

Dissecting Language’s Effect on Bottleneck Models

Stanford CS224N Custom Project

Songlin Li, Qinziyue Xu

Department of Computer Science

Stanford University

svli97@stanford.edu, qinziyue@stanford.edu

Abstract

Language has been shown to be helpful on improving models’ ability to generalize on unseen abstract concepts. However, it’s still unclear why and how language can help the model to generalize. Inspired by the results in Learning with Latent Language (L3) [1], we aim to answer several crucial questions about language’s role on facilitating learning and improve L3’s performance on few-shot classification where models need to learn how to quickly adapt to unseen tasks by learning from a set of similar tasks. We first demonstrate that the accurate descriptions of the spatial relationships can massively improve the models’ performance on few-shot classification by providing correct guidance encoded in natural language. To improve model’s performance, we focus on two directions: 1) enhancing model’s visual reasoning via providing more informative language guidance on spatial relationships, 2) enhancing model’s ability to fuse different modality. Our results demonstrated that 1) we can achieve comparable classification performance by using a simple concept retrieval mechanism 2) a more trainable image model can dramatically improve the classification accuracy.

1 Key Information to include

- External mentor: Jesse Mu

2 Introduction

Human language contains rich information that helps us explore the world. For example, we can easily learn about a spatial relationship between objects not by seeing hundreds of examples, but by knowing the description of the relationship. By encoding abstract concepts into language, humans can easily generalize to various examples even if the examples contain noisy information. We explore the language’s effect on improving few-shot visual classification in an under-explored scenario where natural language descriptions are available during training but inaccessible during testing. Previous works have demonstrated that models’ ability to generalize on unseen abstract concepts can be improved by incorporating a language model. One of the approach is to use language model as a bottleneck where image representations are used to generate concept descriptions which are then re-encoded as hidden representations for classifications. [1] Another approach is using language model as a regularizer [2], where generated concepts are not used for classifications (Figure 1). However, language model’s exact role on improving classification accuracy is unknown. Based on our experiments, accurate concept descriptions can massively increase models’ classification accuracy. We treat this level of performance as the Oracle. To reach a near-Oracle performance level, we explore two directions including improving language model’s ability on generating correct concept descriptions and enhancing the fusion of different modalities’ representations. For the first direction, we experiment with Schedule Sampling, Concept Retrieval. Multimodal approach and Bilinear Scorer are employed as part of the second direction. We propose a new retrieval structure to replace the bottleneck during testing. Instead of generating concept description during testing, we simply retrieve the most relevant concepts. We are able to match the performance of the current state-of-the-art.

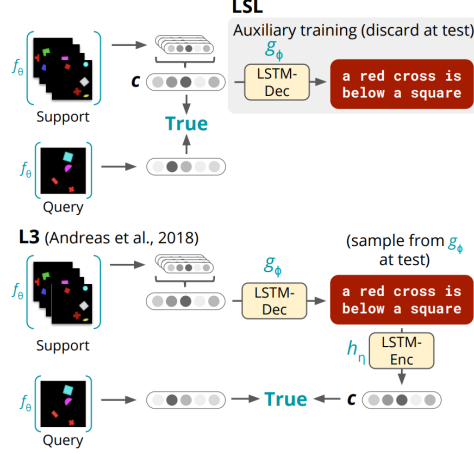


Figure 1: A illustration of L3 and LSL. Red boxes contain descriptions (concept descriptions) of shapes’ spatial relationship [2]

2.1 Few-Shot Classification

In a few-shot classification setting, a task consists of example images and query image. For each task, the model is presented with few images (example images), all associated with the same visual concept such as *a white shape is left of a yellow semicircle*, and asked to decide whether the held-out image (query image) belongs to the same visual concept. Under our setting, models have access to the descriptions of the visual concepts shared by the example images during training (Figure 2).

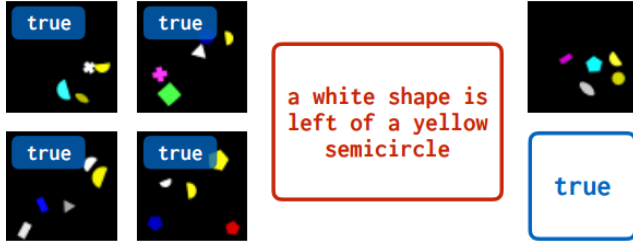


Figure 2: A illustration of our version of few-shot classification problem [1]

3 Related Work

3.1 L3 and LSL

Existing models have indicated that incorporating a language model can improve model’s accuracy for few-shot classification problem. In L3 [1], the model consists of a proposal model and an interpretation model. Proposal model takes example images and proposes a concept description of the images. Interpretation model takes query image and proposed description to classify if the query image and example images share the same concept or not. LSL’s model contains the same proposal model as L3 [2]. However, unlike L3’s interpretation model, LSL makes prediction only based on representations of examples and query image. For LSL, proposal model is used as a regularizer, and only executes in training phase. LSL achieves comparable results to L3. Although L3’s bottleneck structure is more complicated than LSL, in addition to higher accuracy, it serves as an flexible platform for other mechanisms such as explicit retrieval. A bottleneck may also implicitly distill noisy information. However, neither work provide quantitative analysis on the qualities of generated concept descriptions nor the relationship between performances and the description quality.

3.2 Retrieve and Edit

Retrieve and Edit framework built on the assumption that previously seen tasks contains information that’s useful for the current task. Therefore, it might be more optimal to edit the representation of a seen task than generating a new representation from scratch [3]. Inspired by these designs, we incorporated retrieval mechanism into our model. Due to the vocabulary size of concept descriptions

of few-shot classification task is small and all concept descriptions are in a similar format, a well designed retrieval mechanism can replace the proposal model to avoid the difficulties of generating accurate concept descriptions from scratch.

3.3 Scheduled Sampling

Scheduled Sampling indicates that instead of always training RNN models with ground-truth tokens, training the RNN model by passing generated token into next RNN state with a probability can improve model’s performance[4]. One of the key discrepancies between RNN models’ training and testing is that the ground-truth tokens are used as inputs during training while the generated tokens are used during testing time. This discrepancy causes model to be susceptible to compound error that models will be facing during test time. Scheduled Sampling helps models to adapt to the compound errors. Since the interpretation model of L3 is trained with ground-truth concept descriptions and relies on sampled descriptions from proposal model during testing time, we face a similar discrepancy. We thus apply the Scheduled Sampling concept on L3: instead of always training L3’s interpretation model with ground-truth prediction, the team feed sampled description to interpretation model with a probability that increases over training.

4 Approach

Our approaches mainly revolve around two directions: improving model’s classification accuracy via improving the language model or improve reasoning over different modalities.

4.1 L3 Baseline

L3 model includes a proposal model q and an interpretation f [1]. For proposal model q , $\{x_s\}$ is the set of example images for a single task with $|\{x_s\}| = n$. w represents the concept description proposed by proposal model. $rnnDecode$ is a RNN model containing Gated Recurrent Units (GRUs)[5]. rep is an image model with a pretrained 16-layer VGGNet(VGG16) and two fully-connected layers to map input image into an image representation. The proposal model takes in example images as input, and samples a concept description based on the average of image representations.

For interpretation model f , x_q is the query image of a task and w is the proposed concept description. $rnnEncode$ is an embedding layer that maps descriptions into hidden representations. We define input of σ function as $score$. When $score > 0$, σ function predicts True. Magnitude of $score$ indicates the confidence of prediction. For L3, we have $hintScore = rnnEncode(w)^T rep(x_q)$. Therefore, f calculates the probability of query image follows the concept description of example images based on the dot product of representations.

$$q(w|\{x_s\}) = rnnDecode(w) \frac{1}{n} \sum_i rep(\{x_s\})_i \quad (1)$$

$$f(x_q; w) = \sigma(rnnEncode(w)^T rep(x_q)) \quad (2)$$

During inference, two models run in sequence to make predictions. During training time, two models are trained based on ground-truth descriptions w^j and labels y^j for tasks in training dataset indexed by j to minimize loss.

$$Loss = - \sum_j \log q_\theta(w^j|\{x_s\}^j) - \sum_j L(f_\theta(x_q^j, w^j), y^j) \quad (3)$$

4.2 LSL Baseline

LSL’s language model is same as the proposal model in L3. However, its prediction model p is defined as [2]:

$$p(y|x_q, \{x_s\}) = \sigma((\frac{1}{n} \sum_i rep(\{x_s\})_i)^T rep(x_q)) \quad (4)$$

With y represents the label, $\{x_s\}$ represents example images, and x_q represents query image. For LSL, we have $imageScore = (\frac{1}{n} \sum_i rep(\{x_s\}_i))^T rep(x_q)$. During training time, two models are trained based on ground truth descriptions and labels to minimize joint log-likelihood loss.

$$Loss = - \sum_j \log q_\theta(w^j | \{x_s\}^j) - \sum_j \log p_\theta(y = y^j | x_q^j, \{x_s\}^j) \quad (5)$$

During inference, prediction model makes prediction based on example images and query images. LSL relies only on image representation model for prediction and uses language model as a regularizer (1) during training time and does not use language model's output for classification.

4.3 Resnet

Both L3 and LSL baseline image model uses pretrained VGG16 to map image to representation. Since pretrained VGG16 is generally fixed with only 2 fully-connected layers near the output can be trained, it might be very difficult for the image model to adapt to the dataset. We replace the pretrained VGG16 with an existing Resnet18 implementation in LSL's codebase [2] so that the image model can provide better representations for the downstream classification.

4.4 Oracle:

To determine accurate concept descriptions' effect on model performance, we developed the Oracle model based on L3. Instead of using proposed descriptions for classification during test time, oracle uses ground-truth concept descriptions. This method provides a rough upper bound on the potential improvement that can be achieved by incorporating a language model.

4.5 Concept Retrieval:

Concept Retrieval method explicitly retrieves the most relevant concept description as the proposed concept description. For all $task_k$ in the training set, the model records the example images' representations $\mu^k = \frac{1}{n} \sum_i rep(\{x_s\}_i^k)$ and the ground-truth concept descriptions w^k . During inference, for all $task_q$, we use example images' representations $\mu^q = \frac{1}{n} \sum_i rep(\{x_s\}_i^q)$ to retrieve the most relevant concept descriptions by calculating similarity scores between μ^q and all μ^k with a function s .

$$q(w_k | \{x_s\}^q) = \frac{\exp s(\mu^k, \mu^q)}{\sum_k \exp s(\mu^k, \mu^q)} \quad (6)$$

Retrieved concept is then passed to the interpretation model for classification, replacing the proposed concept description. We experiment with dot product, cosine similarity and L2 distance as s .

4.6 Bilinear Scorer:

L3 uses dot product as the scorer to measure the similarity between query image x_q^j and proposed concept description w^j . However, since the concept descriptions and images belong to different modalities, it's very likely that a scorer with more layers is more flexible and can better approximate the distribution of hidden representations. Thus, instead using vanilla dot product. We experiment with Bilinear Scorer which contains an extra fully connected layer that project one of the hidden representations to the same space as the other. Theoretically, this measure should improve score's ability to fuse representations from different modality and consequently increase accuracy.

$$f_{score}(w^j, x_q^j) = rnnEncode(w^j) \cdot rep(x_q^j) W_{proj} \quad (7)$$

4.7 Scheduled Sampling:

We extend Scheduled Sampling's concept to L3. Instead of always training L3's interpretation model with ground-truth descriptions, we modify the existing interpretation model to be trained based on proposed descriptions from proposal model with a probability. We experiment with two probability decay functions to calculate p , the current probability of using ground-truth description for classification.

- Linear Decay: $p = 1 - \frac{i}{I}$, where i represents the number of current iteration and I represents the number of total iterations.
- Inverse Sigmoid Decay: $p = \frac{k}{k + \exp(i/k)}$, where i represents the number of current iteration and k is a constant that determines the speed of convergence.

4.8 Multimodal:

Multimodal approach enhances L3 by incorporating example images' representations into interpretation model. Multimodal's interpretation model can be represented as:

$$f(x_q, \{x_s\}, w) = \sigma(\text{multimodal}(\text{rnnEncode}(w), \frac{1}{n} \sum_i \text{rep}(\{x_s\})_i^T \text{rep}(x_q)) \quad (8)$$

The *multimodal* function represents a function to combine representations of description and the mean of representations of example images. We experiment with following approaches:

- Sum: $\text{multimodal}(x, y) = x + y$
- Element-wise Product: $\text{multimodal}(x, y) = x \odot y$
- Projection Function: Multimodal function concatenates inputs and feed it into 3 linear transformation layers with ReLU activation function.
- Weighted Average: $\text{multimodal}(x, y) = wx + (1 - w)y$, with w is a weight calculated by concatenating inputs and feed into 2 linear transformation layers with ReLU activation function.

5 Experiments

5.1 Data

To get comparable results with L3 [1] and LSL [2], we use the same ShapeWorld dataset [6] which consists of 9000 training, 1000 validation and 4000 test tasks. For each task, dataset contains 4 images to represent a visual concept and 1 query image for classification. Each image contains 4 or 5 entities each identified by one of 8 shapes and 8 colors. Visual concept is a description of spatial relationship between one pair of entities within the image that is common between 4 images provided. Classification result is True or False representing whether the query image belongs to the visual concept defined. Half of the validation and test concepts appear in training set and query image belongs to the visual concept for 50% of the time.

5.2 Evaluation method

Models' hyperparameters are selected based on the best validation accuracy. Models are evaluated based on their classification accuracy on the testing set.

5.3 Resnet

Due to memory constraint, L3 model with Resnet18 runs for 50 epochs with batch size 32. We use Adam Optimizer with learning rate 0.0001. Hidden size of the GRUs and representation dimensions are 512. Original L3 and LSL results are run with batch size 100. To get fair comparison, we rerun L3 with batch size 32 and record the result. Figure 3 indicates the importance of training image model. By replacing fixed VGG16 with Resnet18, the image model can be trained with training data and thus achieves a higher test accuracy than all of the baseline models.

5.4 Oracle

We use the same set of hyper-parameters in 5.3. By supplying the interpretation model with accurate concept description, L3 with VGG16 and batch size 32 was able to match the performance of L3 with Resnet (see Figure 3 for baseline performances). This huge performance gain demonstrates the effect

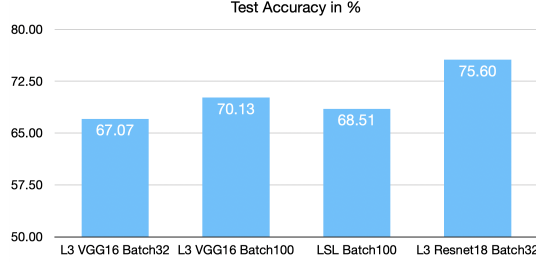


Figure 3: Test accuracies comparison between baseline models and Resnet18 implementation

of correct concept descriptions on improving downstream prediction. On the other hand, we can see that L3 with Resnet’s test accuracy is only increased from 75.6 to 77.66 with Oracle setting. This indicates that with Resnet18 as image model, L3 is able to close the accuracy gap between predicting with ground-truth description (during training) and proposed description (during inference). Based on Table 1, we can also see that Resnet18 Oracle’s test accuracy is higher than VGG16 Oracle’s test accuracy. This indicates that when both interpretation models take ground-truth descriptions as input, the interpretation model with Resnet18 can make better predictions. This demonstrates that Resnet18 benefits the interpretation model and partially justify Resnet18’s massive improvement in test accuracy.

5.5 Concept Retrieval

We use the same training configuration as 5.3. The model performances with different retrieval methods can be found in Table 1. Overall L2 distance leads to the highest classification accuracy. With retrieval, resnet models can match the performance of Oracle, suggesting that retrieved hint can largely model the visual concept of support images. Additionally, the performance of the best retrieval method match the oracle performance for Resnet image model, suggesting retrievals approach works better with a more flexible image model.

	Dot Product	Cosine Similarity	L2 Distance	L3 Oracle
VGG 16	65.76	65.81	66.61	74.96
Resnet 18	73.60	75.29	76.29	77.66

Table 1: Test accuracies comparison between different retrieval methods and Oracle setting

5.6 Bilinear Scorer

With the same training configuration as 5.3, we replace the interpretation model’s dot product scorer with bilinear scorer. The performance compared to the baseline model can be found in Table 2. As expected, with the a more complex scoring function the model is able to achieve a higher classification accuracy, setting additional layer of projection is beneficial to compare representations from distinct modality.

	Dot Product	Bilinear
VGG 16	67.07	69.10
Resnet 18	75.6	77.89
Resnet 18 + L2 Retrieval	76.29	77.39

Table 2: Test accuracies comparison between different scorers

5.7 Scheduled Sampling

For Scheduled Sampling, we first run L3 with Scheduled Sampling modification without Resnet change for 100 epochs and batch size 100. For inverse sigmoid decay function, we use $k = 15$, which

leads to desired decay function. As shown in Table 3, In fixed VGG16 case, Scheduled Sampling improves test accuracy with both decay equations. This is expected because interpretation model is trained with extra proposed descriptions. However, as shown in Table 4, when Scheduled Sampling is applied to L3 with Resnet18 image model, Scheduled Sampling no longer improves test accuracy. This result can be explained by the Oracle experiment. Since L3 with Resnet18 can achieves close to Oracle’s result with current proposed description, feeding in proposed description and feeding in ground-truth description is relatively equivalent, which means Scheduled Sampling approach no longer benefits L3 with Resnet18 model.

Methods	L3 batch 100	LSL batch 100	Linear Decay	Inverse Sigmoid Decay k=15
Test Accuracy	70.13	68.51	71.52	70.44

Table 3: Test accuracies comparison between Scheduled Sampling and baseline models

Methods	L3 Resnet18 batch 32	Linear Decay	Inverse Sigmoid Decay k=15
Test Accuracy	75.60	75.65	74.50

Table 4: Test accuracies comparison between Scheduled Sampling and L3 when use Resnet18 as image model

5.8 Multimodal:

We run various Multimodal functions with L3 and fixed VGG16 for 100 epochs and batch size 100. From Table 5, we can see that all Multimodal functions lead to worse result. This is counter-intuitive since providing extra information to interpretation model should only improve model’s performance. We then conduct analysis on Multimodal Sum function to get an insight on why extra information can hurt model’s performance. The analysis will be available in next section.

Methods	L3 batch 100	Sum	Element-wise Product	Projection Function	Weighted Average
Test Accuracy	70.13	66.33	64.31	69.11	66.33

Table 5: Test accuracies comparison between different Multimodal functions and L3

6 Analysis

6.1 Language Quality

To analyze language’s impact on model performance, we use BLEU score [7] to evaluate the performance of language model. BLEU score is computed between ground-truth concept descriptions and proposed descriptions from language model.

In addition to interpretation model (Section 5.4), Resnet also benefits the proposal model too based on our observation. From first two rows of Table 6, we can see that when running with batch size 32 and 50 epochs, L3 VGG16’s BLEU score is less than L3 Resnet18’s BLEU score. This indicates that with Resnet18, the L3’s proposal model can propose concept descriptions with higher qualities, which contributes to the Resnet18’s massive improvement in test accuracy.

Besides Resnet18’s effect on language quality, we also analyze the language quality of Retrieval approach. Although unable to propose concept description during testing, the retrieval approach can still retrieve concept descriptions whose quality match or surpass the ones of the proposed descriptions, illustrating the effectiveness of the simpler approach (Table 6).

Furthermore, an important observation is that although the proposed / retrieved concept descriptions rarely match the ground-truth concept descriptions exactly(Appendix A), modified L3 model can still make prediction with relatively high accuracy. This behavior proves that partially correct concept descriptions are still very useful to the interpretation model.

	Unigram	Bigram
L3 VGG16	0.677	0.399
L3 Resnet18	0.709	0.448
L3 VGG16 + L2 Retrieval	0.7223	0.4533
L3 Resnet18 + L2 Retrieval	0.7179	0.4599

Table 6: BLEU scores comparison between L3 with VGG16, L3 with Resnet18 and retrieval methods

6.2 Multimodal

As mentioned in previous section, instead of improving the performance of L3, providing extra information to interpretation model actually decreases test accuracy. We analyze Multimodal Sum function due to its interpretability to get an insight of such behavior.

In the Sum case, interpretation model is represented as Equation 9. The *score* of Multimodal Sum function is the sum of *hintScore* of original L3 model and *imageScore* of original LSL model. Multimodal Sum combines the L3-style prediction and LSL-style prediction by summing the scores. If both predictions agree, Multimodal Sum should predict with higher confidence. If two predictions differ, Multimodal Sum follows the prediction with higher confidence (larger magnitude).

However, as shown in Table 7, when applying Multimodal Sum function, Multimodal *hintScore* is quickly trained to be negative for all tasks. On the other hand, Multimodal *imageScore* is quickly trained to be positive for most of tasks. This indicates that both the model calculating image score and the model calculating hint score are not trained properly in Multimodal case, and this may be caused by the additional complexity due to extra dependency between models. The failure of training both models eventually leads to poor test accuracy that we observe.

$$f(x_q, \{x_s\}, w) = \sigma(rnnEncode(w)^T rep(x_q) + (\frac{1}{n} \sum_i rep(\{x_s\})_i)^T rep(x_q)) \quad (9)$$

epoch	20	40	60	80	100
Multimodal image score positive fraction	0.97	0.9	0.93	0.92	0.93
Multimodal hint score positive fraction	0	0	0	0	0

Table 7: Fraction of positive scores calculated by image score model and hint score model through the training process

7 Conclusion

We have analyzed language’s effect in the classification task and enhance a state-of-the-art model based on our analysis. We illustrate that accurate concept description can increase model’s accuracy on few-shot image classification task. We investigate various techniques that can potentially enhance L3 model and analyze their behaviors. We demonstrate how Resnet18 can benefit L3 model, and can be combined with Retrieval and BiLinear Scorer approaches to provide comparable result without generating concept descriptions from scratch. There are various modifications that can be made as a future step. For example, we can potentially implement an edit algorithm to edit the concept description retrieved in Concept Retrieval framework, so that the concept description can better generalize to all concepts at test time. Instead of using current L3 loss function to indirectly enforce images with similar concept to have high similarity, we can potentially implement an algorithm executed in training phase to explicitly train the retrieval model to do retrieval.

References

- [1] Jacob Andreas, Dan Klein, and Sergey Levine. Learning with latent language. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, June 2018.

- [2] Jesse Mu, Percy Liang, and Noah Goodman. Shaping visual representations with language for few-shot classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4823–4830, Online, July 2020. Association for Computational Linguistics.
- [3] Tatsunori B Hashimoto, Kelvin Guu, Yonatan Oren, and Percy S Liang. A retrieve-and-edit framework for predicting structured outputs. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.
- [4] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS’15: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, 2015.
- [5] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. 6 2014.
- [6] Alexander Kuhnle and Ann Copestake. Shapeworld - a new test methodology for multimodal language understanding. April 2017.
- [7] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. pages 311–318, July 2002.

A Appendix

Example comparison between ground-truth concept descriptions and proposed descriptions with L3 Resnet18 model.

ground truth: <sos> a yellow square is above a green circle . <eos>
sampled: <sos> a pentagon is below a yellow shape . <eos>
ground truth: <sos> a circle is below a red triangle . <eos>
sampled: <sos> a red pentagon is below a red shape . <eos>
ground truth: <sos> a red cross is below a red shape . <eos>
sampled: <sos> a red shape is to the left of a red semicircle . <eos>
ground truth: <sos> a magenta shape is below a magenta semicircle . <eos>
sampled: <sos> a magenta square is to the right of a white shape . <eos>
ground truth: <sos> a green ellipse is to the left of a rectangle . <eos>
sampled: <sos> a red shape is to the left of a green circle . <eos>