Refining Visual Question Answering with Selective Prediction on ViLT

TECHNISCHE UNIVERSITÄT DARMSTADT

Group E:
Muhammed Amjad Abdushakkoor
Ansh Prakash
Harshal Bhimani

Deadline: Monday 8th July, 2024

1. Introduction

According to [1], Visual Question Answering (VQA) models take an image and a question as inputs and produce a natural language answer. VQA integrates image understanding with text comprehension, involving tasks like object detection and reasoning [1]. Several models have been developed, including Pythia, VisualBERT, and ViLT, which achieves this task.

For our work, we are focusing on Vision-and-Language Transformer (ViLT). ViLT simplifies traditional Vision-and-Language Pre-training (VLP) by treating image processing similarly to text [5]. ViLT uses a transformer as a backbone, combining textual and visual data through layers to output a feature sequence [5]. It employs patch embedding for images and standard tokenization methods like WordPiece and BPE for text.

ViLT utilizes masked language modeling (MLM) and Image Text Matching (ITM) to enhance its capabilities in understanding and correlating images and text [5]. These mechanisms enable ViLT to generate answers from given image-question pairs with high accuracy. However, it still faces an issue of being overconfident on OOD question-answer pairs. It simply doesn't know when it doesn't know.

1.1. Selective prediction on VQA

For VQA and other classification models, completely trusting their output may not be ideal as they don't provide any indicator about confidence on their predictions. They often try to find short-cut on training set. This situation is even worse when you consider how these models give a prediction out of their potential set of answers, even when its confidence in it is not good enough. There is generally no mechanism where it can say 'I am not confident enough to answer' or 'I don't know' if its not trained well enough to give a confident output for a given input.

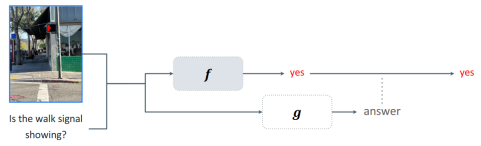
As illustrated in figure 1, selective prediction works with the help of two main functions. Firstly, there is the function f(x) which is the main model that predicts the output text, when the image and question are given as inputs. Then, there is the selection function g(x), which determines whether the model must answer or abstain from answering [8].

If g(x) let's the model answer the given question, then the answer is shown to the user like its done normally. However, if the function makes the model abstain from answering, the prediction from the model is covered and an output such as 'I don't know' or 'Not confident enough to answer' is shown to the user.

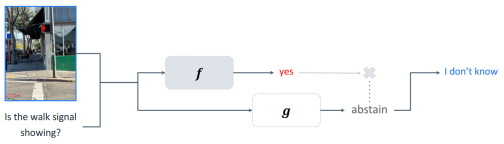
1.2. Selection with Guaranteed Risk Control

A selective classifier is a pair (f, g), where f is a classifier, and g : X → 0, 1 is a selection function, which serves as a binary qualifier for f as follows,

$$(f, g)(x) \triangleq \begin{cases} f(x), & \text{if } g(x) = 1; \\ \text{don't know}, & \text{if } g(x) = 0. \end{cases}$$



(a) When the selection function is confident



(b) When the selection function abstains

Figure 1: Working of selective prediction [7]

Thus, the selective classifier abstains from prediction at a point x if and only if $g(x) = 0$. The performance of a selective classifier is quantified using coverage and risk. Fixing P, coverage, defined to be $\phi(f, g) = \mathbb{E}_P[g(x)]$ is the probability mass of the non-rejected region in X [8]. The selective risk of (f, g) is

$$R(f, g) = \mathbb{E}_P[\ell(f(x), y)g(x)]/\phi(f, g).$$

We are given a classifier f, a training sample S_m, a confidence parameter $\delta > 0$, and a desired risk target $r^* > 0$. Our goal is to use S_m to create a selection function g such that the selective risk of (f, g) satisfies

$$\mathbb{P}_{S_m}\{R(f, g) > r^*\} < \delta, \quad (2)$$

where the probability is over training samples S_m sampled i.i.d. from the unknown underlying distribution P. Among all classifiers satisfying , the best ones are those that maximize the coverage.

Yonatan et al. introduced this algorithm to find solution for the above eq (2) using an additional confidence function. In their work, they mentioned a confidence-rate function $\kappa_f : X \to \mathbb{R}_+$ for f. The idea is that if $\kappa_f(x_1) >= \kappa_f(x_2)$ then f is more confident about x_1 then x_2

So, given a confidence parameter $\delta > 0$, and a desired risk target $r^* > 0$. Based on the training set, our goal is to learn a selection function g such that the selective risk of the classifier (f, g) satisfies (2).

For $\theta > 0$, the original authors' defined the selection function $g_\theta : X \to \{0, 1\}$ as

$$g_\theta(x) = g_\theta(x \mid \kappa_f) \triangleq \begin{cases} 1, & \text{if } \kappa_f(x) \geq \theta; \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

Additionally, *empirical selective risk* with respect to the labeled sample S_m is defined as follow:

$$\widehat{R}(f, g \mid S_m) \triangleq \frac{\frac{1}{m}\sum_{i=1}^{m} \ell(f(x_i), y_i).g(x_i)}{\widehat{\phi}(f, g \mid S_m)}$$

where $\widehat{\phi}$ is the empirical coverage, $\widehat{\phi}(f, g \mid S_m) = \frac{1}{m}\sum_{i=1}^{m} g(x_i)$.

Here, our objective is to minimize *empirical selective risk* and at the same time try to have higher empirical coverage. SGR Algorithm used in our work could be seen in 1. Connecting back to (3), 1 will get us the θ value for the desired risk.

1.2.1. Confidence function candidates

There are different type of confidence functions that can be used for the purpose of SGR. Some of them are listed below.

- **MaxProb**: This is the most basic kind of selection function and it uses the softmax probability of the prediction. Here, the idea is that the higher the value of the softmax, the more confident the model is about its predicted class. Therefore, this probability score can be compared with a certain threshold value to understand if the prediction made is strong enough to be shown [8]. If the probability score is greater than the threshold, it is shown to the user, or else the model is abstained from answering.
- **Platt Scaling**: The Platt scaling technique is only used for binary classifiers and is a technique to adjust the output of a classifier so it can give better probability estimates. Here, logits produced by the model are converted into probability scores and they are used in a simple logistic regression model to produce two variables a and b after its linear transformation. These variables are used to calibrate the logits and they are passed to a sigmoid function to produce the new probability scores [4][6].
- **Calibration with Matrix scaling**: Matrix scaling can be considered as a multi-class classifier version of Platt scaling as a matrix W is being used to calibrate the logits of the model. Therefore, instead of a linear regression based tranformation, the transformation yields: Wzi + b. Then softmax is applied to get the probabilities [4].

Algorithm 1 Selection with Guaranteed Risk (SGR)

procedure SGR($f, \kappa_f, \delta, r^*, S_m$)
  Sort S_m according to $\kappa_f(x_i)$ for $x_i \in S_m$ (and assume w.l.o.g. that indices reflect this ordering).
  $z_{\min} = 1$; $z_{\max} = m$
  for $i = 1$ to $k = \lceil \log_2 m \rceil$ **do**
    $z = \left\lfloor \frac{z_{\min} + z_{\max}}{2} \right\rfloor$
    $\theta = \kappa_f(x_z)$
    $g_i = g_\theta$
    $\widehat{r}_i = \widehat{r}(f, g_i \mid S_m)$ ▷ Estimate risk
    $b_i^* = B^*(\widehat{r}_i, \delta / \lceil \log_2 m \rceil, g_i(S_m))$ ▷ See Lemma 3.1 [3]
    if $b_i^* < r^*$ **then**
      $z_{\max} = z$
    else
      $z_{\min} = z$
    end if
  end for
  Output (f, g_k) and the bound b_k^*.
end procedure

- **Calibration with Vector scaling**: Vector scaling is a simpler version of matrix scaling where the matrix W is restricted to be diagonal. This means each class's logit is only scaled by a specific factor and shifted by a specific amount without mixing information between different classes[4].

We are going to analyses on 'Maxprob' and 'Calibration with Vector scaling' with SGR 1. Our expectation is that calibrated model should perform better as it reflects the model prediction confidence in sync with the reality.

2. Related Work

- **Reliable Visual Question Answering: Abstain Rather Than Answer Incorrectly**: The paper [8] introduces abstention techniques to enhance VQA reliability by avoiding incorrect answers. The authors develop methods to improve both the accuracy and coverage of abstentions and propose the Effective Reliability metric for better VQA performance evaluation. This approach allows VQA models to maintain low error rates while increasing the number of questions they can confidently answer, setting a foundation for more reliable VQA systems.

- **On Calibration of Modern Neural Networks**: In [4], the concept of confidence calibration which is the accuracy of probability estimates in classification models are being investigated. The authors find that modern neural networks are poorly calibrated compared to older models and identify factors such as depth, width, weight decay, and Batch Normalization as influential. Through extensive experiments, they evaluate various post-processing calibration methods and determine that temperature scaling is highly effective for most datasets, providing practical guidance for improving calibration in neural networks.

- **Improving Selective Visual Question Answering by Learning from Your Peers**: The paper [2] enhances the ability of VQA models to abstain from answering when uncertain, especially with out-of-distribution (OOD) inputs. The proposed Learning from Your Peers approach trains selection functions using model predictions from distinct data subsets. This method significantly improves performance, achieving 32.92

3. Main Contribution

We will analyze selective prediction strategies on a ViLT model. Specifically the vilt-b32-finetuned-vqa model from dandelin will be used. This is a fine tuned model that was trained on the VQAv2 dataset and was introduced in [5].

Moreover, we will specifically going into the testing of selective prediction using MaxProb, and calibration by vector scaling selection functions. The results from experimenting with the various techniques of selective prediction will be available in section 5.

Our main contributions include:

- Applying Selection with Guaranteed Risk (SGR) Algorithm [3] in multi-modal setting
- Experimenting with Calibrated logits in addition to MaxProb

4. Experimental Setup

Dataset: We use the VQAv2 dataset, which contains around 3,000 classes. Visual Question Answering (VQA) v2.0 is a dataset with open-ended questions about images that require an understanding of vision, language, and commonsense knowledge to answer.

We randomly divide the validation set into two equal halves: one for training and the other for testing risk and coverage.

Correct Label Choice: Since VQAv2 has 10 labels from different humans, we consider the label occurring the maximum number of times as the correct label. Our results and risk measures are based on this method of determining the correct label.

Model: We use the ViLT model fine-tuned on VQAv2.

Objective: We aim to enable the model to abstain when it is not confident about its prediction while maximizing coverage. We employ the Selection with Guaranteed Risk (SGR) algorithm introduced by Yonatan et al. [3] and investigate whether *calibrated logits perform better than vanilla logits*.

Yonatan et al. used the SGR algorithm in a unimodal setting with Maxprob as the confidence function. We extend this concept to a multimodal setup and experiment with calibrated logits using a vector-scaling method. Temperature scaling was not used as it does not change the order of softmax probabilities and is thus equivalent to the Maxprob method.

To find θ for the correct risk bound, we will apply the SGR algorithm 1.

Flow:

1. Generate logits from the ViLT model using the validation dataset.
2. Train a vector-scaling calibration model using these logits.
3. Treat both the vanilla logits and the calibrated logits (after applying softmax and taking the maximum value) as confidence values.
4. Create a list of confidence values and a boolean list where incorrect predictions are marked as 1.
5. Feed these lists into the SGR algorithm 1.
6. Use the generated θ in equation (3) to create a rejection function.
7. Determine θ for various risk values.

5. Results and Analysis

Observing the results in Table 1, we see that the risk bound, b^*, is always very close to the target risk, r^*, for the Maxprob method. However, this is not the case with the vector-scaling method, which fails to find a reasonable risk bound at 2% in Table 2.

Additionally, at 100% desired risk, we reach the empirical risk without selective prediction.

We notice that coverage increases significantly as the desired risk increases. Surprisingly, vector-scaling calibration performs worse than vanilla logits according to our results. Nonetheless, our coverage values seem to be better than some of the current state-of-the-art work, such as that by Whitehead et al.

Overall, we find that calibrated logits do not perform as well as vanilla logits, proving our hypothesis incorrect.

desired risk	train risk	train coverage	test_risk	testcov	bounded risk
0.02	0.0173	0.3624	0.0177	0.3649	0.0200
0.10	0.0957	0.6810	0.0980	0.6810	0.1000
0.15	0.1453	0.8013	0.1420	0.8002	0.1500
0.20	0.1950	0.9133	0.1940	0.9125	0.2000
0.25	0.2448	0.9996	0.2432	0.9996	0.2500
0.30	0.2444	1.0000	0.2442	1.0000	0.2496
1.00	0.2440	1.0000	0.2446	1.0000	0.2491

Table 1: Risk control results with Maxprob for VQAv2 for $\delta = 0.001$

desired risk	train risk	train coverage	test_risk	testcov	bounded risk
0.02	0.0000	0.0000	0.0303	0.4585	0.9999
0.10	0.0957	0.6785	0.0961	0.6796	0.1000
0.15	0.1452	0.7909	0.1436	0.7905	0.1500
0.20	0.1950	0.8976	0.1937	0.8975	0.2000
0.25	0.2448	0.9996	0.2445	0.9997	0.2500
0.30	0.2446	1.0000	0.2453	1.0000	0.2497
1.00	0.2461	1.0000	0.2437	1.0000	0.2513

Table 2: Risk control results with vector-scaling calibration for VQAv2 for $\delta = 0.001$

6. Conclusions and Limitations

We found that vector-scaling with SGR didn't provide any significant improvement over Maxprob as confidence-rate function. In extreme case, SGR algorithm failed to find a reasonable bounded risk for calibrated logits. Moreover, the biggest limitation of our method is that we are trying to get confidence value from model itself. In research, it has been shown that even after applying calibration, models are still overconfident on OOD data. In future work, we should disentangle the method of getting confidence value from within the classifier itself.

6.0.1. Our Learnings

We have learned :

1. Calibrating models with vector-scaling, temperature-scaling, and matrix-scaling
2. Found a way find a r^*, such that it tries to balance risk and coverage
3. Dissecting research paper and finding what could be relevant to solve a problem
4. Coding practices required in a team

References

[1] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Batra, and D. Parikh. Vqa: Visual question answering, 2016.

[2] C. Dancette, S. Whitehead, R. Maheshwary, R. Vedantam, S. Scherer, X. Chen, M. Cord, and M. Rohrbach. Improving selective visual question answering by learning from your peers, 2023.

[3] Y. Geifman and R. El-Yaniv. Selective classification for deep neural networks, 2017.

[4] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks, 2017.

[5] W. Kim, B. Son, and I. Kim. Vilt: Vision-and-language transformer without convolution or region supervision, 2021.

[6] J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. Large Margin Classif.*, 10, 06 2000.

[7] A. Ruhrbach and M. Ruhrbach. Lecture 11: Quantification. 2024.

[8] S. Whitehead, S. Petryk, V. Shakib, J. Gonzalez, T. Darrell, A. Rohrbach, and M. Rohrbach. Reliable visual question answering: Abstain rather than answer incorrectly, 2022.

A. GitHub Project Info (mandatory)

GitHub Link: GitHub

- **Muhammed Amjad Abdushakkoor** - Amjad-MA5
- **Ansh Prakash** - AnshPrakash
- **Harshal Bhimani** - harshal912

B. Individual Contributions (mandatory)

Muhammed Amjad has worked on the testing of various VQA models which led to the choosing of the model used in this paper. Also, he has developed the initial maxprob based selective prediction part of the topic. Moreover, he was tasked with writing most parts of the report.

Harshal Bhimani has worked on the development and testing of the maxprob and risk bound methodologies, ensuring the code structure for predictions, risk bound, and batch processing was robust. After confirming the model's workflow and outputs, he integrated batch processing and the risk bound components.

Ansh Prakash developed the hypothesis and implemented various critical modules. He generated the final experimental results presented in our report. Furthermore, he worked on various parts of the report. Additionally, he managed the team.

C. Additional Content (optional)

Here, you can include additional details, qualitative examples, etc., which you do not wish to include in the main report.