



TNSDC - GENERATIVE AI FOR ENGINEERING

FINAL PROJECT

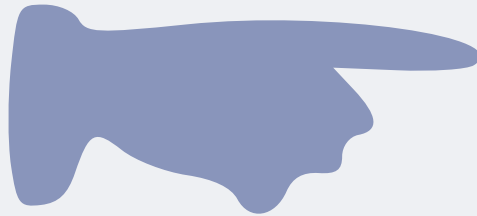
SUBMITTED BY
SAI SHIVANIE M
(311521104047)



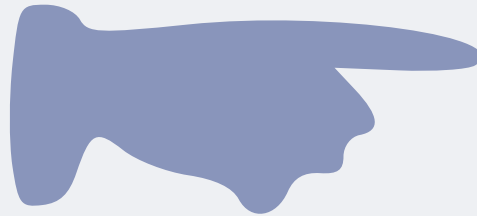
PROJECT TITLE

**AUTOMATIC RECIPE
GENERATOR**

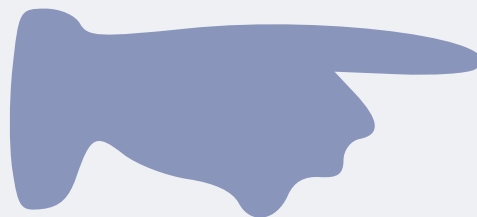
AGENDA



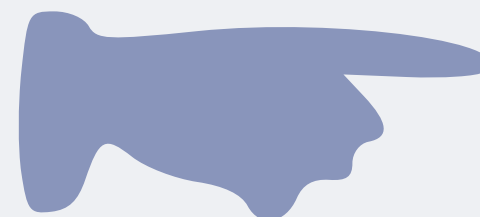
PROBLEM STATEMENT



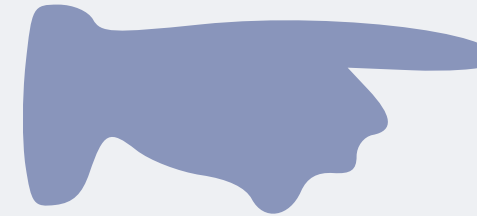
END USERS



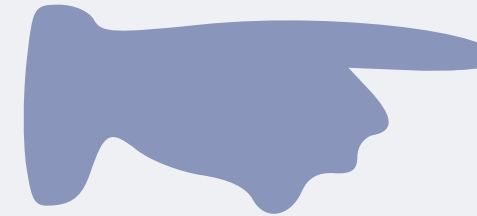
**THE WOW IN THE
SOLUTION**



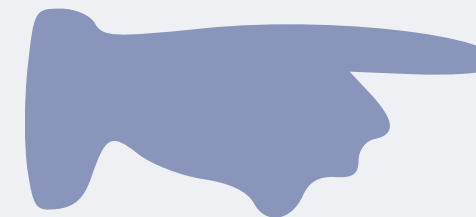
RESULTS



PROJECT OVERVIEW



**SOLUTION AND ITS
VALUE PROPOSITION**



MODELLING



PROBLEM STATEMENT

The task is to develop an automatic recipe generator capable of producing human-like instructions based on provided ingredients. Using advanced natural language processing techniques and the GPT-2 model, the system should generate coherent and realistic recipes tailored to user inputs. This solution aims to streamline the recipe creation process, assist in meal planning, and inspire culinary creativity. The generated recipes should mimic the style and structure of human-generated content while offering a convenient and efficient alternative for recipe discovery and generation.

PROJECT OVERVIEW



The Automatic Recipe Generator is a Python code that leverages the GPT-2 model, a powerful language model, in combination with the Transformers library to automatically create detailed recipes based on a list of ingredients provided by the user. The tokenizer is configured to better suit the recipe generation task. Then a dataset of Indian food recipes is loaded. It performs initial data preprocessing, shuffling the dataset for variety and resetting the index. For model training, a custom dataset class(RecipeDataset) is defined. This class tokenizes and encodes the recipe strings, generating input ID's and attention masks.

The GPT-2 model is fine-tuned on the recipe dataset. The training process includes configuring various training parameters and utilizing the Trainer class from Transformers. The model learns to generate contextual relevant recipes. A function is created to generate recipes where a list of ingredients are given as input, constructs a recipe prompt, and utilizes the fine-tuned GPT-2 model to generate text. The generated text is returned as a complete recipe.

END USERS

01

Home Cooks: Individuals who enjoy cooking at home and seek inspiration for new recipes using ingredients they have on hand.

03

Busy Professionals: Individuals with limited time for meal planning and preparation who rely on quick and convenient recipe suggestions for their daily cooking needs.

02

Food Enthusiasts: People passionate about culinary exploration who are eager to discover unique and creative recipes tailored to their tastes.

04

Cooking Novices: Beginners in the kitchen who benefit from guided recipe suggestions to build their cooking skills and confidence

SOLUTION AND ITS VALUE PROPOSITION



Automatic recipe generator utilizes natural language processing technology to provide users with personalized and quality recipe suggestions. By inputting their available ingredients, users can instantly receive detailed and coherent recipes to their preferences. The solution streamlines the recipe discovery process, offering convenience and inspiration for home cooks, food enthusiasts, and busy individuals alike. With its ability to mimic human-generated instructions, automatic recipe generator empowers users to explore new culinary creations, save time on meal planning, and unleash their creativity in the kitchen.

‘mOm, eHt

Automatic recipe generator brings the magic of human-like creativity to the digital realm. By harnessing the power of advanced natural language processing, the solution offers users an unparalleled experience of discovering, creating, and enjoying delicious recipes effortlessly. With just a few clicks, users can unlock a world of culinary possibilities tailored to their preferences and ingredients on hand. Whether you're a seasoned chef or a kitchen novice, the solution ignites the spark of inspiration and transforms everyday cooking into a delightful adventure. Get ready to be amazed as you explore a treasure trove of mouthwatering recipes that cater to your tastes and preferences like never before.

THE SOLUTION

The automatic recipe generator leverages the GPT-2 model, a cutting-edge language generation model developed by OpenAI. GPT-2 excels at understanding and generating human-like text based on input prompts. By fine-tuning GPT-2 on a dataset of recipes, our model learns the structure and style of human-generated recipes. During generation, the model takes ingredients as input and generates recipe instructions. With its ability to capture nuances of language and culinary knowledge, our model ensures that the generated recipes are not only grammatically correct but also semantically meaningful and aligned with human-generated content.

```
[ ] 1 train_ds = RecipeDataset(train_data)
    2 val_ds = RecipeDataset(val_data)
```

100% 339/339 [00:00<00:00, 709.73it/s]

100% 60/60 [00:00<00:00, 333.10it/s]

```
[ ] 1  trainer.train()
```

[255/255 05:38, Epoch 3/3]

Step	Training Loss
------	---------------

```
TrainOutput(global_step=255, training_loss=1.3587970808440564, metrics={'train_runtime': 343.0826, 'train_samples_per_second': 2.964, 'train_steps_per_second': 0.743, 'total_flos': 531467993088000.0, 'train loss': 1.3587970808440564, 'epoch': 3.0})
```

```
[ ] 1 tokenizer = GPT2TokenizerFast.from_pretrained(model_name,  
2 | | | | | | | | | | | | | | bos_token='<|startoftext|>',  
3 | | | | | | | | | | | | | | eos_token='<|endoftext|>',  
4 | | | | | | | | | | | | | | unk_token='<|unknown|>',  
5 | | | | | | | | | | | | | | pad_token='<|pad|>' )  
6 | | | | | | | | | | | | | | )  
7 model = GPT2LMHeadModel.from_pretrained(model_name)  
8 model.resize_token_embeddings(len(tokenizer))
```

```
1 tokenizer.save_pretrained(model_save_path)
```

```

'./khaanaGPT/tokenizer_config.json',
'./khaanaGPT/special_tokens_map.json',
'./khaanaGPT/vocab.json',
'./khaanaGPT/merges.txt',
'./khaanaGPT/added_tokens.json',
'./khaanaGPT/tokenizer.json')

```

```
Downloading (...)olve/main/vocab.json: 100% 1.04M/1.04M [00:00<00:00, 1.29MB/s]
```

Downloading (...)olve/main/merges.txt: 100% 456k/456k [00:00<00:00, 1.83MB/s]

```
Downloading (...) /main/tokenizer.json: 100% 1.36M/1.36M [00:00<00:00, 1.51MB/s]
```

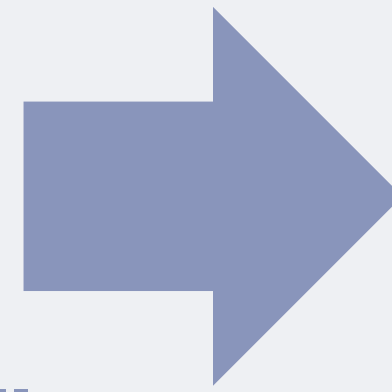
```
Downloading (...)live/main/config.json: 100% 665/665 [00:00<00:00, 40.0kB/s]
```

Downloading model.safetensors: 100% 548M/548M [00:04<00:00, 145MB/s]

```
Downloading (...)neration  config.json: 100% 124/124 [00:00<00:00, 6.61kB/s]
```

Embedding(50260, 768)

RESULTS



The automatic recipe generator successfully generates human-like recipes based on provided ingredients. The generated recipes exhibit readability, closely resembling recipes created by humans. Users can input a list of ingredients, and the model produces detailed instructions for preparing a dish incorporating those ingredients. The generated recipes capture culinary nuances and provide step-by-step guidance, making them easy to follow for users of all skill levels. With its ability to mimic human-generated content, automatic recipe generator offers a seamless and enjoyable experience for discovering new recipes and exploring culinary creativity.



Generating Recipes

```
[ ] 1 for ing in ingredients:
    2     prompt = create_prompt(ing)
    3     print(generator(prompt,
    4         max_new_tokens=512,
    5         penalty_alpha=0.6,
    6         top_k=4,
    7         pad_token_id=50259
    8     )[0]['generated_text'])
    9 def generate_instructions(ingredients):
    10     # Create a prompt with ingredients
    11     prompt = f"Ingredients:\n{' '.join(ingredients)}\n\nInstructions:"
    12     generated_instructions = generated_text[0]['generated_text'].split('Instructions:', 1)[-1].strip()
    13     return generated_instructions
```

```
<|startoftext|>Ingredients:
1/2 teaspoon turmeric powder (haldi)
1 tablespoon coriander (dhania) seeds
4 dry red chillies
1 teaspoon sesame (gingelly) oil
4 cloves garlic
1 teaspoon garam masala powder
salt - to taste
2 tablespoon sesame (gingelly) oil
4 green chillies - slit
500 grams chicken
6 cloves garlic - finely chopped
1 teaspoon fennel seeds (saunf)
2 onion - chopped
2 cups sorrel leaves (gongura) - picked and chopped
1/4 teaspoon methi seeds (fenugreek seeds)
1 inch ginger - finely chopped
1 tomato - chopped
Salt - to taste

To make the Masala Powder, wash the chicken in a pan.
In a small saucepan, add the turmeric powder and coriander and cook until the turmeric powder starts to boil.
Once the turmeric powder is boiling, remove from the heat and let it cook for a minute.
Once the turmeric powder has boiled, add the chicken and cook until it turns translucent from the chicken.
Add the sesame oil and cook till the sesame turns translucent and soft.
Once cooked, drain off excess oil and keep aside.
In a small bowl, add the coriander and sauté for 2 whistles.
Once the coriander becomes translucent, add the chopped ginger and cook until the ginger turns golden and crisp.
Once cooked, remove the chicken and let them rest for 5 whistles.
After 5 whistles, add the coriander and serve.
```

OUTPUT



GITHUB LINK

<https://github.com/SaiShivanie/TNSDC-GENERATIVE-AI.git>

THANK YOU

