LogicNet: A Logical Consistency Embedded Face Attribute Learning Network

Haiyu Wu[1], Sicong Tian[2], Huayu Li[3], Kevin W. Bowyer[1]
[1]University of Notre Dame
[2]Indiana University South Bend
[3]University of Arizona

1. Logical Consistency Evaluation in Real World

To evaluate the performance of logical consistency of predictions in the real-world case, we use the subset of WebFace260M, which contains 603,910 images, as a test set. Since there are no ground truth labels, we only measure the ratio of failed (logically inconsistent) predictions for each method.

Table 1 shows that without the post-processing step, the logical-inconsistency-ignored methods have failed ratios $\{51.43\%$ and $71.53\}$ on FH37K and FH41K, the logical-consistency-aware methods have the failed ratios $\{39.15\%, 40.27\%\}$, the proposed method significantly reduces the number of failed cases, with a failed ratio, $\{25.47\%, 24.36\%\}$, that is less than half of the average failed ratio of no-logic-involved methods and $> 13\%$ lower than logic-involved methods. BF trained with FH41K predicts too many negative labels which causes the outlier ratio, 97%. When we implement the post-processing strategy, all the incomplete cases are gone, which results in a low failed ratio for all methods other than BCE-MOON. This supports the speculation in the main paper, where BCE-MOON over-focuses on positive side and existing methods can somewhat learn the pattern but need to involve post-processing steps.

Table 2 shows the logical consistency test results on CelebA attributes. Since incomplete prediction is not the case for CelebA attributes, only the number of impossible predictions is used to measure the failed ratio. The results show that, except BCE-MOON, all the other methods have $< 2\%$ failed ratio. To dig out the rationale, we also calculate the mean and standard deviation of the number of positive predictions for the attributes that have strong logical relationships. All the methods, other than BCE-MOON, have less than 2 positive predictions on average. This number of positive predictions has limited probability to cause the disobedience of logical consistency. Moreover, the images in WebFace260M are 112x112, which is different from the resolution of the training images of any methods. Therefore, this test result is **not considered** in this paper and **will not be included** in the future work.

2. Comparing with GPT4-V

We also tested the performance of GPT4-V [1], the best Vision-Language model, on these three datasets. Since GPT4-V API has a limitation of monthly spend, all the FH37K and FH41K images and 1,000 randomly sampled CelebA images are tested. Note that the super-resolution option in GPT4-V is not used in this experiment. We tried two types of response formats: 1) returning the binary version attribute predictions of the images, 2) returning the attribute names of the positive predictions. Format 1 works better for FH37K and FH41K and format 2 works better for CelebA. The average accuracy on FH37K, FH41K, and CelebA are $\{51.86\%, 51.61\%, 56.84\%\}$ without checking logical consistency, and $\{48.10\%, 49.29\%, 54.05\%\}$ with checking logical consistency of predictions. The prompt we used for getting the predictions is in the Code 1 and 2. This low performance could be caused by several factors - 1) Accuracy measurement: reporting the average accuracy of positive and negative predictions evenly reveals the uneven performance. For example, in the traditional way, GPT4-V will has 80.62% accuracy, but the accuracy on the positive side is only 18.62%. 2) Attribute ambiguity: there is no documentation guiding GPT4-V to better understand the logical relationships underneath, which makes it hard to make correct/logical predictions. 3) Image resolution: the original images are in 112x112 but GPT4-V needs 512x512 images, so the performance drops significantly. Consequently, the current Vision-Language models cannot handle this challenge well.

Methods	FH37K			FH41K		
	N_{incomp}	N_{imp}	R_{failed}	N_{incomp}	N_{imp}	R_{failed}
With label compensation.						
BCE	0	11,134	1.84	0	7,464	1.24
BCE-MOON	0	330,115	54.66	0	341,114	56.48
BF	0	14,007	2.32	0	3,530	**0.58**
Semantic	0	36,372	6.02	0	44,803	7.42
Constrained	0	38,971	6.45	0	46,601	7.72
LCP	0	5,595	**0.93**	0	5,788	0.96
Ours	0	21,731	3.60	0	19,194	3.18
Without label compensation. (**A general solution**)						
BCE	240,761	6,001	40.86	352,061	585	58.39
BCE-MOON	31,512	313,044	57.05	34,415	321,872	59.00
BF	339,136	1,295	56.37	587,056	0	97.21
Semantic	185,824	21,040	34.25	227,558	21,482	41.24
Constrained	171,319	23,255	32.22	203,063	26,390	37.99
LCP	307,576	300	50.98	248,768	2,416	41.59
Ours	139,184	14,660	**25.47**	133,245	13,838	**24.36**

Table 1. Logical consistency test on predictions. The models are trained with FH37K (left) and FH41K (right). N_{incomp}, N_{imp}, and R_{failed} are the number of incomplete predictions, the number of impossible predictions, and failed ratio. [Keys: **Best**, $> 50\%$, Logic involved methods]

Methods	Considering logical consistency			
	N_{imp}	R_{failed}	N^p_{mean}	$N^p_{std.}$
AFFACT	5,038	0.83	1.65	$\pm$0.68
ALM	85	**0.01**	1.58	$\pm$0.5
BCE	8,723	1.44	1.53	$\pm$0.65
BCE-MOON	167,261	**27.70**	2.41	$\pm$1.09
BF	6,897	1.14	1.22	$\pm$0.45
Semantic	4,187	0.69	1.62	$\pm$0.64
Constrained	4,703	0.78	1.76	$\pm$0.68
BCE+LCP	3,008	0.5	1.81	$\pm$0.62
Ours	5,541	0.92	1.62	$\pm$0.64

Table 2. Logical consistency test of the algorithms trained with CelebA-logic. N_{incomp}, R_{failed}, N^p_{mean}, $N^p_{std.}$ are the number of incomplete predictions, failed ratio, the average and standard deviation number of positive predictions on 9 attributes that have strong logical relationships. [Keys: **Best**, **Worst**, Logic involved methods]

3. Others

Algorithm 1 and 2 are the logical rules used to detect the logically inconsistent predictions. We adopt the rules from [2] for FH37K and FH41K. These rules are used in both accuracy measurement and real-world experiments. Figure 1 shows the attributes in the CelebA dataset that have weak logical relationships, and are independent from the other attributes (e.g., Mouth Slightly Open, Wearing Earrings, Wearing Necktie) or have ambiguous definitions (e.g., Attractive, Oval Face, Blurry).

Algorithm 1 FH37K/41K Failed prediction detection

Attribute groups (Category: $\textbf{List}_{\textbf{attr}}$)

Beard areas: Clean Shaven, Chin Area, Side to Side, Info not Vis
Beard lengths: 5 O'clock Shadow, Median, Long, Info not Vis
Mustache: None, Isolated, Connected-to-beard, Info not Vis
Sideburns: None, Present, Connected-to-beard, Info not Vis
Bald: False, Top only, Sides only, Top and Sides, Info not Vis

Fail conditions

Mutually exclusive:

1. More than one positive prediction in Beard areas (except Info not Vis), Beard lengths (except Info not Vis), Mustache, Sideburns, Bald group
2. Clean Shaven + any of Beard lengths/Mustache Connected-to-beard/Sideburns Connected-to-beard
3. Chin area + Sideburns Connected-to-beard
4. Bald (Top and Sides or Sides only) + having sideburns (Sideburns Present, Sideburns Connected-to-beard)

Dependency:

1. Having beard (Chin Area, Side to Side) + one of the beard lengths must be true
2. Mustache is connected to beard + ¬(Chin Area, Side to Side)
3. Sideburns is connected to beard + ¬Side to Side

Collectively exhaustive

No positive prediction in Beard area/Beard lengths/Mustache/Sideburns/Bald

Algorithm 2 CelebA Failed prediction detection

Attribute Groups (attr: $\textbf{List}_{attr}$)

No Beard: 5 O'clock Shadow, Goatee, Mustache
¬Male: 5 O'clock Shadow, Goatee, Mustache, ¬No Beard
Bangs: Receding Hairline
Bald: Receding Hairline, Bangs, Wearing Hat

Logic rules

Mutually exclusive: $((attr \wedge \neg List_{attr}) \vee (\neg attr \wedge List_{attr}))$

Fail conditions: $attr \wedge List_{attr}$

Listing 1 Prompt for getting CelebA predictions.

```
content = [
    {
        "type": "text",
        "text": f"""Give the predictions of [5_o_Clock_Shadow,Arched_Eyebrows,Attractive,Bags_Under_Eyes,Bald,Bangs,Big_Lips,
            Big_Nose,Black_Hair,Blond_Hair,Blurry,Brown_Hair,Bushy_Eyebrows,Chubby,Double_Chin,Eyeglasses,
            Goatee,Gray_Hair,Heavy_Makeup,High_Cheekbones,Male,Mouth_Slightly_Open,Mustache,
            Narrow_Eyes,No_Beard,Oval_Face,Pale_Skin,Pointy_Nose,Receding_Hairline,Rosy_Cheeks,
            Sideburns,Smiling,Straight_Hair,Wavy_Hair,Wearing_Earrings,Wearing_Hat,Wearing_Lipstick,
            Wearing_Necklace,Wearing_Necktie,Young] for each image. Try your best to predict.
            You should give me the positive predictions all the other attributes will be considered as negative.
            Output format: 'image name - No_Beard,Wavy_Hair,...,Bald' for each image.
            Self-check:
            1) make sure you consider the logical relationship.
            Try you best to describe the attributes! Images could be low resolution.
            DO NOT SHOW ANY UNNECESSARY MESSAGES!
            -------------------------------------------
            image list: {image_names[start:start + num]}"""
    },
]
```

Listing 2 Prompt for getting FH37K/41K predictions.

```
content = [
    {
        "type": "text",
        "text": f"""Give the predictions of
        [Clean_shaven, Chin_area, Side_to_side, Beard_area-Info_not_vis,
            Bread_length-5_o_clock_shadow, Bread_length-Short, Bread_length-Medium,
            Bread_length-Long, Bread_length-Info_not_vis, Mustache-None, Mustache-Isolated, Mustache-Connected_to_beard,
            Mustache-Info_not_vis, Sideburns-None, Sideburns_present, Sideburns-Connected_to_beard,
            Sideburns-Info_not_vis, Bald-FALSE, Bald-Top_only, Bald-Top_and_sides, Bald-Sides_only,
            Bald-Info_not_vis] for each image. Try your best to predict.
            to describe each image with True or False. "
           "You can just give the results (True/False) but should be in the correct order and use single space "
           "to separate each output."
            Output format: 'image name - No_Beard,Wavy_Hair,...,Bald' for each image.
            Self-check:
            1) make sure you consider the logical relationship.
            Try you best to describe the attributes! Images could be low resolution.
            DO NOT SHOW ANY UNNECESSARY MESSAGES!
            -------------------------------------------
            image list: {image_names[start:start + num]}"""
    },
]
```

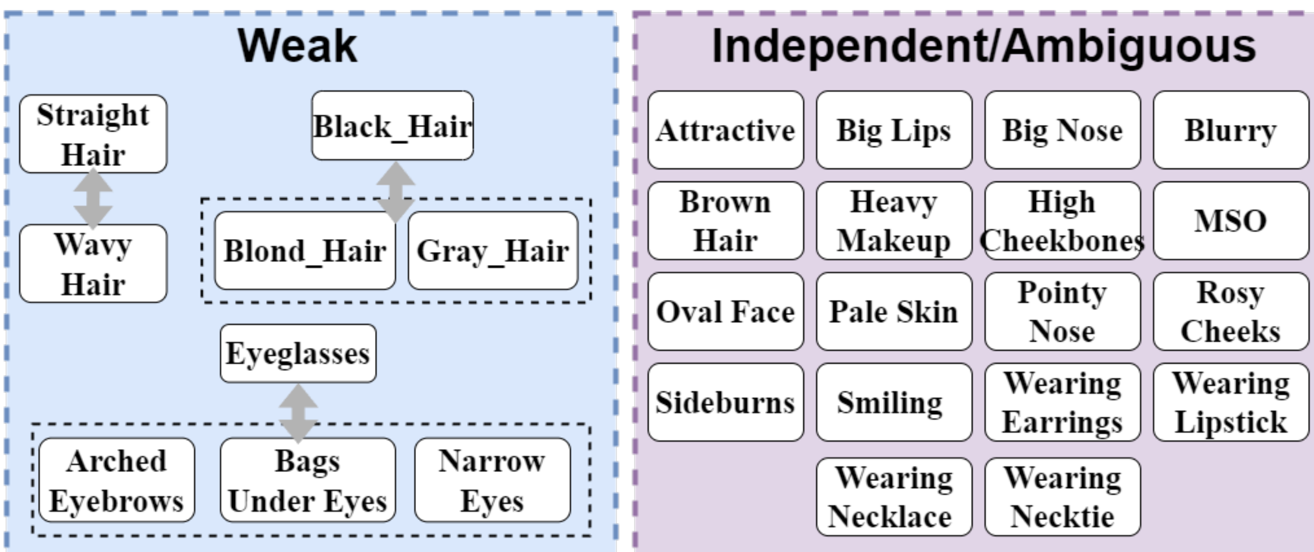

Figure 1. Weak and independent/ambiguous attributes in CelebA.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1

[2] Haiyu Wu, Grace Bezold, Aman Bhatta, and Kevin W. Bowyer. Logical consistency and greater descriptive power for facial hair attribute learning. In *CVPR*, pages 8588–8597, 2023. 2