Master in
Computer Vision
Barcelona

Diffusion models

Module: C5

Group: 7

Students: Cristian Gutiérrez
Iñaki Lacunza
Marco Cordón
Merlès Subirà

Index

1. Task A: Installation and Preliminaries.
2. Task B: Technical Exploration.
3. Task C: Identify a problem.
4. Task D: Generate synthetic samples.
5. Task E: Fine-Tuning Week 4 model.

Task (a): Installation and Preliminaries

Task (a): Installation and Preliminaries

In order to install each Stable Diffusion version we followed their corresponding Hugging Face documentation given to us by the guidelines in which they use the Diffusers [1] lib.

```
$ pip install diffusers transformers accelerate scipy safetensors invisible_watermark
```

In order to run them we will use the default example given in the documentation also.

- `Stable Diffusion 2.1` Enhanced version of the original Stable Diffusion model
- `Stable Diffusion XL` Designed to handle larger image resolutions and produce high-quality images with improved stability and scalability
- Their corresponding `Turbo` versions Accelerated variant of both Stable Diffusion model, optimized for faster training and generation of high-quality images while maintaining stability and scalability.

For each of them we will perform a **quality benchmark** based on human perception.
We will also perform a inference **time benchmark** to see how fast runs each of them.

This will help us choose the model that gives the best trade-off between time and quality.

[1] von Platen, P., Patil, S., Lozhkov, A., Cuenca, P., Lambert, N., Rasul, K., Davaadorj, M., Nair, D., Paul, S., Liu, S., Berman, W., Xu, Y., & Wolf, T. Diffusers: State-of-the-art diffusion models (Version 0.12.1) [Computer software]. https://github.com/huggingface/diffusers

Task (a): Image quality Benchmark

Given the same prompt and their default hyper-parameters all the models will generate **50 images**. We will value their subjective quality based on our opinion, but also the good consistency with the prompt.

Prompt: “A realistic panda with glasses studying at a desk, surrounded by bookshelves in a library.”
(Generated by ChatGPT to assess this benchmark more easily)

Stable Diffusion 2.1

Consistency: **Bad**
(No glasses, no realistic)

Stable Diffusion XL

Consistency: **Very Good**
(glasses, realistic, detailed book)

Stable Diffusion 2.1 Turbo

Consistency: **Good**
(glasses, no realistic,
not variability on the bookshelf)

Stable Diffusion XL Turbo

Consistency: **Good**
(glasses, somewhat realistic,
more variability on the bookshelf)

Task (a): Image quality Benchmark

Generally glasses have not been added, and if added they can be found anywhere but in the panda's face. However, there is more variety in the added information: different colored and shaped books, and the panda is seated in different forms

Very detailed and varied results. All the results look more realistic, the panda is dressing different types of clothes and the style is slightly but perceptively different in each sample. Moreover, apart from shelves, windows have been added in some cases

Curiously enough the turbo 2.1 version **draws the glasses** but at the trade-off of having less variability in the books on the background and the sitting form of the panda.

In the turbo XL version the resolution fails so much, all the images look `blurry` but stays consistent with the prompt and adds the glasses and variability in the bookshelves.

Stable Diffusion 2.1

Stable Diffusion XL

Stable Diffusion 2.1 Turbo

Stable Diffusion XL Turbo

Task (a): Negative Prompt to avoid cartoon style

So far, we have seen that the generated images are of cartoon-ish style, even though we explicitly added the `realisitc` requirement, we will try to add this requirement via a negative prompt:

Prompt: “A realistic panda with glasses studying at a desk, surrounded by bookshelves in a library.”

Negative Prompt: “cartoon”

Stable Diffusion 2.1

Stable Diffusion XL

Stable Diffusion 2.1 Turbo

Stable Diffusion XL Turbo

The negative prompt of “cartoon” does indeed remove the cartoonish look but that **does not imply** that it will make it more realistic, hence this is not the approach. **It changes to other styles that aren’t either realistic**.

Task (a): Other negative prompts

We tried other less subtle negative prompts to check that the system does indeed works and in the latent space we get further away from the negative embeddings. We tried with positive emotions:

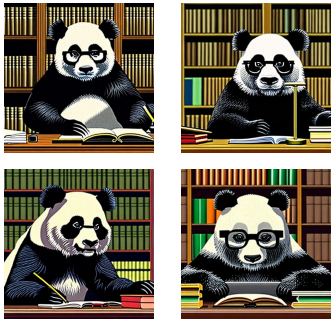
Prompt: “A realistic panda with glasses studying at a desk, surrounded by bookshelves in a library.”

Negative Prompts:
“smile, smiling, happy”

Stable Diffusion 2.1

Stable Diffusion XL

Stable Diffusion 2.1 Turbo



Stable Diffusion XL Turbo

The pandas seem to have **lost expressivity**, it is hard for the machine to determine a panda emotional state. Moreover, on SD 2.1 adding the negative prompt results in worse results with malformed pandas.

Task (a): Inference time Benchmark

We also benchmarked the inference generation time for 20 images for each model. We compare also the negative prompt model we performed on the previous slide. All the tests have been carried using a RTX 3090.

Prompt

	Avg Inference time (s)
SD 2.1	5.41
SD XL	13.81
SD 2.1 Turbo	2.58
SD XL Turbo	4.10

Prompt + Single negative prompt

	Avg Inference time (s)
SD 2.1	5.37
SD XL	13.66
SD 2.1 Turbo	2.56
SD XL Turbo	4.34

Prompt + Multiple negative prompts

	Avg Inference time (s)
SD 2.1	5.30
SD XL	13.59
SD 2.1 Turbo	2.56
SD XL Turbo	4.34

As we can see in the upper benchmark and in the qualitative results, exists a trade-off between quality- and inference time like it was predicted. The model with the highest quality image (SD XL) is also the model with the largest inference time. So, the decision of which model we will use will depend on the purpose of our task and if we need speed or quality.

Task (b): Technical Exploration

Task (b): Technical Exploration

Given our initial benchmark done in **Task (a)**, during this section we will focus on Stable Diffusion 2.1 model because there is more room for improvement. We think that SD XL already gives good results with the default provided model.

Number of inference steps

The number of denoising steps. More denoising steps usually lead to a higher quality image at the expense of slower inference.

20 steps

50 steps

250 steps

500 steps

SD 2.1 Images

DDPM	**20 steps** Inference time (s)	**50 steps** Inference time (s)	**250 steps** Inference time (s)	**500 steps** Inference time (s)
SD 2.1	6.82	5.37	87.40	51.53
SD XL	8.37	13.66	67.94	132.5
SD 2.1 Turbo	2.13	2.56	12.62	23.62
SD XL Turbo	3.49	4.34	20.50	39.66

Task (b): Technical Exploration

The remarks of this section is that, when high number of steps are needed, adding DDIM is a must.

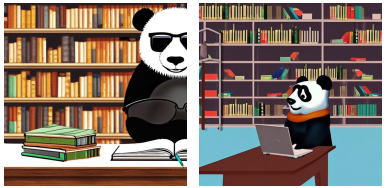
DDPM vs DDIM

We tested using DDIM with 500 inference steps and they produce similar inference time results…

(500 steps)	**DDPM** Inference time (s)	**DDIM** Inference time (s)
SD 2.1	165.25	169.18
SD XL	423.93	437.29
SD 2.1 Turbo	55.39	56.26
SD XL Turbo	39.13	39.40

SD 2.1 w/ 500 inference steps

DDPM

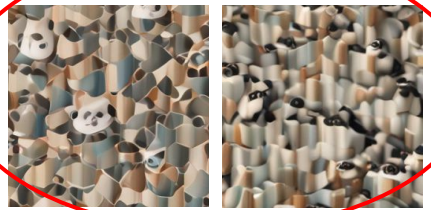


DDIM

SD XL w/ 500 inference steps

???

DDPM



DDIM

✅

Even though both DDPM and DDIM have similar inference times, when using **higher num of steps** DDIM gives **better results** in terms of image quality. We think this is due to the skipping of denoising steps.

Task (b): Technical Exploration

Guidance Scale

Guidance scale (**w** param) reflects how 'loyal' is w.r.t. the prompt. We have tried with different values of guidance to check the different loyalty levels that the created images follow. As we can see, the upper images have different styles, however the high guidance images follow the same style in all the images.

(SD 2.1)

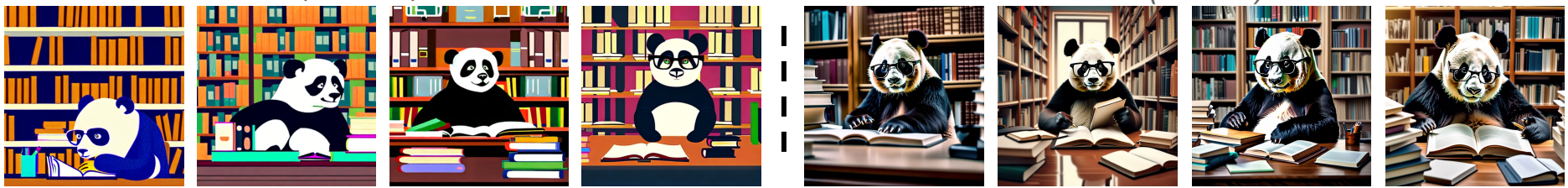
Low guidance (2)

(SD XL)

(SD 2.1)

High guidance (25)

(SD XL)



Task (c): Identify a problem

Task (c): Identify a problem

Analyzing the previous week work we have realized that we did not receive a **correct text retrieval** when there were **animals in the image**:

- Always that appeared an **animal with 4 legs** like a cow, zebra, sheep or even elephants, the model gave us back captions that talked about a **brown horse**.
- On the other hand when in the image appeared a **brown horse with a man riding it**, the model focused on the man and mainly on **the riding action** although the text doesn't speak about horses.

Here there are some examples:

Image with ID 217192

Ground Truth Caption

A herd of elephant forages through a field.

Predicted Captions

A brown horse is grazing in the grass.

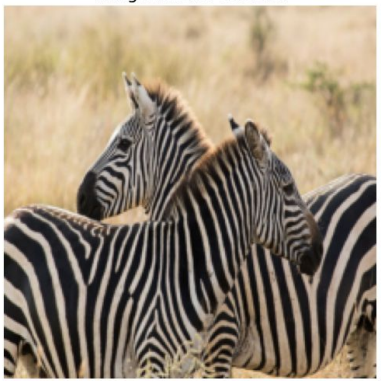
A skinny horse is grazing in a field.

A brown horse is grazing grass near a red house.

Three cats sleeping with their owners on a bed.

A brown horse grazes in an open field next to trees.

Image with ID 361386



Ground Truth Caption

A couple of zebra standing on a grass field.

Predicted Captions

A brown horse is grazing grass near a red house.

Three cats sleeping with their owners on a bed.

A skinny horse is grazing in a field.

A brown horse grazes in an open field next to trees.

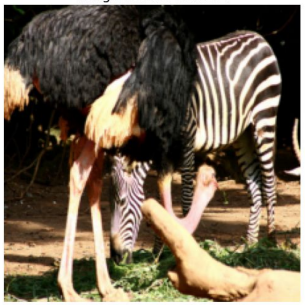
A brown horse is grazing in the grass.

Task (c): Identify a problem

In this image all the captions talk about riding different things, but no one talks about a brown horse.

More examples:

Image with ID 606694



Ground Truth Caption

A zebra standing next to a large bird.

Predicted Captions

A brown horse grazes in an open field next to trees.

A brown horse is grazing grass near a red house.

A brown horse is grazing in the grass.

A cocker spaniel sitting next to a woman holding a baby.

A skinny horse is grazing in a field.

Moreover, the captions are always the same 4.

Image with ID 766432

Ground Truth Caption

Horses are an important asset to city police departments.

Predicted Captions

A person on a bicycle is riding in front of a car.

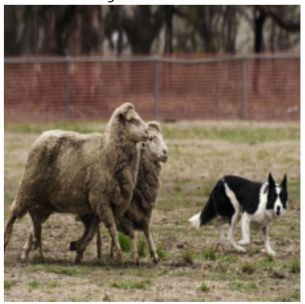
A group of bike riders race down a busy street.

Three people riding bikes on the side of a road.

A woman walks her bike past a city bus.

Two adults and a child riding bicycles on the side of a road.

Image with ID 47050



Ground Truth Caption

two sheep and a god running in a field

Predicted Captions

A brown horse is grazing in the grass.

A brown horse is grazing grass near a red house.

A skinny horse is grazing in a field.

A brown horse grazes in an open field next to trees.

Lady carrying a purse walking along side a man.

Again only one talks about horses, and in this one not even the man is riding.

Image with ID 623260

Ground Truth Caption

A person walking a brown horse down a race track.

Predicted Captions

A person on a bicycle is riding in front of a car.

A brown horse is grazing grass near a red house.

A black car is near someone riding a bike.

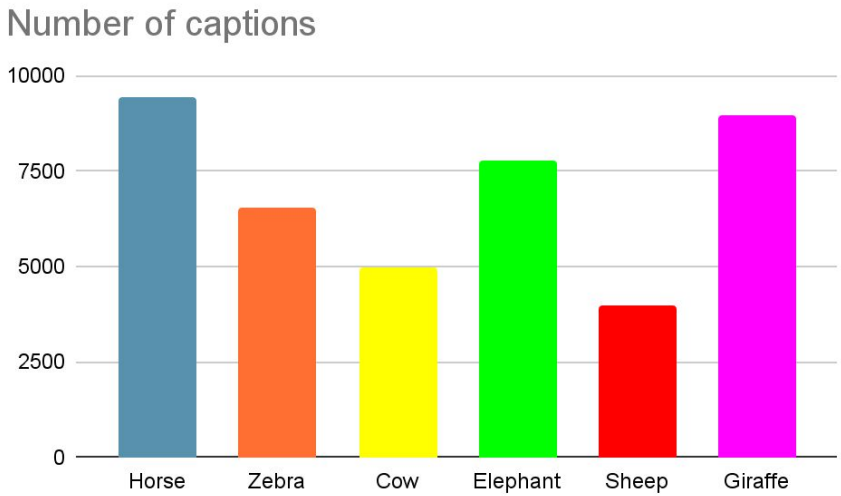
A woman in green is riding a bike.

A lady in a large hat riding a bike.

Task (c): Identify a problem

- We think that the problem is a database imbalance between the number of horse images and the number of images where appear other animals.

- To check if our idea is true we have selected the captions where appear the name of any animal that is confused with a horse to compare their number to the number of horse captions.



In the left graph we can see the difference between the number of captions for every animal. Although, the difference is lower than expected, **there is an imbalance between the images,** so this maybe has caused that the model recognizes every quadruped animal as a horse.

So, our **research question** is:

It is possible to avoid animal misclassifications only with a data-augmentation?

Task (d): Generate synthetic samples

Task (d): Generate synthetic samples

IDENTIFYING CAPTIONS OF INTEREST:

- We have employed the Sentence-Transformer model to group captions of the same context. Note that the used model groups similar context embeddings in the same cluster. But still, the only way of understanding the topic of each cluster relies on looking at its captions.

Some examples showing the first and last 3 elements of each cluster (ordered from closest to farthest):

Cluster n. 1:
3206 elements
tennis

‘a person is on a tennis court with a racket’
‘A person on a court with a tennis racket.’
‘A person on a court with a tennis racket.’
…
‘A woman that is holding a tennis racquet.’
‘A woman that is holding a tennis racquet.’
‘A male tennis player holding his racket, is anticipating the ball.’

Cluster n. 5:
2328 elements
giraffes

‘there are a few giraffes that are standing near each other’
‘the giraffes are standing next to each other ‘
‘a couple of giraffes that are standing next to each other’
…
‘A giraffe standing next to a tree on a field.’
‘there is a large giraffes standing inside of a building’
‘A giraffe that is standing in a grassy area.’

Cluster n. 2:
2886 elements
surf

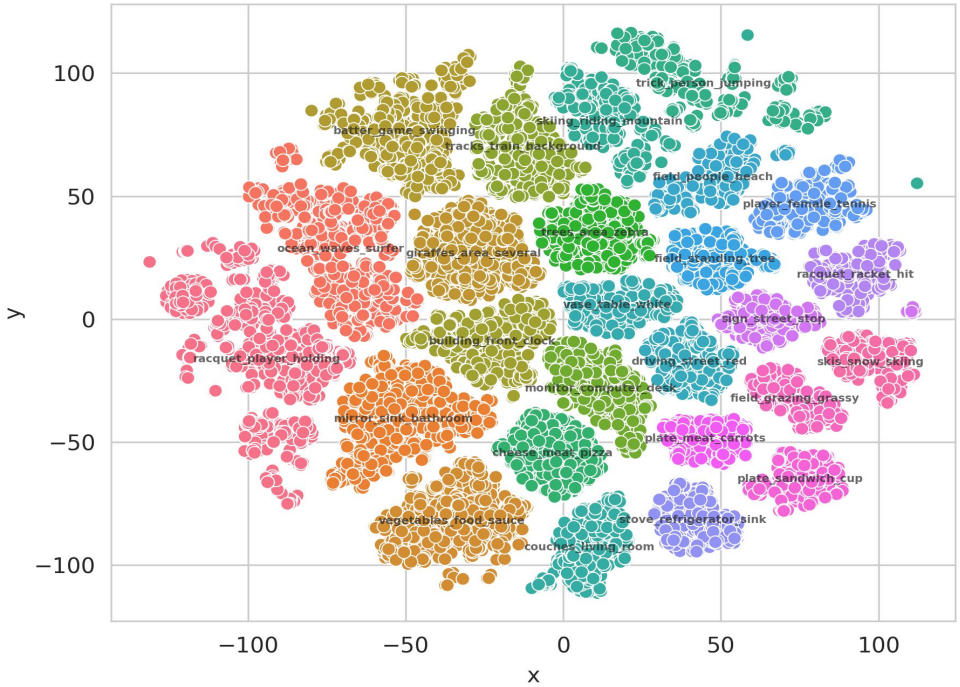
‘A person is on a surfboard in the waves.’
‘A person on a surfboard that is on a wave.’
‘A person on a surfboard in the surf of a wave.’
…
‘a man surfing on a wave at the beach’
‘a person riding a parachute surf board in a body of water’
‘A man is happily riding his surfboard and catching a nice wave.’

Cluster n. 17:
1291 elements
kite

‘there are many people that are flying kites ‘
‘there are several people that are flying a kite’
‘a few people that are flying some kites’
…
‘a kite that is being flown through the air’
‘Groups of people standing on a hill, flying kites.’
‘A bunch of people flying kites in Washington, D.C.’

Task (d): Generate synthetic samples

- In order to get the main topics of each cluster, we have employed BERTopic. Which extracts easily interpretable topics (along with their probabilities). For each cluster, we have retrieved the 3 most probable topics, joining them afterwards using "_".



- We have seen that the topic extractor model does not work as good as we expected. Some of the extracted topics are not useful for characterizing a cluster, for instance: `*the*', `in', '*a*', *with*' … so we have prohibited creating these type of topics.

- Once we have named each cluster, we have created a tsne plot in order to visualize the obtained results. As it can be seen, some very close clusters have very similar labels, as expected.

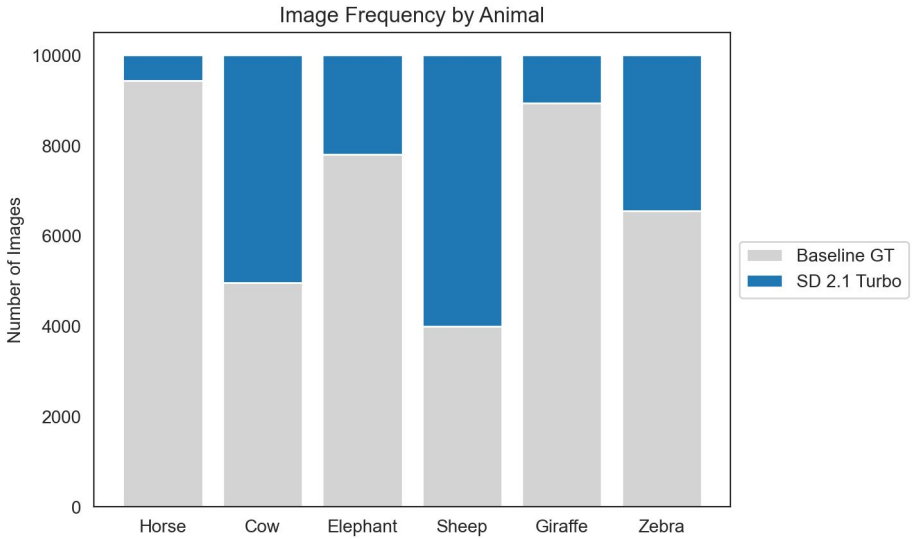
- It seems that what BERTopic has done has just been to choose the most frequent words in each cluster. In short, **Sentence-Transformer** ✓**, BERTopic** ✗

- Note that for easy visualization purposes, only clusters with more than 1000 elements have been shown. If a minimum size of 25 elements is used to create a cluster, more than 3000 clusters are obtained, the largest amount of them containing no more than 40 captions.

Task (d): Generate synthetic samples

In order to **balance our animal database,** we have created **new synthetic animal images** using **diffusion models**. Like we have seen in task c the animals with the lowest number of captions and images are: **cow, zebra and sheep**.

Full Data-Augmentation

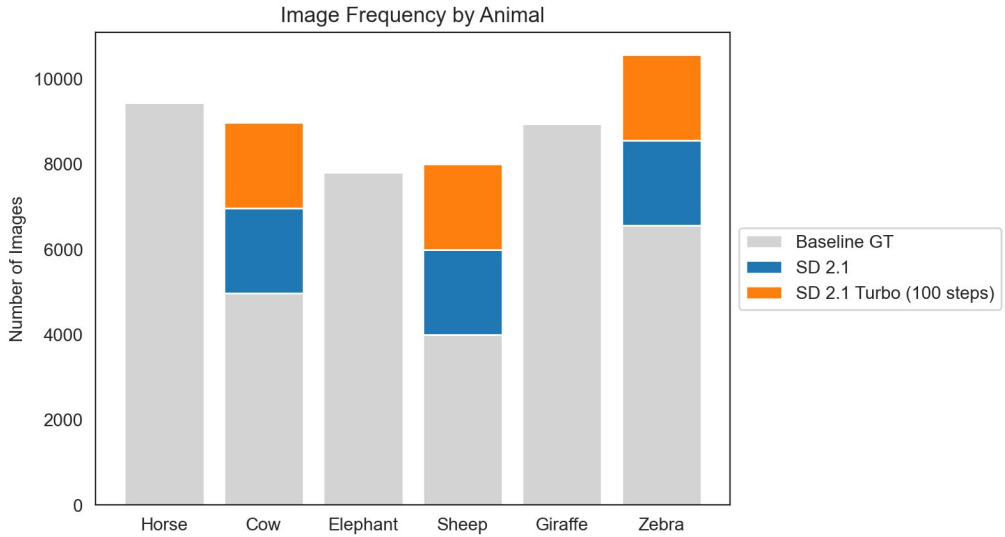


- We selected all these **animals captions** and we employed them as **prompts** for the diffusion models.
- We have employed **3 different models** to create the images: SD 2.1 (50 steps), SD 2.1 Turbo (50 steps), SD 2.1 Turbo (100 steps).
- At the beginning we wanted to balance the database, creating new images so that **all animals had 10k images.** But the problem was that to get that, for instance only for sheep we needed 6k new images and it was needed a lot of hours only for one animal and model. So we only did this solution using the SD 2.1 Turbo (50 steps) model, that was the fastest.
- For the other 2 we created **2000 new synthetic images for each unbalanced animal** with each model. *(Due to the computational time we only could create 2000, since only for each animal and model takes 6 hours and we did not have a lot of time).*

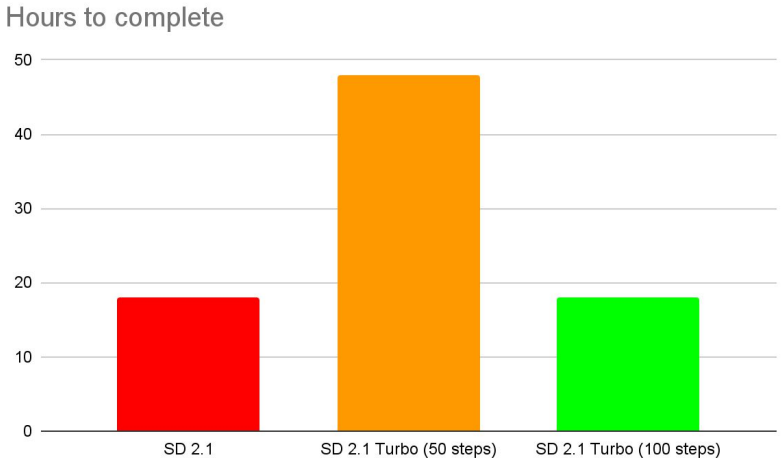
Task (d): Generate synthetic samples

Like we didn't know if we were going to have time to create the full-data augmentation, we decided to create one by increasing the number of images of the animals that appeared the least. We employed 2 different models. As we can see in the right table this strategy is **much faster than the first one**, so that allowed us to save some time.

Mixed Data-Augmentation



Expected time to complete



Task (d): Generate synthetic samples

To create the synthetic images we have sampled the train captions where appear ‘cow’, ‘sheep’ or ‘zebra’. Moreover we have created a database with the original COCO images those captions talk about in order to calculate the FID later. Finally we have randomly selected one of those captions to use it as a prompt for the diffusion models. *(We have repeated this last step 2000 times to obtain all the images of each animal).*

Some example results:

'This is a black cow standing in a grassy field.'

COCO

SD 2.1

'Two cows that are wading in some water.'

COCO

SD 2.1 TURBO (100 steps)

Task (d): Generate synthetic samples

More example results:

'A person is feeding a cow an apple'

Same model and same caption but different generated images.

COCO

SD 2.1 TURBO (50 steps)

COCO

SD 2.1

SD 2.1 TURBO (100 steps)

As we can see in these examples the images with the **highest resolution** are from the **SD 2.1** model but at the same time is the model that needs **more time to extract each image.** However, the quality of the rest is not too low and the images have a lot of relation with their captions.

'The two zebras are standing near one another.

Task (d): Generate synthetic samples

In order to qualitatively assess the generated synthetic samples, we computed the **Fréchet inception distance** of the ground truth images in the dataset and the generated ones. As we can see, SD 2.1 created better images w.r.t. the dataset, furthermore the animals that are more manageable by the generative model are zebras by a big difference compared to the others such as cows and sheeps.

	Fréchet inception distance (FID) [1, 2]					
	Horse	Cow	Elephant	Sheep	Giraffe	Zebra
SD 2.1 *(50 steps)*	–	87.82	–	80.30	–	49.00
SD 2.1 Turbo *(50 steps)*	114.54	139.27	123.34	147.72	98.73	61.87
SD 2.1 Turbo *(100 steps)*	–	141.25	–	147.52	–	62.70

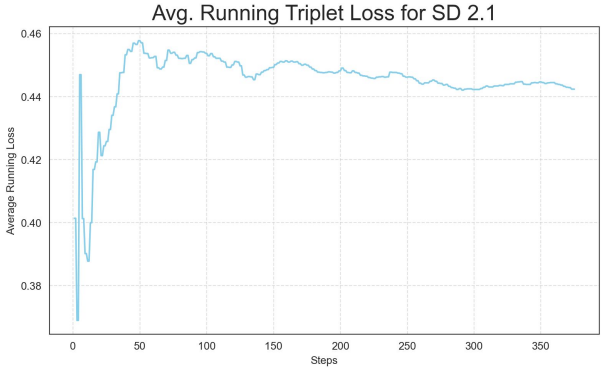
[1] Martin Heusel, et al. (2018). GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium.
[2] Implementation **modified** from https://github.com/mseitzer/pytorch-fid, originally from https://github.com/bioinf-jku/TTUR

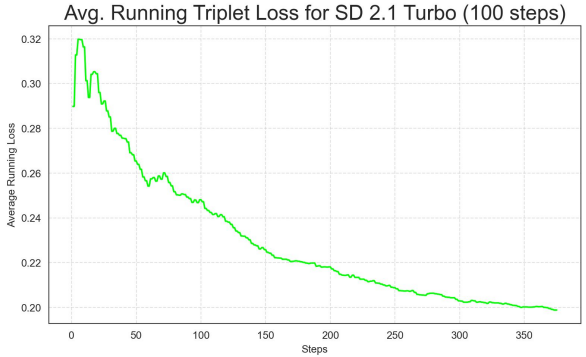
Task (e): Fine-tuning Week 4 model

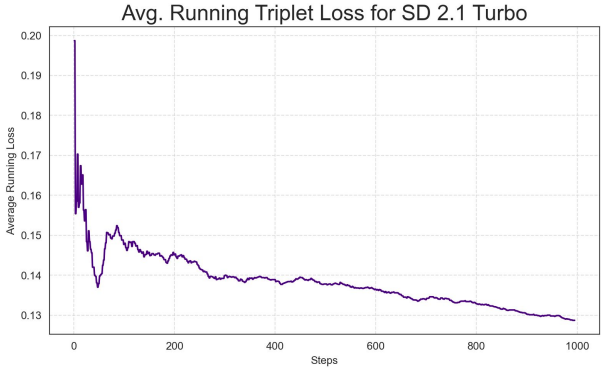
Task (e): Fine-tuning Week 4 model

We have focused on a subproblem, which consisted on improving retrieval with animals.

In the previous week, training the whole model from zero was very time consuming, since each epoch using the original database needed ~9 hours. Given that in the previous week we trained the embedding models on a much larger dataset (the whole dataset), this week we have decided to use as baseline the work of the previous week, fine-tuning it.







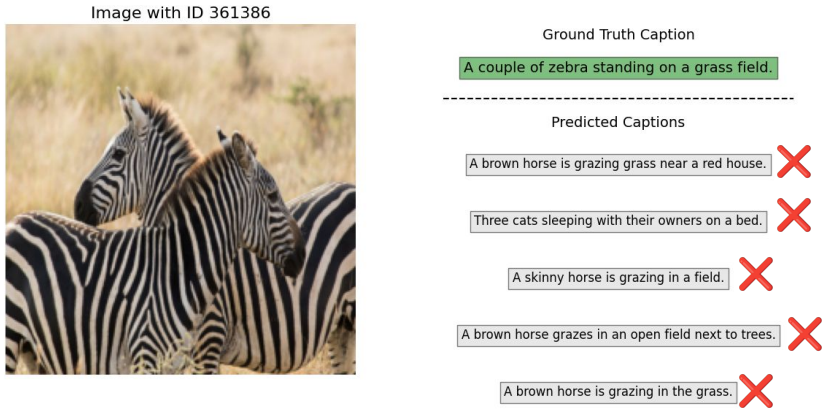
Mixed Data-Augmentation ***Full Data-Augmentation***

We have used the generated samples to tune the models. We have trained (fine-tuned) the model in 5 epochs, needing 50 minutes in total (using a RTX 3090). As in the previous week, we have used the *Adam* optimizer used learning rate has been 2e-5, and the batch size has been 32.

Task (e): Fine-tuning Week 4 model

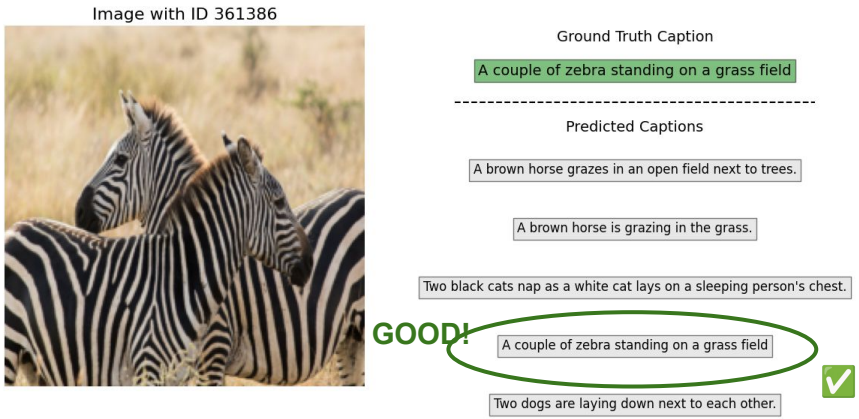
Even though it has not been guessed as the first caption, now the correct caption for zebras appears as the 4th option. Thus the data balancing has made its effect and now zebras are indeed **better represented**.

Before Fine-tuning



The model miss-classifies the zebras for “brown horse” all the times.

After Fine-tuning with Synthetic data

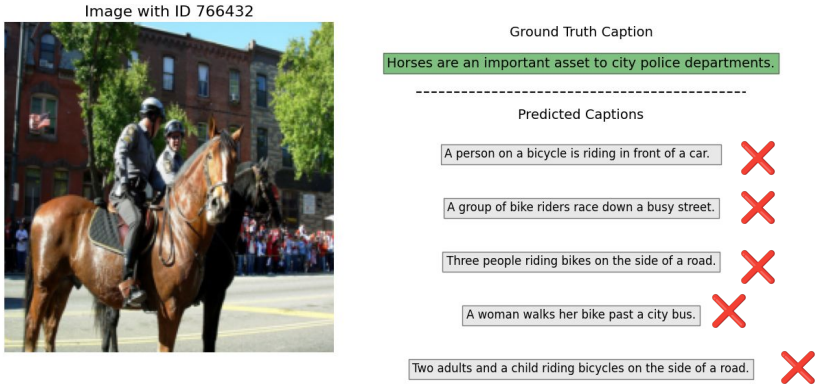


The model miss-classifies the zebras for “brown horse” except one caption that at least retrieves a caption about a zebra.

Task (e): Fine-tuning Week 4 model

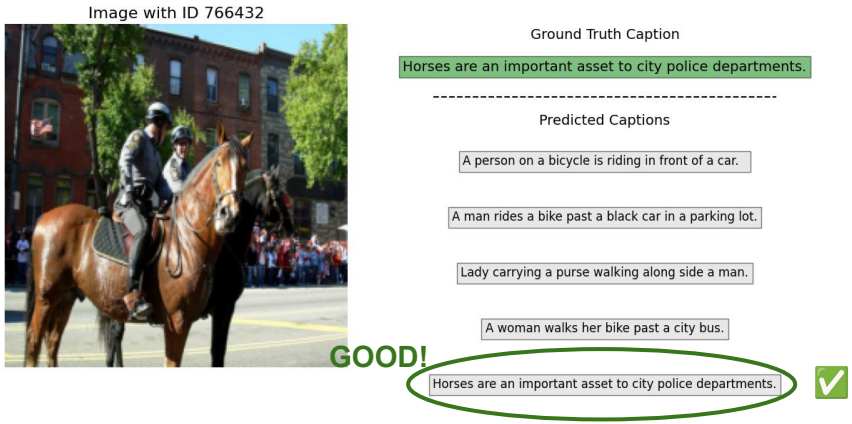
In this example the previous week the model did not retrieve the correct caption but this week at least retrieves the correct in the last position of the Top-5 captions.

Before Fine-tuning



The model miss-classifies all the captions.

After Fine-tuning with Synthetic data



The model now retrieves the correct captain in the fifth position.

Task (e): Fine-tuning Week 4 model

In this example the fine tuning of the model works correctly. The last week model did not retrieve the correct caption in none of the positions, but this week model at least introduces it in the fourth position of the retrieval captions.

Before Fine-tuning

Image with ID 47050

Ground Truth Caption

two sheep and a god running in a field

Predicted Captions

A brown horse is grazing in the grass. ❌

A brown horse is grazing grass near a red house. ❌

A skinny horse is grazing in a field. ❌

A brown horse grazes in an open field next to trees. ❌

Lady carrying a purse walking along side a man. ❌

The model miss-classifies the sheeps for “brown horse” all the times.

After Fine-tuning with Synthetic data

Image with ID 47050

Ground Truth Caption

two sheep and a god running in a field

Predicted Captions

A brown horse is grazing grass near a red house.

A skinny horse is grazing in a field.

A brown horse is grazing in the grass.

GOOD!

two sheep and a god running in a field ✅

A large hours is eating grass in a field.

The model miss-classifies the sheeps for “brown horse” but now at least introduces the correct one in the 4th position.

Task (e): Fine-tuning Week 4 model

Results:

- After the qualitative evaluation of the fine tuned model, we can confirm that we have slightly improved the previous week results with the data-augmentation using synthetic images from diffusion models.

- However, the results are not good as expected, since the most of the captions are not retrieved in the first positions.

- The problem could be the low quality of the synthetic images as we could confirm making the FID score or the low number of new images that we have been able to create.

- Due to lack of time we could not create more images for data augmentation or use more complex models like XL that could have given us better results.

Summary Slide

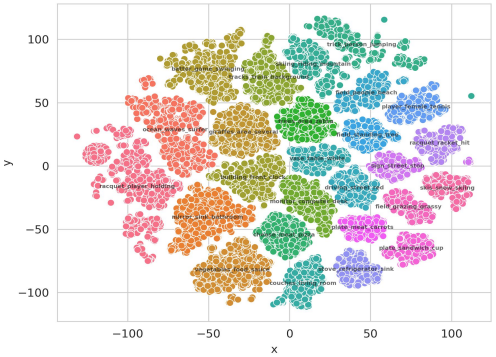
SD 2.1 | SD XL | SD 2.1 Turbo | SD XL Turbo

SD XL w/ 500 inference steps

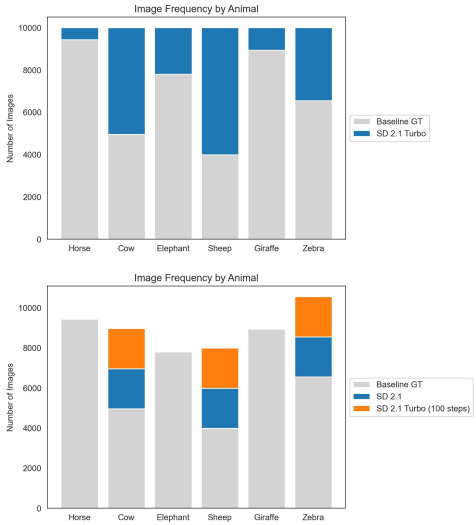
??? DDPM | DDIM

Sentence Transformer ✓

BERTopic ✗



2 Different Data-Augmentation Approaches



- Apart from working and understanding different diffusion models. We have also employed the BERT language model from last week, as well as the Sentence-Transformer model and the BERTopic models.
- We have analyzed the results obtained last week and focused on a concrete problem. We have slightly improved last week results using the data augmentation.
- Due to the short time that we had we cannot create many images as we wanted on a initial time or using the best model of task_a.