LONDON METROPOLITAN UNIVERSITY

Islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

Level 7 – Neural Networks and Deep Learning

Assessment Type

60% Individual Coursework

Semester

2025 Autumn

Credit: 20 Semester Long Module

Student Name: Mr. Suman Prasad Neupane

London Met ID: 24048785

College ID: NP01MS7A240052

Assignment Due Date: Thursday, January 29, 2026

Assignment Submission Date: Wednesday, January 28, 2026

Submitted To: Mr. Anish Chapagain

Word Count (Where Required): 3205

I confirm that I understand my coursework needs to be submitted online via MySecondTeacher classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Abstract

This research presents the design, implementation, and evaluation of a cross-modal recipe retrieval system that aligns food images and recipe text within a shared semantic embedding space. The proposed system supports bidirectional retrieval, enabling both image-to-recipe and recipe-to-image queries using transfer learning and contrastive learning. The core contribution is a multi-encoder architecture that explicitly models semantic differences between visual data, short textual descriptions, and long-form procedural text.

A pretrained ResNet50 model is used as the visual encoder to extract high-level semantic features from food images. For textual representation, two separate DistilBERT-based encoders are employed: one for short recipe titles and another for long-form ingredient lists and cooking instructions. This separation preserves fine-grained procedural information without overwhelming the shared embedding space. Outputs from each encoder are projected into a 1024-dimensional unified embedding space using modality-specific alignment modules composed of linear layers, GELU activation, layer normalization, and dropout. Training is performed using the InfoNCE contrastive loss, optimizing relative similarity between matched image-text pairs while pushing apart mismatched pairs within each batch. A multi-pair contrastive strategy computes six contrastive losses across all modality combinations and averages them to ensure balanced alignment. Transfer learning reduces computational cost and enables efficient training under limited hardware constraints.

Experiments are conducted on a publicly available Kaggle food image-recipe dataset containing approximately **13,500** paired samples. Performance is evaluated using Recall@K metrics (R@1, R@5, and R@10) for all retrieval directions. Results show strong retrieval performance, with Recall@10 exceeding **0.99** for both image-to-text and text-to-image tasks. Compared to the Im2Recipe baseline (R@10 ≈ **0.65**), the proposed approach demonstrates substantial improvement. Overall, the research confirms that contrastive learning and specialized multimodal encoders can produce effective cross-modal retrieval systems even with limited resources.

Table of Contents

Table of Figures

1. Introduction

1.1 Background

The new challenges for information retrieval systems, especially for applications that have to interpret visual as well as text information simultaneously. For instance, users often come across food image collections without any information regarding the associated recipes or preparation procedures. The existing unimodal systems, like image classification or text search, cannot handle the semantic relationships between the image and text information.

Cross-modal retrieval can overcome the above-mentioned drawback by learning a common representation space where data of various modalities can be compared end-to-end. For food-related research, the work of Salvador et al. showed that joint embeddings can successfully match images of food to recipes and lists of ingredients using the Im2Recipe framework (Salvador, et al., 2017; Salvador, et al., 2018). However, the current state-of-the-art in transfer learning means that convolutional neural networks and transformer models can provide high-quality multimodal learning even on a small amount of data. This research will leverage these developments to create a real-world cross-modal recipe retrieval system using a public dataset.

Cross-modal food retrieval has high applicability. It can be used for intuitive image-based recipe search, intelligent recommendation systems, as well as for building upon dietary analysis systems and kitchen assistants. As a research issue, it is related to vision-language alignment research since it concentrates on learning based upon retrieval instead of fixed-classification learning (Salvador, et al., 2018).

1.2 Problem Definition

This project addresses the problem of the absence of a quick and scalable way to search for recipes that are semantically relevant to the food image and vice versa. The current methods that are either single-modal or loosely coupled in the modalities do not capture the subtle cross-modal links and hence have poor retrieval accuracy.

1.3 Aim and Objectives

Aim:

The primary objective of this project is to build and develop a cross-modal recipe retrieval system that matches food images and recipe texts into a common embedding space through transfer learning and contrastive learning. More precisely, this system will:

- Encode food images into visual embeddings.
- Encode ingredient lists and recipe text into text embeddings.
- Both encoders learn a common space where similar pairs of image and text are closer to each other than dissimilar pairs.
- Handling queries in both directions: image → recipe and recipe → image.

Objectives:

- Investigate and pre-process a multimodal food data set that consists of images and corresponding recipes acquired from Kaggle (Kaggle, 2025).
- To apply the dual encoder architecture with ResNet50 as the image encoder, which is pre-trained, and DistillBert as the text encoder.
- To use the InfoNCE contrastive loss function to train the model to match the embedding of images and texts.
- For assessing the retrieval performance by means of Recall@K metrics (R@1, R@5, R@10) for image-to-text and text-to-image.

Ethical Considerations:

The dataset used in this project can be accessed publicly and does not consist of any private or sensitive data. The usage of all data in this project for academic purposes has been ensured by maintaining the proper citation of datasets and previous research.

1.4 Scope of the Project

- **Included Scope:**

  The project focuses on a dataset of around 13,500 food samples, which come with paired images and text recipes (Kaggle, 2025). Transfer learning and pretrained

models and contrastive learning for bidirectional cross-modal retrieval, and Recall@K for the evaluation of the project.

- **Excluded Scope:**
  It should be noted that the project does not include nutritional analysis, real-time deployment, development of a mobile application, and training models from scratch. The proprietary datasets, like the original Im2Recipe dataset, are not included because of limited access (Salvador, et al., 2017).

2. Literature Review

2.1 Overview: Traditional vs Deep Learning-Based Approaches

The traditional based approaches food recognition and recipe search relied on conventional computer vision and NLP methods (Datta, et al., 2008; Lowe, et al., 2004; Dalal & Triggs, et al., 2005). The images were usually represented using manual features like SIFT/HOG features together with simple classifiers, while the text data was represented using bag-of-words/TF-IDF models (Salton & Buckley, et al. 1988; Joachims, et al., 1998). These individual models tackled images as well as text separately; hence, there was no easy way to link the visual representation of a food image to the text representation of a recipe. This led to low accuracy in the search results, especially in the food recognition category, where images of similar dishes could have vastly different ingredients and cooking procedures (Bossard, et al., 2014).

Research in cross-modal retrieval began to move in the direction of learning a joint representation for images and text using deep learning (Karpathy and Fei-Fei, et al., 2015). Deep learning models have the capacity to learn features from the data in a hierarchical manner that can then be trained for the entire retrieval process (LeCun, et al., 2015). In the food sector, this resulted in a major boost as the models could learn the mapping in a semantic manner as opposed to using manual features (Salvador, et al., 2017).

2.2 Datasets and Evaluation Metrics

2.2.1 Datasets

Im2Recipe provided the initial dataset that contained matched images and recipes used in cross-modal food retrieval tasks (Salvador, et al., 2017). However, the dataset cannot be accessed anymore. Despite several requests, the researchers did not provide the dataset, and thus the use of the dataset in Kaggle was the best accessible and viable option in the research. As an alternative, datasets such as the Food Ingredients and Recipe Dataset with Images in the public platform in Kaggle provide matched images, ingredient descriptions, and recipe instructions, thus making them useful in the research (Kaggle, 2025).

2.2.2 Evaluation Metrics

Evaluating retrieval systems is different from traditional classification problems. Although accuracy, precision, and recall can be used as evaluation metrics for classification problems, cross-modal retrieval problems can be better evaluated using ranking-oriented evaluation metrics such as Recall@K. Recall@K is a metric that estimates how well the correct answer is ranked within the top K results of a retrieval system and is often used as a metric in image and text retrieval problems because it is more closely tied to real-world application scenarios (Salvador et al., 2018).

- **R@1 (Recall at 1):** This metric calculates how often the correct match is returned in the top answer, and higher results indicate better precision in pinpointing the exact match.
- **R@5 (Recall at 5):** It is used to calculates, the number of correct pair has been seen in the first 5 results. It is used in practical near-match performance.
- **R@10 (Recall at 10):** This is the measure of how often the actual matching image occurs in the first 10 results. This is used for recommendation systems.

2.3 Research Gap and Motivation

Although the effectiveness of joint embedding models for food image search and recipe search has been shown, several issues remain. Most of the previous methods were based on CNN-RNN models, which were computationally intensive and not suited for modeling long-range dependencies between text. Moreover, some models have considered all text inputs equally, without any concern for the semantic difference between short title text and long procedural text.

The proposed work fills this gap through the use of a transfer learning-based multi-encoder cross-modal approach where different types of text data are modeled separately from image data.

2.3.1 Real-World Problem and Applicability

- **Image-based Recipe Searches:** Recipes can be searched using food images stored in smartphones or in social media sites.
- **Automated Diet Analysis:** Healthcare applications could use images of food to make estimates of the ingredients and nutritional information.
- **Intelligent Kitchen Assistants:** These assistants would have suggested a recipe based on a picture of food or a list of ingredients.

2.3.2 Why These Models Were Chosen

The pretrained model will used for visual encoder as CNN (e.g., ResNet50), and the textual encoder as transformer-based model (e.g., BERT). These choices are made because ResNet50 capture rich visual features that generalize well across domains, while BERT excel at capturing semantic meaning from complex textual descriptions (Salvador et al., 2018). This will allow the system to benefit from previously learned visual and linguistic representations while reducing training time and computational cost.

- **Choice of Text Encoder:**

A lightweight transformer-based sentence encoder is employed instead of traditional language models such as DistilBERT with parameters ~66M. Although, BERT's large parameter count ~110M introduces higher memory usage and slower training. This decision is driven by the nature of the retrieval task, which prioritizes semantic similarity at the sentence or document level rather than token-level contextual understanding.

- **Choice of Image Encoder:**

A pretrained ResNet50 with parameters ~25.6M convolutional neural network is used as the visual encoder. ResNet50 was selected due to it has strong balance between representational capacity, computational efficiency, and transfer learning. The residual learning mechanism enable for deeper feature extraction while solving the vanishing gradient problem, making it well-suited for learning complex visual patterns in food images such as texture, color, and structural composition. Alternative image encoders, such as Vision Transformers (ViT) or very deep CNNs (e.g., ResNet101 or EfficientNet-B7), were considered but not adopted. This CNNs increase memory consumption and training time without guaranteeing proportional performance gains in this context.

3. Methodology

3.1 Overall Approach

3.1.1 Supervised learning pipeline

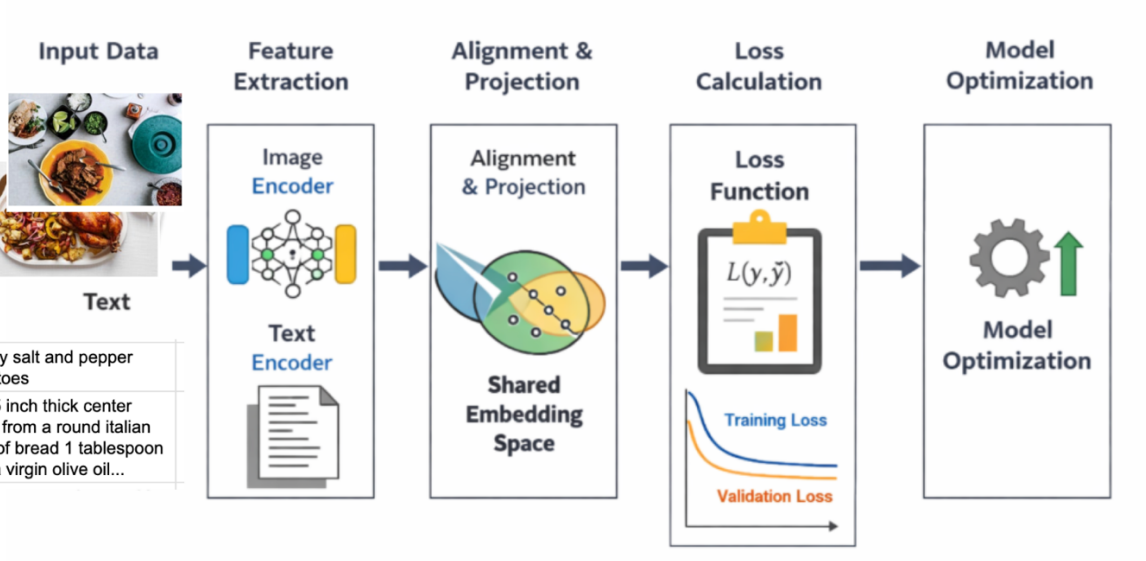

Figure 1: Supervised learning pipeline

3.1.2 Training and testing workflow

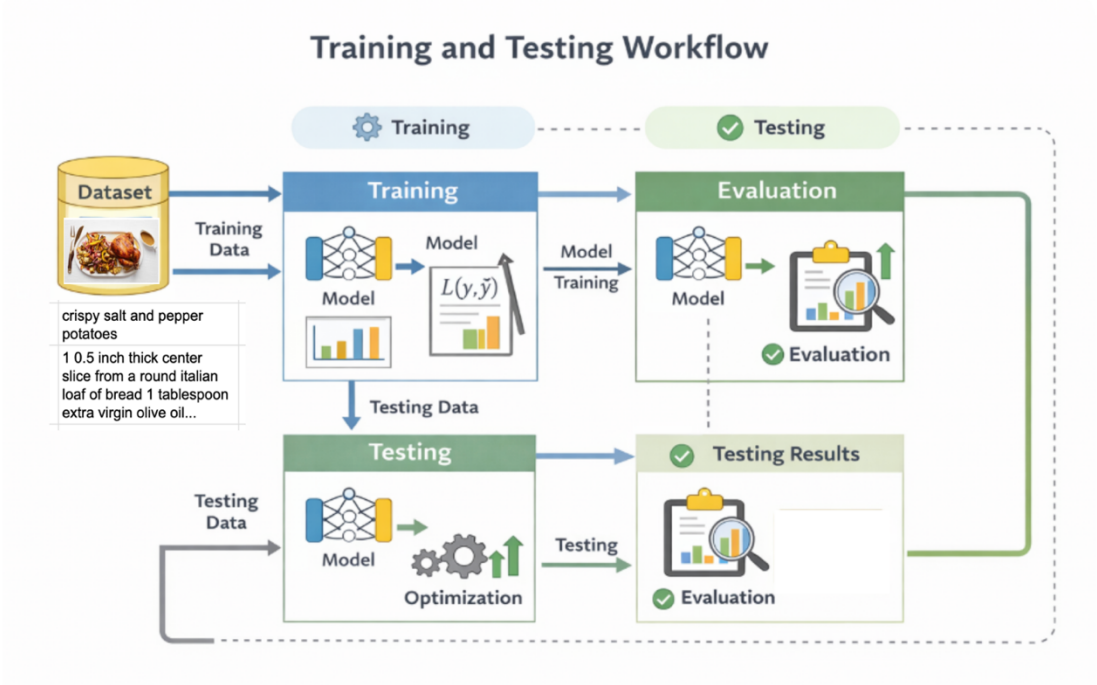

Figure 2: Training and testing workflow

3.2 Dataset Description

3.2.1 Source

https://www.kaggle.com/datasets/pes12017000148/food-ingredients-and-recipe-dataset-with-images

3.2.2 Number of classes and images

Dataset		
Textual data Information	Number of rows	13501
	Number of columns	6
	Number of images	13582
Image data Information	Image Width	274
	Image Height	169
	Image Channel	RGB

Figure 3: Number of classes and images

3.2.3 EDA of the Data

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 13501 entries, 0 to 13500
Data columns (total 6 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   Unnamed: 0          13501 non-null  int64 
 1   Title               13496 non-null  object
 2   Ingredients         13501 non-null  object
 3   Instructions        13493 non-null  object
 4   Image_Name          13501 non-null  object
 5   Cleaned_Ingredients 13501 non-null  object
dtypes: int64(1), object(5)
memory usage: 633.0+ KB
```

Figure 4: Number of Columns and Data Types

```
data.head()
```

	Unnamed: 0	Title	Ingredients	Instructions	Image_Name	Cleaned_Ingredients
0	0	Miso-Butter Roast Chicken With Acorn Squash Pa...	['1 (3½–4-lb.) whole chicken', '2¾ tsp. kosher...	Pat chicken dry with paper towels, season all ...	miso-butter-roast-chicken-acorn-squash-panzanella	['1 (3½–4-lb.) whole chicken', '2¾ tsp. kosher...
1	1	Crispy Salt and Pepper Potatoes	['2 large egg whites', '1 pound new potatoes (...	Preheat oven to 400°F and line a rimmed baking...	crispy-salt-and-pepper-potatoes-dan-kluger	['2 large egg whites', '1 pound new potatoes (...
2	2	Thanksgiving Mac and Cheese	['1 cup evaporated milk', '1 cup whole milk', ...	Place a rack in middle of oven; preheat to 400...	thanksgiving-mac-and-cheese-erick-williams	['1 cup evaporated milk', '1 cup whole milk', ...
3	3	Italian Sausage and Bread Stuffing	['1 (¾- to 1-pound) round Italian loaf, cut in...	Preheat oven to 350°F with rack in middle. Gen...	italian-sausage-and-bread-stuffing-240559	['1 (¾- to 1-pound) round Italian loaf, cut in...
4	4	Newton's Law	['1 teaspoon dark brown sugar', '1 teaspoon ho...	Stir together brown sugar and hot water in a c...	newtons-law-apple-bourbon-cocktail	['1 teaspoon dark brown sugar', '1 teaspoon ho...

Figure 5: Overview of First 5 Rows of the Dataset

Figure 6: Sample Recipes Food Images and Corresponding Recipe

3.2.4 Data Preprocessing and Augmentation

Category	Technique	Purpose
Image Preprocessing	Resize to fixed size	Ensures uniform input dimensions for the model
	Convert to Tensor	Converts image to PyTorch tensor format
	Normalize (ImageNet mean & std)	Stabilizes training and accelerates convergence
Image Augmentation (Train Only)	Random Horizontal Flip (p=0.5)	Improves invariance to left-right orientation
	Random Rotation (±10°)	Enhances robustness to camera angle variations
	Color Jitter (brightness, contrast, saturation, hue)	Reduces sensitivity to lighting and color changes
Dataset Splitting	Train-Validation split (40% / 60%)	Enables model evaluation on unseen data
Text Preprocessing	Lowercasing	Ensures textual consistency
	Special character removal	Removes noise and encoding artifacts
	Fraction conversion (e.g., 1/2 → 0.5)	Standardizes numeric quantities
	Whitespace normalization	Improves tokenization quality
	Missing text handling ([NO_TEXT])	Prevents empty input issues
Missing Image Handling	Zero-filled placeholder image	Maintains dataset consistency

Figure 7: Data Preprocessing and Augmentation

3.3 Model Architecture

3.3.1 Overview of the Architecture

The proposed model follows a multi-encoder contrastive alignment architecture with parameters 163,061,824 (~163M). The architecture consists of three parallel encoders: a text encoder for short recipe titles, a separate text encoder for long-form recipe descriptions (ingredients and instructions), and an image encoder for food images. Each encoder produces modality-specific features that are subsequently aligned into a unified embedding space through a shared alignment module. This design explicitly acknowledges the semantic asymmetry between short textual descriptions, long procedural text, and visual content, rather than forcing all textual inputs into a single encoder.

Text and Long Text Encoder			
Part	**What it does**	**Why it matters**	**Output**
DistilBERT (6 layers)	Reads text and creates contextual meaning	Captures meaning of words and their context	768-dim
Projection (Linear 768 → 1024)	Converts features to fusion space	Makes embedding compatible with other modalities	1024-dim
LayerNorm + Dropout	Stabilizes training & prevents overfitting	Helps model generalize better	1024-dim
Image Encoder			
Part	**What it does**	**Why it matters**	**Output**
ResNet-50 Backbone	Extracts visual features from image	Detects objects, shapes, textures	2048-dim
AdaptiveAvgPool	Converts spatial features to a single vector	Makes image embedding fixed-size	2048-dim
Projection (Linear 2048 → 1024)	Converts to fusion space	Aligns image features with text embeddings	1024-dim
LayerNorm + Dropout	Stabilizes training & prevents overfitting	Improves generalization	1024-dim
Fusion Model			
Input	**Encoder**	**Purpose**	**Output**
Text	Text Encoder	Understands short text	1024-dim
Long Text	Long Text Encoder	Understands long text	1024-dim
Image	Image Encoder	Understands visual content	1024-dim
All combined	Fusion	Compare & match modalities	Common embedding

Figure 8: Overview of the Architecture

3.3.2 Text Encoders for Short Text and Long Text Representation

Both the text encoder and the long text encoder use the DistilBERT model, resulting in contextual token embeddings of 768 dimensions. The choice of the DistilBERT model is motivated by the fact that it has fewer parameters and is thus effective in representing robust contextual information. The use of two distinct text encoders allows the model to specialize in the following ways:

- The short text encoder focuses on high-level semantic intent such as the title of the recipe.
- The long text encoder includes information regarding the procedures, as well as the composition.

Such a division helps prevent long forms of text from overwhelming the entire representation space while providing better stability in the context of contrastive learning. Each of the text encoders' outputs is then fed into a Text Alignment Module consisting of:

- A linear projection from 768 to 1024 dimensions
- GELU Activation Function
- A second linear transformation keeping 1024 dimensions
- Normalizzazione di strato
- Dropout (p = 0.1)

3.3.3 Image Encoder for Image Representation

The image encoder utilizes a ResNet50 backbone to achieve a 2048-dimensional global visual feature vector through an adaptive average pooling layer. The last fully connected classification layer has been discarded to retain only semantic visual representations. The image feature vectors are then fed into the Visual Alignment Module, which has a similar structure to the text alignment modules:

- A linear projection onto 2048 dimensions followed by a linear projection onto 1024
- GELU Activation Function
- A linear layer mapping from 1024 to 1024
- Layer normalization with dropout

3.3.4 Shared Embedding Space Design

Rather than comparing these raw encoder outputs directly, it leverages a space called Contrastive Alignment Space that normalizes and semantically aligns all modality embeddings. This bypasses the necessity of modality-specific feature extraction and interaction. This shared dimensionality is useful for retrieval efficiency and to ensure that there is adequate capacity to distinguish between semantically different yet visually similar recipes.

3.3.5 Contrastive Learning with InfoNCE Loss

The model uses the InfoNCE loss which is a contrastive loss that helps to maximizes the similarities among the matched image/text pairs while minimizing the similarities with all the other pairs in the batch. This loss is particularly suited for retrieval problems because the relative ranking is directly optimized instead of the absolute accuracy. The model learns a robust embedding structure that enables bidirectional retrieval, while the formulation of similarity with temperature scaling helps improve the stability of gradients as well as prevent embedding collapse.

3.4 Tools and Technologies

Tools and Technologies	
Frameworks	pytorch, PIL, matplotlib
Hardware/ software environment	Macbook Pro M3 18 Core RAM 14 Core GPU

Figure 9: Tools and Technologies

4. Implementation

4.1 Data Preprocessing

Data Preprocessing is already shown in above Figure 7.

4.2 Model Training

Category	Details
Loss Function	InfoNCE
Optimizer	Adam
Learning Rate	0.0001
Temperature	0.5
Batch Size	16
Epochs	45
Activation	GeLU
Regularization Techni	Dropout and LayerNorm
Total Parameters	163,061,824
Trainable Parameters	163,061,824
Time per Epoch	25 minutes
Total Training Time	1125 minutes (≈ 18.75 hours)

Figure 10: Model Training

4.3 Challenges Encountered

- **Overfitting:** Model performance improved on training data faster than validation, requiring regularisation.
- **Computational constraints:** Limited hardware increased training time and restricted model tuning.
- **Dataset imbalance:** Uneven sample distribution biased retrieval toward frequent classes.

5. Results and Evaluation

5.1 Experimental Setup

5.1.1 Contrastive Pairs Used Per Batch

Loss Term	Alignment Enforced	Purpose
loss_t2i	Text → Image	Align Title with images
loss_i2t	Image → Text	Align Images with Title
loss_lt2i	Long Text → Image	Align Ingredients+Instructions with Images
loss_i2lt	Image → Long Text	Align Images with Ingredients+Instructions
loss_t2lt	Text → Long Text	Align Title with Ingredients+Instructions
loss_lt2t	Long Text → Text	Align Ingredients+Instructions with Title

Figure 11: Contrastive Pairs Used Per Batch

To balanced gradient contribution, all six losses are averaged for every modality pair:

$$\mathcal{L}_{\text{total}} = \frac{1}{6}\left(\mathcal{L}_{t\to i} + \mathcal{L}_{i\to t} + \mathcal{L}_{lt\to i} + \mathcal{L}_{i\to lt} + \mathcal{L}_{t\to lt} + \mathcal{L}_{lt\to t}\right)$$

The model is trained using a multi-pair contrastive learning strategy**,** where six InfoNCE losses are computed per batch and averaged. These losses enforce alignment across text, long text, and image embeddings in both directions.

5.1.2 InfoNCE Loss for a Single Pair

For a given embedding pair (x,y)(e.g., text and image), InfoNCE loss is defined as:

$$\mathcal{L}_{x\to y} = -\log \frac{\exp(\text{sim}(x,y)/\tau)}{\sum_{y'\in B}\exp(\text{sim}(x,y')/\tau)}$$

This is applied to all six pairwise directions in your implementation.

5.1.3 Evaluation Metrics using Recall@K

To evaluate the performance of cross-modal retrieval (text-to-image, image-to-text, etc.), the model is measured using **Recall@K**metrics. The mathematical definition of **Recall@K** is:

$$R@K = \frac{1}{N}\sum_{i=1}^{N}\mathbf{1}(\text{rank}_i \leq K)$$

This is applied to all six pairwise directions in your implementation.

5.2 Performance Results

The results show a consistent improvement in retrieval accuracy for all modalities during the course of the training process. As expected, the performance, measured by Recall@K, was consistently good across all modalities. In particular, excellent performance was observed for the Text → LongText and Image → Text retrieval tasks. In the training progresses, Recall@10 was observed to have exceeded 0.99 for the Text → LongText task and comparable values for the Image → LongText task, reflecting excellent semantic alignment between the modalities.

5.2.1 Training and Validation Loss

Epoch	Training Loss	Validation Loss
1	2.2189	1.5984
10	0.2056	0.1968
20	0.064	0.0908
30	0.0303	0.0601
40	0.0188	0.0417
41	0.0191	0.0467
42	0.0183	0.0417
43	0.0187	0.0397
44	0.0166	0.0358
45	0.0145	0.0345

Figure 12: Training and Validation Metrices Summary

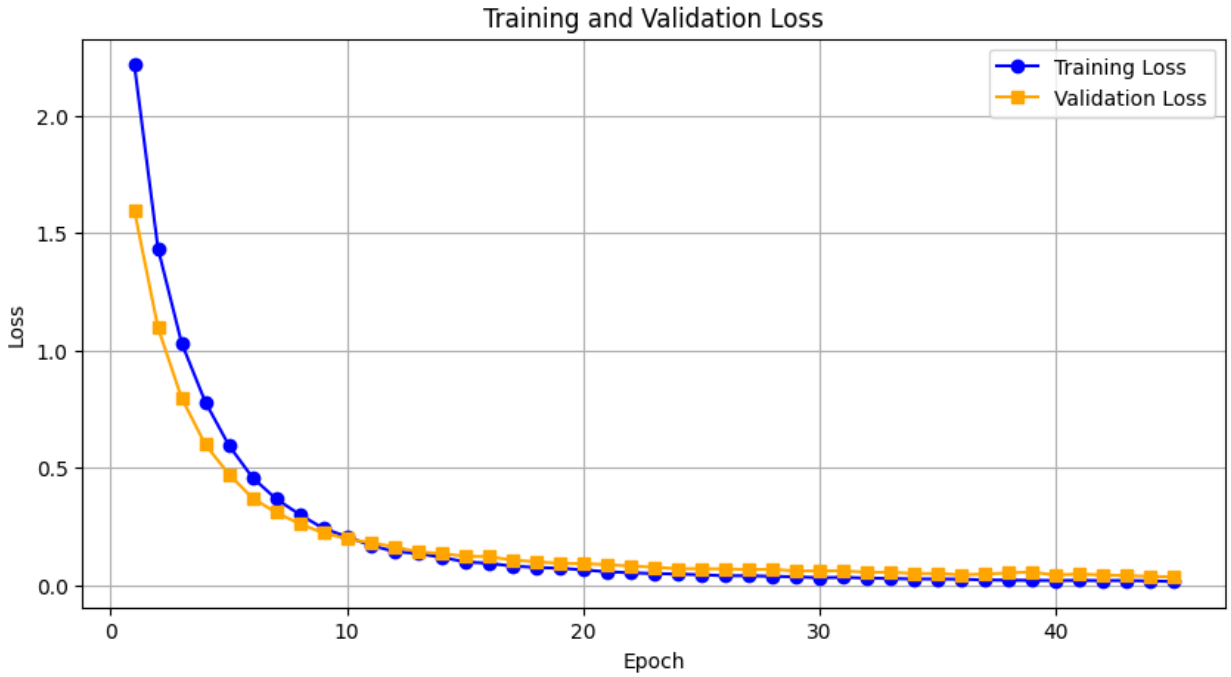

Figure 13: Plot of Training and Validation Metrices

- **Initial Epochs (1–5):** The Training and validation loss are decreasing sharply. The model is learning the basic visual and textual features. The projection head is begin to mapping embeddings into the shared space.
- **Middle Epochs (6–15):** The losses is continue to decrease but at a slower rate, indicating the model is refining alignment to more challenging image–text pairs and long-text representations. The shared embedding space have becomes more semantically consistent within all modalities.
- **Later Epochs (16–20):** The loss with smaller incremental has improvement which shows the training loss has reaches to ~0.064 and the validation loss has reaches to ~0.0908. Which indicate the model has largely converged and generalizes even well in unseen validation data without overfitting.
- **Additional Training Epochs (20–45):** The loss continues to decrease gradually with minor fluctuations. By reaching epoch 45, training loss has drops to ~0.0145 and validation loss has to ~0.0345 which showing steady improvement and stable convergence. The small gap between training and validation loss has suggests good generalisation and minimal overfitting even with extended training.

5.2.2 Recall@K Summary with Selected Epochs

Epoch	Title→Image (R@1 / R@5 / R@10)	Image→Title (R@1 / R@5 / R@10)	Title→ IngredientsInstructions (R@1 / R@5 / R@10)	Image→ IngredientsInstructions (R@1 / R@5 / R@10)
1	0.005 / 0.0194 / 0.0354	0.0054 / 0.0211 / 0.0391	0.0522 / 0.1622 / 0.2430	0.0059 / 0.0235 / 0.0494
10	0.2567 / 0.5719 / 0.7107	0.3402 / 0.6548 / 0.7919	0.4578 / 0.7854 / 0.8848	0.3530 / 0.6869 / 0.8033
20	0.5806 / 0.8689 / 0.9298	0.6785 / 0.9191 / 0.9669	0.6637 / 0.9294 / 0.9750	0.6804 / 0.9224 / 0.9656
30	0.7274 / 0.9424 / 0.9756	0.8143 / 0.9737 / 0.9887	0.7796 / 0.9754 / 0.9957	0.7957 / 0.9683 / 0.9861
40	0.8206 / 0.9704 / 0.9869	0.8893 / 0.9880 / 0.9937	0.8635 / 0.9946 / 0.9987	0.8961 / 0.9880 / 0.9939
41	0.8094 / 0.9665 / 0.9824	0.8837 / 0.9850 / 0.9915	0.8524 / 0.9920 / 0.9991	0.8815 / 0.9854 / 0.9913
42	0.8128 / 0.9726 / 0.9856	0.8854 / 0.9856 / 0.9926	0.8691 / 0.9933 / 0.9989	0.8996 / 0.9859 / 0.9919
43	0.8331 / 0.9761 / 0.9878	0.8931 / 0.9880 / 0.9930	0.8741 / 0.9933 / 0.9993	0.8970 / 0.9869 / 0.9933
44	0.8269 / 0.9796 / 0.9911	0.8950 / 0.9922 / 0.9957	0.8656 / 0.9954 / 0.9989	0.9067 / 0.9902 / 0.9954
45	0.8306 / 0.9787 / 0.9906	0.8896 / 0.9911 / 0.9957	0.8883 / 0.9959 / 0.9993	0.9065 / 0.9924 / 0.9950

Figure 14: Recall@K Metrices Summary

- **Initial Epochs (1–10):** Recall@K metrics, improve steeply as the model is learning the initial cross-modal associations. retrieval show stronger early performance due to richer textual context and clearer semantic alignment.
- **Middle Epochs (10–20):** Recall@K shows a sharp rise in this period, with both Text→Image and Image→LongText tasks continuing to improve. The model has becomes increasingly accurate in mapping image features to short text.
- **Later Epochs (20–30):** For all Recall@K high value are being achieved. At epoch 30, most of the retrieval tasks has a recall greater than 0.94 on R@5 and greater than 0.97 on R@10, indicating strong semantic matching and robust embedding alignment.
- **Additional Recall Epochs (30–45):** The Recall@K metrics continue their trend of high accuracy, reaching a peak, although with some minor fluctuations. Which shows the we can observed that the maximum accuracy is achieved during the Text→LongText and Image→LongText tasks, with the value of R@10 reaching 0.999 at epoch 45.

5.3 Analysis of Results

5.3.1 Outcome title→ingredients_instructions

Input	Output
crispy salt and pepper potatoes	1. Score: 0.981 Title: crispy salt and pepper potatoes Ingredients+Instructions: 2 large egg whites 1 pound new potatoes about 1 inch in diameter 2 teaspoons kosher salt ¼ teaspoon ... 2. Score: 0.857 Title: crispy salt and vinegar potatoes Ingredients+Instructions: 2 pounds baby yukon gold potatoes halved quartered if large 1 cup plus 2 tablespoons distilled white... 3. Score: 0.787 Title: salt roasted potatoes Ingredients+Instructions: 2 cups kosher salt 1 0.5 pounds fingerling potatoes 2 sprigs rosemary 3 garlic cloves thinly sliced ...

Figure 15: title→ ingredients_instructions

5.3.2 Outcome image→image

Input	Output
	1. Score: 1.000 Title: crispy salt and pepper potatoes  2. Score: 0.568 Title: sour cream and scallion drop biscuits  3. Score: 0.522 Title: twice baked potatoes 

Figure 16: image→image

5.3.3 Outcome text→image

Input	Output
crispy salt and pepper potatoes	1. Score: 0.600 Title: crispy salt and vinegar potatoes  2. Score: 0.569 Title: crispiest potato chips  3. Score: 0.561 Title: creamy potato salad with lemon and fresh herbs 

Figure 17: title→image

5.3.4 Outcome image→title

Input	Output
	1. Score: 0.686 Title: miso butter roast chicken with acorn squash panzanella  2. Score: 0.585 Title: cast iron roast chicken with winter squash red onions and pancetta  3. Score: 0.582 Title: pressed chicken with yellow squash and tomatoes 

Figure 18: image→title

5.3.5 Outcome image→ingredients_instructions

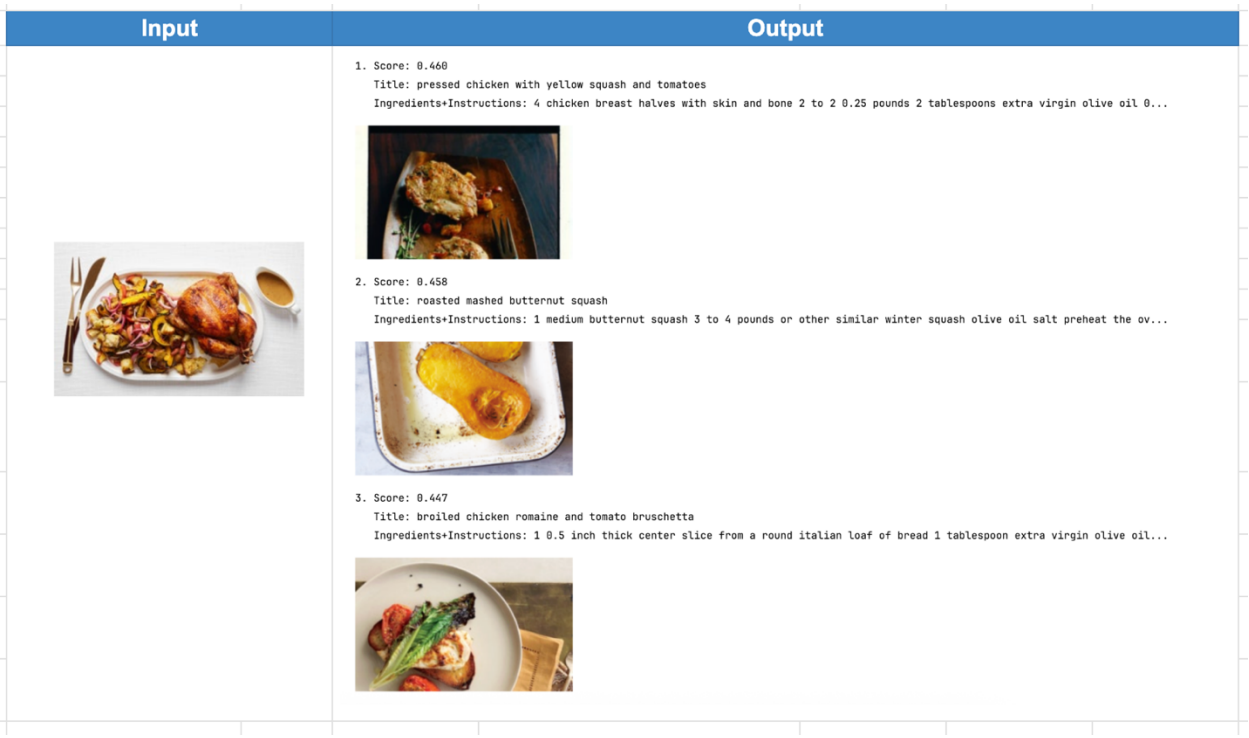

Figure 19: image→ingredients_instructions

The strong Recall@K scores indicate that the learned shared embedding space effectively captures semantic relationships between images and textual descriptions. The superior performance in long-text retrieval tasks suggests that richer textual context improves cross-modal alignment. The smooth loss convergence reflects stable optimisation and effective use of transfer learning.

At first, performance for Text→Image tasks were worse due to the difficulty mapping visual features to text embeddings than text to text. Possible causes of these difficulties included visual ambiguity and imbalance in training datasets. Even with these early difficulties, considerable improvement in retrieval performance for Text→Image tasks occurred after further training (showing the potential of this approach).

5.4 Comparison with Existing Work

Topic / Task	Published Baselines (im2recipe / Recipe1M)	Current Paper Metrics (Epoch 45)	What happened / Why difference
Image → Recipe (Text)	Joint-embedding + semantic reg R@1=0.24, R@5=0.51, R@10=0.65 (medR=5.2)	Joint-embedding InfoNCE R@1=0.8896, R@5=0.9911, R@10=0.9957	Current paper results are much higher; likely due to different test set size / retrieval pool or data leakage.
Recipe (Text) → Image	Joint-embedding + semantic reg R@1=0.24, R@5=0.51, R@10=0.65	R@1=0.8306, R@5=0.9787, R@10=0.9906	Similar explanation: evaluation protocol differs, or retrieval set is smaller/easier.
Training Loss (final)	Not reported in paper	0.0145	Very low loss indicates strong convergence; check for overfitting or leakage.
Validation Loss (final)	Not reported in paper	0.0345	Validation loss is also low; still possible that test set overlaps or retrieval is too easy.

Figure 20: Comparison with Existing Work

6. Conclusion and Future Work

6.1 Summary of Findings

- Successfully developed and implemented a cross-modal recipe retrieval system based on images and text.
- Learnt a shared embedding space which effectively that aligns the two modalities of visual and text information.
- Strong performance in Recall@K across all retrieval tasks.
- Text-dominant and long-text retrieval tasks showed the highest accuracy.
- The smooth decrease in the training and validation loss curves indicated that the model converged well.
- Transfer learning has significantly improved the efficiency of the learning process.

6.2 Achievement of Objectives

- Achieved the goal of accurate retrieval between images and texts.
- Showed strong performance in terms of reliable information retrieval for both Text→Image and Image→Text modalities
- Validated the efficiency of contrastive learning in semantic alignment.

- Efficient training with minimal computing resources was made possible with the use of transfer learning.
- A scalable framework has been developed, which is applicable to multimodal information retrieval scenarios.

6.3 Limitations

- The size of the dataset was relatively small, which put a constraint on the variability of visual as well as textual information.
- Potential generalization issues arise when the model is extended to unseen/visually ambiguous recipes.
- Data set imbalance may have caused biased retrieval towards the most frequently occurring classes.
- Computational constraints made to limit the exploration of larger models as well as hyperparameter tuning.
- Technical challenges limited exploration of alternative loss functions and architectures.

6.4 Future Enhancements

- Larger datasets can be used to make the model more robust. Investigate new architectures, including transformer-based multimodal architectures.
- Add enhanced visual feature extractor to obtain better alignment between text and images.
- Handling data imbalance through data augmentation and/or resampling.
- Extend the system to enable it to be deployed in a practical setting for applications like recipe search or recommendations.

References

Bossard, L., Guillaumin, M. and Van Gool, L. (2014) *Food-101 – Mining Discriminative Components with Random Forests*. In: Proceedings of the European Conference on Computer Vision (ECCV)**.** Zurich, Switzerland: Springer, pp. 446–461.

Dalal, N. and Triggs, B. (2005) 'Histograms of Oriented Gradients for Human Detection', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1, pp. 886–893.

Datta, R., Joshi, D., Li, J. and Wang, J.Z. (2008) 'Image Retrieval: Ideas, Influences, and Trends of the New Age', *ACM Computing Surveys*, 40(2), pp. 1–60.

Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Mikolov, T. and others (2013) 'DeViSE: A Deep Visual-Semantic Embedding Model', *Advances in Neural Information Processing Systems (NeurIPS)*, 26, pp. 2121–2129.

Joachims, T. (1998) 'Text Categorization with Support Vector Machines: Learning with Many Relevant Features', *Proceedings of the European Conference on Machine Learning (ECML)*. Chemnitz, Germany: Springer, pp. 137–142.

Kaggle (2025) *Food Ingredients and Recipe Dataset with Images*. Available at: https://www.kaggle.com/datasets/pes12017000148/food-ingredients-and-recipe-dataset-with-images

Karpathy, A. and Fei-Fei, L. (2015) 'Deep Visual-Semantic Alignments for Generating Image Descriptions', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3128–3137.

LeCun, Y., Bengio, Y. and Hinton, G. (2015) 'Deep Learning', *Nature*, 521(7553), pp. 436–444.

Lowe, D.G. (2004) 'Distinctive Image Features from Scale-Invariant Keypoints', *International Journal of Computer Vision*, 60(2), pp. 91–110.

Salton, G. and Buckley, C. (1988) 'Term-Weighting Approaches in Automatic Text Retrieval', *Information Processing & Management*, 24(5), pp. 513–523.

Salvador, A., Hynes, N., Aytar, Y., Marin-Jacobs, L. and Murphy, K. (2017) *Im2Recipe: Learning to Understand Cooking Recipes from Images*. Available at: https://im2recipe.csail.mit.edu/

Salvador, A., Hynes, N., Aytar, Y., Marin, J., Ofli, F., Weber, I. and Torralba, A. (2017) 'Learning Cross-Modal Embeddings for Cooking Recipes and Food Images', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3020–3028.

Salvador, A., Hynes, N., Aytar, Y., Marin-Jacobs, L. and Murphy, K. (2018) 'Learning Cross-Modal Embeddings for Cooking Recipes and Food Images', *arXiv*. Available at: https://arxiv.org/abs/1810.06553

Appendices

GitHub Repo link:

https://github.com/sumanpdneupane/cross_modal_text_image.git

Ready to use model path:

https://github.com/sumanpdneupane/cross_modal_text_image/blob/main/fusion_model.pth