[23-2] UST Seminar

Exploring Uncertainty-aware Class-wise Thresholds for Recognition Model's Uncertainty Detection

UST-ETRI School Hwang Jihyun
(aribae@etri.re.kr)

ETRI 한국전자통신연구원 UST

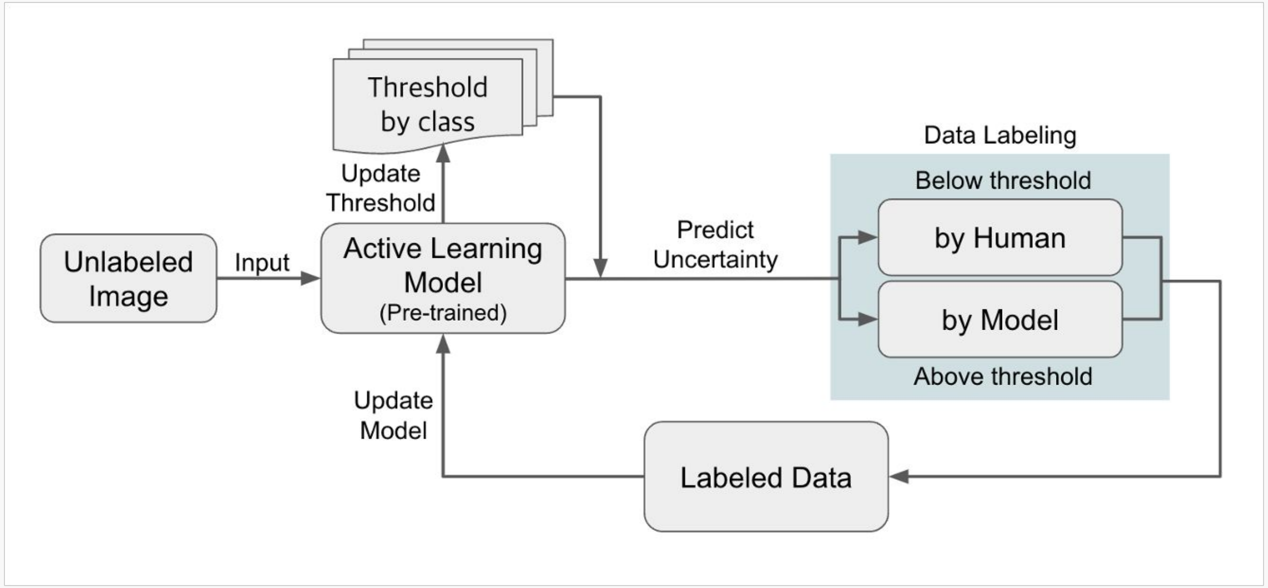
Table of contents

Recap of previous study

Recap of previous study

Setting different thresholds per class for uncertainty measurements

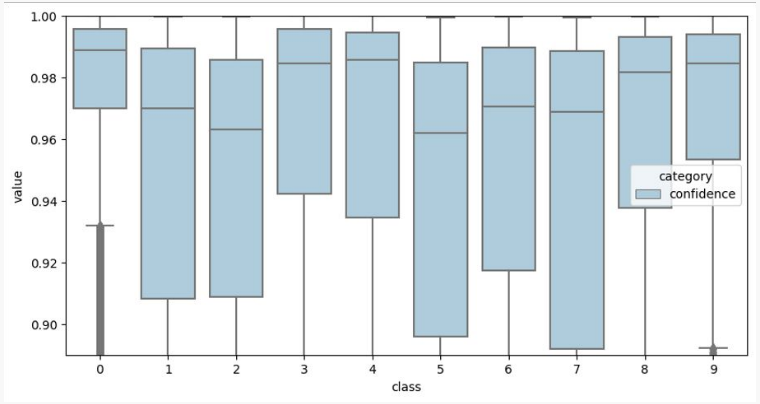
Proposed Architecture:



Recap of previous study

Results:

- Verify that different classes have different confidence distributions
- Different thresholds show improved classification performance



threshold	recall(avg)	precision(avg)
0.90	0.175	1
0.80	0.550	0.969
0.70	0.675	0.888
Q1	0.875	0.571
mean	0.700	0.671
median	0.700	0.675

Comparison of confidence distributions by class

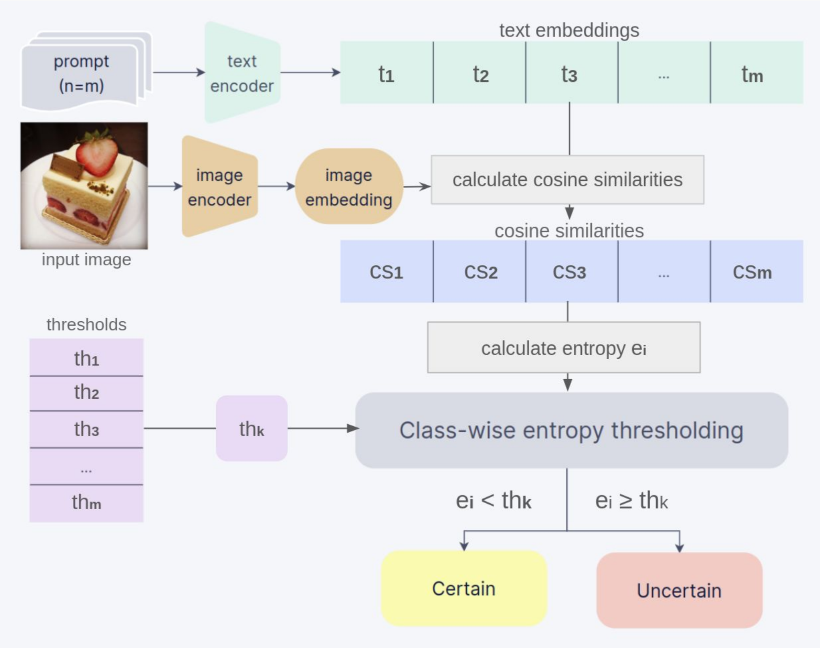
Misclassification Classification Test Results

Introduction

Introduction

Estimating uncertainty in zero-shot image classification using the vision-language model, **CLIP**.

Proposed Architecture:



Introduction

[Open AI's CLIP]

Versatility:

- CLIP is a model for understanding the relationship between images and text, which can be used for a variety of tasks. This model processes **images and text together**
- CLIP can **perform multiple tasks in a single model**, including image classification, text classification, image search, text search, and image creation.

Zero-shot learning:

- CLIP was not explicitly learned for all class labels during training. Instead, it was learned using many image-text pairs with text descriptions.
- This makes CLIP robust to **make predictions about new classes**, and is effective in transfer learning about new tasks or datasets.

Introduction

[Open AI's CLIP]

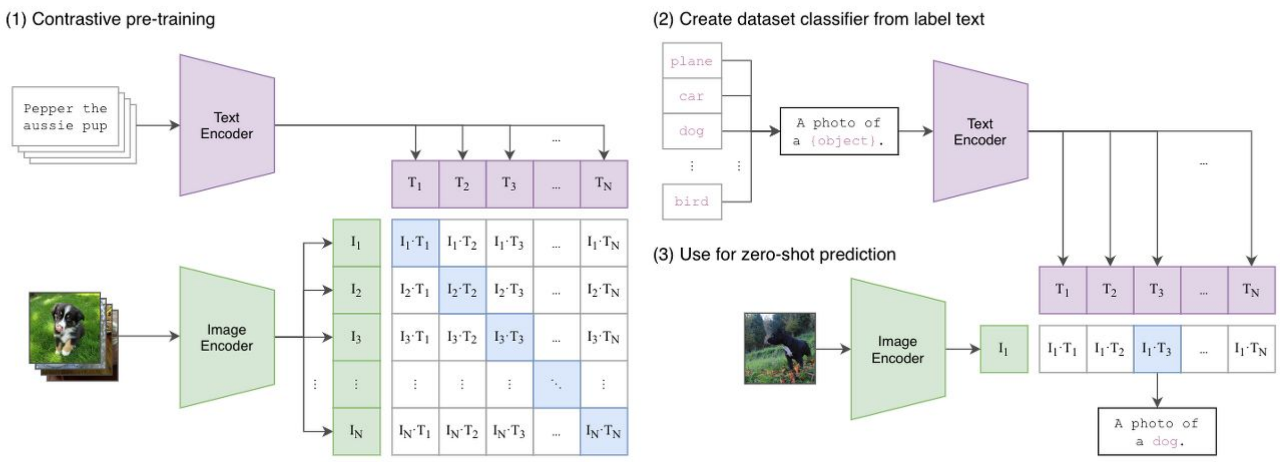


Figure 1. Summary of our approach. While standard image models jointly train an image feature extractor and a linear classifier to predict some label, CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the names or descriptions of the target dataset's classes.

Methodology

Methodology

[Calculate Thresholds using cosine similarity]

Calculate image and text embedding:

- Calculate E*img*, the image embedding for the input image img, and E*text(j)*, the text embedding for each prompt *j*.

Calculate and Normalize Cosine Similarity:

- Calculate cosine similarity s*j* between image embedding E*img* and each text embedding E*text(j)*
- The calculated cosine similarity s*j* is normalized by dividing by 2 to obtain the final similarity score sim*j*

$$s_j = \frac{E_{img} \cdot E_{text}(j)}{\|E_{img}\| \times \|E_{text}(j)\|}$$

$$sim_j = \frac{s_j + 1}{2} \quad for\ j = 1, 2, \ldots, K$$

Methodology

[Calculate Thresholds using cosine similarity]

Select and classify prompts with the highest similarity:

- Compare the normalized similarity *simj* for each prompt to select the prompt with the highest similarity
- Classifies the image into the class corresponding to the selected prompt

Estimating the uncertainty of classification:

- Calculate the entropy(H) of the similarity distribution between the input image and the prompt
- As entropy increases, so does the uncertainty of classification

$$H(img) = -\sum_{j=1}^{K} sim_j \log sim_j$$

Methodology

[Find thresholds by using grid-search method]

Grid search method:

- One of the many classical methods for optimization
- Select the best parameter combination by systematically analyzing all possible parameter combinations

(a) Entropy List Preparation

class_number = n

entropy_list = $[e_1, e_2, e_3, \ldots, e_k]$

(b) Sampling of true positive and false positive

TP = $\{e_i \mid class(e_i) = n \wedge predict(e_i) = n\}$

FP = $\{e_i \mid class(e_i) \neq n \wedge predict(e_i) = n\}$

(c) Find the optimal threshold value.

min_count = ∞

threshold = None

for e in entropy_list:

    TP′ = $\{e_i \in TP \mid e_i > e\}$

    FP′ = $\{e_i \in FP \mid e_i < e\}$

    count = card(TP′) + card(FP′)

    if count < min_count:

        min_count = count

        threshold = e

Methodology

[Set three threshold criteria]

Class-Wise Thresholds:

- Use grid search to find the optimal threshold by considering the uncertainty of each class's sample

Average of Class-Wise Thresholds (Mean of Class-Wise Thresholds):

- The average of thresholds obtained for each class

Single Threshold by Grid Search on Entire Dataset (Grid Search Single Threshold):

- A single threshold obtained by applying grid search to the predictive results of the CLIP model for the entire dataset.

Results

Results

[Misclassification detection results]

- Class-wise entropy thresholding method shows higher performance as a result of synthesizing the entire set of results.

		Dataset	
		CIFAR10	Food101
1	# of images	6000	20200
2	# of correct predictions	7926	12434
3	# of incorrect predictions	2074	7766
4	Accuracy(%)	79.206	61.554

CLIP model accuracy

Results

[Misclassification detection results]

- Class-wise entropy thresholding method shows higher performance as a result of synthesizing the entire set of results.

		Dataset	
		CIFAR10	Food101
1	Class-wise Thresholds	0.779	0.845
2	Mean of class-wise thresholds single threshold	0.456	0.367
3	Grid search single threshold	0.529	0.576

Uncertainty detection performance

Results

[Misclassification detection results]

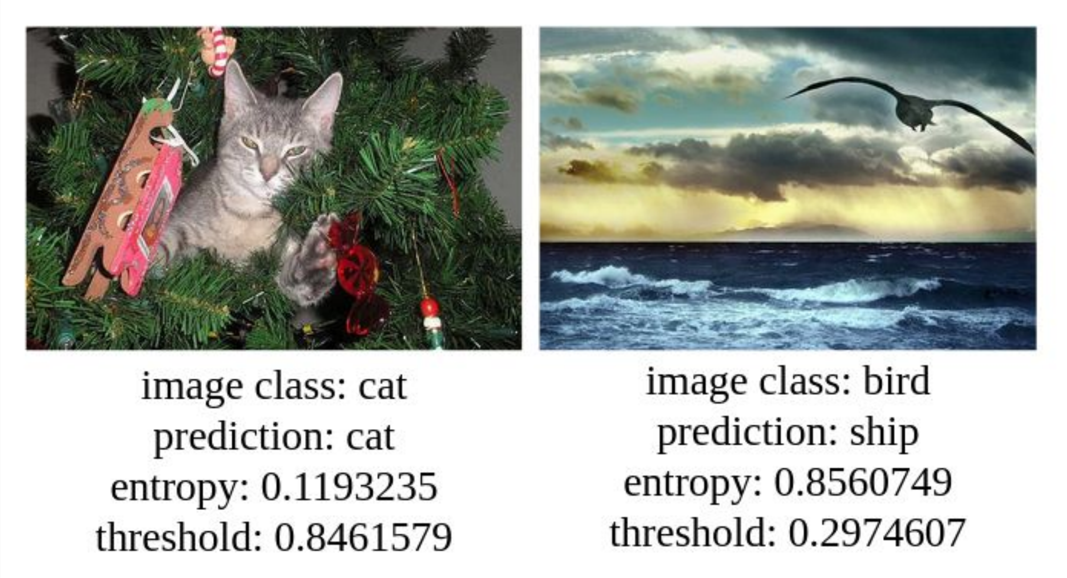
- Out-of-Distribution (OOD): The model represents an untrained data area and is used to assess predictive uncertainty for a given model.
- The results of OOD also confirmed that class-wise entropy thresholds showed the highest performance

		Dataset	
		CIFAR10	Food101
1	Class-wise Thresholds	0.933	0.894
2	Mean of class-wise thresholds single threshold	0.689	0.475
3	Grid search single threshold	0.779	0.751

Uncertainty detection performance In OOD dataset

ETRI 한국전자통신연구원 UST

Results



Uncertainty detection image sample

Conclusion

Conclusion

Transfer Learning:

- Easy to measure classification performance without additional learning

Necessity for fine tuning:

- Performance was not good on the Fine-Grained classification dataset
- No matter what model we use, we will need to learn about the domain data want to use

Thank you