BiMind: A Dual-Head Reasoning Model with Attention-Geometry Adapter for Incorrect Information Detection

Zhongxing Zhang[1,*], **Emily K. Vraga**[1], **Jisu Huh**[1], **Jaideep Srivastava**[1,*]
[1]University of Minnesota, Twin Cities
{zhan8889, ekvraga, jhuh, srivasta}@umn.edu

Abstract

Incorrect information poses significant challenges by disrupting content veracity and integrity, yet most detection approaches struggle to jointly balance textual content verification with external knowledge modification under collapsed attention geometries. To address this issue, we propose a dual-head reasoning framework, **BiMind**, which disentangles *content-internal reasoning* from *knowledge-augmented reasoning*. In BiMind, we introduce three core innovations: (i) an **attention geometry adapter** that reshapes attention logits via token-conditioned offsets and mitigates attention collapse; (ii) a **self-retrieval knowledge mechanism**, which constructs an in-domain semantic memory through kNN retrieval and injects retrieved neighbors via feature-wise linear modulation; (iii) the **uncertainty-aware fusion strategies**, including entropy-gated fusion and a trainable agreement head, stabilized by a symmetric Kullback-Leibler agreement regularizer. To quantify the knowledge contributions, we define a novel metric, **Value-of-eXperience (VoX)**, to measure instance-wise logit gains from knowledge-augmented reasoning. Experiment results on public datasets demonstrate that our BiMind model outperforms advanced detection approaches and provides interpretable diagnostics on when and why knowledge matters. Our BiMind model and the tested datasets are available at https://github.com/cvzh/BiMind.

1 Introduction

Nowadays, with the rapid rise of social media platforms, such as X (Twitter), Instagram, and TikTok, an increasing number of individuals or communities rely on these online platforms for communication, information dissemination, and education, especially during the pandemic (Tsao et al., 2021). Though the conveniences brought by social media,

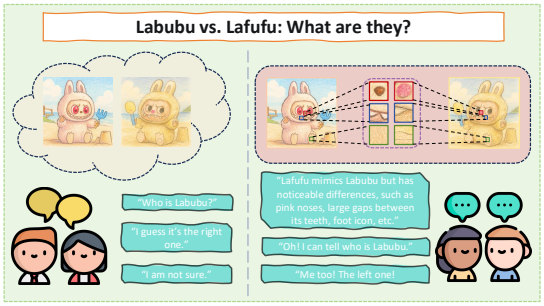

Figure 1: Illustrative case of self-correction via knowledge: without knowledge (left), the content-internal head labels Lafufu (counterfeit Labubu) as Labubu; with knowledge (right), the knowledge-augmented head corrects the label as Lafufu.

the content correctness (i.e., factual accuracy and alignment with verifiable evidence) of information disseminated still falls short of media standards and social expectations, compared to traditional media platforms, e.g., television and newspapers (Shu et al., 2017; Zhou and Zafarani, 2020). A large volume of unverified or distorted content is easily produced and propagated through social media platforms (Ahmed et al., 2022). Given that such incorrect information (e.g., spam (Wang et al., 2016), rumor (Bian et al., 2020), etc.) has significant negative impacts on individuals and society, such as social trust and information credibility (Thorson et al., 2010; Bhattarai et al., 2021; Mazzeo et al., 2021; Lan et al., 2025), addressing incorrect information propagation has become crucial in the areas of social media, mass communication, and public health. Technically, automatic models are developed to identify and detect the incorrect information on social media platforms, thereby mitigating the social effects (Guo et al., 2020; Yang et al., 2023; Shi et al., 2023).

While incorrect information detection methods have achieved significant advances, these methods still struggle with feature complexity, knowledge

*Corresponding authors

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 15674–15693

injection, and attention collapse. Specifically, prior work focuses either on textual content features (e.g., linguistic features and contextual embeddings) or on external knowledge (knowledge graphs and retrieval-augmented generation), which integrates all the feature streams into a classifier, without any disentanglement between what is learned from textual content and what is contributed by external knowledge. As illustrated in Figure 1, without knowledge inputs, the reader potentially accepts the incorrect information (i.e., Lafufu) as correct (i.e., Labubu) from the raw content; once the reader obtains relevant knowledge, the information is identified as incorrect.

To uncover the interplay between content reasoning and knowledge reasoning, we propose a new view: disentangling content reasoning from knowledge reasoning in an explicit and structured way. In this paper, we introduce a novel dual-head model architecture, **BiMind**, for incorrect information detection. Our model employs two separate heads to explore content and knowledge features, respectively, where the knowledge is retrieved from an in-domain memory. This separation mechanism allows us to measure, analyze, and apply the two streams of features in a structured way. Technically, our three contributions drive BiMind's novelty:

- First, we introduce an attention geometry adapter (AGA) that reshapes attention distributions at the pre-softmax logit level, stabilizing text encoding by mitigating attention collapse.
- Second, we design a self-retrieval knowledge module that encodes the training set into an in-domain semantic memory and then injects nearest-neighbor features via feature-wise linear modulation (FiLM).
- Third, we propose two uncertainty-aware fusion strategies, i.e., entropy-gated fusion and a trainable agreement head, where we adapt a symmetric Kullback–Leibler (KL) regularizer to ensure consistency between heads.
- Finally, we define a novel metric, Value-of-eXperience (VoX), to quantify the contributions from external knowledge, improving model interpretability.

Experimental results demonstrate that our model enhances detection accuracy and interpretability, especially when external knowledge contributes to model predictions.

2 Related Work

2.1 Content-based Methods

Today, machine learning (ML) and natural language processing methods (Kadhim, 2019; Su et al., 2020) have emerged as advanced tools to classify textual information in news articles into one or more predefined classes, such as correct or incorrect. Traditional ML methods, such as support vector machine, random forest, and decision tree, are commonly used in news content classification; however, these methods require hand-crafted features and struggle with complex text features, thus compromising performance (Minaee et al., 2021).

Along with neural networks being boosted, deep learning frameworks have further enhanced the classification performance by extracting complex content features and capturing nuanced semantic features, such as convolutional neural networks (CNNs) (Kim, 2014; Wang, 2017; Kaliyar et al., 2020), recurrent neural networks (RNNs) (Ma et al., 2016; Ruchansky et al., 2017), and long short-term memory (LSTM) (Sachan et al., 2019; Ma et al., 2020). Kaliyar et al. (Kaliyar et al., 2020) proposed a deep CNN model for incorrect information detection compared to classical CNN and LSTM structures, where it explores pre-trained word embeddings and multiple hidden layers to extract text features.

Additionally, attention networks integrated different features extracted from different latent aspects of news articles to improve detection accuracy (Yang et al., 2016; Linmei et al., 2019; Sun and Lu, 2020; Yun et al., 2023). For example, a hierarchical attention network (HAN) (Yang et al., 2016) was proposed to capture hierarchical structure of documents and employ the word-level and sentence-level attentions. To construct structured graphs based on texts, graph convolutional networks (GCNs) (Yao et al., 2018; Haider Rizvi et al., 2025) have been applied to textual content classification tasks, which construct document-level and corpus-level graphs to learn relationships among words, documents, and corpus.

With the aid of pre-trained knowledge embeddings, the transformer-based models have advanced the detection accuracy of incorrect information in news articles (Croce et al., 2020; Kaliyar et al., 2021; Xiong et al., 2021; Van Nooten and Daelemans, 2025). Combining the bidirectional encoder representations from transformers (BERT) (Devlin et al., 2019) with a CNN structure, Kaliyar et al.

(Kaliyar et al., 2021) proposed a BERT-based incorrect information detection model, where it inputs the BERT embeddings into one-dimensional CNN layers and then detects incorrect information using local features and global dependencies. Along with the data structure and modality extending, multimodal approaches are proposed to handle more intricate detection tasks for incorrect information content across text, image, video, audio data, or multiple languages (Conneau and Lample, 2019; Abdali et al., 2024; Wu et al., 2024; Lu and Koehn, 2025). For instance, Wu et al. (Wu et al., 2024) emphasized the substantive content over stylistic features, using Large Language Models (LLMs) to reframe news articles and focus on content veracity. Though LLMs emerged with impressive capability of processing multimodal features, LLMs still require a large volume of data to update the known knowledge and maintain performance.

2.2 Knowledge-based Methods

Traditional detection methods focus on internal content features and external fact-checking resources to detect incorrect information (Vlachos and Riedel, 2014; Hassan et al., 2015; Guo et al., 2022). For instance, the fact-checking approaches can identify and classify the texts by using the external knowledge sources to fact-check the news content (Etzioni et al., 2008; Wu et al., 2014; Shi and Weninger, 2016; Vo and Lee, 2018). But, these fact-checking approaches are time-consuming and demand human annotations, limiting the scalability and efficiency.

For further exploiting the content and external knowledge features to detect incorrect information, the credibility-based knowledge methods (Popat, 2017; Zhang et al., 2018; Deng et al., 2025) were proposed, which extract the source and content credibility features to identify factual news from non-credible ones, thereby enhancing model performance.

To explore the user behavior, engagements, and interactions on social media, the social relationship-aware approaches (Ghenai and Mejova, 2018; Shu et al., 2019; Dou et al., 2021; Teng et al., 2022) were proposed, which can capture user relationships, news content, and dissemination patterns to improve detection accuracy. For instance, Shu et al. (Shu et al., 2019) presented a tri-relationship-based detection framework of incorrect information content, where it explores the tri-relationship among publishers, news pieces, and users to differentiate reliable and unreliable articles. Zhang et al. (Zhang et al., 2024) explored the heterogeneous subgraph transformer (HeteroSGT) to detect incorrect information via the heterogeneous graph by unearthing the relationships among news topics, entities, and content.

To understand the propagation patterns of incorrect information within social networks, the network-based methods (Zhou and Zafarani, 2019) were suggested, where these methods focus on the interactions among spreaders and their influence on information propagation. Ma et al. (Ma et al., 2018) presented tree-structured recursive neural networks to model the propagation pattern of tweets for detecting rumors on social media. Typically, graph-based approaches were proposed (Bian et al., 2020; Fu et al., 2022) to explore the potential of graph structure in modeling social context structures, including knowledge-driven (Wang et al., 2018; Dun et al., 2021), propagation-based (Zhu et al., 2024), and context-aware approaches (Shang et al., 2024; Li et al., 2025).

Another direction of incorrect information detection approaches focuses on enhancing model performance with knowledge generation. Retrieval-augmented methods (Guu et al., 2020; Lewis et al., 2020; Chen et al., 2026) apply nearest-neighbor retrieval into LLMs to improve factual reasoning. Though achieving expected performance, these methods are computationally intensive and entangle retrieved knowledge with raw content in an opaque way. In contrast, our model, BiMind, disentangles content-internal reasoning from knowledge-augmented reasoning using a single yet transparent architecture. This separation strategy allows us to explicitly quantify the value of external knowledge through our proposed uncertainty-aware fusion and VoX metric, which differentiates our model from generic knowledge embedding frameworks.

3 Methodology

In this section, we introduce the fundamental framework of our proposed BiMind model, as shown in Figure 2. Here, we define the raw input text x_i as an internal information unit; all auxiliary information beyond the raw content, such as that retrieved from in-domain memory or linked to external resources, is treated as external knowledge unit. Our objective is to disentangle *content-internal reasoning* from *knowledge-augmented reasoning* in a structured way, and to provide interpretable diagnostics on

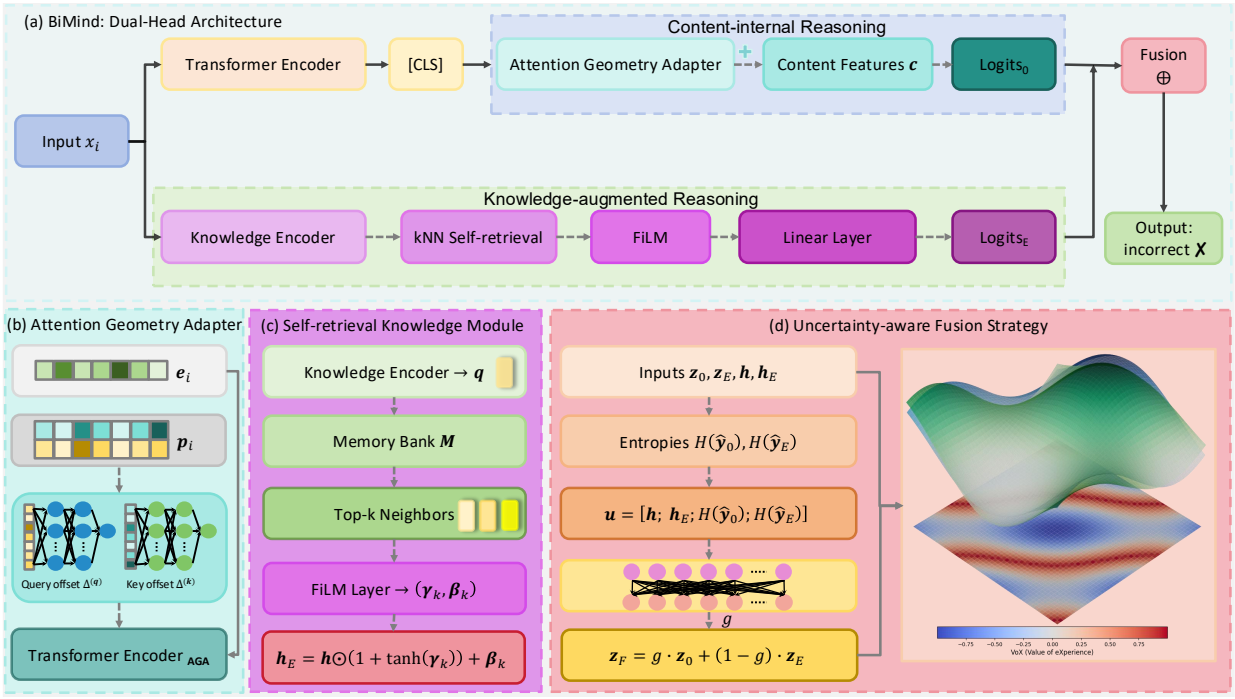

Figure 2: **An illustration of our proposed BiMind framework.** (a) Dual-head architecture with a content-internal head (top) and a knowledge-augmented head (bottom). (b) Attention geometry adapter reshapes pre-softmax attention logits via token-conditioned offsets. (c) Self-retrieval knowledge module retrieves top-k neighbors and injects knowledge via FiLM to provide knowledge-augmented representations. (d) Uncertainty-aware fusion combines head logits via an entropy gating, with the VoX metric quantifying the knowledge contributions by comparing head outcomes, where blue and green surfaces in the instance space represent the content and knowledge reasoning heads, respectively.

when and why external knowledge contributes to incorrect information detection. We present **BiMind**, a dual-head model with five key ingredients: (1) an attention geometry adapter (AGA) that reshapes pre-softmax attention geometry; (2) a self-retrieval knowledge module that constructs an in-domain memory through kNN retrieval; (3) a FiLM-based layer that injects retrieved external knowledge into the text representations; (4) the uncertainty-aware fusion strategies, including entropy-gated fusion strategy and a trainable agreement head, restrained by a symmetric KL regularizer; and (5) a VoX metric that quantifies knowledge contributions. More details can be found in the Appendix.

3.1 Problem Definitions

In this paper, we define the incorrect information detection as assessing whether a given information unit x_i is correct, where i is the i-th piece of information. Our detection foundation is that x_i is correct if no detected incorrectness exists. Therefore, the detection task is reframed as identifying incorrect elements within x_i. Formally, we define:

$$y(x_i) = \begin{cases} 1 & \text{if } \mathcal{I}(x_i) = \varnothing \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Here, x_i represents title, sentence, article, or narrative. $\mathcal{I}(x_i)$ is a set of incorrectness identified in x_i, such as linguistic elements (tokens or phrases), representation elements (feature embeddings), or knowledge elements (retrieved neighbors). $y(x_i)$ denotes correctness, i.e., 1 (correct information) or 0 (incorrect information). For incorrect information detection, we model it as a binary classification function:

$$f(x_i) \rightarrow y(x_i) \tag{2}$$

using a set of labeled training textual data, i.e.,

$$D_{\text{train}} = \{(x_i, y(x_i))\}_{i=1}^{|D_{\text{train}}|} \tag{3}$$

$y(x_i)$ is the label of x_i. $|D_{\text{train}}|$ is the total number of information units in the training dataset. We aim at learning the classification function:

$$f(x_i; \boldsymbol{\theta}) = \hat{y}_i \tag{4}$$

where $\hat{y}_i \in \{0, 1\}$ denotes the predicted label of x_i and $\boldsymbol{\theta}$ is the learnable parameter vector.

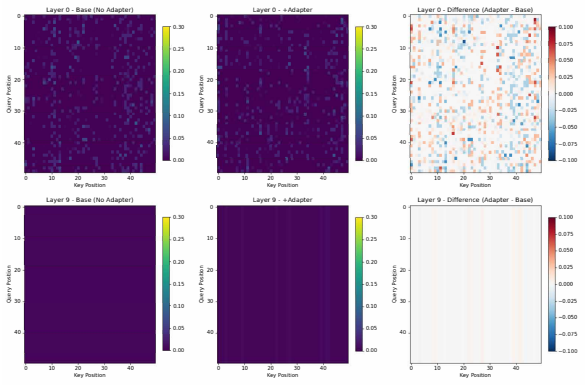

Figure 3: Illustration of the attention maps with (+) and without AGA across shallow (Layer 0) and deep (Layer 9) layers. AGA induces a structured, token-conditioned attention pattern at deeper layers, reshaping attention distribution by injecting global, key-focused inductive bias.

3.2 Attention Geometry Adapter

Let $x_i = (t_1, t_2, \ldots, t_L)$ be a tokenized text sequence. t_i denotes the i-th token and L is the length of the token sequence. Each token t_i is mapped to an embedding $\boldsymbol{e}_i \in \mathbb{R}^d$ through an embedding matrix $\boldsymbol{E}$, where d is the dimension of token embedding:

$$\boldsymbol{E}(x_i) = [\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_L] \in \mathbb{R}^{L \times d} \tag{5}$$

Instead of directly modifying token embeddings, we introduce an AGA module that reshapes attention distributions at the pre-softmax logit level. It's a lightweight module that improves how the model focuses on tokens using grammar rules. For each token t_i, we assign a token-level attribute vector $\boldsymbol{p}_i \in \{0,1\}^{D_{pos}}$ by using part-of-speech (POS) categories, where D_{pos} is the number of POS tags. In our POS tag set, we set $D_{pos} = 5$, including VERB/AUX, NOUN, ADJ, ADV, and OTHER. This representation provides an interpretable, low-dimensional basis for conditioning attention geometry, where attention geometry denotes the structural pattern of attention distributions across tokens, beyond individual attention weights.

Then, the Transformer encoder projects token embeddings $\boldsymbol{E}(x_i)$ into queries, keys, and values as in the standard self-attention. For each attention head h, AGA computes token-conditioned logit offsets $\boldsymbol{\Delta}$ of query and key via lightweight multilayer perceptrons (MLPs):

$$\boldsymbol{\Delta}^{(\boldsymbol{q})} = f_q(\boldsymbol{p}_i), \qquad \boldsymbol{\Delta}^{(\boldsymbol{k})} = f_k(\boldsymbol{p}_i) \tag{6}$$

where $f_q(\cdot)$ and $f_k(\cdot)$ are two-layer MLPs. The final pre-softmax attention logits are updated as:

$$\widetilde{\boldsymbol{A}}_{i,j}^{(h)} = \frac{\boldsymbol{q}_i^{(h)\top} \boldsymbol{k}_j^{(h)}}{\sqrt{d_k}} + \boldsymbol{\Delta}_{h,i}^{(\boldsymbol{q})} + \boldsymbol{\Delta}_{h,j}^{(\boldsymbol{k})}, \tag{7}$$

where $\tilde{\boldsymbol{A}}_{i,j}^{(h)}$ denotes the pre-softmax attention logit between query token $\boldsymbol{q}_i^{(h)}$ and key token $\boldsymbol{k}_j^{(h)}$ in attention head h; d_k is the dimension of the key vectors; $\boldsymbol{\Delta}_{h,i}^{(q)}$ and $\boldsymbol{\Delta}_{h,j}^{(k)}$ are token-conditioned offsets for head h applied to the query and key logits, respectively. Through Eq.(7), AGA adjusts attention score by adding token-level biases before softmax. More details are provided in the Appendix.

By injecting structured offsets, AGA reshapes the attention distributions, increasing entropy and mitigating attention collapse while leaving token embeddings unchanged, as shown in Figure 3. The attention outputs are then computed following standard multi-head attention and passed through the subsequent layers of Transformer $\mathcal{T}_{AGA}(\cdot)$:

$$\boldsymbol{H}_i = \mathcal{T}_{\text{AGA}}(\boldsymbol{E}(x_i)) \in \mathbb{R}^{L \times d} \tag{8}$$

in which $\boldsymbol{H}_i$ is the sequence representation. Finally, we apply max-pooling to capture the most salient features $\boldsymbol{h}$ in the sequence:

$$\boldsymbol{h} = \max_{i=1}^{L} \boldsymbol{H}_i \in \mathbb{R}^d \tag{9}$$

3.3 Semantic Neighbor Retrieval

In this section, we construct an in-domain semantic memory $\boldsymbol{M} = \{\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_N\}$ by encoding all training information units with a pre-trained SentenceTransformer (Reimers and Gurevych, 2019), then feeding knowledge vectors into backbone models like LLaMA-7B (Touvron et al., 2023). This memory serves as a lookup module that helps model find similar examples. $\boldsymbol{m}_j \in \mathbb{R}^{d_s}$ is the embedding of one information unit, normalized to unit length. d_s is the dimension of knowledge embeddings, and N is the number of training units stored in $\boldsymbol{M}$. For an input x_i, we encode it as query $\boldsymbol{q} \in \mathbb{R}^{d_s}$ and compute cosine similarity s_i between q and m:

$$s_j = \boldsymbol{q}^\top \boldsymbol{m}_j, \quad j = 1, \ldots, N \tag{10}$$

Next, we select the top-k neighbors with indices $\mathcal{N}(x_i)$ and aggregate:

$$\bar{\boldsymbol{m}} = \frac{1}{k} \sum_{j \in \mathcal{N}(x_i)} \boldsymbol{m}_j. \tag{11}$$

To inject retrieved knowledge neighbors, we map $\bar{\boldsymbol{m}}$ into modulation parameters $\boldsymbol{\gamma}_k$ and $\boldsymbol{\beta}_k$:

$$\boldsymbol{\gamma}_k = \boldsymbol{W}_\gamma \bar{\boldsymbol{m}} + \boldsymbol{b}_\gamma, \quad \boldsymbol{\beta}_k = \boldsymbol{W}_\beta \bar{m} + \boldsymbol{b}_\beta \quad (12)$$

Here, $\boldsymbol{W}_\gamma, \boldsymbol{W}_\beta \in \mathbb{R}^{d \times d_s}$ are projection matrices; $\boldsymbol{b}_\gamma$ and $\boldsymbol{b}_\beta$ are bias terms. Then, we apply FiLM layers (Perez et al., 2018) to produce the knowledge-augmented representation $\boldsymbol{h}_E$:

$$\boldsymbol{h}_E = \boldsymbol{h} \odot (1 + \tanh(\boldsymbol{\gamma}_k)) + \boldsymbol{\beta}_k \quad (13)$$

where $\odot$ is element-wise multiplication, and FilM adaptively scales $(1 + \tanh(\boldsymbol{\gamma}_k))$ and shifts $(\boldsymbol{\beta}_k)$ each dimension of $\boldsymbol{h}$ based on knowledge retrieved from semantic memory.

3.4 Dual-head Prediction

Combining text representations with content features $\boldsymbol{c} \in \mathbb{R}^{d_c}$, we define two reasoning heads:

$$\boldsymbol{z}_0 = \boldsymbol{W}_0[\boldsymbol{h}; \boldsymbol{c}] + \boldsymbol{b}_0, \quad \boldsymbol{z}_E = \boldsymbol{W}_E[\boldsymbol{h}_E; \boldsymbol{c}] + \boldsymbol{b}_E \quad (14)$$

where $\boldsymbol{z}_0$ and $\boldsymbol{z}_E \in \mathbb{R}^K$ are the outputs of the content-internal and knowledge-augmented reasoning heads, respectively. K is the number of labels. $\boldsymbol{W}_0$, $\boldsymbol{W}_E \in \mathbb{R}^{d \times (d+d_c)}$ are projection matrices; $\boldsymbol{b}_0$ and $\boldsymbol{b}_E$ are bias terms. Then, $\boldsymbol{z}_0$, $\boldsymbol{z}_E$ are transformed into probability distributions:

$$\hat{\boldsymbol{y}}_0 = \text{softmax}(\boldsymbol{z}_0), \quad \hat{\boldsymbol{y}}_E = \text{softmax}(\boldsymbol{z}_E) \quad (15)$$

$\hat{\boldsymbol{y}}_0$ is the prediction result from content-internal reasoning Head_0, and it's what model "believes" without external knowledge. Head_E reflects knowledge-augmented reasoning and derives $\hat{\boldsymbol{y}}_E$, and $\hat{\boldsymbol{y}}_E$ is what model "believes" with knowledge. Together, we can explore how knowledge affects predictions.

3.5 Uncertainty-aware Fusion Strategy

We fuse two predictions from Head_0 and Head_E using an entropy-gated strategy and a trainable agreement head, respectively. Using the entropy-gated fusion strategy, we first compute entropy for each head:

$$\mathcal{H}(\hat{\boldsymbol{y}}) = -\sum_{k=1}^{K} \hat{\boldsymbol{y}}_k \log \hat{\boldsymbol{y}}_k \quad (16)$$

where $\hat{\boldsymbol{y}}$ is the predicted probability distribution produced by a reasoning head and $\hat{\boldsymbol{y}}_k$ denotes the probability of class k. Then, we formulate the gate input vector $\boldsymbol{u}$, and feed it into MLP gate:

$$g = \sigma(\boldsymbol{W}_g \boldsymbol{u} + \boldsymbol{b}_g) \in (0, 1) \quad (17)$$

g is the fusion weight and σ is the activation function. $\boldsymbol{W}_g$ and $\boldsymbol{b}_g$ are parameters of the MLP gate. Finally, we fuse logits:

$$\boldsymbol{z}_F = g \cdot \boldsymbol{z}_0 + (1 - g) \cdot \boldsymbol{z}_E, \quad \hat{\boldsymbol{y}}_F = \text{softmax}(\boldsymbol{z}_F) \quad (18)$$

where $\boldsymbol{z}_F$ is the fused logits combining content-internal and knowledge-augmented reasoning. $\hat{\boldsymbol{y}}_F$ is the final prediction results. Therefore, if Head_0 has high entropy (uncertainty), the gate shifts the weight toward Head_E, and vice versa. More details can be found in the Appendix.

3.6 Agreement Regularization

To stabilize training process, we enforce the agreement between heads while preserving differences. We define the symmetric KL regularizer as:

$$\mathcal{L}_{\text{agree}}^{(i)} = \tfrac{1}{2} \left[D_{\text{KL}}(\hat{\boldsymbol{y}}_0^{(i)} \,\|\, \hat{\boldsymbol{y}}_E^{(i)}) + D_{\text{KL}}(\hat{\boldsymbol{y}}_E^{(i)} \,\|\, \hat{\boldsymbol{y}}_0^{(i)}) \right] \quad (19)$$

where $\mathcal{L}_{\text{agree}}^{(i)}$ is agreement loss. $D_{\text{KL}}(\hat{\boldsymbol{y}}_0^{(i)} \,\|\, \hat{\boldsymbol{y}}_E^{(i)})$ is the KL divergence between distributions $\hat{\boldsymbol{y}}_0^{(i)}$ and $\hat{\boldsymbol{y}}_E^{(i)}$. Based on the symmetric KL regularizer, our training objective is:

$$\mathcal{L} = \frac{1}{n} \sum_{i=1}^{n} \Big[\mathcal{L}_{\text{CE}}(\boldsymbol{z}_F^{(i)}, y(x_i)) + \tfrac{1}{2} \mathcal{L}_{\text{CE}}(\boldsymbol{z}_0^{(i)}, y(x_i)) + \tfrac{1}{2} \mathcal{L}_{\text{CE}}(\boldsymbol{z}_E^{(i)}, y(x_i)) + \lambda \mathcal{L}_{\text{agree}}^{(i)} \Big] \quad (20)$$

where $\mathcal{L}_{\text{CE}}^{(i)}$ is cross-entropy (CE) loss, and λ is the agreement loss weight. Through loss function $\mathcal{L}$, we encourage both heads to produce consistent predictions while maintaining their distinct reasoning.

3.7 Value-of-eXperience Metric

To measure knowledge contributions, we define VoX at the instance level. Terminologically, we refer to the retrieved knowledge as "experience" in our framework, to clarify its role as external evidence augmenting content-internal predictions. Given the correctness label $y(x_i)$, VoX is:

$$\text{VoX}(x_i) = \boldsymbol{z}_E[y(x_i)] - \boldsymbol{z}_0[y(x_i)]. \quad (21)$$

Our interpretations are summarized as follows:

- $\text{VoX}(x_i) > 0$: knowledge increases prediction confidence in correct class.
- $\text{VoX}(x_i) < 0$: knowledge decreases prediction confidence, suggesting potential noise.

- VoX(x_i) ≈ 0: knowledge has little effect.

Unlike accuracy or F1 score, VoX highlights when and why knowledge matters and provides interpretable diagnostics on knowledge augmentation.

4 Experiments

In this section, we conducted extensive experiments on four datasets collected from real-world scenarios, and experimental results demonstrate that our model has superior performance and efficiency to most tested models. We first introduced the experimental setup, including the datasets, tested models, and experimental settings. Then, we reported the experiment results and VoX values, and then analyzed these results for further exploration. Furthermore, the ablation study shows the modules contributing to the performance improvement. More details are provided in the Appendix.

4.1 Experimental Setup

Datasets. For conducting extensive experiments, we used four datasets to broadly test our model and other advanced models, including health datasets (MM COVID (Li et al., 2020) and RoCOVery (Zhou et al., 2020)), news content dataset (LIAR (Wang, 2017)), and a multi-domain dataset (MC Fake (Min et al., 2022)).

Experimental Models. To fairly perform the comparison experiments, we compared our proposed BiMind model with five models, which include a CNN-based model (Kim, 2014), a GCN-based model (Yao et al., 2018), HAN (Yang et al., 2016), BERT (Devlin et al., 2019), and HeteroSGT (Zhang et al., 2024).

4.2 Experiment Settings

For training and testing our proposed model, we split all the datasets into train, validation, and test datasets using a ratio of 80%, 10%, and 10%, respectively. To validate the generalizability of tested methods, we performed 10 rounds of tests with random seeds for each model and then recorded the average results and standard deviation. Here, all the experiments were conducted on 1 NVIDIA A100 GPU with 40 GB RAM. We quantitatively evaluated our model's performance compared to the other five tested models, using classification metrics such as accuracy (Acc), Macro-precision (Pre), Macro-F1 (F1), and Macro-recall (Rec).

4.3 Experimental Results

In Table 1, we reported the experimental results of all the tested models across the four datasets. From Table 1, one can see that our model achieves superior performance across all the metrics on the MM COVID, LIAR, and ReCOVery datasets, and suboptimal performance on the dataset MC Fake. It shows that our modules can improve the model performance and have a significant impact on detecting incorrect information. Additionally, we can see that our model achieves higher recall values on all four datasets, typically on the MM COVID, LIAR, and ReCOVery datasets. A higher recall value indicates that less incorrect information is missed. Furthermore, it should be noted that our model has robust and consistent performance across all the datasets, compared with other tested models. Though HeteroSGT achieves the optimal results, such as Rec and F1 on MC Fake due to its subgraph structure, it still drops performance by 19.1% on Acc and 19.1% on Pre, 5.2% on Acc and 5.7% on Pre, respectively, compared to our proposed model on ReCOVery and LIAR datasets. More details are provided in the Appendix.

4.4 Case Study

As illustrated in Table 2, the performance of BiMind varies across benchmark datasets due to the gate routes between Head_0 and Head_E. On datasets ReCOVery (+0.47 / 84.24% VoX, ≈0.04 Gate) and MM COVID (+0.97 / 83.29% VoX, ≈0.03 Gate), the gate leans strongly towards Head_E because the retrieved knowledge positively aligned with ground-truth labels. On MC Fake (−0.08 / 39.91% VoX, ≈0.22 Gate), the fusion has partial reliance on knowledge and generates mixed results, which improve minority-class recall but introduce noise. In contrast, on LIAR (+0.07 / 60.56% VoX, ≈0.19 Gate), we can see that when external knowledge is noisy, it leads fusion to weaken the predictions with a significant drop in F1 (58.78 → 47.30), highlighting that low gate values are not always effective and thus must be interpreted in the context of knowledge integrity and veracity. Additional analysis can be found in the Appendix.

4.5 Ablation Study

We conducted an ablation study on the ReCOVery dataset to evaluate the performance of four core modules in our model: AGA, self-retrieved knowledge module via FiLM, fusion strategies (entropy-

Dataset	CNN		GCN		BERT		HAN		HeteroSGT		BiMind	
	Acc	Pre	Acc	Pre	Acc	Pre	Acc	Pre	Acc	Pre	Acc	Pre
MM COVID	0.582±0.035	0.478±0.170	0.717±0.156	0.735±0.236	0.730±0.093	0.727±0.094	0.855±0.005	0.854±0.005	0.915±0.009	0.905±0.011	0.951±0.008	0.950±0.011
ReCOVery	0.658±0.011	0.460±0.104	0.718±0.037	0.691±0.178	0.682±0.030	0.441±0.213	0.722±0.021	0.462±0.197	0.727±0.023	0.731±0.047	0.918±0.013	0.922±0.013
MC Fake	0.825±0.001	0.544±0.156	0.724±0.138	0.516±0.169	0.827±0.006	0.713±0.271	0.825±0.005	0.463±0.098	0.883±0.002	0.812±0.003	0.887±0.005	0.827±0.006
LIAR	0.546±0.019	0.432±0.181	0.487±0.039	0.493±0.047	0.537±0.007	0.513±0.017	0.546±0.025	0.493±0.036	0.581±0.002	0.580±0.003	0.633±0.001	0.637±0.002
Dataset	Rec	F1	Rec	F1	Rec	F1	Rec	F1	Rec	F1	Rec	F1
MM COVID	0.547±0.039	0.474±0.101	0.685±0.178	0.621±0.184	0.722±0.101	0.720±0.103	0.854±0.006	0.853±0.005	0.883±0.013	0.893±0.011	0.951±0.009	0.951±0.008
ReCOVery	0.501±0.020	0.422±0.107	0.609±0.102	0.516±0.021	0.722±0.081	0.416±0.032	0.506±0.002	0.457±0.013	0.585±0.036	0.571±0.049	0.918±0.013	0.919±0.013
MC Fake	0.501±0.002	0.455±0.004	0.552±0.169	0.470±0.039	0.502±0.001	0.451±0.002	0.500±0.004	0.453±0.001	0.762±0.002	0.783±0.003	0.700±0.099	0.738±0.109
LIAR	0.502±0.005	0.377±0.049	0.494±0.029	0.423±0.055	0.510±0.012	0.483±0.014	0.502±0.018	0.445±0.053	0.575±0.002	0.571±0.003	0.636±0.003	0.633±0.002

Table 1: Detection performance on four datasets (best in red, second-best in blue).

Dataset	Head/Mode	Acc	F1	Pre	Rec	VoX (mean / pos%)	Gate Mean (%<0.3 / %>0.7)
ReCOVery	Head_0	85.22	85.15	85.10	85.22	–	–
	Head_E	87.19	86.93	86.98	87.19	+0.47 / 84.24%	0.04 (100.00% / 0.00%)
	Fused	87.19	86.93	86.98	87.19	–	0.04 (100.00% / 0.00%)
MC Fake	Head_0	86.50	84.85	84.79	86.50	–	–
	Head_E	87.35	86.90	86.63	87.35	-0.08 / 39.91%	0.22 (81.24% / 0.00%)
	Fused	87.42	86.82	86.52	87.42	–	0.22 (81.24% / 0.00%)
MM COVID	Head_0	81.70	81.78	82.13	81.70	–	–
	Head_E	90.45	90.51	91.18	90.45	+0.97 / 83.29%	0.03 (100.00% / 0.00%)
	Fused	90.72	90.77	91.38	90.72	–	0.03 (100.00% / 0.00%)
LIAR	Head_0	59.90	58.78	59.66	59.90	–	–
	Head_E	57.70	47.30	66.32	57.70	+0.07 / 60.56%	0.19 (100.00% / 0.00%)
	Fused	58.29	48.66	66.51	58.29	–	0.19 (100.00% / 0.00%)

Table 2: Cross-dataset VoX results. We report mean VoX value and percentage of samples with positive gain (pos%); we also show mean gate value and routing mass below/above thresholds, with %<0.3 indicating the gate leaned strongly toward knowledge head Head_E, and %>0.7 indicating it leaned strongly toward content head Head_0.

Model Variant	Acc	Pre	Rec	F1
Full BiMind model (w/ knowledge)	**0.897**	**0.895**	**0.897**	**0.895**
Baseline (content w/o knowledge)	0.852	0.849	0.852	0.848
– Attention geometry adapter	0.872	0.870	0.872	0.870
– Knowledge retrieval	0.847	0.847	0.847	0.847
– Gated fusion	0.862	0.861	0.862	0.861
– Trainable agreement head	0.867	0.868	0.867	0.867
– Symmetric KL Regularizer	0.872	0.881	0.872	0.874

Table 3: Ablation study on ReCOVery dataset. One module is removed at a time from the full BiMind model.

gated scheme and trainable agreement head), and symmetric KL regularizer. From Table 3, we can see that the BiMind model, with all these four modules, achieved the best performance, i.e., Acc of 0.897, Pre of 0.895, Rec of 0.897, and F1 score of 0.895. More results can be found in the Appendix.

5 Limitations

Though our proposed BiMind framework has superior performance in the incorrect information detection task by integrating textual and knowledge features, several limitations remain. First, AGA conditions attention geometry on token-level attributes, which might be less efficient for inputs with limited salient lexical signals. Secondly, BiMind does not incorporate social credibility or propagation patterns into the detection pipeline. Then, when the detection model has prediction errors, it might weaken the correct information flow. Finally, with the growing prevalence of multimodal information on social media platforms, incorrect information is not limited to textual content but involves images, videos, and audio. Therefore, multimodal incorrect information detection has become a critical challenge. BiMind can be extended to a multimodal reasoning framework by the disentanglement principle: one head focuses on content-internal textual reasoning, and another head encodes visual signal, followed by uncertainty-aware fusion to calibrate their contributions.

6 Conclusion

Incorrect information significantly disrupts content quality and integrity on social media platforms, and therefore, it's increasingly important to develop detection models that are efficient and interpretable. Compared to most detection approaches that blend textual content and external knowledge, we proposed **BiMind**, a dual-head framework that explicitly disentangles *content-internal reasoning* from *knowledge-augmented reasoning* for incorrect in-

formation detection. In this work, we first designed an attention geometry adapter that reshapes attention distributions to prevent attention collapse. Secondly, an in-memory semantic knowledge base was constructed to retrieve and encode external knowledge features through the FiLM layer. Then, we introduced two uncertainty-aware fusion strategies, including an entropy-gated strategy and a trainable agreement head, regularized by a symmetric KL regularizer. Finally, the VoX metric was defined, which quantifies the knowledge contributions, providing interpretable diagnostics at the instance level on *when* and *why* knowledge impacts detection. Experimental results across benchmark datasets show that BiMind achieves competitive performance while providing interpretable insights on knowledge injections.

Acknowledgments

This work is supported in part by the National Science Foundation under Award No. 2343387.

References

Sara Abdali, Sina Shaham, and Bhaskar Krishnamachari. 2024. Multi-modal misinformation detection: Approaches, challenges and opportunities. *ACM Computing Surveys*, 57(3):1–29.

Sajjad Ahmed, Knut Hinkelmann, and Flavio Corradini. 2022. Combining machine learning with knowledge engineering to detect fake news in social networks-a survey. *arXiv preprint arXiv:2201.08032*.

Bimal Bhattarai, Ole-Christoffer Granmo, and Lei Jiao. 2021. Explainable tsetlin machine framework for fake news detection with credibility score assessment. *arXiv preprint arXiv:2105.09114*.

Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 549–556.

Xi Chen, Wei Xue, and Yike Guo. 2026. Actormind: Emulating human actor reasoning for speech role-playing. *arXiv preprint arXiv:2604.11103*.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pages 7059–7069.

Danilo Croce, Giuseppe Castellucci, and Roberto Basili. 2020. GAN-BERT: Generative adversarial learning for robust text classification with a bunch of labeled examples. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2114–2119, Online. Association for Computational Linguistics.

Boyi Deng, Wenjie Wang, Fengbin Zhu, Qifan Wang, and Fuli Feng. 2025. Cram: Credibility-aware attention modification in llms for combating misinformation in rag. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 23760–23768.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, volume 1 (long and short papers)*, pages 4171–4186.

Yingtong Dou, Kai Shu, Congying Xia, Philip S. Yu, and Lichao Sun. 2021. User preference-aware fake news detection. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2051–2055.

Yaqian Dun, Kefei Tu, Chen Chen, Chunyan Hou, and Xiaojie Yuan. 2021. Kan: Knowledge-aware attention network for fake news detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 81–89.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S Weld. 2008. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74.

Dongqi Fu, Yikun Ban, Hanghang Tong, Ross Maciejewski, and Jingrui He. 2022. Disco: Comprehensive and explainable disinformation detection. *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4848–4852.

Amira Ghenai and Yelena Mejova. 2018. Fake cures: user-centric modeling of health misinformation in social media. In *Proceedings of the ACM on Human-Computer Interaction*, pages 1–20.

Bin Guo, Yasan Ding, Lina Yao, Yunji Liang, and Zhiwen Yu. 2020. The future of false information detection on social media: New perspectives and trends. *ACM Computing Survery*, 53(4):1–36.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, pages 3929–3938.

Syed Mustafa Haider Rizvi, Ramsha Imran, and Arif Mahmood. 2025. Text classification using graph convolutional networks: A comprehensive survey. *ACM Computing Survery*, 57(8).

Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2015. Detecting check-worthy factual claims in presidential debates. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1835–1838.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Ammar Ismael Kadhim. 2019. Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review*, 52:273–292.

Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. 2021. Fakebert: Fake news detection in social media with a bert-based deep learning approach. *Multimedia Tools and Applications*, 80(8):11765–11788.

Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang, and Soumendu Sinha. 2020. Fndnet–a deep convolutional neural network for fake news detection. *Cognitive Systems Research*, 61:32–44.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Guangchen Lan, Huseyin A Inan, Sahar Abdelnabi, Janardhan Kulkarni, Lukas Wutschitz, Reza Shokri, Christopher G Brinton, and Robert Sim. 2025. Contextual integrity in llms via reasoning and reinforcement learning. *arXiv preprint arXiv:2506.04245*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, pages 9459–9474.

Guoyi Li, Die Hu, Zongzhen Liu, Xiaodan Zhang, and Honglei Lyu. 2025. Semantic reshuffling with LLM and heterogeneous graph auto-encoder for enhanced rumor detection. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8557–8572, Abu Dhabi, UAE. Association for Computational Linguistics.

Yichuan Li, Bohan Jiang, Kai Shu, and Huan Liu. 2020. Mm-covid: A multilingual and multimodal data repository for combating covid-19 disinformation. *arXiv preprint arXiv:2011.04088*.

Hu Linmei, Tianchi Yang, Chuan Shi, Houye Ji, and Xiaoli Li. 2019. Heterogeneous graph attention networks for semi-supervised short text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 4821–4830, Hong Kong, China. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

TaiMing Lu and Philipp Koehn. 2025. Learn and unlearn: Addressing misinformation in multilingual LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10191–10206, Suzhou, China. Association for Computational Linguistics.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J. Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, pages 3818–3824.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor detection on Twitter with tree-structured recursive neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1980–1989, Melbourne, Australia. Association for Computational Linguistics.

Qianli Ma, Zhenxi Lin, Jiangyue Yan, Zipeng Chen, and Liuhong Yu. 2020. MODE-LSTM: A parameter-efficient recurrent network with multi-scale for sentence classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 6705–6715, Online. Association for Computational Linguistics.

Valeria Mazzeo, Andrea Rapisarda, and Giovanni Giuffrida. 2021. Detection of fake news on covid-19 on web search engines. *Frontiers in Physics*, 9:685730.

Erxue Min, Yu Rong, Yatao Bian, Tingyang Xu, Peilin Zhao, Junzhou Huang, and Sophia Ananiadou. 2022. Divide-and-conquer: Post-user interaction network for fake news detection on social media. In *Proceedings of the ACM Web Conference*, pages 1148–1158.

Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep learning–based text classification: A comprehensive review. *ACM Computing Survery*, 54(3).

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. 2018. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3942–3951.

Kashyap Popat. 2017. Assessing the credibility of claims on the web. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 735–739.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Natali Ruchansky, Sungyong Seo, and Yan Liu. 2017. Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 797–806.

Devendra Singh Sachan, Manzil Zaheer, and Ruslan Salakhutdinov. 2019. Revisiting lstm networks for semi-supervised text classification via mixed objective function. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6940–6948.

Lanyu Shang, Yang Zhang, Bozhang Chen, Ruohan Zong, Zhenrui Yue, Huimin Zeng, Na Wei, and Dong Wang. 2024. Mmadapt: A knowledge-guided multi-source multi-class domain adaptive framework for early health misinformation detection. In *Proceedings of the ACM Web Conference*, pages 4653–4663.

Baoxu Shi and Tim Weninger. 2016. Fact checking in heterogeneous information networks. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 101–102.

Chongyang Shi, Yijun Yin, Qi Zhang, Liang Xiao, Usman Naseem, Shoujin Wang, and Liang Hu. 2023. Multiview clickbait detection via jointly modeling subjective and objective preference. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11807–11816, Singapore. Association for Computational Linguistics.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: a data mining perspective. *SIGKDD Explor. Newsl.*, 19(1):22–36.

Kai Shu, Suhang Wang, and Huan Liu. 2019. Beyond news contents: The role of social context for fake news detection. In *Proceedings of the 12nd ACM International Conference on Web Search and Data Mining*, pages 312–320.

Qi Su, Mingyu Wan, Xiaoqian Liu, and Chu-Ren Huang. 2020. Motivations, methods and metrics of misinformation detection: an nlp perspective. *Natural Language Processing Research*, 1(1):1–13.

Xiaobing Sun and Wei Lu. 2020. Understanding attention for text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3418–3428, Online. Association for Computational Linguistics.

Xian Teng, Yu-Ru Lin, Wen-Ting Chung, Ang Li, and Adriana Kovashka. 2022. Characterizing user susceptibility to covid-19 misinformation on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 1005–1016.

Kjerstin Thorson, Emily Vraga, and Brian Ekdale. 2010. Credibility in context: How uncivil online commentary affects news credibility. *Mass Communication and Society*, 13(3):289–313.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Shu-Feng Tsao, Helen Chen, Therese Tisseverasinghe, Yang Yang, Lianghua Li, and Zahid A Butt. 2021. What social media told us in the time of covid-19: a scoping review. *The Lancet Digital Health*, 3(3):e175–e194.

Jens Van Nooten and Walter Daelemans. 2025. Jump to hyperspace: Comparing Euclidean and hyperbolic loss functions for hierarchical multi-label text classification. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4260–4273, Abu Dhabi, UAE. Association for Computational Linguistics.

Andreas Vlachos and Sebastian Riedel. 2014. Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 18–22.

Nguyen Vo and Kyumin Lee. 2018. The rise of guardians: Fact-checking url recommendation to combat fake news. In *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 275–284.

William Yang Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

Xuepeng Wang, Kang Liu, Shizhu He, and Jun Zhao. 2016. Learning to represent review with tensor decomposition for spam detection. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 866–875, Austin, Texas. Association for Computational Linguistics.

Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. 2018. Eann: Event adversarial neural networks for multi-modal fake news detection. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 849–857.

Jiaying Wu, Jiafeng Guo, and Bryan Hooi. 2024. Fake news in sheep's clothing: Robust fake news detection against llm-empowered style attacks. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3367–3378.

You Wu, Pankaj K. Agarwal, Chengkai Li, Jun Yang, and Cong Yu. 2014. Toward computational fact-checking. In *Proceedings of the VLDB Endowment*, pages 589–600.

Yijin Xiong, Yukun Feng, Hao Wu, Hidetaka Kamigaito, and Manabu Okumura. 2021. Fusing label embedding into BERT: An efficient improvement for text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1743–1750, Online. Association for Computational Linguistics.

Chang Yang, Peng Zhang, Wenbo Qiao, Hui Gao, and Jiaming Zhao. 2023. Rumor detection on social media with crowd intelligence and ChatGPT-assisted networks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5705–5717, Singapore. Association for Computational Linguistics.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2018. Graph convolutional networks for text classification. *arXiv preprint arXiv:1809.05679*.

Jungmin Yun, Mihyeon Kim, and Youngbin Kim. 2023. Focus on the core: Efficient attention via pruned token compression for document classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13617–13628, Singapore. Association for Computational Linguistics.

Amy X. Zhang, Aditya Ranganathan, Sarah Emlen Metz, Scott Appling, Connie Moon Sehat, Norman Gilmore, Nick B. Adams, Emmanuel Vincent, Jennifer Lee, Martin Robbins, Ed Bice, Sandro Hawke, David Karger, and An Xiao Mina. 2018. A structured response to misinformation: Defining and annotating credibility indicators in news articles. In *Companion Proceedings of the The Web Conference*, pages 603–612.

Yuchen Zhang, Xiaoxiao Ma, Jia Wu, Jian Yang, and Hao Fan. 2024. Heterogeneous subgraph transformer for fake news detection. In *Proceedings of the ACM Web Conference*, pages 1272–1282.

Xinyi Zhou, Apurva Mulay, Emilio Ferrara, and Reza Zafarani. 2020. Recovery: A multimodal repository for covid-19 news credibility research. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 3205–3212.

Xinyi Zhou and Reza Zafarani. 2019. Network-based fake news detection: A pattern-driven approach. *arXiv preprint arXiv:1906.04210*.

Xinyi Zhou and Reza Zafarani. 2020. A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Survery*, 53(5).

Junyou Zhu, Chao Gao, Ze Yin, Xianghua Li, and Juergen Kurths. 2024. Propagation structure-aware graph transformer for robust and interpretable fake news detection. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4652–4663.

A Methodology Details

Pipeline Overview. Given an input x_i, BiMind first encodes the text via a Transformer encoder with an AGA module to produce a text representation $\boldsymbol{h}$ from the content itself. In parallel, BiMind retrieves top-k semantically similar examples from an in-domain memory bank and injects them into $\boldsymbol{h}$ through FiLM to form a knowledge-augmented representation $\boldsymbol{h}_E$. The two heads then perform separate predictions and generate $\boldsymbol{z}_0$ and $\boldsymbol{z}_E$, where the fusion weight g is computed using an uncertainty-aware fusion strategy and $\boldsymbol{z}_0$ and $\boldsymbol{z}_E$ are combined into the final prediction $\boldsymbol{z}_F$. Finally, we quantify how much the retrieved knowledge helps by using the VoX metric.

Here, we chose these POS categories based on empirical patterns observed in our preliminary analysis, which show that certain word types, such as adjectives and pronouns, provide strong discriminative signals. Beyond improving accuracy, our AGA design makes the model's attention behavior more stable and interpretable by grounding it in linguistic patterns. A universal POS tagging framework is adopted, and the formulation is language-agnostic, with only the tagging tool needing to be adapted for different languages.

AGA. In the AGA module, a learnable per-head temperature τ_h is applied before softmax to normalize the logits and produce attention weights $\alpha_{i,j}^{(h)}$:

$$\alpha_{i,j}^{(h)} = \mathrm{softmax}_j\left(\frac{\widetilde{\boldsymbol{A}}_{i,j}^{(h)}}{\tau_h}\right) \tag{22}$$

Each head output is then computed as a weighted sum $\boldsymbol{o}_i^{(h)}$ of values $\boldsymbol{v}_j^{(h)}$:

$$\boldsymbol{o}_i^{(h)} = \sum_{j=1}^{L} \alpha_{i,j}^{(h)}\, \boldsymbol{v}_j^{(h)} \tag{23}$$

and the final output of multi-head attention (MHA) is:

$$\mathrm{MHA}(\boldsymbol{E}(x_i)) = \mathrm{Concat}\Big(\boldsymbol{o}_i^{(1)}, \ldots, \boldsymbol{o}_i^{(H)}\Big)\, \boldsymbol{W}_o \tag{24}$$

where H is the number of attention heads in MHA and $\boldsymbol{W}_o$ is the weight matrix.

Entropy-gated Fusion. Uncertainty-aware fusion strategy is a weighting strategy that generates the final prediction based on the model's entropy. A higher entropy indicates the prediction is less confident. Here, we formulate the gate input vector $\boldsymbol{u}$ as:

$$\boldsymbol{u} = [\boldsymbol{h}; \boldsymbol{h}_E; \mathcal{H}(\hat{\boldsymbol{y}}_0); \mathcal{H}(\hat{\boldsymbol{y}}_E)] \tag{25}$$

where $\mathcal{H}(\hat{\boldsymbol{y}}_0)$, $\mathcal{H}(\hat{\boldsymbol{y}}_E)$ are the entropy of Head_0 and Head_E.

Trainable Agreement Head. In the trainable agreement head scheme, we combine two streams of features from both heads and add a new classifier to learn how to jointly leverage them, instead of directly fusing predictions. To construct agreement features, we combine:

- hidden states $\boldsymbol{h}$ and $\boldsymbol{h}_E$,
- elementwise interaction $(\boldsymbol{h} \odot \boldsymbol{h}_E)$,
- and absolute difference $(|\boldsymbol{h} - \boldsymbol{h}_E|)$.

Formally, the agreement feature vector is defined as:

$$\boldsymbol{\phi}_{\text{agree}} = [\,\boldsymbol{h};\ \boldsymbol{h}_E;\ \boldsymbol{h} \odot \boldsymbol{h}_E;\ |\boldsymbol{h} - \boldsymbol{h}_E|\,]. \tag{26}$$

Then, the agreement features are fed into the MLP layers:

$$\boldsymbol{z}_A = \boldsymbol{W}_2\, \sigma(\boldsymbol{W}_1 \boldsymbol{\phi}_{\text{agree}} + \boldsymbol{b}_1) + \boldsymbol{b}_2, \quad \boldsymbol{z}_A \in \mathbb{R}^K, \tag{27}$$

where $\boldsymbol{z}_A$ is agreement logits. $\boldsymbol{W}_1, \boldsymbol{W}_2$ and $\boldsymbol{b}_1, \boldsymbol{b}_2$ are learnable parameters. Finally, the agreement head outputs predictions as:

$$\boldsymbol{y}_A = \mathrm{softmax}(\boldsymbol{z}_A). \tag{28}$$

Here, $\boldsymbol{y}_A$ is the "agreement head" prediction, which learns to explore the consistency and discrepancy between the two reasoning heads.

B Experimental Model

Model Configuration. Our detection pipeline employs a BiMind design, which effectively integrates textual and knowledge features. Each input token is represented by a 128-dimensional embedding vector. The maximum sequence length is 5000. In the transformer-based classifier module, our model consists of 2 stacked transformer encoder layers with a multi-head attention scheme (i.e., 16 attention heads), producing pooled text representations. This custom 2-layer transformer encoder is trained from scratch as the backbone for content-internal prediction heads. Unlike our LLM variants, this architecture does not use pretrained models, and it encodes text only and learns task-specific representations end-to-end.

Then, we construct two heads: a content-internal head that incorporates the text representations with the AGA module, and an external knowledge head that injects the knowledge vectors via FiLM before MLP. Here, we set k to 3 in the knowledge retrieval module, based on SentenceTransformer (Reimers and Gurevych, 2019). In the feature fusion function, we set the entropy-gated strategy as the default, where other options include a trainable agreement head, standard logit average, and product-of-experts. For the two heads, we employ ReLU as an activation function and set dropout regularization to 0.3, where both heads are trained with CE loss and a symmetric-KL agreement regularizer. To handle the class imbalance issue, we adopt class-balanced weights on the CE loss. The final output layer with a softmax function is designed to provide the probability distribution indicating the likelihood of the content being labeled 1 (i.e., correct information) or 0 (i.e., incorrect information). Here, we use Adam optimizer with learning rate 1×10^{-5} and batch size 64 to train our model, where we employ the mixed-precision training, gradient clipping, and early-stopping (patience = 3) to tune the hyperparameters.

Experimental Models. In the experiment setup, we compared our BiMind model with five tested models, including CNN-based model (Kim, 2014), GCN-based model (Yao et al., 2018), HAN (Yang et al., 2016), BERT (Devlin et al., 2019), and HeteroSGT (Zhang et al., 2024). Technically, the CNN-based model employs CNN layers to extract text features from article content and then uses the extracted features to detect incorrect information. The GCN-based model explores the weighted graph built on news articles, which uses a GCN for identifying incorrect information. HAN applies word-level and sentence-level features in news content for incorrect information detection. Here, BERT is a transformer-based model, similar to our transformer-based classifier. HeteroSGT explores the heterogeneous subgraph transformer to classify articles via the heterogeneous graph.

Model Parameters. In the main body of this paper, we used the LLaMA-7B model as the backbone. We reported both total and trainable parameters for all models we tested and compared the SentenceTransformer-based baseline model against several LLMs:

- Small/Medium models (110M-184M parameters): BERT, RoBERTa, and DeBERTa .
- Large model (7B parameters): LLaMA-7B.

Model	Model Size	Trainable Params	Ratio
BiMind-Custom (Tiny)	3.5M	3.5M	100%
BiMind-RoBERTa (Medium)	125M	3.3M	2.6%
BiMind-LLaMA (Large)	7B	44.4M	<1%

Table 4: Parameter efficiency across BiMind variants.

Detailed results are provided in Appendix Tables 9 and 10. To demonstrate model scalability, we designed three versions of BiMind, as shown in Table 4. In this work, we can see that BiMind has good performance, not just due to the model size. Our results show:

- Model efficiency: BiMind outperforms full-sized models like BERT even though it has far fewer parameters that need training.
- Performance consistency: Our system shows performance improvements whether we use a small, medium, or large model as the base.

To find the best settings for both the standard BiMind and the LLM-based BiMind, we use Optuna to test a set of different hyperparameter settings and keep random seeds fixed for reproducibility, such as:

- For the standard BiMind: We test different model structures, including: how many layers and heads, learning rate, and dropout.
- For the LLM-based BiMind: Since the LLM is frozen, we focus on tuning the parts we added, such as the AGA module.

To ensure our comparison between models is fair, we follow these rules:

- Training setup: Every model is evaluated using the same data split strategy and learning objectives.
- No peeking: We never look at the final test results when picking the best settings.

C Dataset Statistics

Here, we present the statistics of the datasets we used, listed in Table 5.

Dataset-level Knowledge Impact Analysis. We conducted a comprehensive statistical analysis of knowledge impact across these datasets, as shown in Table 6. Based on these datasets, we observed

Dataset	# Label 0	# Label 1	# Total	Avg. Length (words)
MM COVID	1,888	1,162	3,048	25
RoCOVery	605	1,294	1,899	500
LIAR	2,507	2,053	4,560	17
MC Fake	2,671	12,621	15,292	300

Table 5: Statistics of the datasets used in our experiments.

substantial but different levels of vocabulary alignment between test instances and the knowledge bank (ranging from 75.60% to 81.68%), showing that retrieved knowledge is largely in-domain. But, retrieval relevance and its effects differ significantly. LIAR presents the lowest retrieval similarity (mean 0.6077), indicating weaker semantic alignment between its short claims and retrieved knowledge, which limits the efficiency of knowledge injection. In addition, MM COVID shows moderate similarity (mean 0.6870) with higher variance, revealing that knowledge retrieval is more sensitive and selective: for short and noisy social media posts, knowledge injection yields large positive VoX gains when relevant knowledge is retrieved.

In contrast, MC Fake and ReCOVery both exhibit consistently high retrieval similarity (means 0.7751 and 0.7567, respectively), suggesting that retrieval quality is not the primary bottleneck. Instead, linguistic complexity is the dominant factor: retrieved knowledge in these datasets presents extremely low Flesch Reading Ease scores, and knowledge impact varies primarily with how such dense content is integrated rather than how relevant it is. In summary, these statistical results illustrate a spectrum of knowledge integration regimes, ranging from knowledge-limited (LIAR), to retrieval-sensitive (MM COVID), and finally to complexity-dominated settings (MC Fake, ReCOVery), motivating adaptive and uncertainty-aware mechanisms for mediating the impact of external knowledge.

D Data Leakage

In our pipeline, we implemented several methods to avoid the data leakage issues, such as:

- Data split: We follow the official splits for all the datasets, such as a standard 80/10/10 split and 10-run averaging for evaluation.
- Knowledge base construction: The knowledge base is only constructed from training data. It never has access to the test set during the retrieval process.
- Knowledge dropout (0.3): We train the model to predict outputs even when knowledge is unreliable or absent.
- Uncertainty-aware fusion: When knowledge head has high entropy (uncertainty), the gate shifts weight toward the content head.

Additionally, we conducted additional experiments, including duplicate check, cheating test, and leakage-aware experiment. The results are summarized as below:

- Duplicate check: We removed near-duplicate training samples using cosine similarity $\geq$ 0.85, filtering out 178/1643 samples (10.83%). But, the train-test similarity is still significant (mean 0.71, 22.1% $\geq$ 0.80), showing potential leakage risk.
- Cheating test: When test samples are included in the knowledge base, retrieval becomes perfect (top-1 similarity = 1.0, 100% $\geq$ 0.95). A KNN method explores the leakage and improves from 78.82% to 89.66% (+10.84%), while our full model is unchanged (90.15% $\rightarrow$ 90.15%), showing it does not use leakage.
- Leakage-aware experiment: Full model performance is stable across different setups:
  - Setup A (deduplicated knowledge base): 90.15%
  - Setup B (external Wikipedia): 90.64%
  - Setup C (leakage baseline): 90.15%

  In contrast, KNN model shows A $\approx$ 78.82% < C $\approx$ 89.66%, confirming that leakage is detectable but not utilized by our model.

Empirical Results. From the results in Table 7, we can see that the fusion mechanism significantly mitigates negative transfer, where fused performance exceeds both individual heads. From our analysis in Table 8, we observed: If data leakage existed, we would expect >50% "Helps" cases (i.e., the model simply retrieving answers). The 15% improvement aligns with semantic knowledge transfer, not memorization. In addition, we measured retrieval similarity, and if the model is cheating or memorizing answers, we would see very high similarity scores (i.e., near 1.0). Instead, our results show:

Dataset	Vocab Alignment (%)	Max Sim.	Mean Sim.	Flesch (helps)
LIAR	76.55	0.6603±0.0924	0.6077±0.0843	37.94
MM COVID	75.60	0.7586±0.1228	0.6870±0.1082	31.68
MC Fake	79.91	0.8176±0.1084	0.7751±0.1082	-280.47
ReCOVery	81.68	0.8056±0.1071	0.7567±0.1026	-585.00

Table 6: Statistical comparison of knowledge attributes across datasets. Vocabulary alignment measures lexical intersection between inputs and the knowledge base. Retrieval relevance is reported as cosine similarity (mean ± std). Flesch Reading Ease for the *helps* category reflects the linguistic complexity of retrieved knowledge.

- Low similarity: The average similarity between what the model finds and what it's searching is quite low (around 0.52), showing that it is finding related ideas, not exact copies.
- Mixed results: Though the retrieved information helps the model 15% of the time, it actually confuses the model 8% of the time. If the model were "leaking" the correct answers, it would be right almost 100% of the time.

Metric	Content Head	Knowledge Head	Fused
Accuracy	68.2%	61.5% (-6.7%)	70.1% (+1.9%)
F1-Score	0.65	0.58	**0.68**

Table 7: Empirical results on fusion strategy.

Knowledge Impact	% of Test Set	Interpretation
Helps (wrong → correct)	∼15%	Genuine knowledge benefit
Hurts (correct → wrong)	∼8%	Knowledge introduces noise
Neutral / Both Wrong	∼77%	No leaked advantage

Table 8: Knowledge impact on data leakage.

For future work, we plan to design and evaluate other knowledge filtering mechanisms, such as confidence-based knowledge injection or learning a classifier to predict whether retrieval knowledge will help.

E Experimental Results

For the five comparison models, CNN has poor performance on all the datasets, which may result from its fixed convolutional kernels. Due to these kernels focusing on local features, the global features or dependencies might not be effectively explored in news articles and social contexts. GCN presents different results across multiple datasets and receives better detection accuracy on the MC Fake dataset. In addition, HAN and BERT are transformer-based models with attention mechanisms, and thus, the performance is comparable between them. Though HeteroSGT achieves optimal results, such as Rec and F1 on MC Fake due to its subgraph structure, it still drops performance by 19.1% on Acc and 19.1% on Pre, 5.2% on Acc and 5.7% on Pre, respectively, compared to our proposed model on ReCOVery and LIAR datasets. Typically, on the LIAR dataset, our model achieves consistent performance across seeds, with a low standard deviation (±0.001).

Extended Experiments. Here, we extended the dual-head design to other LLMs, i.e., RoBERTa (Liu et al., 2019) and DeBERTa (He et al., 2020). From Table 9, we can see that separating content-internal reasoning (Head_0) from the knowledge-augmented reasoning (Head_E) shows significant dataset-relevant behavior. On knowledge-aligned datasets, like ReCOVery and MM COVID, Head_E consistently improves recall and F1 score, suggesting that external knowledge provides complementary contextual signals beyond textual content alone. In contrast, on the LIAR dataset with short claims and weak retrieval alignment, Head_E is not generally helpful, supporting our motivation to disentangle content inference from knowledge-based reasoning rather than enforcing unconditional knowledge injection. Additionally, the uncertainty-aware fusion strategy achieves either the best or second-best performance across models and datasets. Notably, it reduces variance and receives gains when one head performs poorly, especially on the LIAR dataset. These results validate our design choice to treat knowledge as an auxiliary, selectively trusted signal, with a fusion strategy adapting to instance-level uncertainty rather than relying on static feature concatenation alone.

Together, these results demonstrate that the effective and reliable knowledge injection (i) conditions on data-inherent attributes, including vocabulary alignment, retrieval relevance, and sample-level linguistic complexity, and (ii) requires a principled fusion mechanism with uncertainty and agreement measurement.

Model	Dataset	Head/Mode	Acc	F1	Pre	Rec
LLaMA-7B	ReCOVery	Head_0	91.13 ± 2.34	91.03 ± 2.59	91.35 ± 2.32	91.13 ± 2.34
		Head_E	91.04 ± 1.18	91.09 ± 1.21	91.53 ± 1.44	91.04 ± 1.18
		Fused	**91.82 ± 1.29**	**91.86 ± 1.27**	**92.20 ± 1.26**	**91.82 ± 1.29**
	MM COVID	Head_0	94.69 ± 1.01	94.70 ± 1.01	94.46 ± 1.45	94.67 ± 0.99
		Head_E	94.80 ± 0.77	94.81 ± 0.76	94.48 ± 1.21	94.61 ± 0.64
		Fused	**95.12 ± 0.78**	**95.12 ± 0.78**	**94.98 ± 1.06**	**95.08 ± 0.86**
	LIAR	Head_0	**63.26 ± 0.13**	**63.26 ± 0.19**	63.70 ± 0.20	63.60 ± 0.30
		Head_E	62.70 ± 0.41	62.42 ± 0.53	**64.00 ± 0.20**	63.60 ± 0.30
		Fused	62.93 ± 0.29	62.89 ± 0.29	63.80 ± 0.20	**63.70 ± 0.20**
DeBERTa-v3	ReCOVery	Head_0	81.38 ± 2.95	81.64 ± 2.71	83.85 ± 1.80	81.38 ± 2.70
		Head_E	85.72 ± 1.43	85.92 ± 1.25	86.39 ± 1.30	85.32 ± 1.30
		Fused	**85.81 ± 1.27**	**85.94 ± 1.16**	**86.42 ± 1.20**	**85.81 ± 0.90**
	MM COVID	Head_0	93.00 ± 1.23	92.98 ± 1.24	93.07 ± 1.20	92.73 ± 1.40
		Head_E	**94.59 ± 1.05**	**94.59 ± 1.05**	**94.47 ± 1.20**	**94.56 ± 1.00**
		Fused	94.53 ± 1.09	94.54 ± 1.08	94.43 ± 1.30	94.47 ± 1.10
	LIAR	Head_0	59.75 ± 0.89	59.14 ± 1.21	61.70 ± 0.50	60.80 ± 0.50
		Head_E	**62.05 ± 0.77**	**62.02 ± 0.94**	**62.50 ± 0.50**	**62.50 ± 0.60**
		Fused	61.91 ± 0.78	61.87 ± 1.00	62.50 ± 0.50	62.40 ± 0.60
RoBERTa	ReCOVery	Head_0	81.28 ± 2.96	81.67 ± 2.60	80.36 ± 2.20	82.84 ± 1.74
		Head_E	84.53 ± 2.10	84.71 ± 2.19	83.97 ± 1.16	85.76 ± 0.74
		Fused	**84.93 ± 2.45**	**85.12 ± 2.54**	**83.92 ± 1.76**	**86.13 ± 1.06**
	MM COVID	Head_0	91.87 ± 1.23	91.88 ± 1.23	92.25 ± 1.44	91.87 ± 1.23
		Head_E	**94.49 ± 0.26**	**94.49 ± 0.27**	**94.59 ± 0.26**	**94.49 ± 0.26**
		Fused	94.21 ± 0.44	94.22 ± 0.44	94.40 ± 0.55	94.21 ± 0.44
	LIAR	Head_0	61.18 ± 0.77	60.83 ± 0.83	62.37 ± 0.17	61.91 ± 0.25
		Head_E	**61.88 ± 0.63**	**61.84 ± 0.73**	**62.71 ± 0.25**	**62.55 ± 0.34**
		Fused	61.83 ± 0.66	61.68 ± 0.78	62.67 ± 0.29	62.47 ± 0.41

Table 9: Extended performance comparison (mean ± std, in %, across 10 runs, best in **bold**).

Dataset	Model	Acc	Pre	Rec	F1	Training	Testing	Graph
MM COVID	HeteroSGT	0.915±0.009	0.905±0.011	0.883±0.013	0.893±0.011	55.11	–	13.62
	BiMind	**0.902±0.116**	**0.902±0.110**	**0.898±0.142**	**0.900±0.132**	15.19	0.08	–
ReCOVery	HeteroSGT	0.727±0.023	0.731±0.047	0.585±0.036	0.571±0.049	21.94	–	9.03
	BiMind	**0.879±0.017**	**0.862±0.028**	**0.843±0.203**	**0.854±0.208**	14.99	0.10	–
MC Fake	HeteroSGT	0.883±0.002	0.812±0.003	0.762±0.002	0.783±0.003	478.53	–	40.04
	BiMind	**0.887±0.051**	**0.827±0.016**	**0.700±0.099**	**0.798±0.109**	153.01	0.45	–
LIAR	HeteroSGT	0.581±0.002	0.580±0.003	0.575±0.002	0.571±0.003	116.22	–	14.48
	BiMind	**0.605±0.041**	**0.601±0.045**	**0.595±0.037**	**0.595±0.037**	73.51	0.52	–

Table 10: Performance and runtime comparison between **BiMind** and **HeteroSGT** across four benchmark datasets. Performance is reported as mean ± std. Runtime is measured in seconds per run.

F Ablation Study

When removing the AGA, it leads to a significant drop in accuracy (0.897 → 0.872) and F1 (0.895 → 0.870), showing the importance of reshaping attention logits to prevent attention collapse. Without the knowledge retrieval function, it also reduces the performance, such as a larger drop in accuracy

and recall (0.897 → 0.847), indicating the significance of semantic knowledge neighbors in grounding short or ambiguous content. Additionally, replacing the uncertainty-aware fusions with a simple logit average, it causes performance degradation in F1 (0.895 → 0.861 or 0.867), showing that our fusion strategies help the model adaptively trust knowledge-augmented predictions when content-internal predictions are uncertain. Finally, removing the symmetric KL regularizer, it reduces F1 from 0.895 to 0.874, demonstrating that agreement between heads stabilizes training and improves predictions.

In conclusion, ablation results show that each component contributes complementary benefits: content features build strong baselines, attention geometries sharpen token-level salience, knowledge retrieval contextualizes content, and uncertainty fusion with an agreement regularizer ensures robust integration. In a modular and structured way, these modules jointly enable BiMind to achieve both competitive performance and interpretable diagnostics on when and why knowledge matters.

G Retrieval-based Methods

We tested our model against three retrieval-based methods on ReCOVery and provided detailed comparison results in Table 11. All models are tested using the same data and experiment settings. Below are the key findings:

- Retrieval is insufficient: The simple retrieval-based methods (like kNN) cannot reach expected performance, showing how retrieved knowledge is used matters more than retrieval itself.
- Simple fusion doesn't work: The RAG model performs slightly worse than kNN, suggesting that just stacking knowledge does not help the model learn better.
- BiMind is significantly better: Our model is nearly 4% more accurate than the baselines, indicating that BiMind correctly predicts a larger number of cases where other models failed.

Why BiMind Wins. Our dual-head design is more effective because:

- Two separate paths: BiMind has one branch for the text itself and one for external knowledge, allowing it to learn which head to trust based on input features.
- Conditional fusion using gating: Instead of treating all external knowledge as equally important, our model uses a gating strategy to dynamically weight retrieved knowledge, especially when knowledge is noisy.
- Knowledge dropout during training: We train the model to handle cases where knowledge is unreliable, improving robustness.

Method	Accuracy	F1	Improvement
kNN (k=5)	83.74%	83.54%	-
kNN-BERT (k=5)	83.74%	83.54%	-
Standard RAG (k=3)	83.25%	83.00%	-0.49%
BiMind (Ours)	**87.70%**	**87.90%**	**+3.96%**

Table 11: Comparison with retrieval-based baselines on ReCOVery.

Knowledge Safeguards. Here, BiMind explores multiple safeguards across retrieval, training, and architecture, such as knowledge dropout (e.g., 0.3) and an uncertainty-aware fusion strategy to prevent noisy or irrelevant retrieval from hurting performance in low-resource or short-text scenarios. Additional safeguards are summarized as follows:

- Similarity-aware knowledge vector: We augment the knowledge vector with similarity signals, enabling the model to down-weight low-quality retrieval.
- Bounded FiLM injection: FiLM injection is bounded via $tanh$ to prevent feature distortion from noisy inputs.
- Agreement loss: An agreement loss regularizes divergence between reasoning heads, stabilizing learning under noisy retrieval.
- Leakage-aware evaluation: The leakage evaluation is performed with similarity-based deduplication (threshold = 0.85) and a cheating test to ensure robustness is not driven by retrieval leakage.

In addition, we evaluated the safeguard's robustness using the Knowledge Rejection Rate (KRR) and the Fusion Recovery Rate (FRR). When the gating mechanism does not reject irrelevant knowledge (KRR = 0.0%), the model recovers correct predictions in 68.6% of irrelevant cases, demonstrating strong robustness. Further analysis shows

that recovery is maximized at moderate gate values (FRR = 81.8%, gate values ≈ 0.44-0.45), indicating that performance relies on balancing internal and external knowledge rather than hard filtering. Additionally, retrieval similarity is not a reliable indicator of correctness (FRR: 76.9% (medium similarity) vs. 66.7% (high similarity)), as high-similarity neighbors can still hurt the predictions. These results highlight the importance of uncertainty-aware fusion over retrieval-only strategies.

To further validate our findings with RAG, we also experimented with LLaMA-7B as the backbone and compared our BiMind to LLaMA-based RAG. Even though LLaMA is a big model and improves over standard RAG (85-86% range), BiMind still surpasses it by 1-2%. These results show that BiMind does not just retrieval knowlegde; it can incorporate all the information from content itself and external knowledge in a structured manner. An important finding from our dual-head design is that knowledge contribution varies across instances. By analyzing the learned fusion gates, we observed that:

- For easy cases (high content clarity), content head dominates (gate ≈ 0.8-0.9).
- For ambiguous cases requiring context, the knowledge head contributes more (gate ≈ 0.4-0.6).

From these findings, we can see that this adaptive behavior is absent in fixed-weight RAG approaches. This additional ablation study shows that removing the gating mechanism (i.e., fixed 50-50 fusion) reduces accuracy by 2.5%, confirming that dynamic fusion is essential.

H VoX Analysis

To further interpret the VoX values, we visualized four types of knowledge impacts in Figure 4, which demonstrates how knowledge can impact prediction confidence over the reasoning path. Typically, knowledge can help (e.g., MM COVID), be neutral, hurt (e.g., LIAR), or produce mixed patterns (e.g., MC Fake) from the dataset-level outcomes. To validate VoX interpretability, we conducted additional experiments using two metrics: Δ_{margin} and $\Delta_{entropy}$. From our results, we observed that VoX is strongly correlated with Δ_{margin} (Pearson r = 0.54, p < 0.001) and weakly correlated with $\Delta_{entropy}$ (r = 0.02), showing that higher VoX values are related to larger margin improvements and

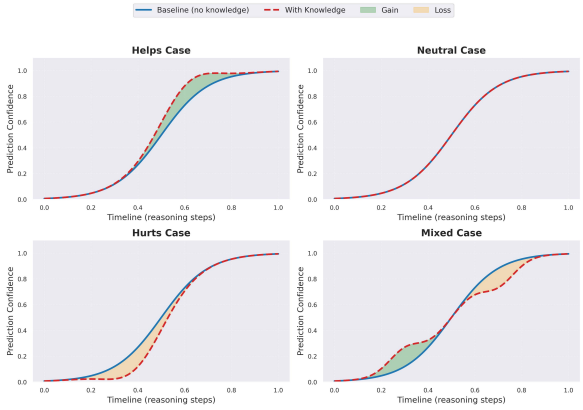

Figure 4: An illustration of knowledge impacts at the instance level. Blue curve = baseline (no knowledge), red dashed curve = with knowledge, shaded regions indicate gain (green) or loss (orange).

reduced uncertainty. It demonstrates that VoX represents model confidence shifts rather than calibration impact. In addition, $\Delta_{confidence}$ shows stronger alignment with performance gain (ρ = 0.353, p < 0.001) and demonstrates very high agreement with VoX itself (ρ = 0.754, p < 1e-38), showing that VoX effectively captures confidence differentials rather than random calibration noise. Spearman correlation between VoX and per-instance probability improvement is ρ = 0.23 for gain and ρ = 0.41 for correctness (both p < 0.001), statistically supporting its diagnostic validity.

I Quantitative Analysis

Here, we describe the running time comparison of our BiMind framework with backbone Sentence-Transformer and HeteroSGT. Beyond superior detection accuracy, we compared the runtime of BiMind against HeteroSGT, as shown in Table 10. In our framework, we skip the graph construction phase, resulting in training and testing that is nearly 4× faster (e.g, on MM COVID). HeteroSGT requires additional graph construction time (e.g., 40.04s on MC Fake) and retraining to adapt to new topics; but, BiMind generalizes with lightweight attention signals and knowledge features, showing BiMind's efficiency and scalability merits in incorrect information detection applications.

J Attention Head Specialization

Figure 5 compares attention head specialization at Layer 9 of the Transformer with and without AGA. For the baseline model, the attention heads present severe representational collapse: all heads have

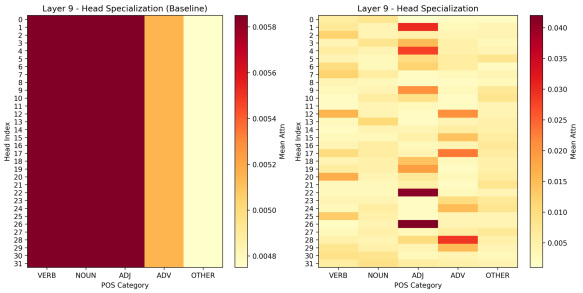

Figure 5: Layer 9 attention head specialization with and without AGA. Left: Baseline Transformer without AGA. Right: Transformer with AGA.

nearly identical attention patterns, with uniformly high focus on several categories (VERB, NOUN, ADJ) and minimal head-level variance. It shows that, without AGA, the self-attention mechanism tends to flatten linguistic structure in deeper layers. However, with AGA, it shows a significantly different geometric pattern. The number of active heads is reduced, but the head specialization is selectively preserved where a small number of heads focus on different categories (like ADJ and ADV). Specifically, head specialization remains diverse rather than uniform. It demonstrates that AGA transforms attention collapse into geometry-aware concentration, compressing distributed signals into a low-rank but structured representation.