CS536 - Final Project
Cross Modal Representation Learning

February 23, 2022

1 Introduction

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. In machine learning, multi-view analysis [1, Chapter 20.2.8] refers to the setting where data about a single concept comes in multiple views (image, text, audio signal, graph, table, ...).

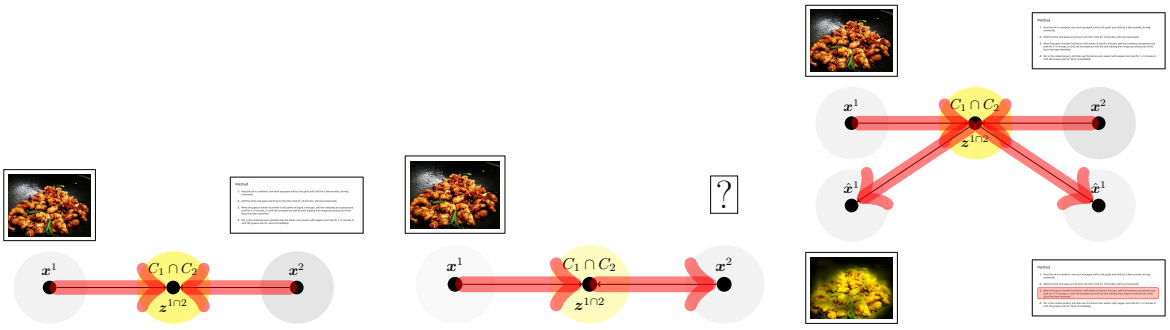
In this project, you will explore models and methods related to cross-modal representation learning, where the goal is to learn a common (and possibly private) representation from multiple views. Learning this representation is essential for several key tasks:

1. Cross-modal retrieval: given a query in one view (e.g., text), retrieve similar instances from the gallery in another view (e.g., images).

2. Cross-modal translation: given an instance in one view (e.g., image), reconstruct that instance in another view (e.g., produce a text caption for the given image).

3. Cross-modal alignment: given different views of an instance, align the subsets of features in the two views that correspond to each other.

These tasks are illustrated in Fig. 1 on an example of image and text views.



(a) Retrieval (b) Translation (c) Alignment (correspondences)

Figure 1: Three tasks of Cross-Modal Representation Learning: (a) Predicting one view from another. (b) Finding subset of features in one view that are attributed to another view. (c) Finding correspondence between subsets of features in two views that give rise to the views' conceptual similarity.

Suppose there are $M(\geq 2)$ **views**, $\{C_1, C_2, \ldots, C_M\}$ associated with an entity $\boldsymbol{x}$, represented as different **views** $\boldsymbol{x}^m$, $m = 1, \ldots, M$, of entity $\boldsymbol{x}$ in each respective view. In the case, for instance, the underlying entity might be a prepared meal, and one view could be the image of the meal, another the list of ingredients (in

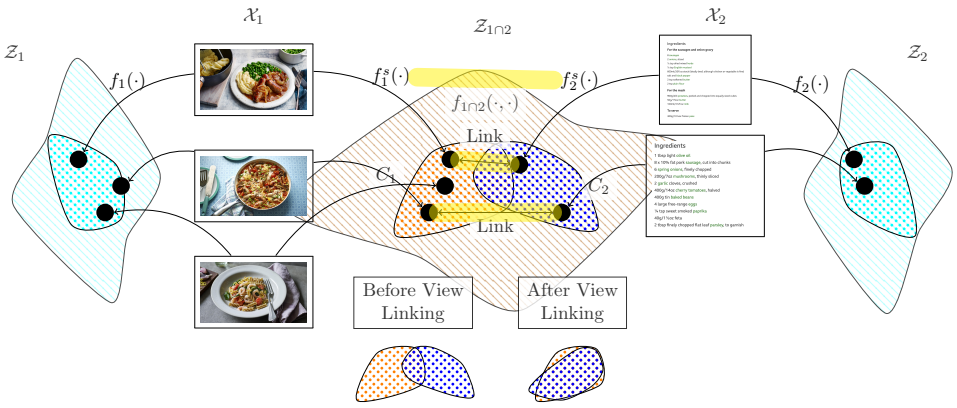


Figure 2: Illustration of Cross-modal Representation Learning. Image view, C_1, and text view, C_2, should be linked (paired) through the shared representation in $\mathcal{Z}_{1\cap 2}$ space. Those shared representations can be recovered using mappings $f_1^s(\cdot)$ and $f_2^s(\cdot)$, proxies for joint mapping $f_{1\cap 2}(\cdot,\cdot)$. Individual views also contain private representations in spaces $\mathcal{Z}_1$ and $\mathcal{Z}_2$, recoverable from instances by private mappings $f_1(\cdot)$ and $f_2(\cdot)$. Examples of paired as well as unpaired instances are given in training data. Once paired, the view representations become matched in the shared space. This will enable information and knowledge transfer across views.

text) needed to prepare the meal, the recipe instructions describing how to combine those ingredients, and yet another view the meta information about the food, like flavors or nutritional facts.

What links those views together is the set of **common representations** $\boldsymbol{z}^{1\cap 2\cap 3\ldots\cap M}$, sometimes also called factors. This representation describes the common aspects of different views. For convenience we assume that $\boldsymbol{z}^{1\cap 2\cap 3\ldots\cap M} \in \Re^{D_z} = \mathcal{Z}_{1\cap 2\cap 3\ldots\cap M}$, the canonical **shared view space**. In , those could be the canonical representations of ingredients that can be visually detected in a food image as well as found in the list of ingredients or the recipe instructions. While those representations could be flat and sparse, i.e., each ingredient is represented as a one-hot vector in $\mathcal{Z}$, it is much more likely that those representations are distributed (and locally dense), as an indicator of similarity of different views.

Oftentimes, individual views of the same entity also contain **private representation**. We denote them by $\boldsymbol{z}^m \in \Re^{D_m} = \mathcal{Z}_m$ for view m. For instance, $\boldsymbol{z}^{\text{Image}}$ could be the factors that are specific to food images, such as the pose or the size of food objects/ingredients in the image, the serving dish, or decorative elements, all of which are not reflected in other views, such as the recipe text. This is illustrated in Fig. 2.

1.1 Food AI

A compelling example of multi-view representation learning can be found in the task of linking together the image and text representations of data, for problems such as image captioning, visual question answering, instructional video analysis, etc. In the domain of cooking and nutrition, the text-image pairs are the recipe and meal image counterparts. We will use this application domain to ground the multi-view representation learning in this project. To learn more about Food AI, see e.g, [2; 3; 4; 5].

2 Classical Multi-View Representation Learning

2.1 Concept

In this task you will explore classical multi-view representation learning. You will assume known feature extractors, such as the ResNet features in images or the word2vec features in text, then learn common representations using linear models for paired data, the Canonical Correlation Analysis (i.e., Supervised PCA), [1, Chapter 20.2.8].

2.2 Task

Learn the CCA model that links the text and the image views of a recipe. You will use ResNet50 or ResNext101 backbone to extract visual features from images. You will use the WordPiece [6] to tokenize the full sequence of recipe words, and average them to create the text feature.

2.3 Evaluation

Model will be evaluated on the retrieval tasks, both text-to-image and image-to-text. You will use standard retrieval metrics: median rank (medR) and recall rate at top K (RK). Here, 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. See [5] for more details.

As a part of the evaluation you will need to conduct ablation studies regarding the optimal dimension of the shared space. For the text view, you will want to consider which element of the recipe text (title, ingredients, instructions) most significantly impacts the retrieval performance. You should also visualize and investigate the learned embeddings to explain the model's performance.

2.4 Datasets

You will use the Recipe 1 Million (R1M) [2; 7]. 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 238999 train, 51119 validation and 51303 test image-recipe pairs, in accordance to the official data release provided in R1M.

3 Nonlinear Multi-View Representation Learning

3.1 Concept

You will build upon the linear CCA multi-view model using (a) non-linear deep models [8; 9] and (b) the triplet loss (see e.g., [5]). The goal will be to understand what gains one can obtain from using the models beyond the baseline linear CCA.

3.2 Evaluation

You will use the same evaluation strategy as in Sec. 2.3. Your ablation studies should consider the hyper-parameters of the new models. Like in Sec. 2.3, you will want to use visualization to give support to your findings.

3.3 Datasets

You will use the same data as in Sec. 2.4.

4 Option 1: Correspondence Based Multi-View Models for Image-Text

4.1 Concept

In previous tasks, you used holistic representations of the image and text views (i.e., a single vector for the whole image and a single vector for the body of text). Your aim now is to investigate whether fine-grain representation (e.g., objects/regions in images and words/groups of words in text) and correspondences between those entities within image and text improve both the retrieval performance as well as the interpretability of results (e.g., highlight that only certain parts of image and certain words in text are relevant for learning the shared cross-modal representation; other image and text elements may be private to individual modalities).

Models such as transformers that incorporate attention mechanisms [10], a form of dense correspondences, offer the type of representation amenable to this task. Transformers have been designed to primarily represent data withing individual views, but have also been extended to work across data views, c.f., [11, Sec. 3.7, Transformers for Multi-Modal Tasks].

You will investigate transformer architectures that are appropriate for the task above. After choosing an appropriate architecture, you will apply it to the data from Sec. 2.4.

In this task, you have the freedom to both use existing models (architectures) or propose and evaluate your own.

4.2 Datasets

You will use the R1M dataset from Sec. 2.4. However, this dataset lacks correspondence (e.g., object localization) labels.

For localization evaluation, you can use the dataset r-FG-BB in [12]. Unfortunately, this dataset contains text in Japanese, possibly incompatible with the English-trained models on R1M. Additional credit will be given for teams that suggest the means to bridge the gap between the unlabeled (R1M) and the labeled (r-FG-BB) dataset, necessary for quantitative evaluation.

4.3 Evaluation

You will evaluate the performance of your chosen model based on the retrieval metrics described in Sec. 2.3. However, you will also seek to evaluate, qualitatively, whether the learned correspondences (attention is a proxy for those correspondences) lead to meaningful image-region-to-word-entity associations (e.g., the word "egg" in the text of a recipe is associated with the image region where the egg is visible.) Because R1M lacks the localization or correspondence labels, your evaluation can be either qualitative or can be based on the above-referenced Japanese text dataset r-FG-BB.

5 Option 2: Correspondence Based Multi-View Models for Video-Text

5.1 Concept

While in the image-text multiview pair the correspondence may be between image regions and words in text (as in Sec. 4), in video-text pairs the correspondence will be between video frames (or short video clips) and words or sentences in the text. This task aims to investigate the multiview representation learning for video-text pairs.

The correspondence task in video-text is often formulated as the video-to-text alignment problem. In this context, the correspondence task is often referred to as the video-text segmentation task or the text-video clip localization task. Some examples of prior work in this direction can be found in [13; 14; 15]. These models treat video and text as streams that need to be aligned together in order to find correspondences.

Your goal will be to chose or propose a model that can accomplish the video-text alignment for instructional data, as described below.

5.2 Datasets

The R1M dataset described in Sec. 2.4 does not contain references to videos of collected recipes. Instead, you will use the YouCookII dataset [16] of transcribed cooking videos.

5.3 Evaluation

Your evaluation should be based on the fine-grained version of retrieval known as the Step Localization, c.f., [15].

5.4 Advanced option

Most of the aforementioned approaches treat text and videos as stream. However, instructional text (e.g., in recipes) is not unstructured in the same way regular written text may be (e.g., in stories). Structured text of recipes or other instructional text can be effectively represented using Flow Graphs [17; 18]. Thus, instead of stream-to-stream alignment, one can consider the fundamental problem of graph-to-stream alignment. Unfortunately, this task is notoriously difficult and has not been studied extensively. Few solutions have been proposed in the hypertext analysis community [19] and computational biology [20]. Your goal here would be to extend these solutions to the problem at hand.

References

[1] K. P. Murphy, *Probabilistic Machine Learning: An Introduction*. MIT Press, Mar. 2022.

[2] A. Salvador, N. Hynes, Y. Aytar, J. Marin, F. Ofli, I. Weber, and A. Torralba, "Learning cross-modal embeddings for cooking recipes and food images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3020–3028, 2017.

[3] A. Salvador, M. Drozdzal, X. G. i Nieto, and A. Romero, "Inverse cooking: Recipe generation from food images," in *CVPR*, 2019.

[4] M. Fain, A. Ponikar, R. Fox, and D. Bollegala, "Dividing and conquering cross-modal recipe retrieval: from nearest neighbours baselines to sota," *CoRR*, vol. abs/1911.12763, 2019.

[5] R. Guerrero, H. X. Pham, and V. Pavlovic, *Cross-Modal Retrieval and Synthesis (X-MRS): Closing the Modality Gap in Shared Subspace Learning*, p. 3192–3201. New York, NY, USA: Association for Computing Machinery, 2021.

[6] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean, "Google's neural machine translation system: Bridging the gap between human and machine translation," *CoRR*, vol. abs/1609.08144, 2016.

[7] J. Marín, 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*, 2019.

[8] Q. Tang, W. Wang, and K. Livescu, "Acoustic feature learning via deep variational canonical correlation analysis," in *Interspeech 2017*, 2017.

[9] M. Suzuki, K. Nakayama, and Y. Matsuo, "Joint multimodal learning with deep generative models," Nov. 2016.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. U. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), pp. 5998–6008, Curran Associates, Inc., 2017.

[11] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in vision: A survey," Jan. 2021.

[12] T. Nishimura, S. Tomori, H. Hashimoto, A. Hashimoto, Y. Yamakata, J. Harashima, Y. Ushiku, and S. Mori, "Visual grounding annotation of recipe flow graph," in *Proceedings of the 12th Language Resources and Evaluation Conference*, (Marseille, France), pp. 4275–4284, European Language Resources Association, May 2020.

[13] D. Zhukov, J.-B. Alayrac, R. G. Cinbis, D. Fouhey, I. Laptev, and J. Sivic, "Cross-task weakly supervised learning from instructional videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3537–3545, 2019.

[14] V. Gabeur, C. Sun, K. Alahari, and C. Schmid, "Multi-modal transformer for video retrieval," in *Computer Vision – ECCV 2020*, pp. 214–229, Springer International Publishing, 2020.

[15] N. Dvornik, I. Hadji, K. G. Derpanis, A. Garg, and A. D. Jepson, "Drop-DTW: Aligning common signal between sequences while dropping outliers," Aug. 2021.

[16] L. Zhou, C. Xu, and J. J. Corso, "Towards automatic learning of procedures from web instructional videos," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[17] S. Mori, H. Maeta, Y. Yamakata, and T. Sasada, "Flow graph corpus from recipe texts," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, (Reykjavik, Iceland), pp. 2370–2377, European Language Resources Association (ELRA), May 2014.

[18] Y. Yamakata, S. Mori, and J. Carroll, "English recipe flow graph corpus," in *Proceedings of the 12th Language Resources and Evaluation Conference*, (Marseille, France), pp. 5187–5194, European Language Resources Association, May 2020.

[19] G. Navarro, "Improved approximate pattern matching on hypertext," *Theoretical Computer Science*, vol. 237, no. 1-2, pp. 455–463, 2000.

[20] V. N. S. Kavya, K. Tayal, R. Srinivasan, and N. Sivadasan, "Sequence alignment on directed graphs," *J. Comput. Biol.*, vol. 26, pp. 53–67, Jan. 2019.