Probabilistic Concept Graph Reasoning for Multimodal Misinformation Detection

Supplementary Material

A. Algorithm for Concept Growth

Algorithm 1 summarizes the alternating procedure and stops early after two non-improving rounds (typically $\#rounds \leq 6$).

Algorithm 1 Alternating Concept Growth (per round)

1: **Input:** concept graph $(\mathcal{C}, \{\mathcal{L}_r\})$, encoders, train set Tr, validation set Va
2: Propose up to 5 new concepts via LLM prompting (§4.1)
3: Initialize new edges $\mathcal{L}_r \to \mathcal{L}_{r+1}$ via Eq. 3 and Eq. 4
4: **Warm-up:** train concept classifier for 2 epochs; freeze previous layers
5: **Joint training:** unfreeze current and preceding layers for 6 epochs
6: **Validation check:** retain round only if $\Delta\text{NLL} \geq \eta$ and ΔAUC is significant
7: Consolidate valid concepts (3 epochs), save checkpoint
8: **Return:** concept graph $(\mathcal{C}, \{\mathcal{L}_{r+1}\})$

B. Experiment Baselines

General MLLMs: We use the following general purpose MLLMs to do zero-shot misinformation detection in our paper.

- **LLaVA-1.6** [20]: A large multimodal model that combines a vision encoder and Vicuna for general-purpose visual and language understanding.
- **Qwen3-omni** [47]: A large multimodal model capable of processing multiple modalities, and generating real-time text or speech response.
- **Llama 4** [26]: A natively multimodal that enable text and multimodal experiences.
- **Gemini 2.5 Pro** [4]: A multimodal reasoning model aims to solve complex problems.
- **GPT-5** [28]: A deeper reasoning multimodal model for harder problems such as writing, research, analysis, and so on.

Multimodal Detectors:

- **MPFN** [12]: A multimodal detector captures and fuses both shallow and deep level information text and images for misinformation detection.
- **BMR** [49]: A multimodal detector that bootstrap each modality's representation by intial predictions for single modality to get the final misinformation detection result.
- **HAMMER** [35]: A multimodal detector utilizes the fine-grained interaction between different modalities for mistinformation detection.
- **MiRAGe** [9]: A simple multimodal detector that fuses text- and image-level concept bottleneck models' results to combat the spread of AI-generated misinformation.
- **FKA-Owl** [22]: A multmodal detector that leverages forgery-specific knowledge to augment MLLMs to reason about manipulations.
- **GLPN-LLM** [7]: A multimodal detector that integrates LLM capabilities via label propagation techniques to enhance prediction accuracy.
- **C3N** [32]: A multimodal detector that captures learnable patterns of cross-modal content correlations to facilitate news classification.
- **MGCA** [6]: A multimodal detector that aligns multi-granularity clues for both misinformation and distortion detection.

C. Implementation Details

We implement PCGR with PyTorch [30] and trained it on four NVIDIA A100 GPUs. Each dataset was split into 70% training, 20% validation, and 10% testing. The learning rate was set to 1×10^{-4} and batch size to 32. Concept generation was performed using the GPT-5 API. All parameters were optimized using the Adam optimizer [3], and training stopped upon convergence or after 150 epochs. For automatic concept growth, MLLM prompting occurs only during training and is never

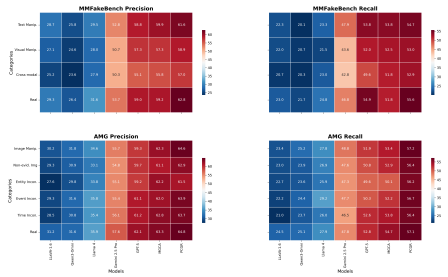

Figure 7. Precision and recall heatmap for fine-grained level detector results on MMFakeBench and AMG datasets. Score magnitudes are chromatically encoded, ranging from blue (lowest) to red (highest).

used at inference. And each round queries only the previous round's mistake log, which shrinks over rounds. And Table 3 shows PCGR has comparable training/inference time to MGCA on 4×A100 GPUs.

Table 3. Training and inference time of MGCA and PCGR.

Method	Training time			Inference time		
	MiRAGeNews	MMFakeBench	AMG	MiRAGeNews	MMFakeBench	AMG
MGCA	193mins	300mins	180mins	20mins	33mins	18mins
PCGR	**220mins**	**350mins**	**178mins**	**20mins**	**32mins**	**20mins**

D. Fine-grained Detection Results

Figure 7 shows the precision and recall for each subcategory in MMFakeBench and AMG datasets. We can observe that PCGR beats all baselines across two datasets, demonstrating the superior performance of our model for fine-grained classification tasks.

E. Sensitivity Study

We conduct a sensitivity study to investigate the impact of the maximum number of concepts per layer. Specifically, we evaluate the F1 and accuracy scores of our PCGR model on the MiRAGeNews, MMFakeBench, and AMG datasets as the concept capacity increases. As illustrated in Figure 8, we observe that: (1) increasing the concept budget initially yields performance gains across all

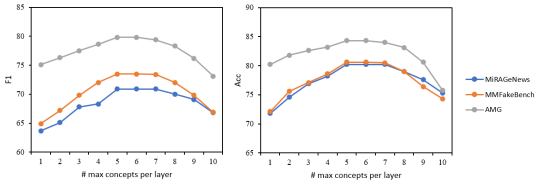

Figure 8. Impact of the max concepts constraint per layer.

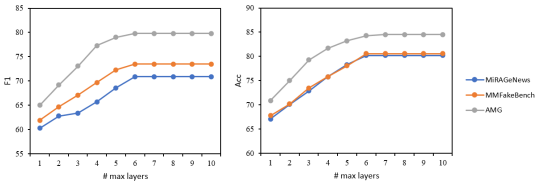

Figure 9. Impact of the max layers constraint (hierarchical depth).

three datasets; (2) performance reaches an optimum when the maximum concept constraint is set to 5; and (3) performance degrades when the constraint exceeds 5.

To get an intuitive understanding of the hierarchical depth's impact, we evaluate PCGR's performance across varying numbers of concept layers. The F1 and Accuracy trajectories for the MiRAGeNews, MMFakeBench, and AMG datasets are illustrated in Figure 9. Our study reveals that: (1) PCGR's performance initially correlates positively with an increase in the maximum layer constraint; and (2) our model's performance saturates as the constraint reaches 6 layers, indicating that PCGR effectively captures necessary abstractions with moderate depth, eliminating the need for excessively deep concept graphs.