Storyteller

From images to stories

Mingfei Cui, Silvia Cardani

31.1.2023

LMU – WS2022/2023 – Seminar Computational Creativity, Prof. P. Wicke

Contents

Motivation and first steps

- Building a model that *from a sequence of images* is able to *generate a short story in a certain framing*

- *Several pre-existing image-captioning models* based on COCO caption dataset *were tested to choose the best* one. Some of them: Grit, Oscar, Xmodal-Ctx, Meshed-Memory Transformer, OFA, BLIP, Lavis (that integrates BLIP)

1. The original models were first run to check their functionality
2. then a new image was used as input and the code was adapted
3. Basing on the results (quality of the caption obtained), *BLIP* was chosen

OFA – test result

Generated caption:
„a *woman* and a woman sitting on the beach in front of the ocean"

BLIP – test result

Generated caption:
„a woman and her dog on the beach“

LAVIS (with BLIP) – test result

Generated caption:
„there is a woman sitting on the beach with her dog“

Test result comparison

OFA	BLIP	LAVIS (with BLIP)
„a *woman* and a woman sitting on the beach in front of the ocean”	„a woman and her dog on the beach“	„there is a woman sitting on the beach with her dog“

Storyteller

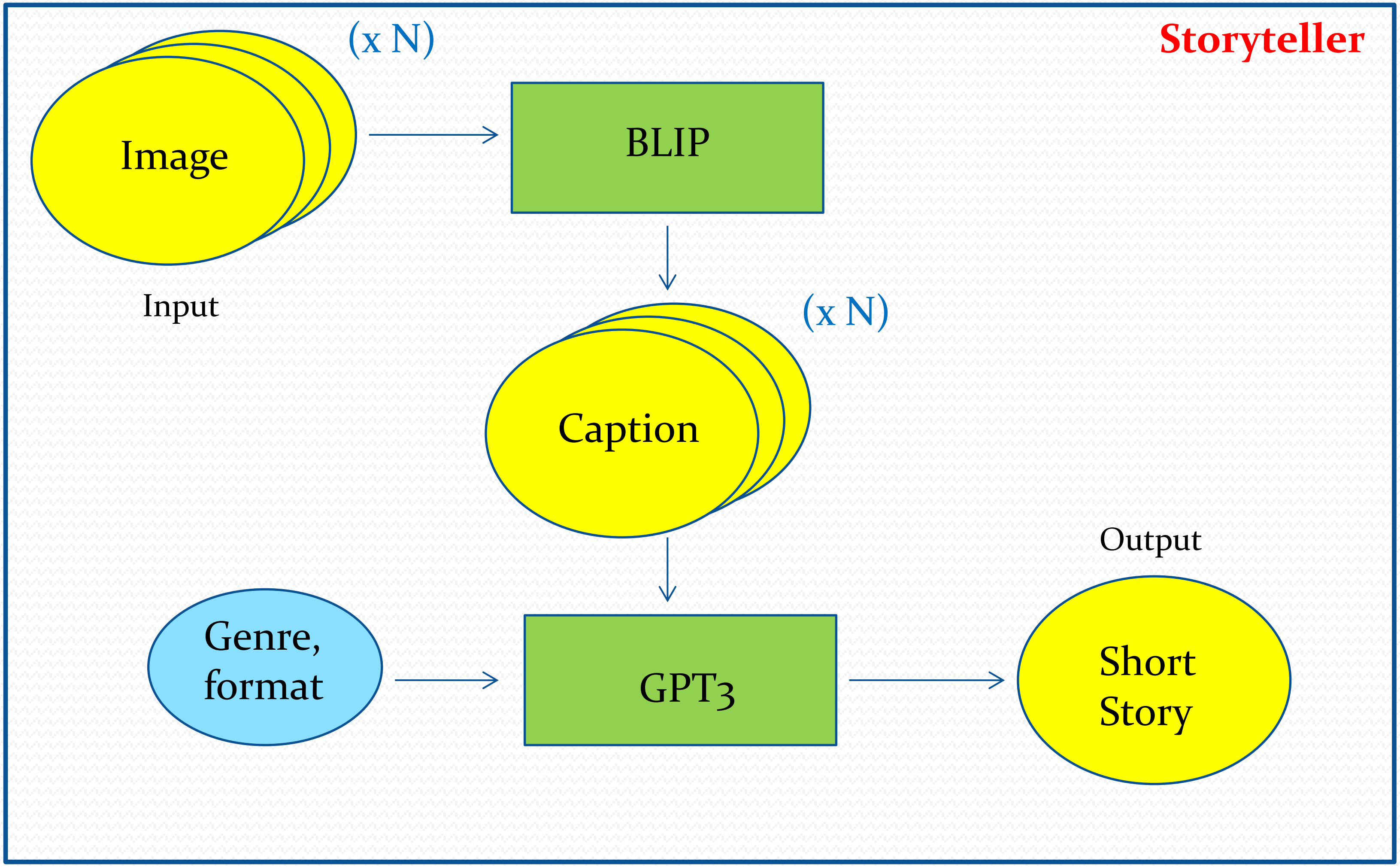
In Storyteller, the generation of a short story is done in several steps:

1. uploading a sequence of *images as input*
2. random *generation of captions* for the images
3. *a story is output* using these captions

Architecture of Storyteller

Storyteller is made of different parts, reflecting the image-to-text generation process:

- *BLIP* (Bootstrapping Language-Image Pre-training) is first used to generate captions for the prompted images
- *GPT3* (Generative Pre-trained Transformer 3) is then employed to generate a story by using Text Completion, basing on the captions generated by BLIP and on the *user's choice of the story's genre and format*



Architecture of Storyteller

BLIP

BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation

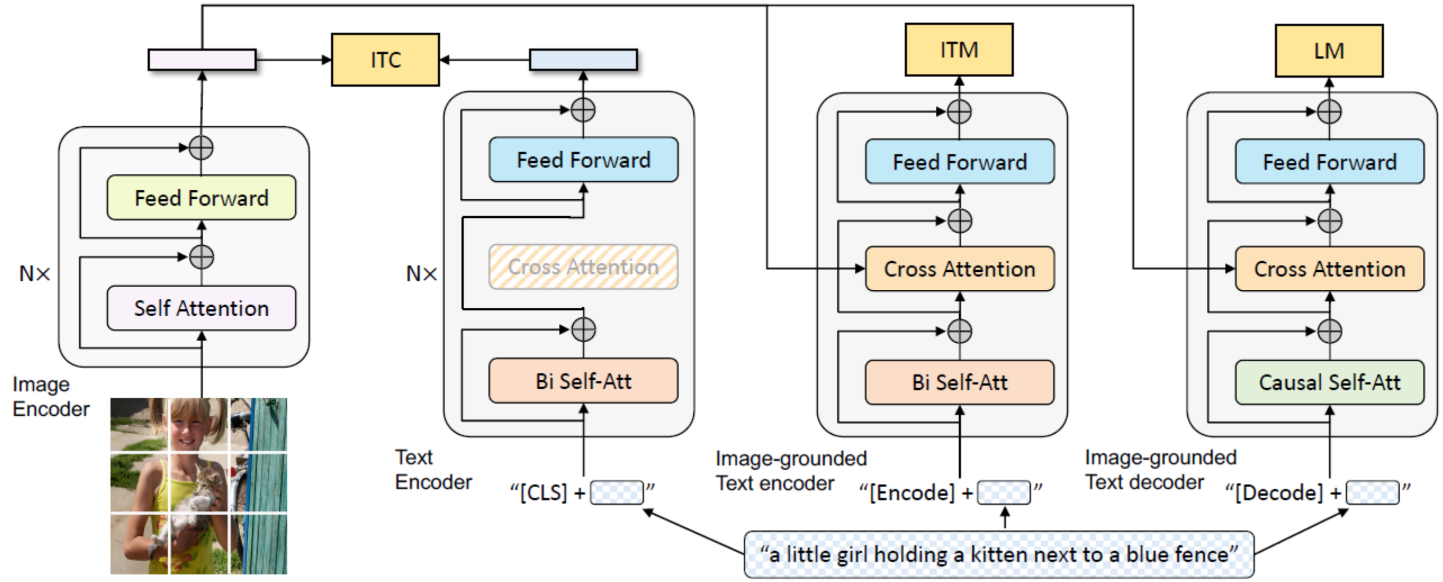


Figure 2. Pre-training model architecture and objectives of BLIP (same parameters have the same color). We propose multimodal mixture of encoder-decoder, a unified vision-language model which can operate in one of the three functionalities: (1) Unimodal encoder is trained with an image-text contrastive (ITC) loss to align the vision and language representations. (2) Image-grounded text encoder uses additional cross-attention layers to model vision-language interactions, and is trained with a image-text matching (ITM) loss to distinguish between positive and negative image-text pairs. (3) Image-grounded text decoder replaces the bi-directional self-attention layers with causal self-attention layers, and shares the same cross-attention layers and feed forward networks as the encoder. The decoder is trained with a language modeling (LM) loss to generate captions given images.

BLIP

- First a *Visual Trasformer* dividing the input image into patches and encoding them as embeddings
- Then, Multimodal mixture of Encoder/Decoder:

1. BERT-based *Text Encoder*, to align the two feature text and image in the semantic space (input: image/text pair)
2. *Image-grounded Text Encoder*, to capture the fine-grained grounded alignment between vision and language. It predicts the match positive and unmatch negative pair, acting as a filter
3. *Image-grounded Text Decoder*, to predict the next token. The image-grounded text decoder is activated to generate textual descriptions given an image

BLIP

- To perform efficient pre-training, the *encoder* employs *bi-directional self-attention* to build representations for the *current* input tokens
- The *decoder* employs *causal self-attention* to predict the *next* tokens
- CapFilt: *Captioner + Filter* to improve the quality of the text corpus, as noisy data (wrong description of images) affects the learning of the vision-language alignment signal. They are fine-tuned individually on the high quality human-annotated COCO dataset. Bootstrapping is performed
- *Nucleus sampling* was chosen as a sampling method, instead of beam search, for larger variety and better creativity

LAVIS (with BLIP) – Nucleus sampling example

Multiple captions generated with nucleus sampling:

1) „a cat is on its back playing with a Christmas tree“
2) „an orange cat reaching up to a Christmas tree“
3) „a cat on a tree with an orange Christmas star“

BLIP – effects of CapFilt

T_w: "from bridge near my house"

T_s: "a flock of birds flying over a lake at sunset"

Human caption:

T_w: "in front of a house door in Reichenfels, Austria"

Synthetic caption:

T_s: "a potted plant sitting on top of a pile of rocks"

T_w: "the current castle was built in 1180, replacing a 9th century wooden castle"

T_s: "a large building with a lot of windows on it"

Figure 4. Examples of the web text T_w and the synthetic text T_s. Green texts are accepted by the filter, whereas red texts are rejected.

Synthetic texts produced by CapFilt are often more precise than human descriptions of images found in the web

GPT3

- Very large Transformer model from OpenAI, with 175 billions parameters
- Trained using the following datatsets: Common Crawl, WebText2, Books1, Books2, Wikipedia
- After training GPT3, a task description, eventual examples and a prompt are input to get a prediction
- In the training the model should have seen the same structure and somehow match the pattern
- The model returns a Text Completion in natural language (predicts the next most likely word)

Transformer

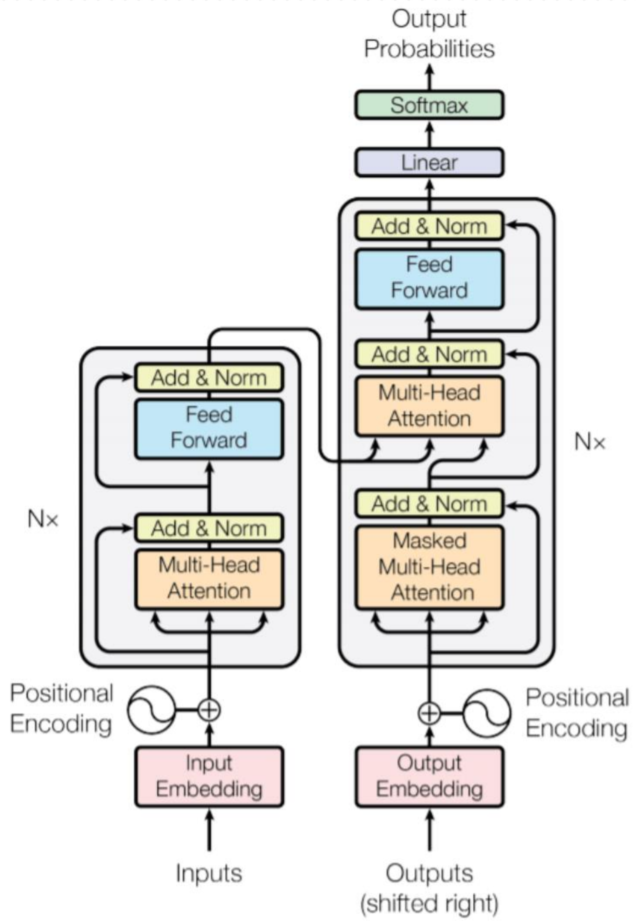


Figure 1: The Transformer - model architecture.

GPT3 as a Transformer

- *Positional Encoding*: to encode the position of the current token in the sequence
- *Multi-head Attention*: predicts which input tokens to focus on and how much. E.g. in a sequence of 3 tokens, weight matrices are learnt. These transform our sequence embeddings into 3 separate 3x64 matrices, each one for a different task. The first 2 matrices (queries and keys) are multiplied together, which yields a 3x3 matrix. This matrix (normalized through softmax) represents the importance of each token to each other tokens. This is then multiplied with the third matrix (values), giving, for each token, a mix of all other token values weighted by the importance of their respective tokens. The process is repeated 96 times in GPT-3

GPT3 as a Transformer

- *Feed forward*: a multi-layer-perceptron (the input is multiplied with learnt weights, learnt bias is added) with 1 hidden layer
- Decoding: After passing all the 96 layers of GPT-3, the start encoding mapping (word to vectors) is reversed to transform our output vector embedding back into a word encoding

GPT3: Parameters

- ***Model***: engine employed to generate predictions
- ***Max_tokens***: max nr. of tokens generated by the model
- ***Temperature***: to control randomness and creativity of the model. Before applying softmax, output values can be scaled with temperature. If close to 1, no values modifications happen before softmax. If close to 0, the model becomes more deterministic, outputting the same tokens after a given sentence (highest probable token will become very likely)
- ***Frequency_penalty***: controls the model's tendency to repeat predictions. It reduced the probabilities of words that have already been generated, depending on their frequency. Default is 0

GPT3: Parameters

- ***Presence_penalty***: encourages the model to make new predictions. It lowers the probabilities of a word if it already appreared in the prediction, not depending on frequency. Default is 0
- ***Top_p***: also *nucleus sampling*, the sampling threshold during inference time. It limits the amount of possible words to sample in the output to the k most likely predicted words. E.g. with a top-k parameter of 1 (neuter), it always picks the most likely word. If set e.g. to 3.5%, the model will sample and select randomly among the objects in their cumulative distribution of 3.5%, according to their likelihood (e.g. one 2%, the other 1.5%). This also controls originality and randomness. OpenAI recommends to variate either top_p or temperature and leave the other at 1

Features of Storyteller

Storyteller:

- friendly introduces itself to the user
- lets the user choose the images for the story
- also lets choose the genre and the format of the story
- is able to *recognize typing errors*, so that the user can correct them. This refers to the number of images, genre and format
- If the user is not satisfied with the story proposed, he/she can make the system generate other stories

Choice of genre and format

To diversify genre and format of the story, Storyteller gives the user the following possibilities:

- For category *Book*:
  - Lord of the Rings
  - Harry Potter
- For category *TV-show*:
  - Sitcom
  - Doctor Who
  - Stranger Things
- For category *Play*:
  - Shakespeare
  - West End Musical
  - Pantomime

Output examples: Input Images

Output ex. 1: Harry Potter Book

welcome

caption

info from H.P. books

```
silvia@silvia-pc: ~/CC/cc22-23-storyteller/src

Welcome to Storyteller Program!
In our program, you will be able to input a sequence of images and
we will based on the images generate a story in the format and genre of your choosing.
Let's get started!

How many images would you like to use?
1
Please select 1 images one by one from the file picker, which just opened.
  0%|                                                                                   | 0/1 [00:00<?, ?it/s]
{'name': 'blip_image_train'}
{'name': 'blip_image_eval'}
{'name': 'blip_caption', 'prompt': 'a picture of '}
{'name': 'blip_caption'}
100%|██████████████████████████████████████████████████████████████████████| 1/1 [00:09<00:00,  9.86s/it]
['the panda bear is laying down on the bench']
What genre would you like the story to be? Enter:
[0] for Book
[1] for TV Show
[2] for Play
0

What genre option would you like the story to be? Enter:
[0] for Lord of the Rings Story
[1] for Harry Potter Story
1

The sun shone brightly on the grounds of Hogwarts, and a gentle breeze rustled through the trees. The Great Lake glimmered in the distance and the familiar towers of the castle ca
st long shadows on the dewy grass.

Harry Potter stepped out of the Entrance Hall and onto the grounds. It had been a long year, and he was ready for a well-deserved break from classes and studying. He had just begu
n to make his way towards the lake when something caught his eye.

Standing on a nearby bench was a large, black and white panda bear. It was lying down with its paws in the air and its head resting on the back of the bench. It seemed to be takin
g a nap, oblivious to the students and teachers walking by.

Harry stopped in his tracks, taken aback by the sight. He had never seen a panda bear at Hogwarts before. He slowly approached the animal, trying not to startle it.

"Hey there," he said in a gentle voice. "What are you doing here?"

The panda bear stirred, slowly lifting its head to look at Harry. Its eyes were a deep, dark brown and its fur was soft and luxurious. It blinked a few times before turning its he
ad away, as if it was ignoring him.

Harry smiled. "I guess you don't want to talk," he said. He reached out and gently stroked the animal's fur. The panda bear's fur was surprisingly warm to the touch, and it felt c
omforting.

Suddenly, the panda bear let out a loud yawn and slowly stood up. It stretched its legs and gave Harry one last look before turning and ambling away.

Harry watched as the panda bear disappeared into the distance. He shook his head in amazement, wondering what had brought the animal to Hogwarts. He couldn't help but feel a stran
ge connection to the animal - as if it was a sign of something to come.

Would you like to generate another story? (y/n)
```

Output ex. 2: West End Musical

```
silvia@silvia-pc: ~/CC/cc22-23-storyteller/src

Welcome to Storyteller Program!
In our program, you will be able to input a sequence of images and
we will based on the images generate a story in the format and genre of your choosing.
Let's get started!

How many images would you like to use?
3
Please select 3 images one by one from the file picker, which just opened.
  0%|                                                          | 0/3 [00:00<?, ?it/s]
{'name': 'blip_image_train'}
{'name': 'blip_image_eval'}
{'name': 'blip_caption', 'prompt': 'a picture of '}
{'name': 'blip_caption'}
 33%|███████████                                     | 1/3 [00:11<00:23, 11.52s/it]
{'name': 'blip_image_train'}
{'name': 'blip_image_eval'}
{'name': 'blip_caption', 'prompt': 'a picture of '}
{'name': 'blip_caption'}
 67%|██████████████████████                          | 2/3 [00:21<00:10, 10.67s/it]
{'name': 'blip_image_train'}
{'name': 'blip_image_eval'}
{'name': 'blip_caption', 'prompt': 'a picture of '}
{'name': 'blip_caption'}
100%|█████████████████████████████████████████████████| 3/3 [00:32<00:00, 10.71s/it]
['a black and white panda bear sleeping on a wooden platform', 'a rabbit on the grass looking to its left', 'a squirrel sits on top of a branch and stares at the camera']
What genre would you like the story to be? Enter:
[0] for Book
[1] for TV Show
[2] for Play
2

What genre option would you like the story to be? Enter:
[0] for Shakespeare Play
[1] for West End Musical
[2] for Pantomime
1

ACT I

Scene 1:

[The stage is lit. A black and white panda bear is sleeping on a wooden platform. The lights dim and a spotlight appears on the panda bear.]

[The rabbit on the grass looks to its left as the music begins. The music is soft and dreamy.]

SOLILOQUY:
 My life is so peaceful, so calm and serene.
I look around and I know that I'm living the dream.
But I can't help but feel that something's missing.
There's something more to life that I'm still wishing.
```

captions

Output ex. 2: West End Musical

silvia@silvia-pc: ~/CC/cc22-23-storyteller/src

```
[0] for Shakespeare Play
[1] for West End Musical
[2] for Pantomime
1

ACT I

Scene 1:

[The stage is lit. A black and white panda bear is sleeping on a wooden platform. The lights dim and a spotlight appears on the panda bear.]

[The rabbit on the grass looks to its left as the music begins. The music is soft and dreamy.]

SOLILOQUY:
"My life is so peaceful, so calm and serene.
I look around and I know that I'm living the dream.
But I can't help but feel that something's missing.
There's something more to life that I'm still wishing.

[The music crescendos as the spotlight intensifies and the squirrel sits on top of the branch and stares at the camera.]

SOLILOQUY:
I want to find my purpose, to discover my place.
To know what I'm meant for and to find my own grace.
But I know I must go and take a chance,
To find the answers that I'm looking to ask.

[The music fades. The lights dim and the curtain closes.]

Scene 2:

[The curtain opens and the stage is lit. The black and white panda bear is standing in the center of the stage.]

[The music begins. It is a powerful and emotional ballad. The panda bear is singing.]

SONG:
I'm searching for a place, a place to call my own.
Where I can be free, and I can be who I want to be.
I'm searching for a home, a place to call my own.
Where I can be loved and I can be who I want to be.

[The music builds as the panda bear moves across the stage. The rabbit and the squirrel join in the background, dancing and singing in harmony.]

SONG:
I'm searching for a way, a way to break the chains.
To be who I am, and to never feel the same.
I'm searching for a dream, a dream to make me whole.
Where I can find love, and I can be who I want to be.

[The music fades and the lights dim as the curtain closes.]


Would you like to generate another story? (y/n)
```

Output ex. 3: Stranger Things

silvia@silvia-pc: ~/CC/cc22-23-storyteller/src

```
END OF ACT I

Would you like to generate another story? (y/n)
Y

Invalid input. Please enter y or n.


Would you like to generate another story? (y/n)
y

What genre would you like the story to be? Enter:
[0] for Book
[1] for TV Show
[2] for Play
1

What genre option would you like the story to be? Enter:
[0] for Sitcom Opening
[1] for Doctor Who Episode
[2] for Stranger Things Episode
2


Scene 1:
We open on the small town of Hawkins, Indiana. It's a beautiful summer day, and the sun is high in the sky. We see a young girl, Eleven, walking through a park. As she walks, she
notices an adorable rabbit sitting on the grass with its mouth open. She smiles and kneels down to get a closer look.

Scene 2:
Suddenly, Eleven hears a rustling in the tree above her. She looks up to see a squirrel perched on the branch. Before she can react, the squirrel leaps from the tree and lands on
the rabbit, startling Eleven. She jumps back, and the squirrel scurries off into the woods.

Scene 3:
Just then, Eleven hears a strange noise coming from the woods. She stands up and peers into the darkness. She can make out a faint light, and as she gets closer she realizes it's
coming from a doorway in the side of a hill. She takes a deep breath, and cautiously steps through the doorway.

Scene 4:
Eleven finds herself in a dark and mysterious underground laboratory, filled with strange and mysterious technology. Suddenly, she hears a loud noise and turns to see a group of D
emogorgons emerging from the shadows. She gasps, and the Demogorgons start to move towards her.

Scene 5:
Eleven quickly jumps back and grabs a nearby pipe. She swings it wildly at the Demogorgons, and manages to keep them at bay. Just then, a group of her friends arrive and join the
fray. Together, they manage to fight off the Demogorgons, and escape the laboratory.

Scene 6:
We cut back to the park, where Eleven and her friends are safe. They share a relieved hug, and the sun begins to set. As they walk away, we see the rabbit and squirrel from earlie
r, happily sitting together on the grass. The End.


Would you like to generate another story? (y/n)
```

Conclusion

- Storyteller can generate a story from images, letting the user select genre and format
- By modifying GPT3 parameters like temperature or top_p (nucleus sampling), the model can become more creative, generating more diverse words in the text
- Using frequency_penalty and presence_penalty can also lead to less repetition in the output

Bibliography

1. Li et al., *BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation*, arXiv: 2201.12086v2, Feb. 2022
2. Brown et al., *Language Models are Few-Shot Learners*, arXiv:2005.14165v4, Jul. 2020 *(GPT3)*
3. Vaswani et al., *Attention is all you need*, arXiv:1706.03762v5, Dec. 2017 *(Tranformers)*