Image Captioning & Diffusion Models

Team 4:

- María José Millán
- Agustina Ghelfi
- Laila Aborizka

DATA EXPLORATION AND CLEANING

Dataset overview: 13501 images.

- Image Name : The filename of the image.
- Title : The caption or title describing the image.

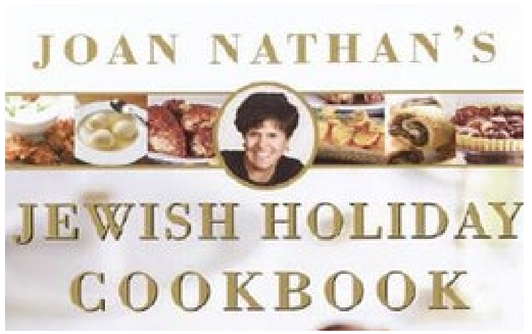
Data Cleaning Steps

- Missing titles: 5 missing titles. Titles with missing values were removed and their respective images also.
- Non-Existent Images: Some entries had Image Name values that didn't match any actual image files. These rows were removed to ensure the dataset only contains valid image-caption pairs.

The word cloud shows the frequency of words in the titles, with "Salad" and "Chicken" appearing most frequently.

During our data exploration, we observed that some images are not food-related. Among these, some have captions describing food, while others have captions unrelated to food, leading to inconsistencies between images and annotations.

Zombies Rising



Zamosc Gefilte Fish

Whipped Sweet Potatoes with Honey

ARCHITECTURE OVERVIEW

ENCODER

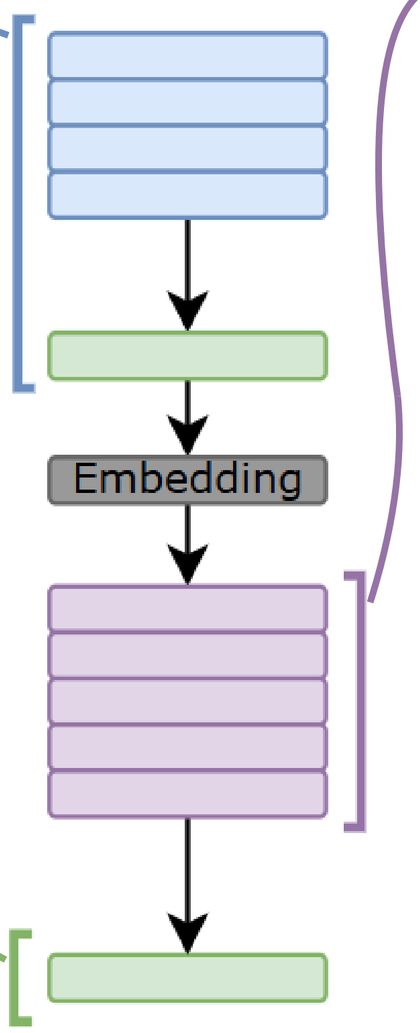
Extracts meaningful visual representations from the input image, typically using CNNs to process and output high-level feature maps.

ResNet-18

As a lightweight CNN, ResNet-18 serves as a strong baseline, capable of extracting essential visual features while keeping computational cost low.

ResNet-50

The increased depth of ResNet-50 allows for capturing more abstract and detailed visual representations, which can support richer and more accurate caption generation.



HEAD

Fully Connected layer.

DECODER

Takes encoded features and sequentially generates the caption one token at a time, typically using RNNs due to their ability to model sequential data.

GRU

Offers a balance between expressive power and training efficiency, making it a practical choice for the initial decoder in image captioning experiments.

LSTM

Designed to retain relevant information over longer sequences, which may enhance fluency and cohesion in caption generation.

Transformer Decoder

Introduced to assess the benefits of attention-based decoding, the Transformer can generate captions that better reflect the overall meaning and structure of the image.

TEXT LEVEL REPRESENTATION ANALYSIS

We explored different levels of word representation, character-level, word-level, and WordPiece-level, to better understand their impact on model performance. We explored each of these representation levels for every combination of the mentioned encoder-decoder architectures. This allowed us to compare how each representation performs across different model setups.

EXAMPLE WITH RESNET-50 + GRU

	GroundTruth	Character	Word	WordPiece
	Our Favorite Chocolate Chip Cookies	Caaaee	Chocolate Chocolate	the -
	lemon icebox pie	Eea e	lemon and	-

	CHAR	**WORD**	WORDPIECE
BLEU-1	0	**0.101**	0.068
ROUGE-1	0	**0.127**	0.086
METEOR	0	**0.055**	0.047

We observed that similar trends in the evaluation metrics were consistent across all encoder-decoder combinations. In every case, word-level representation consistently outperformed character-level and WordPiece-level, making it the best-performing representation overall.

While the generated captions are far from accurate, especially for character and WordPiece levels, we can see that the word-level model is able to capture some relevant content words—like chocolate and lemon—which suggests a stronger semantic alignment despite the overall limitations.

TEACHER FORCING

Teacher Forcing is a training technique used in the decoder of sequence models. Instead of using the model's own predictions in every input for the next step, we sometimes provide the ground truth sequence from the dataset, using the correct token from the training data.

- Standard Approach: The model generates an output and feeds it into the next step.
- Teacher Forcing: The ground truth token is used as input at each step, guiding the model.

RESNET-50 + GRU - WORD-PIECE LEVEL

	GroundTruth	WordPiece	WordPiece (TF)
	Our Favorite Chocolate Chip Cookies	the -	chocolate -
	lemon icebox pie	-	lemon - pie

	WordPiece	**WordPiece (TF)**
BLEU-1	0.068	**0.127**
ROUGE-1	0.086	**0.106**
METEOR	0.047	**0.063**

These results suggest that teacher forcing can help stabilize and guide the learning process for word-piece representations, enabling the model to generate more meaningful and complete captions. While the improvement is modest, it highlights the potential of this technique for enhancing lower-performing representation levels.

QUANTITATIVE RESULTS

MODEL	BEST REPRESENTATION	BLEU-1	ROUGE-1	METEOR
ResNet-18 + GRU	WORD	0.071	0.120	0.047
ResNet-18 + LSTM	WORD	0.101	0.127	0.055
ResNet-18 + Transformer	WORD	0.150	0.115	0.113
ResNet-50 + GRU	WORD	0.126	0.159	0.062
ResNet-50 + LSTM	WORD	0.087	0.107	0.044
Teacher forcing: Resnet-50 + GRU	WORD-PIECE	0.127	0.106	0.063

QUALITATIVE RESULTS

Groundtruth	Zucchini-Lentil Fritters With Lemony Yogurt	Grilled Chicken with Board Dressing
R18 + GRU	with with	grilled with
R18 + LTSM	with with	grilled and with
R50 + GRU	grilled grilled with	chicken with with
R50 + LTSM	grilled with	chicken with with
R18 + Transf	chicken - chicken chicken chiken	grille - chicken chicken with with with
R50 + GRU(TF)	cchini - yogurt	Chicken with -

- Models with GRU and LSTM often generate repetitive or incomplete outputs: “with with”, “chicken with with”, failing to capture full dish names.
- R50+GRU, despite being the best performing model quantitatively, still shows repetition generating outputs like “grilled grilled with”.
- In contrast, teacher forcing produces more distinct phrases “Chicken with -”, “cchini - yogurt” suggesting benefits from word-piece tokenization and improved decoding.

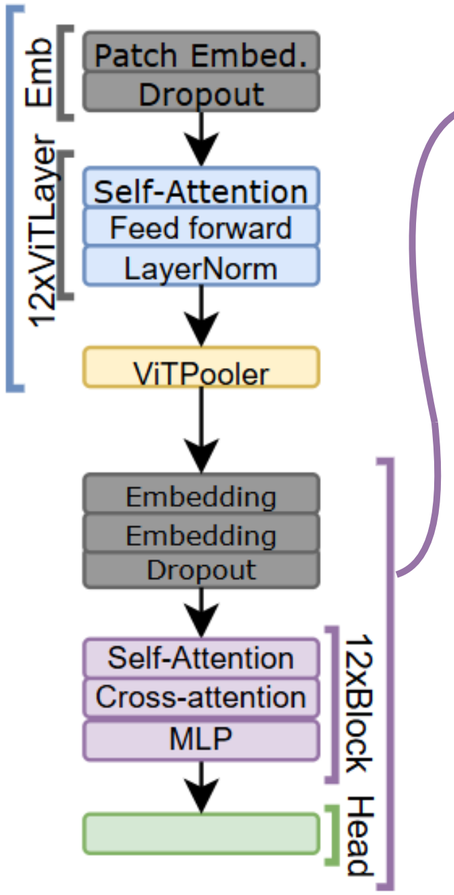
ARCHITECTURE OVERVIEW: VIT + GPT-2

ENCODER: VIT

The ViT encoder processes images by dividing them into fixed-size patches, embedding each patch into a feature vector, and capturing spatial dependencies using self-attention

Why ViT?

The shift from CNNs to ViT allows the model to capture long-range dependencies and complex relationships across different parts of the image, providing richer and more detailed visual features that can enhance caption generation.



DECODER: GPT-2

Trained on diverse text sources, GPT-2 generates text autoregressively by predicting the next token based on previous context.

Why GPT-2?

As a powerful language model, GPT-2 excels at generating coherent, fluent text, and its autoregressive nature makes it well-suited for generating captions one word at a time, ensuring linguistic consistency and relevance to the image.

Why This Combination?

By combining ViT with GPT-2, we leverage the strengths of both models: ViT's ability to capture detailed visual features and GPT-2's prowess in generating coherent, fluent text. This combination can potentially improve the quality and accuracy of generated captions, offering a more flexible and robust solution compared to previous CNN-RNN architectures.

STRATEGIES AND RESULTS

We first performed direct evaluation of the model using pretrained weights from huggingface. We then explored three training strategies with several experiments conducted for each strategie, the best for each strategie where:

1. Fine-tuning ViT while keeping GPT-2 frozen: Taking into consideration all the metrics, including the loss, in this trategie the best model is the one where the embedding layers and the ViT blocks were frozen up to block number 5.
2. Freezing ViT and fine-tuning GPT-2: the best model here is the one where the embedding layers and the GPT-2 blocks were frozen up to block number 7.
3. Fine-tuning both ViT and GPT-2: after freezing different combinations of layers with an approach where we use a specific learning rate for the encoder and another for the decoder, we determined that our best model involved freezing up to layer 10 in ViT and GPT-2.

	Best model so far: ResNet-50 + GRU	ViT🔥 GPT❄️	ViT❄️ GPT🔥	ViT🔥 GPT🔥
BLEU_1	0.126	0.155	0.196	**0.218**
BLEU_2	-	0.058	0.095	**0.113**
ROUGE_L	0.159	0.119	0.175	**0.196**
METEOR	0.062	0.079	0.111	**0.130**

🔥Fine-tune ❄️Freeze

QUALITATIVE RESULTS

Groundtruth	Zucchini-Lentil Fritters With Lemony Yogurt	Grilled Chicken with Board Dressing
R18 + GRU	with with	grilled with
R18 + LTSM	with with	grilled and with
R50 + GRU	grilled grilled with	chicken with with
R50 + LTSM	grilled with	chicken with with
R18 + Transf	chicken - chicken chicken chiken	grille - chicken chicken with with with
R50 + GRU(TF)	cchini - yogurt	Chicken with -
ViT ft GPT fz	pucchini andFemoniled-illedter with Carong andurt	ailled Chicken with Greeningressing
ViT fz GPT ft	Cucchini andandimeil Chickenritters Baconong Butterurt	Grilled Chicken with Lemon-ressing and
ViT ft GPT fz	Cucchini-Stimeil Friedritters Baconony Yogurt	Grilled Chicken with Lemon-ressing

- ViT+GPT fine-tuned models produce significantly more coherent and complete dish names.
- While previous models struggled with repetition and incomplete phrases, successfully capture the structure and components of the original titles like "Grilled Chicken with Lemon-dressing".

QWEN

Qwen 2.5 VL 7B is a 7-billion-parameter multimodal language model based on a Transformer architecture with dense attention and an integrated visual encoder. In our setup, we used the instruct-tuned variant (Qwen2.5-VL-7B-Instruct) to generate image-based titles, following an image captioning approach guided by structured prompts.

We noticed that being more descriptive and providing examples in our prompt enhanced the performance on our test set, so we try different prompts:

Prompt 1: "Give 1 phrase describing the food in the image from its key ingredients."

- **Ground truth:** Zucchini-Lentil Fritters With Lemony Yogurt
- **Prompt 1:** Zucchini and Cheese Fritters

- **Ground truth:** Anchovy and Rosemary Roasted Lamb
- **Prompt 1:** The image depicts a plate of roasted meat with potatoes and carrots, garnished with herbs.

	Best model so far: ViT🔥 GPT🔥	Prompt 1 (baseline)
BLEU_1	0.218	0.139
BLEU_2	0.113	0.064
ROUGE_L	0.196	0.205
METEOR	0.130	0.177

QWEN

Qwen 2.5 VL 7B is a 7-billion-parameter multimodal language model based on a Transformer architecture with dense attention and an integrated visual encoder. In our setup, we used the instruct-tuned variant (Qwen2.5-VL-7B-Instruct) to generate image-based titles, following an image captioning approach guided by structured prompts.

We noticed that being more descriptive and providing examples in our prompt enhanced the performance on our test set, so we try different prompts:

Prompt 2: "Give a recipe title describing the food in the image. If no title is found give an empty string."

- Ground truth: Zucchini-Lentil Fritters With Lemony Yogurt
- **Prompt 1**: Zucchini and Cheese Fritters
- **Prompt 2**: Zucchini Fritters

- Ground truth: Anchovy and Rosemary Roasted Lamb
- **Prompt 1**: The image depicts a plate of roasted meat with potatoes and carrots, garnished with herbs.
- **Prompt 2**: Roast Beef with Root Vegetables and Herbs

	Best model so far: ViT🔥 GPT🔥	Prompt 1 (baseline)	Prompt 2
BLEU_1	0.218	0.139	0.237
BLEU_2	0.113	0.064	0.125
ROUGE_L	0.196	0.205	0.228
METEOR	0.130	0.177	0.176

QWEN

Qwen 2.5 VL 7B is a 7-billion-parameter multimodal language model based on a Transformer architecture with dense attention and an integrated visual encoder. In our setup, we used the instruct-tuned variant (Qwen2.5-VL-7B-Instruct) to generate image-based titles, following an image captioning approach guided by structured prompts.

We noticed that being more descriptive and providing examples in our prompt enhanced the performance on our test set, so we try different prompts:

Prompt 3: "generate exactly 1 title for the recipe/food/drink based on its key ingredients. Each title should be clear, catchy, and descriptive, highlighting the most important or unique ingredients in the dish. For example, if the key ingredients are potatoes and seasoning, a title might be 'Crispy Salt and Pepper Potatoes'. For a dish with mac and cheese and Thanksgiving flavors, a title could be 'Thanksgiving Mac and Cheese'. You can also include elements that reflect the style or theme of the recipe (e.g., 'Italian Sausage and Bread Stuffing'). Return only the title."

- Ground truth: Zucchini-Lentil Fritters With Lemony Yogurt
- **Prompt 1:** Zucchini and Cheese Fritters
- **Prompt 2:** Zucchini Fritters
- **Prompt 3:** Zucchini and Feta Fritters

- Ground truth: Anchovy and Rosemary Roasted Lamb
- **Prompt 1:** The image depicts a plate of roasted meat with potatoes and carrots, garnished with herbs.
- **Prompt 2:** Roast Beef with Root Vegetables and Herbs
- **Prompt 3:** Herb-Crusted Roast Beef with Garlic Mashed Potatoes

	Best model so far: ViT🔥 GPT🔥	Prompt 1 (baseline)	Prompt 2	Prompt 3 (descriptive +examples)
BLEU_1	0.218	0.139	0.237	**0.244**
BLEU_2	0.113	0.064	0.125	**0.126**
ROUGE_L	0.196	0.205	**0.228**	0.221
METEOR	0.130	0.177	0.176	**0.180**

The quantitative results from both Prompt 2 and Prompt 3 were quite similar. Prompt 3, being more detailed and including examples, provided extra guidance that led to slightly more descriptive titles. However, Prompt 2, which was simpler and more direct, also produced high-quality results. Compared to Prompt 1, both generated titles that were closer to the ground truth, being less general and more aligned with the expected output.

QUALITATIVE RESULTS

Groundtruth	Zucchini-Lentil Fritters With Lemony Yogurt	Grilled Chicken with Board Dressing
R18 + GRU	with with	grilled with
R18 + LTSM	with with	grilled and with
R50 + GRU	grilled grilled with	chicken with with
R50 + LTSM	grilled with	chicken with with
R18 + Transf	chicken - chicken chicken chiken	grille - chicken chicken with with with
R50 + GRU(TF)	cchini - yogurt	Chicken with -
ViT ft GPT fz	pucchini andFemoniled-illedter with Carong andurt	ailled Chicken with Greeningressing
ViT fz GPT ft	Cucchini andandimeil Chickenritters Baconong Butterurt	Grilled Chicken with Lemon-ressing and
ViT ft GPT fz	Cucchini-Stimeil Friedritters Baconony Yogurt	Grilled Chicken with Lemon-ressing
QWEN (prompt 3)	Zucchini and Feta Fritters	Herb-Crusted Roasted Chicken with Garlic and Lime

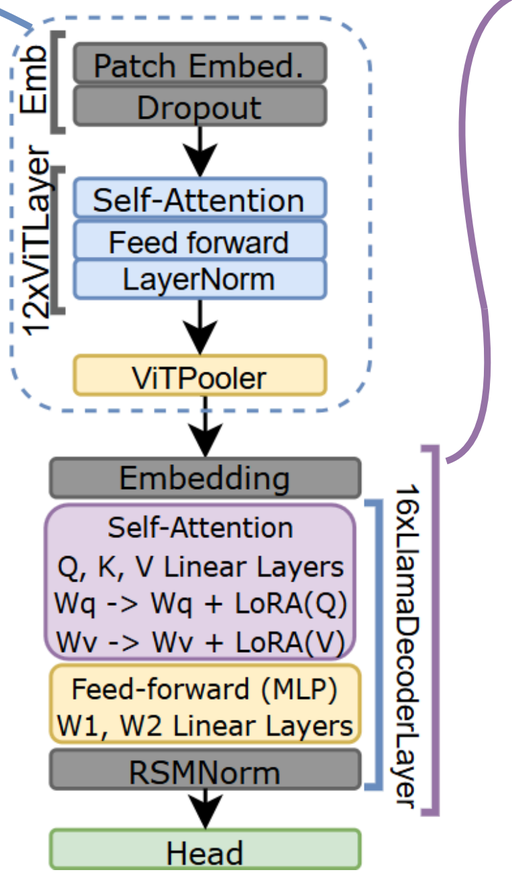
ARCHITECTURE OVERVIEW: LLAMA + LORA

VIT: FEATURE EXTRACTOR

In this setup, ViT serves purely as a feature extractor.

The output of the final transformer layer is pooled into a single vector that summarizes the visual content.

This approach allows for global reasoning over the image without relying on inductive biases inherent to convolutional architectures.



LLAMA 3.2

Llama 3.2 is a 16-layer autoregressive transformer that generates captions token by token, conditioned on the projected ViT feature embeddings.

Why Llama?

LLaMA offers a more efficient transformer architecture than GPT-2. LLaMA's architecture allows it to generate more fluent, contextually coherent captions while requiring fewer computational resources, making it a more efficient option for fine-tuning on specific tasks.

Why LoRA?

It enables to fine-tune LLaMA with minimal computational cost. By injecting low-rank adaptation layers into the attention mechanism, we can adapt the model for our specific task with fewer trainable parameters.

LORA ANALYSIS

We analyzed the impact of different Lora configurations. Specifically, with :

- **r** (rank): that controls the dimensionality of the low-rank matrices used to approximate the original weight updates.
- **alpha**: acts as a scaling factor for the LoRA updates, affecting how strongly the low-rank updates influence the model.

We also tested two different dropout rates (0.05 and 0.1) to evaluate regularization effects, and kept the target modules fixed at ["q_proj", "v_proj"], which are key projection layers in transformer architectures where LoRA is applied.

We tested two configurations:

1. α=8 with r=4, tested in both versions of the model.
2. α=16 with r=4. we were able to experiments with this configuration only with 1B,version 3B with this setup resulted in a CUDA out-of-memory error during training.

QUANTITATIVE RESULTS

	Best model so far: QWEN	Llama 3.2 1B	Llama 3.2 3B
BLEU_1	0.244	0.260	**0.269**
BLEU_2	0.126	0.134	**0.141**
ROUGE_L	0.221	0.223	**0.228**
METEOR	0.180	0.155	**0.160**

QUALITATIVE RESULTS

Groundtruth	Zucchini-Lentil Fritters With Lemony Yogurt	Grilled Chicken with Board Dressing
R18 + GRU	with with	grilled with
R18 + LTSM	with with	grilled and with
R50 + GRU	grilled grilled with	chicken with with
R50 + LTSM	grilled with	chicken with with
R18 + Transf	chicken - chicken chicken chiken	grille - chicken chicken with with with
R50 + GRU(TF)	cchini - yogurt	Chicken with -
ViT ft GPT fz	pucchini andFemoniled-illedter with Carong andurt	ailled Chicken with Greeningressing
ViT fz GPT ft	Cucchini andandimeil Chickenritters Baconong Butterurt	Grilled Chicken with Lemon-ressing and
ViT ft GPT fz	Cucchini-Stimeil Friedritters Baconony Yogurt	Grilled Chicken with Lemon-ressing
QWEN (prompt 3)	Zucchini and Feta Fritters	Herb-Crusted Roasted Chicken with Garlic and Lime
LLAMA 3.2 1B	Pucchini andemonil Fritters Lemonemony Yogurt	Roilled Chicken with Lemonering
LLAMA 3.2 3B	Grucchini andentil Saladritters with Yogemony Yogurt Sauce	Grilled Chicken with Lemon Buttering

MODELS EXPLORATION

Prompt	sd-turbo	stable-diffusion-2-1	sdxl-turbo	stable-diffusion-xl-base-1.0	stable-diffusion-3.5-large-turbo	stable-diffusion-3.5-medium
“Strawberry Coconut Cake”						
“People walking in the street of a city”						

1 Initial Model Comparison and Exploration

- We compared various models using two prompts.
- Turbo models generated images faster but with lower detail and quality. Non-turbo models (like XL and 3.5 series) produced higher quality, with 3.5 showing superior detail in complex city scenes.

MODELS EXPLORATION

Prompt	sd-turbo	stable-diffusion-2-1	sdxl-turbo	stable-diffusion-xl-base-1.0	stable-diffusion-3.5-large-turbo	stable-diffusion-3.5-medium
“Strawberry Coconut Cake”						
“People walking in the street of a city”						

2 Diffusion Techniques and Schedulers

- Schedulers manage noise addition and removal across timesteps, affecting both image quality and speed.
- DDPM requires many steps (~1000) for high quality, making it slower.
- DDIM offers faster, high-quality generation by reducing steps significantly (~50).

MODELS EXPLORATION

Prompt	sd-turbo	stable-diffusion-2-1	sdxl-turbo	stable-diffusion-xl-base-1.0	stable-diffusion-3.5-large-turbo	stable-diffusion-3.5-medium
“Strawberry Coconut Cake”						
“People walking in the street of a city”						

3 Key Parameters and Their Effects

- Number of Inference Steps: More steps improve image detail but slow down generation; around 50-80 steps offer a good quality-speed balance.
- CFG : Higher guidance_scale values increase prompt adherence but can reduce diversity if set too high.

MODELS EXPLORATION

Prompt	sd-turbo	stable-diffusion-2-1	sdxl-turbo	stable-diffusion-xl-base-1.0	stable-diffusion-3.5-large-turbo	stable-diffusion-3.5-medium
“Strawberry Coconut Cake”						
“People walking in the street of a city”						

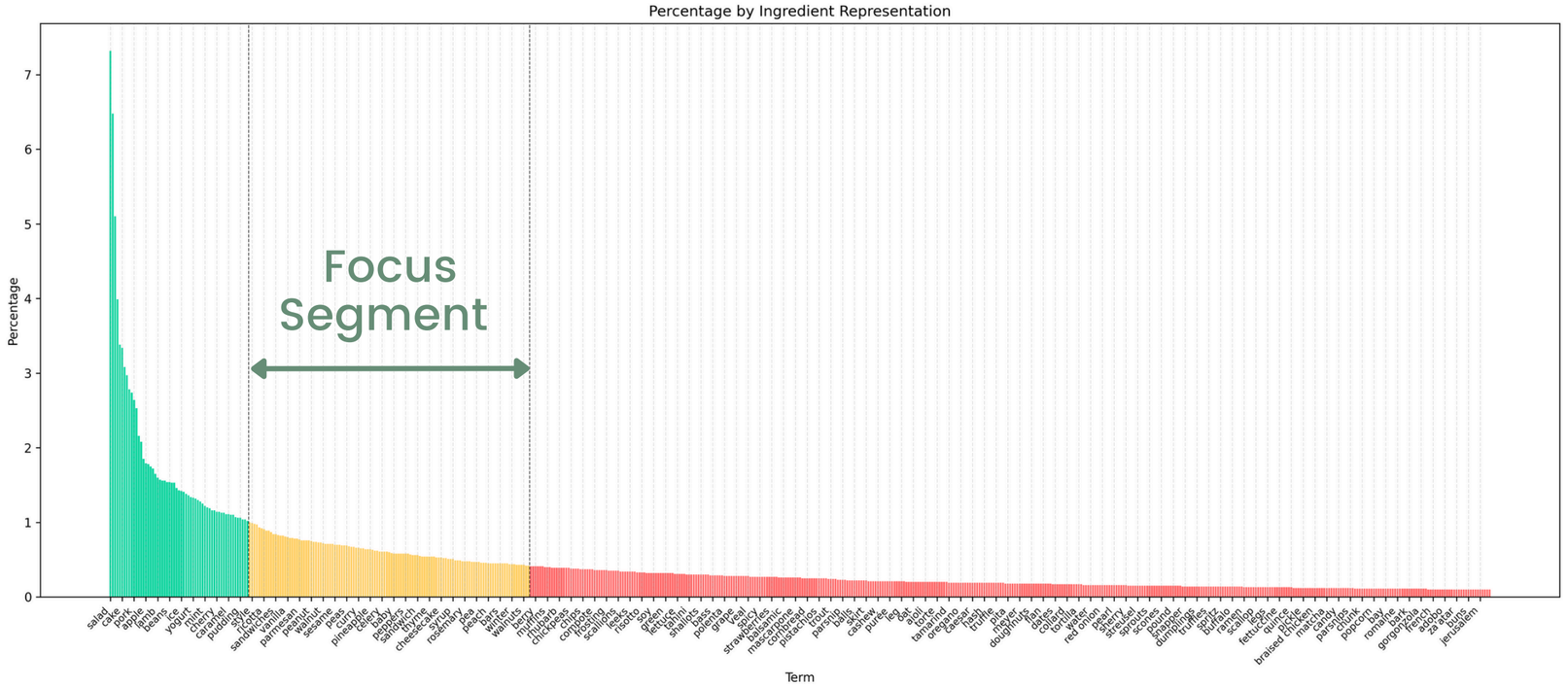
4 Conclusions

- Sheduler: DDIM is more effective than DDPM.
- More denoising steps improved detail.
- Guidance techniques enhanced prompt alignment and output quality.
- negative prompts helped suppress unwanted elements.

Chosen configuration based on the balance between image quality, prompt adherence, and generation time:

- Model: stabilityai/stable-diffusion-xl-base-1.0
- Include negative prompts.
- Scheduler: DDIM
- Number of inference steps: 80
- Guidance Scale: 8

PROBLEM FORMULATION



Common Appear in over 130 images

Under-represented ★ Targeted Oversampling

Discarded We discarded ingredients with less than 52 images

1 Dataset Exploration

Using **spaCy's 'en_core_web_lg'** pipeline, we extracted and analyzed food terms from image captions

2 Term Classification

We classify our food terms into 3 classes:

- **Common (>1% frequency)**: Dominant ingredients
- **Underrepresented (0.4-1%)**: Less frequent ingredients **(Focus Segment)**
- **Discarded (<0.4%)**: Rare terms

3 Problem Formulation

How does **targeted oversampling of underrepresented ingredients using Diffusion Models** affect model predictions in an image captioning task?

GENERATION PIPELINE

The goal of this pipeline is to automatically generate high-quality, photorealistic images along with captions of food dishes containing specified ingredients, specifically for training our food captioning model.

Ingredient Selection

- Number of ingredients selected: 87
- Selection criteria: Ingredients that belong to our predefined focus segment.
- Image generation per ingredient: 20 images each.
- Total images generated: 1,740 photorealistic images

Prompt Engineering (LLM Phase)

- Utilizes meta-llama/Llama-3.2-1B-Instruct, a compact LLM for generating short, vivid food image prompts.
- Ensures each prompt includes a specific ingredient in a dish context rather than as a standalone item.
- Prompts include details about cooking style, plating, and food presentation.

Image Generation (Diffusion Model)

- Model: stabilityai/stable-diffusion-xl-base-1.0
- Prompt: from LLM
- Negative Prompt: "deformed, unappetizing, incorrect texture, malformed , blurry, low quality, cartoon, CGI, ,..."
- Scheduler: DDIM
- Number of inference steps: 80
- Guidance Scale: 8

Caption Generation

- Qwen2.5-VL to generate captions for the 1,740 synthetically generated images.
- It handles complex visual inputs and generates coherent titles, improving the realism and usefulness of our synthetic dataset.
- It address the common issue of non-descriptive or generic image captions

RESULTS - EXAMPLES

Grilled Zucchini with Garlic and Herbs

Zucchini Fritters with Lemon and Basil

Zucchini Lasagna with Basil and Cheese

Teriyaki Chicken and Zucchini Stir-Fry Bowl

Asparagus and Zucchini Boats with Creamy Sauce

Zucchini Salad with Almonds and Basil

ZUCCHINI-BASED PROMPTS

- The model successfully produced a diverse array of realistic food dishes featuring zucchini.
- Zucchini was meaningfully integrated into the recipes (e.g., fritters, salad).
- Results demonstrated strong ingredient-context integration.
- The generated images showed high visual realism

RESULTS

Indian Butter Chicken with Naan and Coriander

Chicken Tikka Masala with Cilantro Rice

Grilled Pineapple Skewers with Curry Dip

Chicken Tikka Masala with Basmati Rice

CURRY-BASED PROMPTS

- Curry-based outputs lacked diversity across generations.
- Nearly all images depicted chicken curry with slight variations (e.g., with rice or naan).
- Despite different prompts, the model fixated on a single dish archetype.
- This fixation reduced the variability and effectiveness of augmentation.

RESULTS

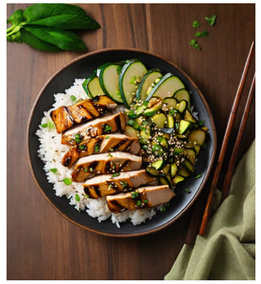
ZUCCHINI-BASED PROMPTS


Grilled Zucchini with Garlic and Herbs


Zucchini Fritters with Lemon and Basil


Zucchini Lasagna with Basil and Cheese


Zucchini Salad with Almonds and Basil


Teriyaki Chicken and Zucchini Stir-Fry Bowl

CURRY-BASED PROMPTS


Indian Butter Chicken with Naan and Coriander


Chicken Tikka Masala with Cilantro Rice


Chicken Tikka Masala with Basmati Rice

Bias Reinforcement

The generation model might reinforce existing biases in food image-caption pairs (e.g., "curry = chicken curry"), thus failing to inject new concepts into the dataset.

Low Caption Variance

Even though the images are photorealistic, the captions are often too similar or predictable, especially when the generated dishes are repetitive (as in the curry case).

Synthetic Realism Gap

It's not just about making images that look realistic—what really matters is whether they introduce new and varied patterns the model can learn from. If the synthetic data is too repetitive or too similar to what's already in the training set, the model may not gain much from it, even if the images appear high-quality.

FINAL MODEL

	llama 3.2-B	llama 3.2-B with augmentation
BLEU_1	0.269	0.277
BLEU_2	0.141	0.150
ROUGE_L	0.228	0.248
METEOR	0.160	0.172

These improvements suggest a minor benefit from the augmentation strategy, but not enough to conclude that it had a significant positive impact on the model's captioning ability.

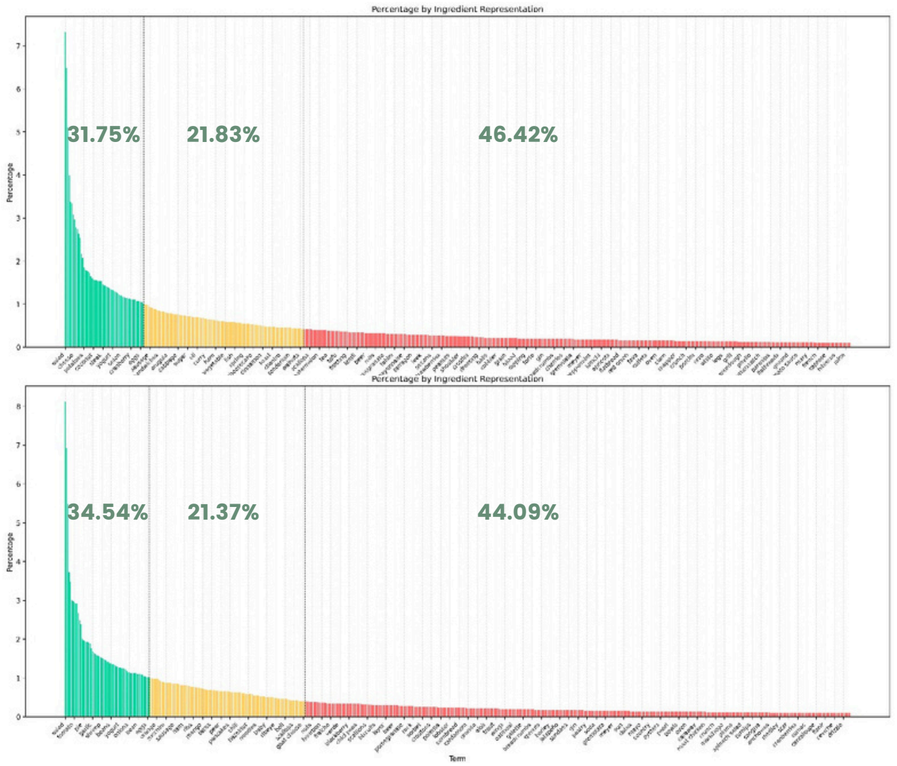
BLEU-1: +0.008

METEOR: +0.012

ROUGE-L: +0.02

KEY TAKEAWAYS

- Synthetic samples were visually promising and modestly improved ingredient balance.
- Limitations such as repetitive dish types, lack of linguistic diversity, and minimal novelty likely hindered performance gains.
- Exploration of cuisine-level imbalance revealed many underrepresented styles (e.g., French, Mexican, Chinese).
- Initially, cuisine was considered less important than ingredients, but combining both could improve balance and diversity in augmentation.



- Augmentation helped smooth ingredient distribution in some areas but unintentionally reinforced imbalance in others.
- The filtering step, which excluded extremely rare ingredients, likely contributed to this issue.
- Although individually rare, these ingredients collectively represent a significant portion of the dataset.

Future work could involve clustering rare ingredients into conceptual groups to enable more scalable augmentation.