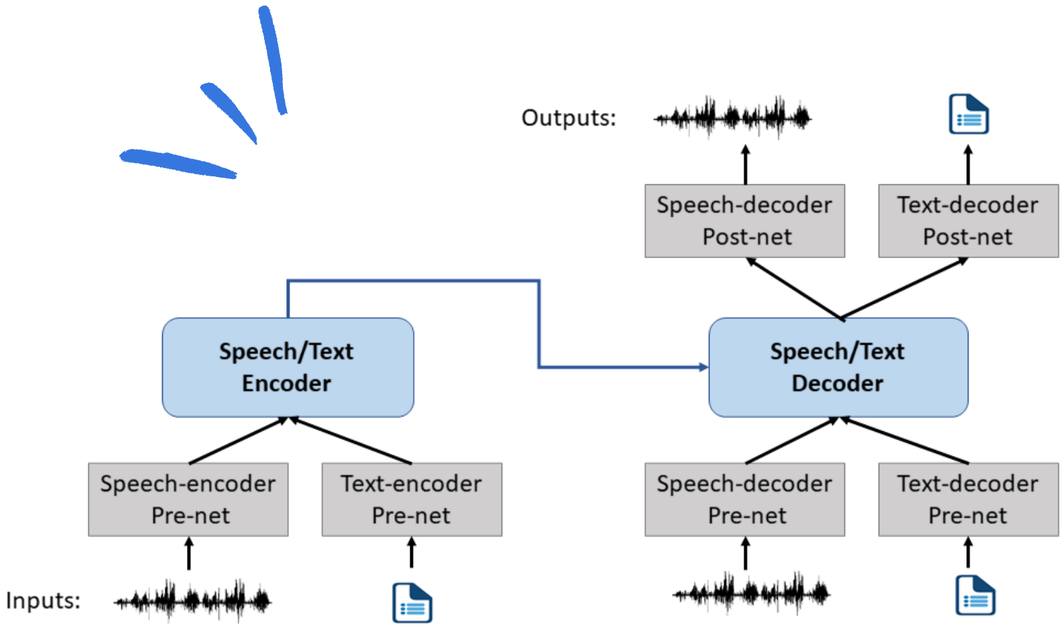


Why use Text-Davinci-003?



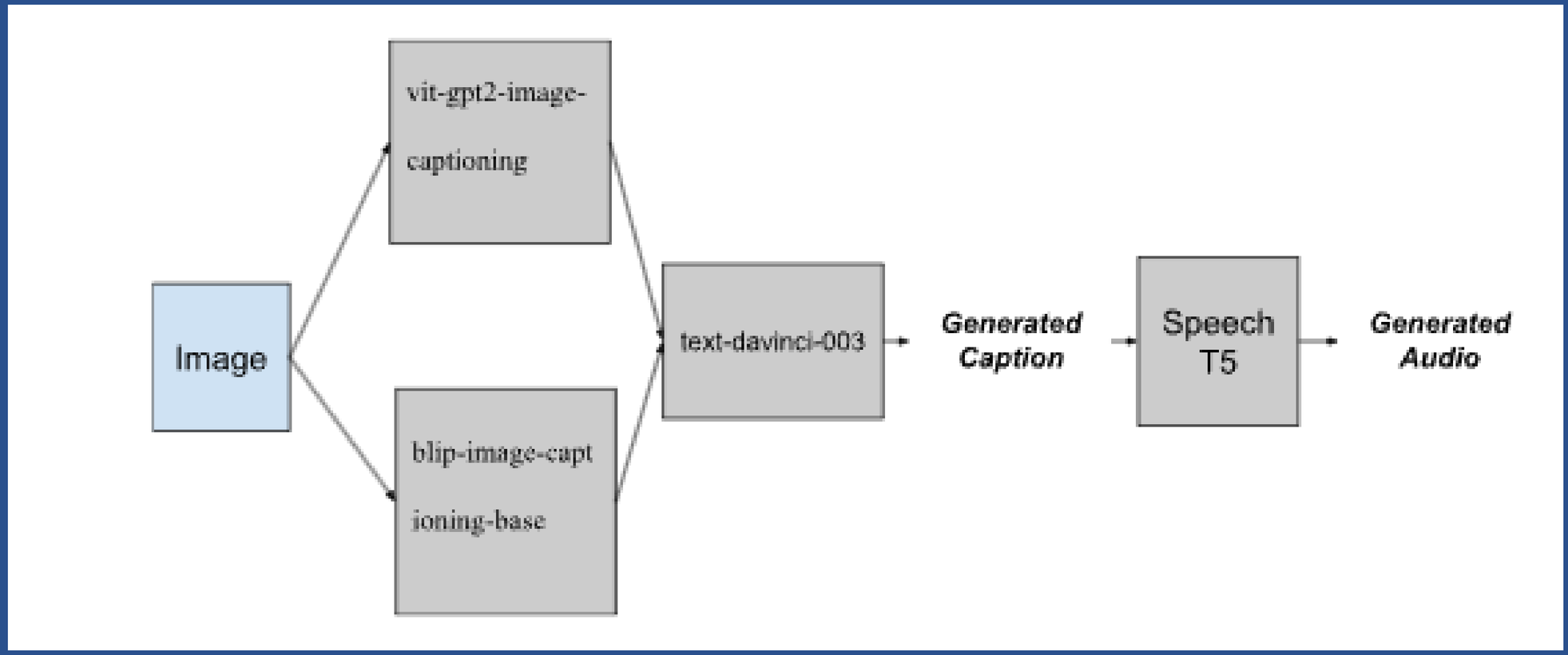
- **Produces higher quality writing**
- **Handle more complex instructions**
- **Better at longer form content generation**

Why use Speech-T5?



- First text-to-speech model
- Multilingual Support
- Integration with other NLP Models

Finalised Pipeline



How do we go about Fine-tuning?

- **Pre-Training**
- **Prompt-Engineering**

Promp Engineering

PROMPT USED			
"Combine both text:" + vit_gpt2_output + "and " + blip_base_output	A white refrigerator freezer is sitting in a kitchen next to a counter and a sink.	lake AI: A bench on a rocky mountain with a view of both the mountains and a mountain lake.	A large jetliner flying through a cloudy sky and a plane flying in the sky.
"Combine and refine the text:" + vit_gpt2_output + "and " + blip_base_output	A white refrigerator freezer is situated in a kitchen next to both a counter and a sink.	A bench on a rocky mountain with a view of both the mountains and the surrounding landscape.	A large jetliner and a plane flying through a cloudy sky.
"Combine both:" + vit_gpt2_output_example1 + "and " + blip_base_output_example1 + " Please refine the text."	A white refrigerator freezer sitting in a kitchen next to both a counter and a sink.	A bench on a rocky mountain with a view of both the mountains and the surrounding landscape.	A large jetliner and a plane flying through a cloudy sky.

Promp Engineering

PROMPT USED			
"You are an AI assistant that combines information. The following are the outputs:" + vit_gpt2_output_example1 + "and "+ blip_base_output_example1 + " Please refine the text."	A white refrigerator freezer sitting in a kitchen next to both a counter and a sink.	A bench on a rocky mountain with a view of both the mountains and the surrounding landscape.	I can see a large jetliner and a plane flying through the cloudy sky.
"Act as a human, provide output that a human would say to a visually impaired from the combination of both text:" + vit_gpt2_output + "and " + blip_base_output	I can see a white refrigerator freezer sitting in a kitchen next to both a counter and a sink.	I can see a bench on a rocky mountain with a spectacular view of both the mountains and the landscape around it.	I can see a large jetliner and a plane soaring through the cloudy sky, their wings cutting through the clouds as they gracefully fly through the sky.
"Act as a human, be as elaborative as possible, combined and refined the text output that a human would say to a visually impaired:" + vit_gpt2_output_example1 + "and " + blip_base_output_example1	I can see a white refrigerator freezer sitting in a kitchen, it's right next to a counter on one side and a sink on the other.	I can see a bench on a rocky mountain with a beautiful view of the distant mountains and the rolling hills of the surrounding landscape.	I can see a large jetliner and a plane gracefully soaring through the cloudy sky, the jetliner's wings cutting through the clouds as they fly through the sky.

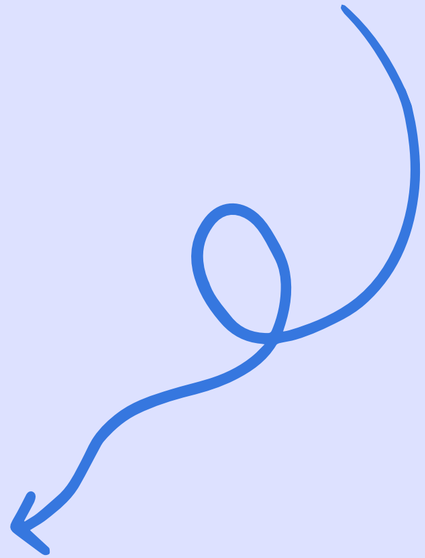
Demo

Loading ...

Future Work

- Further Prompt Engineering
- Pre-training with a customomized dataset
- Creation of a new pipeline

Thank you for your time!



Any Questions?