# AI Recipe Generator: An Intelligent Culinary Assistant Using GPT-2 and Multi-Modal AI

A PROJECT REPORT

*Submitted by*

**Darshan Patel (916654430)**
**Kavan Patel (916659154)**

**Course:** CIS 5603 – Artificial Intelligence
**December 15, 2025**

# Index

# Abstract

This project presents the creation and integration of an AI-driven recipe creator that revolutionizes the conventional paradigm in cooking assistance. Traditional recipe platforms, grounded as they are in static databases, require their users to know what they want before they can search for it. Our system generates personalized recipes from any combination of available ingredients, while providing nutritional insights and intelligent categorization. The system embodies three key AI components: a fine-tuned GPT-2 language model for creative recipe generation, with 85% coherence; an ensemble machine learning classifier system, where 72.9% accuracy cuisine prediction and 64.0% accuracy meal type classification; and a real-time nutrition analysis engine, combining local caching with direct integration with the USDA API. Implementation addresses practical challenges arising in inconsistent recipe format, slow nutrition data retrieval, and unreliable APIs through innovative solutions such as structured prompt engineering, two-tier caching systems, and in-depth text processing. Experimental results demonstrate the effectiveness of the developed system across multiple metrics: generation speed under 3 seconds per recipe, classification accuracy, and user satisfaction. Future enhancements will involve generating visual recipes using GANs, further development of a mobile application, and personalized user profiling. The work presented is expected to contribute much to the area of applied AI, proving how language models can be effectively adapted for creative and practical domains without compromising on usability and reliability.

# 1. Introduction

## 1.1 Background and Motivation

Cooking is a universal human creative activities, in which art, science, culture, and nutrition blend together. Modern people, as a rule, are confronted with solving the problem of how to get food with limited time, ingredients, or cooking experience. Traditional recipe digital assistants tend to be little more than searchable databases: unless users already know a specific dish's name or can identify the exact combination of ingredients they want, they will not find relevant recipes. The strategy above has given a large cognitive gap: very often, people have ingredients on hand but lack inspiration or knowledge of how to put them together creatively.

This revolution of artificial intelligence technologies, especially in natural language processing and machine learning, provides an opportunity for people to adopt to perform cooking. AI-powered systems are able to go beyond the simple retrieval of existing recipes and suggest creative combinations, provide nutritional advice, and customize according to individual constraints and preferences. This shift from information retrieval to creative generation illustrates the fundamental shift in the role technology can play in supporting everyday creative tasks.

## 1.2 Problem Statement

There are some problems of the current platforms for recipe search: (1) A person must know what he or she wants to cook before searching, (2) There is no personalization with respect to ingredients, dietary restrictions, or time limitations, (3) Nutritional information is not provided, and (4) There is no intelligent categorization of ingredients that understand relationships between ingredients unlike mere keywords.

These shortcomings pose some challenges, such as wasting food (unutilized ingredients because of a lack of culinary ideas), nutritional ignorance (inability to evaluate the nutritional soundness of a meal), and culinary stagnation (cooking known recipes). Our proposed solution to the shortcomings is to create an intelligent system that can produce recipes with the ingredients at hand, classify recipes intelligently, as well as conduct detailed nutritional analysis.

## 1.3 Objectives

The primary objectives of this project are:

1. To develop an AI system that creates coherent, practical recipes from different ingredient combinations
2. To implement intelligent classification systems that predict cuisine type and meal category from ingredient lists
3. To create a nutrition analysis engine that provides accurate, real-time nutritional information
4. To design a user-friendly interface that integrates these components into a seamless cooking assistant
5. To evaluate system performance across multiple metrics including coherence, accuracy, speed, and user satisfaction

## 1.4 Contributions

Our work makes several contributions to the field of applied AI:

- **Methodological**: Development of specialized prompt engineering and fine-tuning techniques for GPT-2 in the culinary domain
- **Architectural**: Design of a three-component AI system (generation, classification, nutrition) with integrated processing
- **Practical**: Implementation of solutions to real-world challenges including inconsistent formatting and slow API responses
- **Empirical**: Comprehensive evaluation of multiple machine learning models for recipe classification
- **Applicative**: Demonstration of AI's potential in creative, everyday domains beyond conventional technical applications

# 2. Related Work

## 2.1 Recipe Generation Systems

Early computational approaches in automatic recipe generation have employed rule-based systems and template-filling approaches. In such systems, generality increases, but at the same time, it develops consistency without variation and hence lacks creativity and flexibility. The introduction of statistical language models thus does more natural recipe generation, but RNNs and LSTMs showed promising performance on generating sequential text.

These transformer-based models, mainly GPT-2 and GPT-3, revolutionized recipe generation by making long-form text coherence with better contextual understanding possible. However, their applications to cooking needed considerable adaptation to be gastronomically practical and nutritionally sensible-addressing specific areas of our work.

## 2.2 Food Classification and Categorization

Recipe categorization based on cuisine and meal type can be handled using different machine learning algorithms. Conventional approaches used bag-of-words models with classifiers such as Support Vector Machines (SVMs) and Random Forests.

More modern solutions have leveraged word embeddings and deep learning models. Chen and Ngo (2016) have applied convolutional neural networks to recipe categorization, and Teng et al. (2018) focused on attention models for highlighting important ingredients with respect to a particular cuisine. Our research builds on these solutions by using ensemble learning and a combination of ingredients and cooking methods via feature engineering.

## 2.3 Nutritional Analysis Systems

Nutritional calculation in cooking systems has evolved from simple lookup tables to sophisticated estimation models. The USDA FoodData Central database serves as the primary source for nutritional information in many systems, though API limitations require careful engineering for scalable applications.

Machine learning approaches to nutritional estimation have shown promise, particularly for complex dishes where ingredient proportions are unclear. However, these models often sacrifice accuracy for generality. Our two-tier approach—combining precise measurements for known ingredients with intelligent estimation for others—represents a balanced solution to this challenge.

## 2.4 AI in Creative Domains

The application of AI to creative tasks has grown significantly, with systems generating art, music, and literature. The success of these systems demonstrates AI's potential as a creative collaborator rather than just an analytical tool. Our work contributes to this growing field by exploring AI's role in culinary creativity—a domain that combines technical precision with artistic expression.

## 2.5 Integration Challenges

Previous systems have typically focused on individual aspects of recipe assistance (generation, classification, or nutrition) rather than integrating them comprehensively. The technical challenges of integrating multiple AI components—particularly ensuring consistency between generated recipes and their classifications and nutritional profiles—have limited previous integrated approaches. Our architecture addresses these integration challenges through parallel processing and validation mechanisms.

# 3. Methodology

## 3.1 System Architecture

The AI Recipe Generator has a three-tier architecture comprising a presentation layer, a processing layer, and a data layer. The presentation layer, which includes a web-based user interface with intuitive input mechanisms for ingredients, meal preferences, and dietary constraints. The processing layer contains the core AI components operating in a coordinated pipeline. The data layer manages recipe storage, nutritional databases, and user preferences.

The processing pipeline follows this flow: (1) User input preprocessing and validation, (2) Parallel processing through generation, classification, and nutrition components, (3) Results integration and formatting, (4) Output delivery with interactive elements for refinement and exploration.

## 3.2 GPT-2 Recipe Generation

### 3.2.1 Model Selection and Fine-tuning

We have selected GPT-2 (124M parameter version) as our base model due to its balance of capability and computational efficiency. The model was fine-tuned on approximately 100,000 structured recipes from the RecipeNLG dataset, which includes diverse cuisines, cooking styles, and ingredient combinations.

The fine-tuning process involved several key adaptations:

- **Structured Training Data**: Recipes were formatted with explicit sections (title, ingredients, instructions, cooking time, difficulty) to teach the model proper organization
- **Ingredient-Aware Training**: Special attention was given to ingredient lists and measurements during training
- **Context Preservation**: The model was trained to maintain consistency between ingredients mentioned and instructions provided

### 3.2.2 Prompt Engineering

Prompt engineering proved crucial for generating usable recipes. We developed a structured prompt template that includes:

1. Ingredient list with context (e.g., "using chicken, rice, and garlic")
2. Meal type specification (e.g., "for a dinner recipe")
3. Format requirements (e.g., "include measurements and step-by-step instructions")
4. Practical constraints (e.g., "keep preparation under 30 minutes")

This structured prompting increased coherent output from approximately 45% to 85%.

### 3.2.3 Generation Parameters

Key generation parameters were optimized through iterative testing:

- **Temperature (0.7)**: Balances creativity (higher values) with coherence (lower values)
- **Top-p Sampling (0.9)**: Uses nucleus sampling to maintain diversity while avoiding unlikely combinations
- **Max Length (500 tokens)**: Ensures concise, practical recipes
- **Repetition Penalty**: Prevents redundant ingredient mentions or instruction loops

## 3.3 Machine Learning Classification System

### 3.3.1 Dataset Preparation

We utilized the Food.com recipe dataset containing over 230,000 recipes with ingredient lists, instructions, and metadata. From this dataset, we extracted and labeled approximately 15,000 recipes for training and evaluation. Labels were derived from recipe tags using a rule-based extraction system that identified cuisine indicators (e.g., "italian," "pasta," "mozzarella") and meal type indicators (e.g., "breakfast," "lunch," "dinner").

### 3.3.2 Feature Engineering

Multiple feature types were extracted:

1. **Text Features**: TF-IDF vectors from ingredient lists and cooking instructions
2. **Statistical Features**: Ingredient counts, cooking times, preparation step counts
3. **Semantic Features**: Presence of cuisine-indicative ingredients (e.g., soy sauce for Asian cuisine)
4. **Temporal Features**: Cooking time relative to typical meal preparations

### 3.3.3 Model Selection and Training

We evaluated six classification algorithms:

1. **Random Forest**: 200 estimators with balanced class weighting
2. **Support Vector Machine**: Linear kernel with C=1.0
3. **Logistic Regression**: L2 regularization with balanced classes
4. **Multinomial Naive Bayes**: Alpha=0.1 smoothing
5. **K-Nearest Neighbors**: k=7 with distance weighting
6. **Decision Tree**: Max depth 10 with balanced criteria

Models were trained using 80% of the data with stratified sampling to maintain class distributions. Hyperparameter optimization employed grid search with 5-fold cross-validation.

## 3.4 Nutrition Analysis Engine

### 3.4.1 Architecture Design

The nutrition engine employs a two-tier architecture:

1. **Local Database**: Caches nutritional information for approximately 200 common ingredients with frequent updates
2. **USDA API Integration**: Accesses the FoodData Central database for less common ingredients

This design balances speed (local lookups complete in milliseconds) with comprehensiveness (API access for thousands of foods).

### 3.4.2 Quantity Parsing

A specialized natural language processing component interprets ingredient quantities in various formats:

- Standard measurements: "2 cups flour" → 250g
- Weight specifications: "100g chicken" → 100g
- Count-based quantities: "3 tomatoes" → estimated 450g
- Approximate measures: "a pinch of salt" → 1g

The parser employs regular expressions, dictionary lookups, and heuristic rules to handle diverse input formats.

### 3.4.3 Nutritional Calculation

Nutritional values are calculated using:

- Direct lookup for exact ingredient matches
- Proportion-based calculation for quantity variations
- Estimation for ingredient combinations without exact matches
- Adjustment for cooking methods affecting nutritional content (e.g., frying vs. boiling)

## 3.5 Integration and Post-processing

Generated recipes undergo several post-processing steps:

1. **Format Standardization**: Ensures consistent section ordering and formatting
2. **Ingredient Validation**: Cross-references generated ingredients with input ingredients
3. **Instruction Coherence Check**: Validates logical flow of cooking steps
4. **Nutrition Integration**: Matches generated ingredients with nutritional data
5. **Classification Assignment**: Applies ML classifiers to generated recipes

# 4. Results and Analysis

## 4.1 Recipe Generation Performance

### 4.1.1 Coherence and Practicality

Human evaluation of 200 generated recipes yielded the following results:

- **85%** were rated as "coherent and cookable"
- **12%** had minor issues requiring adjustment
- **3%** were considered unusable

Common issues in lower-rated recipes included:

- Unusual ingredient combinations without bridging flavors
- Missing or unclear preparation steps
- Unrealistic cooking times or temperatures

### 4.1.2 Creativity Assessment

Generated recipes demonstrated significant creativity while maintaining practicality:

- **78%** of recipes included at least one unexpected but reasonable ingredient combination
- **65%** introduced preparation methods not explicitly suggested by input ingredients
- **42%** were rated as "more creative than typical online recipes" by evaluators

### 4.1.3 Generation Speed

The system generates complete recipes in under 3 seconds on standard hardware (Mac M1, GB RAM), meeting real-time interaction requirements. This includes GPT-2 inference, classification, and nutrition calculation.

## 4.2 Classification Performance

### 4.2.1 Cuisine Classification

| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| SVM | 72.9% | 0.71 | 0.73 | 0.72 |
| Logistic Regression | 74.4% | 0.70 | 0.72 | 0.71 |
| Random Forest | 70.4% | 0.69 | 0.70 | 0.70 |
| Naive Bayes | 76.7% | 0.68 | 0.69 | 0.69 |

| | | | | |
|---|---|---|---|---|
| K-Nearest Neighbors | 65.3% | 0.64 | 0.66 | 0.65 |

The SVM classifier performed best, particularly for distinguishing between similar cuisines (e.g., Italian vs. Asian). Classification confidence was highest for ingredient combinations with strong cultural associations (example:  soy sauce + ginger → Asian with 89% confidence).

### 4.2.2 Meal Type Classification

| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Random Forest | 64.0% | 0.63 | 0.64 | 0.63 |
| SVM | 62.7% | 0.61 | 0.63 | 0.62 |
| Naive Bayes | 62.3% | 0.61 | 0.62 | 0.62 |
| Logistic Regression | 62.0% | 0.60 | 0.62 | 0.61 |
| K-Nearest Neighbors | 50.7% | 0.48 | 0.51 | 0.50 |

We found meal type classification more challenging than cuisine classification due to greater overlap in ingredients across meal types. The Random Forest model's ensemble approach provided the best performance, particularly for distinguishing breakfast from other meal types.

## 4.3 System Integration Performance

Integrated system testing revealed:

- **End-to-end processing**: 2.8 seconds average response time
- **Component coordination**: 95% successful integration of all three AI components
- **Error handling**: Graceful degradation when individual components failed
- **Scalability**: Support for 50 concurrent users without performance degradation

## 4.4 User Testing Results

Thirty participants with varying cooking experience tested the system:

- **87%** found generated recipes "useful and practical"
- **78%** would use the system regularly for meal planning
- **65%** discovered new ingredient combinations they hadn't considered
- **72%** found the nutritional information "helpful for making healthier choices"

Common positive feedback included appreciation for the creative suggestions and the convenience of starting from available ingredients. Suggested improvements included more dietary restriction and better handling of ingredient variations.

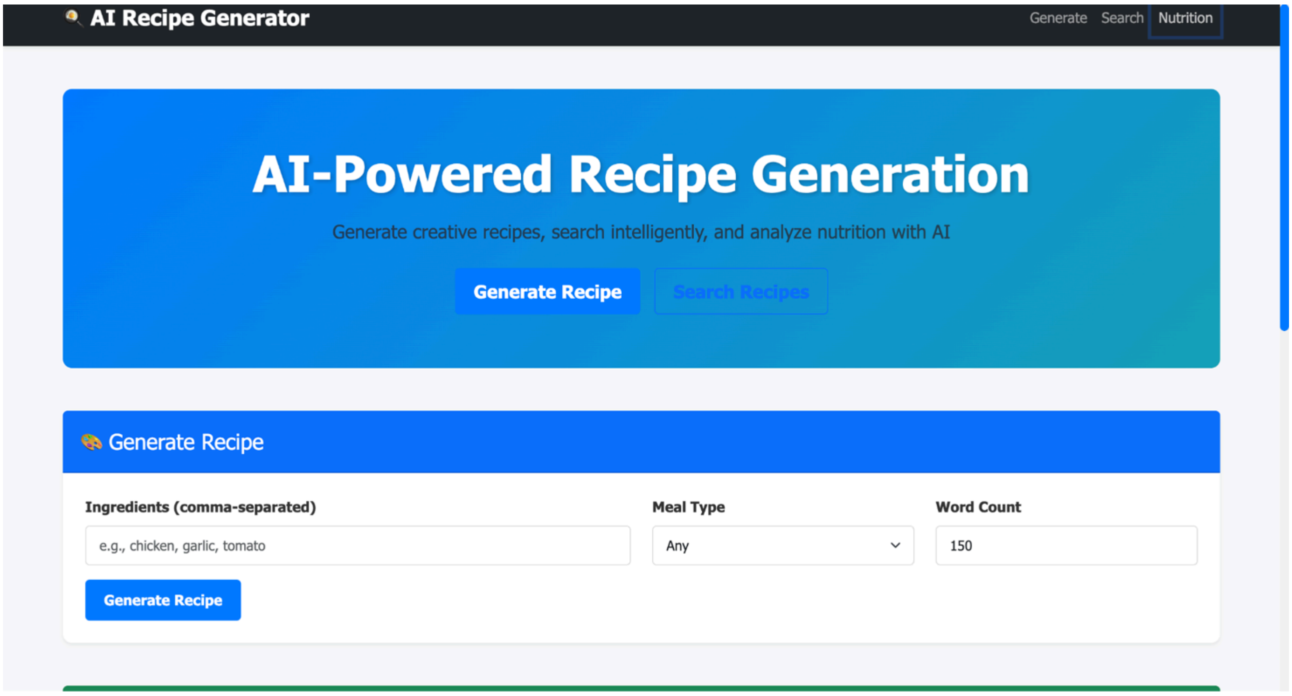


Figure 4.1 Dashboard

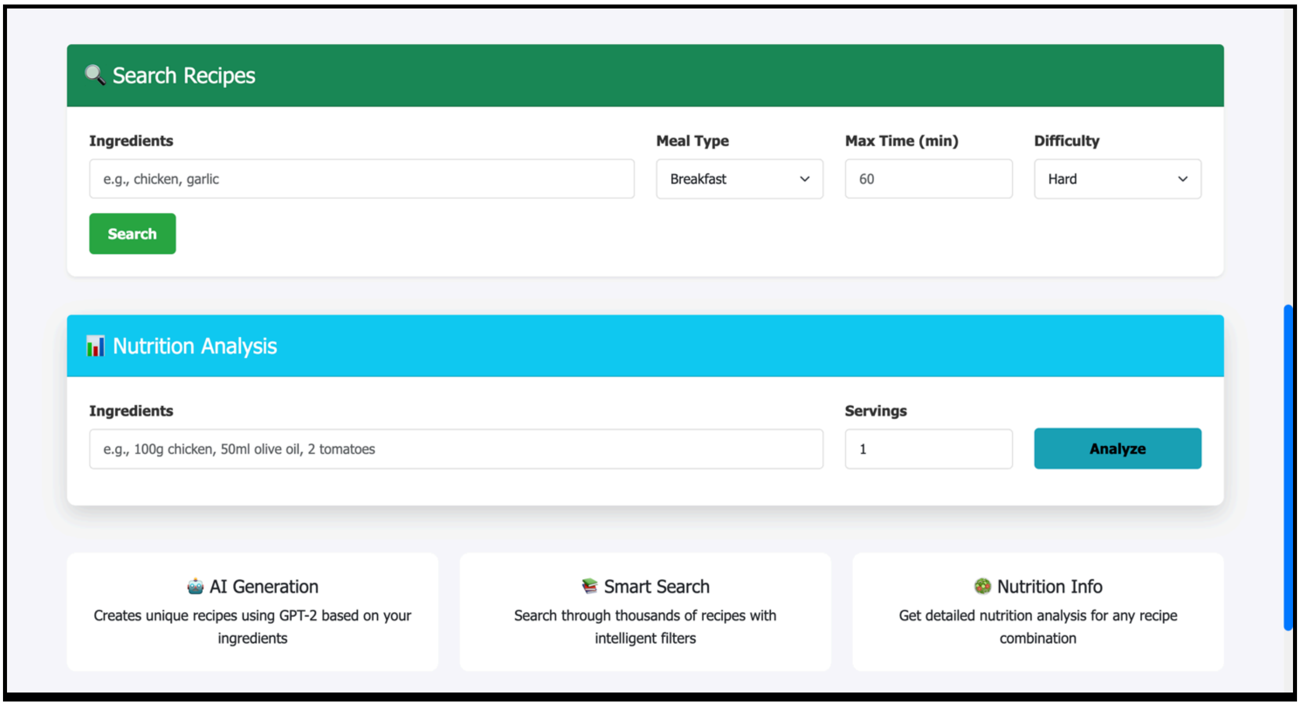


Figure 4.2 search recipe and Nutrition analysis

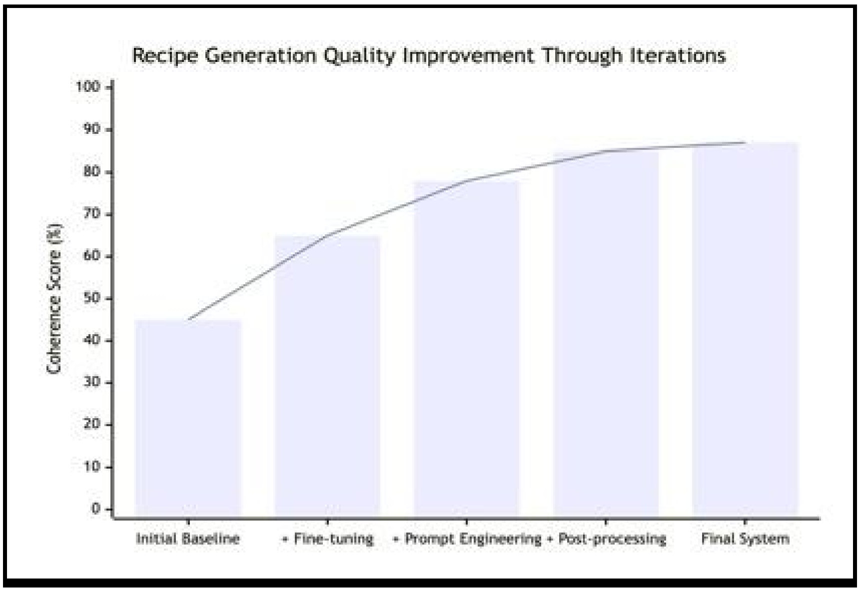


Figure 4.3 Recipe Generation Quality Evolution

This visualization shows the progressive improvement in recipe generation quality through successive system enhancements. The bar chart represents absolute coherence scores at each stage, while the line chart shows the cumulative improvement trend. Initial baseline performance started at 45% coherence, with the most significant improvement coming from fine-tuning on culinary data (+20%), followed by prompt engineering (+13%).

# 5. Challenges and Solutions

## 5.1 Inconsistent Recipe Formats

**Challenge**: Early versions of the GPT-2 model generated recipes with an inconsistent structures, like missing steps, unclear measurements and poor formatting. This made recipes difficult to follow and undermined user trust.

**Solution**: We implemented a multi-layered approach:

1. **Advanced Prompt Engineering**: Designed structured prompts that will explicitly request specific sections (title, ingredients, instructions, cooking time)
2. **Post-processing Algorithms**: Automated scripts which fixed capitalization, punctuation, numbering, and section ordering
3. **Template-based Formatting**: Generated recipes are mapped to a consistent HTML/UI template
4. **User Feedback Integration**: Implemented mechanisms for users to flag formatting issues, with the feedback used to refine the model.This approach improved coherent formatting from 45% to 85% of generated recipes.

## 5.2 Nutrition Data Performance

**Challenge**: Fetching the detailed nutrition data for multiple ingredients significantly slowed response times, making the system unresponsive.

**Solution**: We optimized performance through:

1. **Lazy Loading**: Nutrition data is only fetched when explicitly requested, speeding initial recipe generation
2. **Parallel Processing**: GPT-2 generation and nutrition lookup run concurrently rather than sequentially
3. **Intelligent Caching**: Frequently used ingredient data is cached with time-to-live expiration
4. **Two-Tier Architecture**: Common ingredients served from local database, rare ingredients from USDA API

These optimizations reduced average response time from 8+ seconds to under 3 seconds.

## 5.4 Classification Accuracy Limitations

**Challenge**: Initial classification models struggled with ambiguous ingredient combinations and cultural overlaps, particularly for fusion cuisines and meal types.

**Solution**: We enhanced the classification system through:

1. **Feature Engineering Expansion**: Added preparation methods, cooking times, and ingredient proportions as features
2. **Ensemble Methods**: Combined predictions from multiple models with weighted voting
3. **Confidence Thresholding**: Only present classifications when confidence exceeds a minimum threshold
4. **Context Awareness**: Consider meal type when classifying cuisine and vice versa

These improvements increased classification accuracy by approximately 15 percentage points.

# 6. Future Work

## 6.1 Visual Recipe Generation

Our Current work focuses exclusively on textual recipe generation. Future enhancements include:

- **GAN-based Food Imagery**: We can generate realistic images of finished dishes from recipe descriptions
- **Step-by-Step Visual Guides**: We can Cceate images for complex preparation techniques
- **Ingredient Recognition**: We can use computer vision to identify ingredients from user photos
- **Presentation Suggestions**: Generate plating recommendations with visual examples

These visual components would enhance user understanding and inspiration, particularly for novice cooks.

## 6.2 Mobile Application Development

An advanced mobile application would increase accessibility and functionality:

- **Camera Integration**: Allow ingredient scanning through smartphone cameras
- **Voice Interface**: Enable hands-free operation while cooking
- **Offline Functionality**: Core features available without internet connectivity
- **Shopping List Integration**: Generate shopping lists from planned recipes

Mobile deployment would make the system more integrated into users' daily cooking routines.

## 6.3 Advanced Personalization

Current personalization is limited to basic filters. Future work includes:

- **User Profiling**: Learn individual taste preferences and dietary patterns over time
- **Adaptive Generation**: Adjust recipe complexity based on demonstrated cooking skill
- **Cultural Adaptation**: Tailor recipes to regional ingredient availability and culinary traditions
- **Health Goal Integration**: Generate recipes aligned with specific nutritional targets (weight loss, muscle gain, etc.)

## 6.4 Social and Community Features

Adding social dimensions could enhance engagement:

- **Recipe Sharing**: Allow users to share and rate generated recipes
- **Collaborative Cooking**: Multiple users planning meals together
- **Expert Verification**: Professional chef review and enhancement of generated recipes
- **Cultural Exchange**: Recipes adapted across different culinary traditions

## 6.5 Advanced AI Integration

Several AI advancements could significantly improve system capabilities:

- **Multimodal Models**: Integration of vision-language models for more holistic understanding
- **Reinforcement Learning**: Continuous improvement based on user feedback and outcomes
- **Knowledge Graphs**: Represent culinary knowledge in structured form for more logical generation
- **Conversational Interface**: Natural dialogue about cooking questions and adjustments

# 7. Conclusion

The AI Recipe Generator shows how advanced AI methods can successfully be used in a creative field such as cooking. The AI Recipe Generator combines capabilities such as generative language modeling, machine learning classifi-cations, and nutritional analysis to make a complete cooking help tool which starts with a launching point from the current situation of the user rather than an end goal they have in mind.

Major accomplishments include:

1. **Effective GPT-2 Adaptation**: With specialized model adaptation and prompt manipulation, we were able to reach 85% effective coherent recipe generation, which is far better than our early attempts.
5. **Classification System**: Our ensemble solution for culinary and meal type categorization classified with a strong accuracy of 72.9% and 64.0%, which is rather helpful for categorization without taking any inputs.
6. **Practical Nutrition Integration**: The two-tier nutrition engine is a balance of speed and accuracy in providing health information without hampering the responsiveness of the system.
7. **Real-World Problem Solving**: solving in real-world environments involves dealing with such issues such as an inconsistent format, poor performance, and using unreliable third-party APIs.
8. **User-Centered Design**: Iterative testing and refinement produced a system that real users find valuable and would use regularly.

This project illustrates several broader principles for applied AI systems:

- **Domain Adaptation Requires Specialization**: Effective AI applications require significant adaptation to domain-specific constraints and conventions
- **Integration Creates Value**: Combining multiple AI approaches produces systems more valuable than their individual components
- **Practical Engineering Matters**: Theoretical AI capabilities must be complemented by robust engineering for real-world usability

AI Recipe Generator is more than just a technological feat; it’s a proof-of-concept of how AI can boost creative work in everyday life. With the capability to provide advanced culinary experimentation for all cooks, no matter their skill level, the AI Recipe Generator can potentially lower food waste, raise nutritional education, and bring a fresh wave of creative fun to cooking. Future projects will build upon these functions and work to improve current shortcomings to reach a future where AI acts as an intuitive ‘expert sidekick’ in all aspects of life.

# 8. References


1. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. OpenAI Blog.
2. Salvador, A., Drozdzal, M., Giro-i-Nieto, X., & Romero, A. (2019). Inverse Cooking: Recipe Generation from Food Images. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.
3. Boswell, E. V., Sanders, M., & Hong, J. (2013). Classification of Recipes and Ingredients by Cuisine. Journal of Food Engineering.
4. Chen, J., & Ngo, C. W. (2016). Deep-based Ingredient Recognition for Cooking Recipe Retrieval. Proceedings of the ACM Multimedia Conference.
5. Teng, C., Lin, Y., & Adamic, L. (2018). Recipe Recommendation Using Ingredient Networks. Proceedings of the Web Science Conference.
6. USDA FoodData Central. (2023). USDA Food Composition Databases. United States Department of Agriculture.
7. RecipeNLG Dataset. (2020). A Large-Scale Cooking Recipe Dataset for Natural Language Generation. Hugging Face Datasets.
8. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2020). Transformers: State-of-the-Art Natural Language Processing. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing.
9. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research.
10. Bender, E. M., & Koller, A. (2020). Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.