Classical Multi-View Representation Learning

Utsav Patel
utsav.patel10@rutgers.edu

Aniket Sanap
aniket.sanap@rutgers.edu

Sneh Desai
sd1324@scarletmail.rutgers.edu

Sanchit Thakur
st976@scarletmail.rutgers.edu

Abstract

Most real-world problems are characterized by data simultaneously collected from several sensors. For instance, an activity can be recorded by a video camera with image and audio sensors. Web pages contain text, images, audio clips, tables, all of which describe a related concept in that document. Image collections often contain tags or even complete captions written in natural text that describe the content of those images. This project deals with the exploration of different models (both linear and non linear) for the task of cross-model representation learning. Cross-modal representation learning is an essential part of representation learning, which aims to learn latent semantic representations for modalities including texts, audio, images, videos, etc. The main aim is to learn a common representation from multiple views. The project starts with studying and exploring the task of classical multi-view representation learning using Canonical Correlation Analysis-a linear model for paired data to extract shared representation between features in text and images. Because CCA finds correlations between two multivariate data sets, CCA data structures are a good fit for exploring relationships between the input and output variables found in ensemble data sets (such as those generated for sensitivity studies, uncertainty quantification, model tuning, or parameter studies). In this project, we explore classical multi-view representation learning by learning the CCA model that links the text and the image views of a recipe, and using ResNet50 backbone to extract visual features from images.

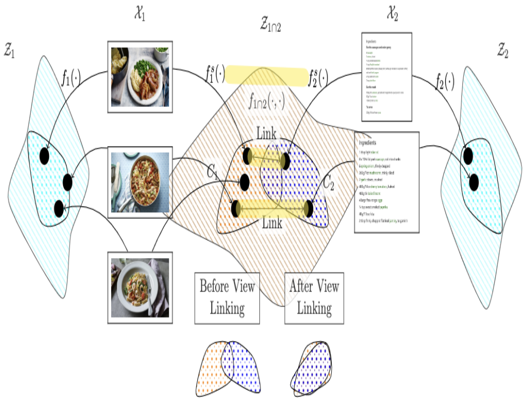


Illustration of Cross-modal Representation Learning

1. Cross Modal Representation Learning

In real world, information about a particular concept or entity is usually obtained through various means, for e.g. an encyclopedia which is known by all age groups contains information in both image and textual format. In accordance with this, most of the real world problems have information coming from multiple media which all describe a related concept. The branch related to analysis of this type of pool of common information fetched from various mediums is called as multi-view analysis. Multi-view, the word is self explanatory which tells that information is seen from multiple views(e.g. text, video and images). More colloquially, cross-model representation learning refers to learning of information coming from multiple modalities or views. Learning such representations form the backbone of various specialized tasks such as: (1) Cross-model Retrieval, (2) Cross-model Translation, and (3) Cross-model Alignment.

(1) Cross-model Retrieval: It refers to predicting information in one view while using the input information in another view. It aims at producing flexible retrieval across various modalities. So if we are given some query in one view, the task is to generate representations in other view which comes down to analysing the similarity of information enclosed in various types of data.

(2) Cross model Translation: It refers to generating representation or information in one view while using the input information provided in another view. It uses the fact that information enclosed in various views contain similarities or similar features as they are in-fact describing a common entity. So the task is to find those common subset of features in one view that are in-fact related to another view.

(3) Cross-model Alignment: It refers to retrieve common subset of features in various views. This is in-fact a usage of conceptual similarity between the various views. So as the name suggests, we actually align the various views and come up with a space of representation that has common subset of features or representation in all those views under inspection.

This project focuses on developing models in 3 steps all addressing to the common problem of ”Cross-model Representation Learning.” The models are made in steps meaning that each successive models is built on top of the predecessor model and this successive model delivers greater performance on the preceding one.

2. Model Used

Then model being used here is the linear model named Canonical Correlation Analysis(CCA) to performs the task of classical multi-view representation learning. CCA is actually like PCA which deals with extracting Principle Components from a single dataset while keeping the amount of variation at maximum but CCA differs from PCA by the fact that it involves extracting correlation between multiple datasets. CCA uses Canonical Variables at its core to analyse the similarity between the datasets. Canonical Variables are basically linear combinations of various variables in one dataset and as the task is to extract dataset similarity, CCA uses pairs of Canonical Variables wherein each pair contains Canonical Variables from both the datasets under consideration. So as a first step, we use CCA- a linear model for pairwise data to handle the task of multi-view representation learning. Here we have feature extractors at our disposal such as ResNet50 [2] for image features and BERT [1] for textual features and then employ CCA to learn common representation. As the first step, we link the text and image views of a recipe and so we use the dataset Recipe 1 Million(R1M) [3].

3. Dataset Used

The Recipe 1 million(R1M) dataset is used for all the models developed developed throughout the scope of this project. This dataset consists of ∼1M text recipes that contain titles, instructions and ingredients in English. Additionally, a subset of ∼ 0.5M recipes contain at least one image per recipe. Data is split in 281598 train, 60422 validation and 60740 test image-recipe pairs, in accordance to the official data release provided in R1M.

4. Data Preprocessing

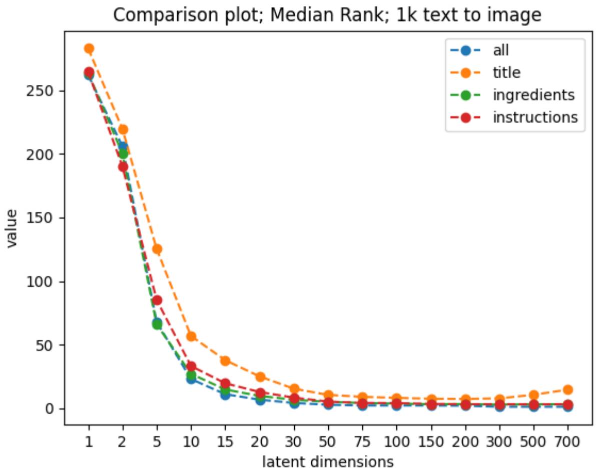
For preprocessing of the data from the dataset described above, the wordpiece tokenizer was used on the text. It works by splitting words either into the full forms (e.g., one word becomes one token) or into word pieces — where one word can be broken into multiple tokens. Also, for the images, center crop was taken by using a preprocessing layer which crops images. This layers crops the central portion of the images to a target size. If an image is smaller than the target size, it will be resized and cropped so as to return the largest possible window in the image that matches the target aspect ratio. In order to expand the dataset for training a deep learning model, small amounts of image augmentation was performed as well. This would help in improving performance and outcomes of our machine learning model by forming new and different examples to train datasets.

5. Evaluation

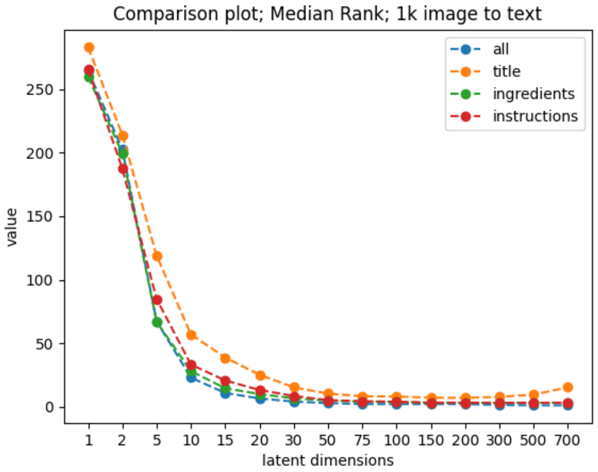
For Step 1, we evaluate the CCA model on standard retrieval metrics names median rank(medR) and recall rate at top K(RK) as the model has to be evaluated on the retrieval tasks both on text-to-image and image-to-text representation retrieval. If the retrieval performance of the CCA model are of superior quality then the medR is low in value and RK is high in value i.e, RK measures the percentage of true positives being ranked within the top K returned results and inline with previous works we report values at k= 1, 5, 10. Both medR and RK are calculated based on a search pool of either 1k or 10k test samples, with the average over 10 different subset reported. We also perform ablation study wherein we check for various settings of dimensions of shared space. Also for text view, we check which elements from 3 main ones namely: title, ingredient and instruction, has high impact on the performance. We even visualize and gain further insights into the learned embeddings to explain the model’s performance behaviour in a sensible manner by visualizing ”Muffin” and ”Salad”’s latent representation using TSNE.

6. Results and Conclusions

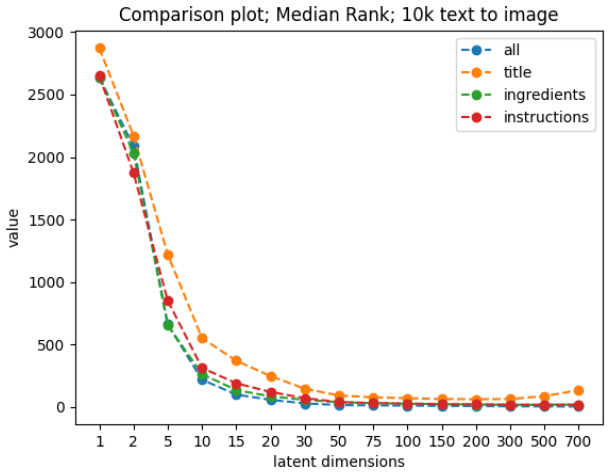
The results and graphs obtained are shown below:



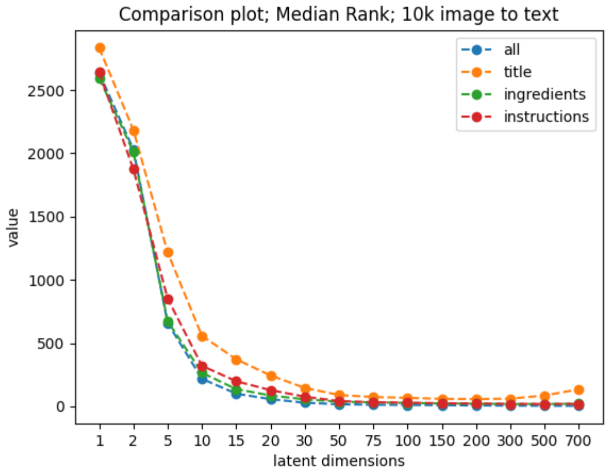
As we can see, the medR value for the 1k test sample and text to image representation falls sharply for all 3 elements, upto a specific latent dimension, beyond which it becomes more or less constant.



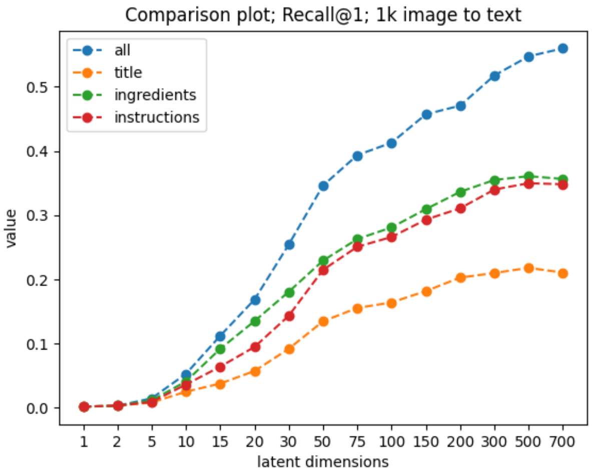
As seen in the above plot, the medR value for the 1k test sample and image to text representation falls sharply for all 3 elements, upto a specific latent dimension, beyond which it becomes more or less constant.



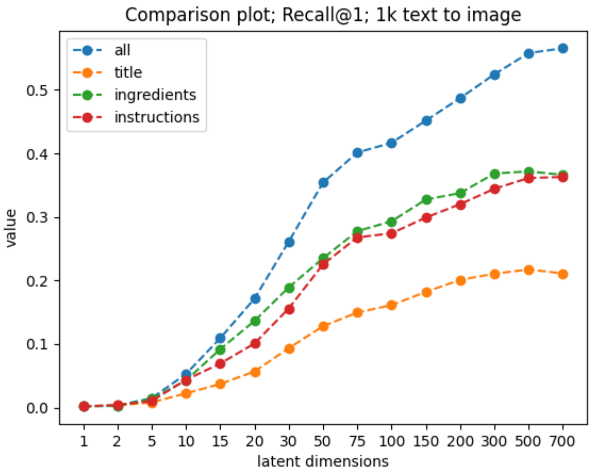
Here, the medR value for the 10k test sample and text to image falls sharply for all elements, upto a specific latent dimension, beyond which it becomes constant.



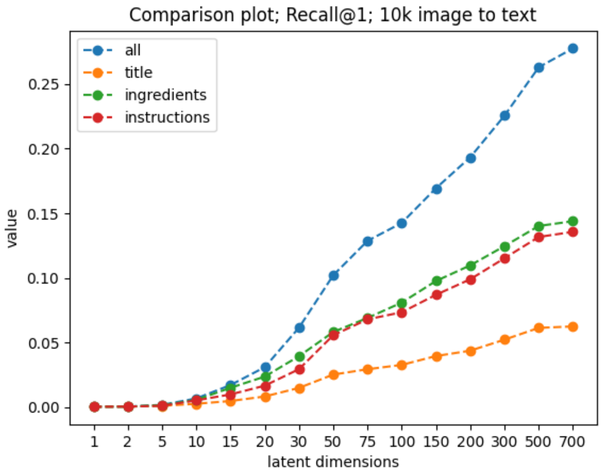
As seen in the above plot, the medR value for the 10k test sample and image to text representation falls sharply for all 3 elements, upto a specific latent dimension, beyond which it becomes more or less constant.



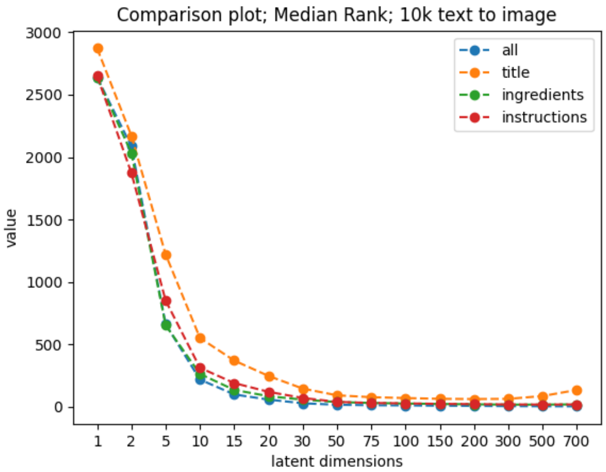
Here, for the 1k test sample and image to text representation, we compare the recall value at k=1 for all 3 elements. As we can see, the value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



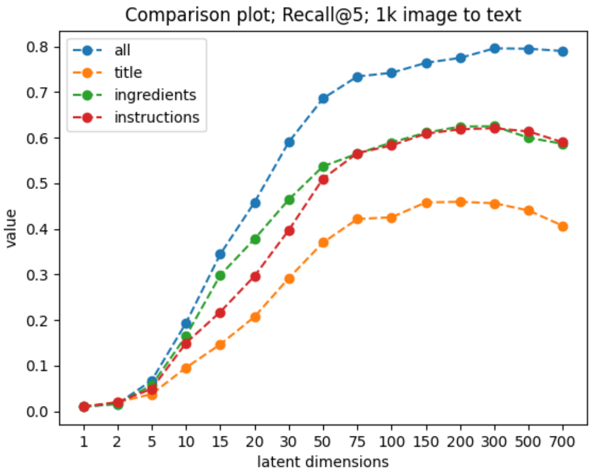
Here, for the 1k test sample and text to image representation, we compare the recall value at k=1 for all 3 elements. As we can see, the value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



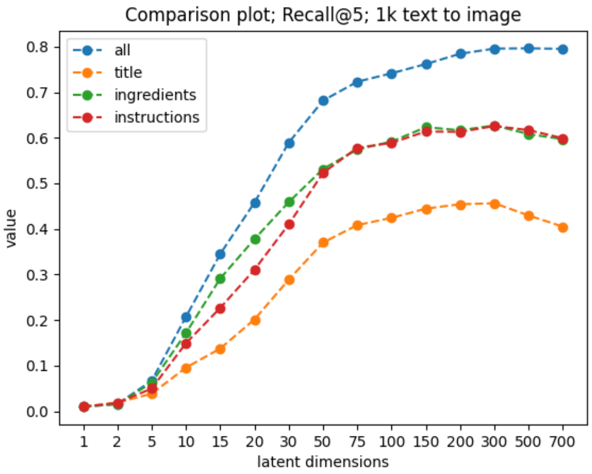
Here, for the 10k test sample and image to text representation, we compare the recall value at k=1 for all 3 elements. As we can see, the value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



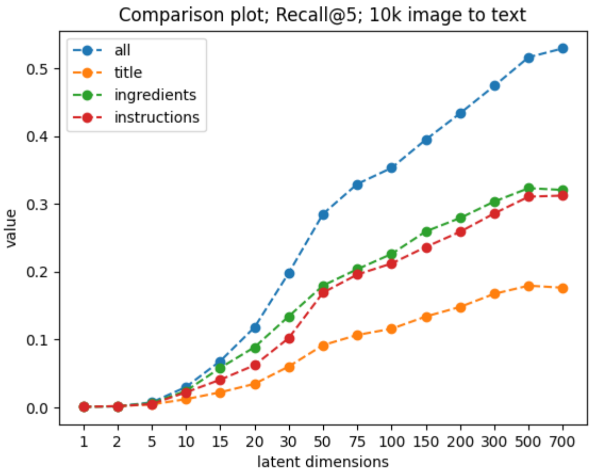
Here, for the 10k test sample and text to image representation, we compare the recall value at k=1 for all 3 elements. Here, the value decreases with the latent dimension upto a certain point, beyond which it stabilizes.



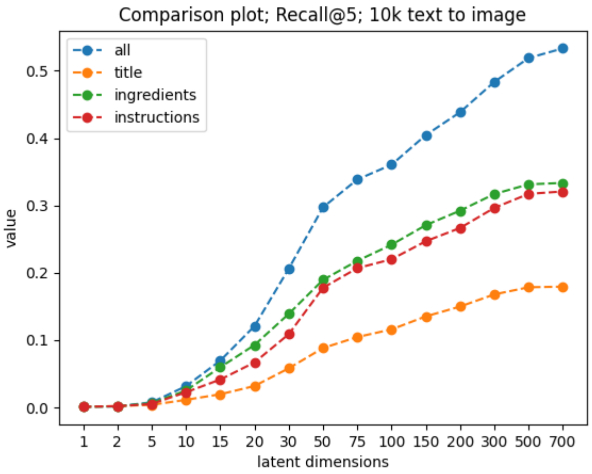
Here, for the 1k test sample and image to text representation, we compare the recall value at k=5 for all 3 elements. The value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



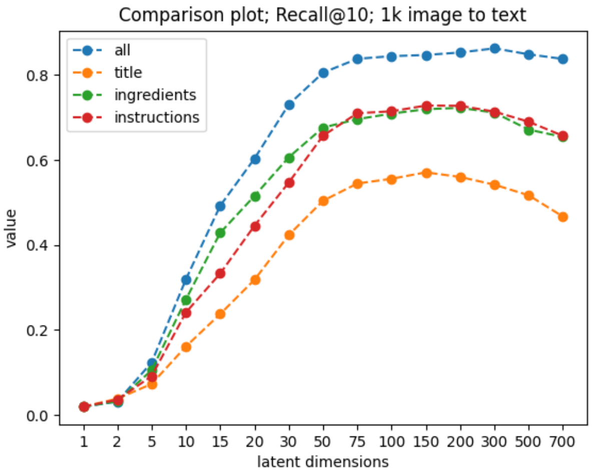
Here, for the 1k test sample and text to image representation, we compare the recall value at k=5 for all 3 elements. The value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



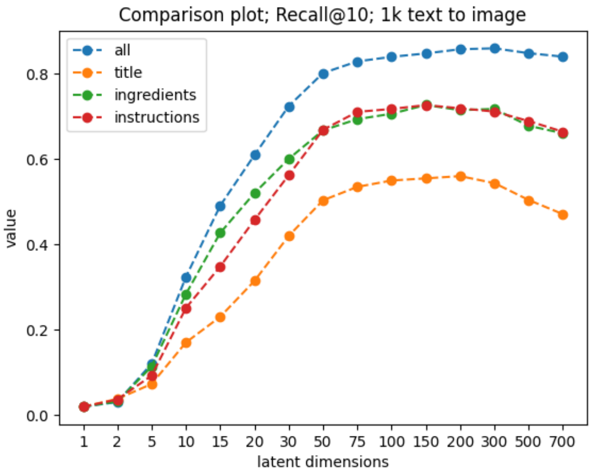
Here, for the 10k test sample and image to text representation, we compare the recall value at k=5 for all 3 elements. The value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



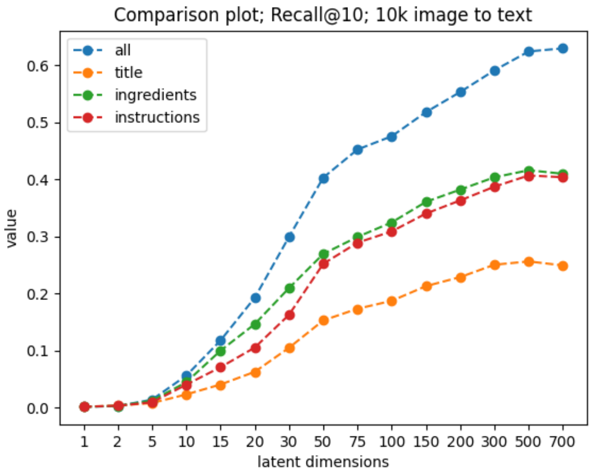
Here, for the 10k test sample and text to image representation, we compare the recall value at k=5 for all 3 elements. The value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



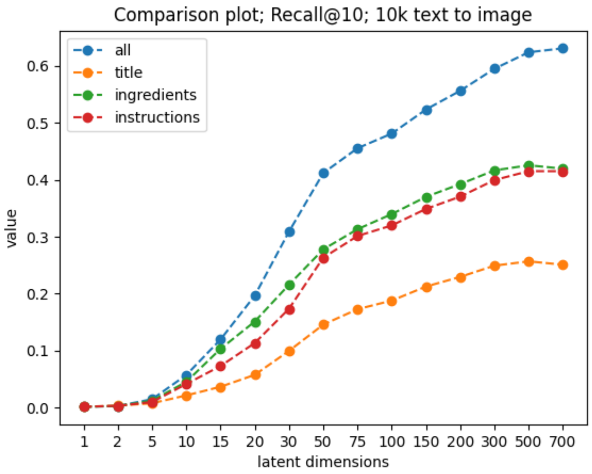
Here, for the 1k test sample and image to text representation, we compare the recall value at k=10 for all 3 elements. The value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



Here, for the 1k test sample and text to image representation, we compare the recall value at k=10 for all 3 elements. The value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.

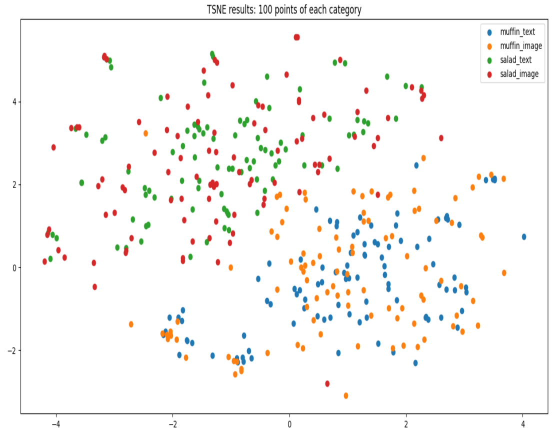


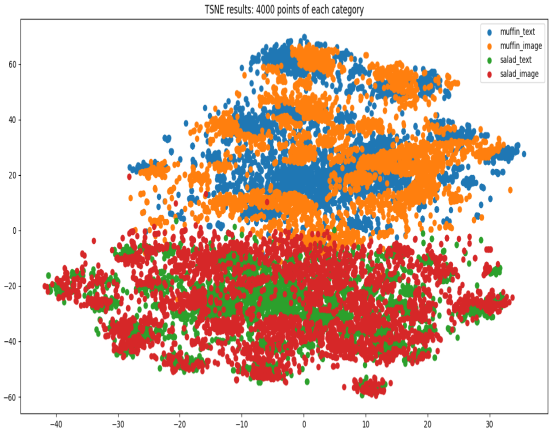
Here, for the 10k test sample and image to text representation, we compare the recall value at k=10 for all 3 elements. The value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.



Here, for the 10k test sample and text to image representation, we compare the recall value at k=10 for all 3 elements. The value increases with the latent dimension, with the value being the highest for all elements together compared to each element individually, whereas the value for the title alone is the lowest.

We have also visualized the latent dimension of linear CCA by reducing its dimension to 2 using TSNE. We extracted embeddings with ”muffin” and ”salad” in their title. We reduced the dimension of those latent space features generated by linear CCA using TSNE. Our model can clearly distinguish between ”muffin” and ”salad,” both in image and text.





References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[3] J. Marin, A. Biswas, F. Ofli, N. Hynes, A. Salvador, Y. Aytar, I. Weber, and A. Torralba. Recipe1m+: A dataset for learning cross-modal embeddings for cooking recipes and food images. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):187–203, 2019. 2