Are Multimodal Transformers Robust to Missing Modality?

Mengmeng Ma1  Jian Ren2  Long Zhao3  Davide Testuggine2  Xi Peng1

1University of Delaware  2Snap Inc.  3Google Research

{mengma,xipeng}@udel.edu, jren@snap.com, longzh@google.com, davide.testuggine@gmail.com

Abstract

Multimodal data collected from the real world are often imperfect due to missing modalities. Therefore multimodal models that are robust against modal-incomplete data are highly preferred. Recently, Transformer models have shown great success in processing multimodal data. However, existing work has been limited to either architecture designs or pre-training strategies; whether Transformer models are naturally robust against missing-modal data has rarely been investigated. In this paper, we present the first-of-its-kind work to comprehensively investigate the behavior of Transformers in the presence of modal-incomplete data. Unsurprisingly, we find Transformer models are sensitive to missing modalities while different modal fusion strategies will significantly affect the robustness. What surprised us is that the optimal fusion strategy is dataset dependent even for the same Transformer model; there does not exist a universal strategy that works in general cases. Based on these findings, we propose a principle method to improve the robustness of Transformer models by automatically searching for an optimal fusion strategy regarding input data. Experimental validations on three benchmarks support the superior performance of the proposed method.

1. Introduction

Multimodal Transformers are emerging as the dominant choice in multimodal learning across various tasks [7], including classification [21, 29], segmentation [34], and cross-model retrieval [18]. They have become the driving force in obtaining better performance on these tasks through a pre-train-and-transfer [3] paradigm.

Although Transformers have demonstrated remarkable success in processing multimodal data, they generally require modal-complete data. The completeness of modality may not always hold in the real world due to privacy or security constraints. For example, a social network might be unable to access location information if users decline to share their private location [20]; a healthcare application might not have all the records available when patients are unwilling to undergo risky or invasive examinations [37]. For this reason, it is important that Transformer models are robust against missing-modal data, i.e., the model performance does not degrade dramatically.

Despite its real-world importance, the robustness against missing modalities in multimodal Transformers is seldom investigated in the literature. So far, research on Transformer models has been limited to developing new architectures for fusion [29, 35, 38] or exploring better self-supervised learning tasks [1, 6, 8, 47, 48]. Recent work on Transformer robustness has primarily focused on noisy inputs rather than missing modalities [23].

A question naturally arises: Are Transformer models robust against missing-modal data? We empirically evaluate this problem across multiple datasets in Table 1. Unsurprisingly, we find that Transformer models degrade dramatically with missing-modal data. As shown, the multimodal performance drops when tested with modal-incomplete data, and, surprisingly, the multimodal performance is even worse than the unimodal when text are missing severely, i.e., only 30% of text are available.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training</th>
<th>Testing</th>
<th>Evaluation</th>
<th>Δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>MM-IMDb [2]</td>
<td>100%</td>
<td>100%</td>
<td>55.3</td>
<td>0%</td>
</tr>
<tr>
<td></td>
<td>100%</td>
<td>100%</td>
<td>31.2</td>
<td>43.6%</td>
</tr>
<tr>
<td></td>
<td>100%</td>
<td>0%</td>
<td>35.0</td>
<td>36.7%</td>
</tr>
<tr>
<td>UPMC Food-101 [43]</td>
<td>100%</td>
<td>100%</td>
<td>91.9</td>
<td>0%</td>
</tr>
<tr>
<td></td>
<td>100%</td>
<td>100%</td>
<td>65.9</td>
<td>28.3%</td>
</tr>
<tr>
<td></td>
<td>100%</td>
<td>0%</td>
<td>71.5</td>
<td>22.2%</td>
</tr>
<tr>
<td>Hateful Memes [17]</td>
<td>100%</td>
<td>100%</td>
<td>70.2</td>
<td>0%</td>
</tr>
<tr>
<td></td>
<td>100%</td>
<td>100%</td>
<td>60.2</td>
<td>14.2%</td>
</tr>
<tr>
<td></td>
<td>100%</td>
<td>0%</td>
<td>56.3</td>
<td>19.8%</td>
</tr>
</tbody>
</table>
2. Related Work

Multimodal learning. Different modalities, e.g., natural language, visual signals, or vocal signals, are often complementary in content while overlapping for a common concept. Multimodal learning aims to utilize the complementary information of each modality to improve the performance of various computer vision tasks. A key aspect of multimodal learning is exploring efficient methods for multimodal fusion. Simple methods like concatenation have been widely studied in [32, 42]. For efficient cross-modality interaction, a tensor fusion [46] mechanism is proposed by Zadeh et al. Following this effort, efficient low-rank fusion [25] is proposed to address the exponential dimension explosion of tensor fusion.

The aforementioned fusion mechanisms heavily depend on the completeness of modality, making multimodal fusion impossible with modality-incomplete data. Therefore, another important direction in multimodal learning is to build models that are robust against the modality-incomplete data [27, 39]. For example, Ma et al. [27] propose a method based on Bayesian Meta-Learning to estimate the latent feature of the modality-incomplete data. However, the existing endeavors usually adopt modality-specific models for each modality, such as ResNet [12] for images and LSTM [13] for texts, which may lead to a larger set of architectural decisions and training parameters. Instead, we use Transformers as general architectures to jointly model each modality, leading to a simple design and reduced training parameters.

Multimodal transformer. Multimodal Transformers have been used in various tasks such as cross-modal retrieval [18, 22], action recognition [29], and image segmentation [34, 45]. They provide several advantages over conventional backbones, e.g., ResNet [12], regarding to flexibility and training load.

The flexibility to accommodate modality-incomplete samples is crucial for multimodal backbones, as the real-world multimodal data are often imperfect due to missing modality. Conventional backbones [31, 39] are generally not flexible. These backbones output the joint multimodal representation by explicitly fusing the features of each modality via concatenation [32], tensor fusion [46], and others mechanisms. However, explicit fusion requires the presence of all modalities. Missing any modality will break the training pipeline. In contrast, multimodal Transformers use the self-attention mechanism [40] to generate a holistic representation of all modalities, allowing the absence of any modalities. When dealing with modality-incomplete samples, it can ignore the absent modalities by applying a mask on the attention matrix. Therefore, multimodal Transformers are more flexible in dealing with missing modalities. Besides, an easy-to-train model is vital for multimodal learning. The training load of conventional multimodal backbone grows as the number of modalities in-
Table 3. Multi-label classification scores (%) on the MM-IMDb [2] under different settings: train and test with full modality (100% Image + 100% Text); train and test with single modality (100% Image or 100% Text). ✓ indicate our implementation.

<table>
<thead>
<tr>
<th>Method</th>
<th>Image</th>
<th>Text</th>
<th>F1 Micro</th>
<th>F1 Macro</th>
<th>F1 Weighted</th>
<th>F1 Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFAS [30]</td>
<td>✓</td>
<td>✓</td>
<td>47.8</td>
<td>25.6</td>
<td>42.1</td>
<td>48.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>60.2</td>
<td>48.9</td>
<td>58.5</td>
<td>60.6</td>
</tr>
<tr>
<td>CentralNet  [41]</td>
<td>✓</td>
<td>✓</td>
<td>33.5</td>
<td>49.2</td>
<td>45.9</td>
<td>57.5</td>
</tr>
<tr>
<td>ViLT [18]†</td>
<td>✓</td>
<td>✓</td>
<td>51.8</td>
<td>45.9</td>
<td>49.2</td>
<td>57.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>63.3</td>
<td>52.5</td>
<td>62.0</td>
<td>62.9</td>
</tr>
</tbody>
</table>

Table 4. Classification accuracy (%) on the UPMC Food-101 [43]. ✓ indicate our implementation.

<table>
<thead>
<tr>
<th>Method</th>
<th>Modality</th>
<th>Image</th>
<th>Text</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT+LSTM [9]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>71.7</td>
</tr>
<tr>
<td>ViLT [18]†</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>92.5</td>
</tr>
<tr>
<td>BERT+LSTM [9]</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>92.1</td>
</tr>
<tr>
<td>ViLT [18]†</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>92.0</td>
</tr>
</tbody>
</table>

 Increases since the backbone usually consists of modality-specific sub-models that need to be trained independently for each modality [27, 41]. Instead, Transformer models process all modalities simultaneously using a single model [18, 24], which greatly reduces the training load.

**Dynamic neural networks.** Our work is also related to dynamic neural networks, which adapt the network structure to different inputs, resulting in noticeable gains in accuracy, computation efficiency, or flexibility [11]. We follow the spirit of the Dynamic Depth [11] method. Numerous methods have been proposed to dynamically select layers for inference to reduce computation costs. Our idea is inspired by AdaShare [36], which focuses on learning a policy that selects layers for sharing in multi-task learning. Its main idea is to use Gumbel Softmax Sampling [14,28] to learn the policy and network parameters without relying on Reinforcement Learning [44] or extra policy network [10]. However, the direct application of Gumbel Softmax Sampling to our problem results in a large search space with many invalid policies. As a result, we develop an efficient method without using Gumbel Softmax Sampling.

### 3. Analysis of Multimodal Transformer

#### 3.1. Background

In this paper, we focus on the multimodal Transformer that adopts Vision Transformer (ViT) [7] as the backbone. ViT consists of a sequence of \( L \) Transformer layers, each of which contains a Multi-Head Attention (MHA) layer, Multi-layer Perceptron (MLP), and Layer Normalization (LN). The MHA computes the dot-product attention [40] on the input sequence, resulting in an attention matrix indicating the similarity between each token.

We follow the vision-language Transformer [18] to preprocess the data. The input text is mapped into the word embedding through a word embedding codebook and a position embedding codebook. The input image is first partitioned into patches and then flattened into vectors. These vectors are then transformed into latent embedding using linear projection and position embedding. Finally, the image and text embedding are integrated with their corresponding modality-type embedding [15, 18]. The final multimodal input sequence is the concatenation of vision and text embedding.

#### 3.2. Robustness Against Missing Modality

**Question:** Are Transformer models robust against modal-incomplete data?

**Observation:** Unsurprisingly, Transformer models degrade dramatically with modal-incomplete data.

We begin by defining how Transformer robustness is measured. Specifically, we adopt two different evaluation settings: a “full” test set with full-modal data and a “missing” test set with missing-modal data. We evaluate Transformer robustness by comparing model performance on the “missing” test set to the “full” test set: the smaller the difference, the better the robustness.

First, we empirically verify that model performance degrades dramatically in the presence of missing-modal data. Table 1 shows the evaluation results on three widely-used multimodal datasets. As shown, when only 30% text modality is observed, the multimodal performance drops by 43.6%, 28.3%, and 14.2%, respectively. Moreover, when the modality is severely missing, the multimodal performance is even worse than the unimodal one on MM-IMDb and UPMC Food-101.

Second, we observe that the modality importance varies on different datasets. We use unimodal performance to indicate the importance of each modality. Results of unimodal performance are shown in Tables 3, 4, and 5. As shown, the text modality is more important than the image modality on MM-IMDb and Food-101, while text and image are equally important on Hateful Memes. In specific, in the first two datasets, text has higher performance than the image. Moreover, when the modality is severely missing, the multimodal performance is even worse than the unimodal one on MM-IMDb and UPMC Food-101.

In contrast, in the Hateful Memes dataset, the performances of text and image are comparable,
and the performance gap between multimodal and unimodal is large (>20%), demonstrating that the two modalities are equally important.

Finally, we empirically observe that Transformer models tend to overfit to dominate modalities. Specifically, we first train our model with multimodal data and test with different unimodal data. Then we examine the performance gap between unimodal and multimodal testing — the larger the gap, the more severe the overfitting. Experimental results are shown in Table 6. As shown, for MM-IMDb dataset, text-only testing performs better than image-only testing, which means that text-only testing is closer to full-modal testing. Therefore text-only testing has a smaller gap than image-only testing, indicating that models trained on this dataset tend to overfit to text modality.

### 3.3. Optimal Fusion Strategy

**Question:** Will the fusion strategy affect Transformer robustness against modal-incomplete data?

**Observation:** Different fusion strategies do affect the robustness of Transformer models. Surprisingly, the optimal fusion strategy is dataset-dependent; there does not exist a universal strategy that works in general cases.

Typically, there exists two widely used fusion strategies: early and late fusion. For early fusion, cross-modal interaction happened in early layers, ensuring the model to have sufficient capacity to exploit multimodal information, but at the expense of larger computing costs. For late fusion, cross-modal interaction happened in later layers, which significantly lower computation costs, but the resulting model might have limited capacity to take the full advantage of multimodal information.

It remains an open question on how to determine the optimal layer for fusion [3]. Existing solutions in multimodal Transformer adopt a fixed fusion strategy [18,21,24,26,33]. However, the one-size-fits-all approach may not be optimal on all datasets. As discussed in Sec. 1, the optimal fusion strategy is dataset-dependent.

### 4. Robust Multimodal Transformer

Without loss of generality, we consider a multimodal dataset with two modalities. Formally, let $D = \{x_1^i, x_2^i, y_i\}_i$ denote the multimodal dataset, where $x_1^i$ and $x_2^i$ represent two different modalities and $y_i$ is the corresponding label. Our target is to improve the Transformer robustness against modal-incomplete data, i.e., model performance does not degrade dramatically. To this end, we propose to leverage multi-task optimization and the optimal fusion strategy to improve the robustness.

**Multi-task learning.** We intend to improve the performance of Transformer models in dealing with modal-incomplete data. In the missing-modal scenario, training data are modal-complete, while testing samples are modal-incomplete. This discrepancy motivates us to incorporate missing-modal data in the training process. By doing so, the Transformer model will be more confident in its prediction on modal-incomplete data, resulting in a robust Transformer. The key idea is to leverage the masking mechanism to “generate” the modal-incomplete data during training and jointly optimize the Transformer model with modal-complete and modal-incomplete data via multi-task optimization. Our method is simple to implement with minimal modification to the Transformer.

**Optimal fusion strategy.** The goal is to automatically search for the optimal fusion strategy on different datasets. Manually finding the optimal strategy is not practical, especially for large-scale models [5,7,47], due to the heavy training load. However, designing such an algorithm is non-trivial in light of the non-differentiable nature of the discrete searching space [36]. Existing methods, such as Reinforcement Learning (RL) [44] and policy network (PN) [10], are either inefficient in training or adding additional parameters to the model. We propose a differentiable method to obtain the fusion strategy through standard back-propagation. The key idea is to learn a policy to obtain optimal layers for fusion. Specifically, each layer is assigned with a policy parameter to decide fusion or not. The fusion strategy is

---

1 Policy parameters are negligible compared to model parameters.

---

**Table 5. AUROC (%) on the unseen test set of Hateful Memes [17].

<table>
<thead>
<tr>
<th>Method</th>
<th>Modality</th>
<th>AUROC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Image</td>
<td>Text</td>
</tr>
<tr>
<td>Unimodal*</td>
<td>✓</td>
<td>54.6</td>
</tr>
<tr>
<td>VILBERT [18]†</td>
<td>✓</td>
<td>56.3</td>
</tr>
<tr>
<td>MMBT-Grid [15]†</td>
<td>✓</td>
<td>67.3</td>
</tr>
<tr>
<td>MMBT-Region [15]†</td>
<td>✓ ✓</td>
<td>72.2</td>
</tr>
<tr>
<td>ViLBERT CC [26]†</td>
<td>✓ ✓</td>
<td>73.4</td>
</tr>
<tr>
<td>Visual BERT [24]†</td>
<td>✓ ✓</td>
<td>72.8</td>
</tr>
<tr>
<td>VILT [18]†</td>
<td>✓ ✓</td>
<td>73.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>70.2</td>
</tr>
</tbody>
</table>

| Table 6. Evaluation on the overfitting issue of Transformer models on MM-IMDb and Hateful Memes. Transformer models tend to overfit to dominate modality. |
|-----------------|---------|-----------|-----------|-----------|
| Dataset         | Training | Testing | Evaluation |
|                 | Image   | Text     | Image     | Text      |
| MM-IMDb         | 100%    | 100%     | 0%        | 100%      | 55.3      |
| Hatful Memes    | 100%    | 100%     | 0%        | 100%      | 70.2      |
|                 | 100%    | 100%     | 0%        | 100%      | 55.7      |
|                 | 100%    | 100%     | 0%        | 100%      | 54.9      |
4.1. Improve Robustness via Multi-task Learning

On a bimodal dataset, e.g., image and text, multi-task learning can have up to three distinct tasks: full-modal (image + text) task, image-only task, and text-only task. Let $f_\theta$ denote a Transformer parameterized by $\theta$. The total loss function is defined as follows:

$$L = \lambda_1 L_{img}(x^1; \theta) + \lambda_2 L_{txt}(x^2; \theta) + \lambda_3 L_{it}(x^1; x^2; \theta),$$

(1)

where $L_{img}$ is the loss for image only task; $L_{txt}$ is the loss for text only task; $L_{it}$ is the loss for image + text task; $\lambda_1$, $\lambda_2$, and $\lambda_3$ are hyperparameters to balance each loss. Transformer models leverage classification tokens [18, 26, 40] to generate embeddings for classification. For the three tasks, we add three classification tokens to the Transformer model. Each classification token will output task-specific embedding for the target task. The model overview is shown in Figure 1 Left. For multi-task learning, each task is expected to use only corresponding modalities for classification, e.g., text modality for the text-only task. Therefore we apply masks on the attention matrix, ensuring that the output embedding of each classification token contains only information from corresponding modalities. For instance, in the text-only task, we mask out all the self-attention to the image and the cross-attention between image and text. The attention masks are shown in Figure 1 Right.

4.2. Search for the Optimal Fusion Strategy

We first introduce the formulation of the search problem. Let $\alpha = \{\alpha_m\}_{m=1}^M$ denote the policy parameters, where $M$ is the total number of layers. To learn the optimal policy parameters, we formulate the parameter learning into a bilevel optimization problem. The object of optimization is to minimize the loss on a validation set $L_{val}(\alpha, \theta^*)$, where

$\text{Algorithm 1: Search for Optimal Fusion Policy.}$

**Input:** Multimodal dataset $D^{tr}$, $D^{val}$; inner-level learning rate $\gamma$; outer-level learning rate $\beta$; initialized policy parameter $\alpha$; number of iterations $K$.

1. while not converged do
2. $\{x_i^1, x_i^2, y_i\} \sim D^{tr}$; $\{x_j^1, y_j\} \sim D^{val}$
3. $\theta_0 \leftarrow \theta$
4. **Lower-Level:**
5. for $k = 0$ to $K - 1$ do
6. Sample policy $s$ with $\alpha$ using Eqn. 3
7. $\theta_{k+1} \leftarrow \theta_k - \gamma \nabla_\theta L^{tr}(x_i^1, x_i^2; s; \theta)$
8. end
9. $\theta^* \leftarrow \theta_K$
10. **Upper-Level:**
11. Sample policy $s$ with $\alpha$ using Eqn. 3
12. $\alpha^* \leftarrow \alpha - \beta \nabla_\alpha L^{val}(x_i^1, x_i^2, s; \theta^*)$
13. end

the optimal weights $\theta^*$ are obtained by minimizing the the training loss $L^{tr}(\theta, \alpha^*)$. The optimization problem is formulated as follows:

$$\min_{\alpha} L^{val}(\theta^*, \alpha),$$

$$\text{s.t. } \theta^* = \arg \min_{\theta} L^{tr}(\theta, \alpha^*).$$

(2)

Next, we describe how to generate the fusion strategy using policy parameters. Existing work on policy learning generally assumes $\alpha_m$ to be bivariate [36], resulting in a search space with $2^M$ possible policies². However, the search space can be significantly reduced. In multimodal fusion, one usually conducts fusion starting from a certain

²Two actions, fusion or not fusion, for each layer and $M$ layers in total.
layer until the last layer. Following this convention, we set $\alpha_m$ to be univariate, leading to a search space with $M$ policies. Let $Q$, a lower triangular matrix with all non-zero elements equal to one and with the size of $M \times M$, denote all the policies. Let $s$ denote the final policy. To obtain a policy, we first softmax the policy parameter to get a soft policy: $s_h = \text{softmax}(\alpha)$. Then, we convert $s_h$ into a hard policy using one-hot encoding with the differential trick\(^3\): $s_h = \text{onehot}(s_h)$. The final policy can be obtained by sampling from $Q$ using the hard policy $s_h$.

$$s = (Q, s_h).$$  \hspace{1cm} (3)

Our method significantly reduces the search space, resulting in an differentiable and easy-to-train policy learning process. The overall method is shown in Algorithm 1. Once the optimal policy is learned, we fixed the policy to retrain the model $\theta$ using the whole training set.

5. Experiments

In this section, we analyze the performance of our approach on three multimodal datasets and aim to answer the following questions: (1) Does the Transformer model perform well with modal-complete data? (Sec. 5.3) (2) Does the proposed method improve the robustness of backbone against missing-modal data? (Sec. 5.4) (3) Why different datasets prefer different layers for multimodal fusion? (Sec. 5.5) (4) What factors affect the effectiveness of our method? (Sec. 5.6)

5.1. Datasets and Metrics

Datasets. MM-IMDb [2] has two modalities: image and text. The target task is to predict the genres of a movie using image, text, or both. This task is multi-label classification, as each movie might have multiple genres. This dataset contains 25,956 image-text pair and 23 classes.

UPMC Food-101 [43] is a classification dataset composed of text and images. The UPMC Food-101 categories are identical to one of the largest publicly available food image datasets: the ETHZ Food-101 [4]. In the UPMC Food-101, image and text pairs are noisy since all the images are obtained in an uncontrolled environment. This dataset contains 90,704 image-text pairs and 101 classes.

Hateful Memes [17] is another challenging multimodal dataset that focuses on identifying hate speech in memes. It is constructed to fail models that rely on single modality and multimodal models are likely to perform well: challenging samples (“benign confounders”) are added to the dataset to make relying on unimodal signals more difficult. Hateful Memes contain exactly 10k memes.

Metrics. For MM-IMDb dataset, following previous works [15, 27, 41], we use F1 Micro, F1 Macro, F1 SAM-

5.2. Implementation Details

Multimodal backbone. We use ViLT as the backbone since it represents the common design of multimodal transformer. ViLT [18] is a pure Transformer-based model that does not rely on modality-specific sub-models to extract features, and multiple objectives are used to pre-train the model, e.g., Image Text Matching (ITM) and Masked Language Modeling (MLM).

Inputs. For image modality, we resize the input image into $384 \times 384$. Following [7], we extract $32 \times 32$ patches from the input images, yielding a total of $12 \times 12 = 144$ patches per image. For text modality, we adopt bert-base-uncased tokenizer to tokenize text inputs. The maximum length of the text sequences is various across different datasets: 1024 (MM-IMDb), 512 (Food-101), and 128 (Hateful Memes).

Network training. We use Adam optimizer [19] in all experiments with different learning rates for network training and policy learning. For network training, the base learning rate is $3 \times 10^{-5}$ and weight decay is $2 \times 10^{-2}$. For policy learning, the base learning rate is $3 \times 10^{-3}$ and weight decay is $3 \times 10^{-5}$. Model parameters are initialized using the pre-trained weights provided by ViLT [18].

5.3. Performance on the Full Test Set

We compare our model with other baseline models under the “full” evaluation setting, where all modalities are observed.

In Tables 3, 4, and 5, we report results on MM-IMDb, UPMC Food-101, and Hateful Memes, respectively. Our results are either state-of-the-art or on par with other methods: on MM-IMDb, compared to CentralNet, we achieve an F1 Weighted of 64.4 in comparison to 63.1; on UPMC Food-101, compared to MMBT, the most similar model to ours, we achieve an accuracy of 92.0 in comparison to 92.1; on Hateful Memes, we achieve comparable performance to MMBT (70.2 vs. 72.2). The results demonstrate the superiority of our model.

5.4. Performance on the Missing Test Set

The “missing” test set. We follow a conventional setting [39] to evaluate the model robustness against missing-modal data, in which the training data are modal-complete, while the testing data are modal-incomplete. We denote the full-modal train/test set as 100% Image + 100% Text and the missing-modal test set as 100% Image + $\eta$% Text, where
Figure 2. Comparison of Transformer robustness [18] on MM-IMDb [2] (left), UPMC Food-101 [43] (middle), and Hateful Memes dataset [17] (right). We adopt ViLT [18] as the backbone. Models are trained with 100% text + 100% image and tested with $\eta\%$ text + 100% image. “Image only” refers to the single modality setting – only image modality is used for training and testing. Our method significantly improves model robustness, especially when the modality is severely missing.

$\eta\%$ is percentage of observed modality and $\eta$ indicates the severity of modality missing. The smaller the $\eta$, the severer the modality missing. When $\eta = 0$, we evaluate the model’s unimodal performance.

**Robustness to missing modality.** We compare Transformer robustness on three multimodal datasets. Results are shown in Figure 2. As shown, our method improves Transformer robustness on datasets with dominant modalities. Specifically, on MM-IMDb and UPMC Food-101, the model performance degrades as $\eta$ decreases. When text modality is severely missing, i.e., only 10% are available, the multimodal performance is even worse than the unimodal one. For example, on MM-IMDb, the baseline model achieves an F1 Macro of 23.1, which is 34.0% lower than the unimodal one of 35.0; on UPMC Food-101, the baseline model achieves Accuracy of 58.4 which is 18.3% lower than the unimodal Accuracy of 73.3. However, in our method, the Transformer model still maintains good performance when modality is severely missing (only 10% text modality is observed), i.e., on MM-IMDb, our model yields F1 macro of 37.3 which is 6.0% larger then unimodal F1 Macro of 35.0; on UPMC Food-101, our method achieves an accuracy of 73.3, which is 2.5% greater than the unimodal accuracy of 71.5.

Our method improves Transformer robustness on datasets with equally important modalities. Different from MM-IMDb and UPMC Food-101, the Hateful Memes dataset does not have dominant modalities. We observe that the multimodal performance is always superior to the unimodal one. In this dataset, our method outperforms the baseline model when tested with modal-complete and modal-incomplete data. As shown in Figure 2, when only 10% of text is observed, our model yields an AUROC of 59.6, which is 2.8% higher than the baseline (58.0); when full modalities are observed, our model outperforms the baseline by 2.3%.

**5.5. Analysis of Optimal Fusion Strategy**

We visualize the optimal policy of three datasets in Figure 3. We observe that late fusion is preferred on MM-IMDb, while early fusion is preferred on Hateful Memes. The learned policy is consistent with the characteristics of each dataset. Recall that in Sec. 3.3, the depth of the fusion layer influences Transformer capacity to model cross-modality relations. The deeper the fusion layers, the lower the capacity. On MM-IMDb, the dominant modality text (plot descriptions) provides more details of the movie genres than the image modality (poster). It is reasonable that the model adopts a late fusion strategy since the prediction task can easily be addressed utilizing the dominant modality, and modeling cross-modal relations only brings marginal gains. In contrast, Hateful Memes dataset is constructed to fail models that rely on a single modality by adding challenging samples (“benign confounders”) to the dataset. Therefore, to handle this dataset, the model should have enough capacity to model cross-modal relations. For a dataset that relies on both modalities to make an accurate prediction, it is reasonable for our method to learn an early fusion strategy.

**5.6. Ablation Study**

**Comparison with a new baseline.** Training with missing modalities is a simple way to improve model robustness. We implement this method as a new baseline. Results are shown in Table 7. Our experiments show that this simple method does not work. As shown, the performance of the new baseline is even worse than the unimodal baseline (Image only) on Food-101 and Hateful Memes.
Learned optimal policy for MM-IMDb. MM-IMDb: F1 Macro = 23.3, Hateful Memes: AUROC = 58.0.


Figure 3. Left: Visualization of the learned policy. Right: Example sample from MM-IMDb and Hateful Memes. Late fusion yields the best robustness on MM-IMDb, while early fusion leads to the most robust model on Hateful Memes. The reported results were obtained using the following settings: training with 100% Image + 100% Text, testing with 100% Image + 10% Text.

Table 7. Results of the new baseline: training and testing with 100% image + 30% text.

<table>
<thead>
<tr>
<th>Method</th>
<th>MM-IMDb</th>
<th>Food-101</th>
<th>Hateful Memes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image only</td>
<td>31.2</td>
<td>65.9</td>
<td>60.2</td>
</tr>
<tr>
<td>New baseline</td>
<td>40.4</td>
<td>44.3</td>
<td>59.7</td>
</tr>
<tr>
<td>Ours</td>
<td><strong>46.6</strong></td>
<td><strong>77.5</strong></td>
<td><strong>61.2</strong></td>
</tr>
</tbody>
</table>

Table 8. Ablation study on multi-task learning and optimal fusion on MM-IMDb.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>✓</td>
<td>✓</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>10%</td>
<td>23.0</td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>✓</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>10%</td>
<td><strong>37.3</strong></td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>✓</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>10%</td>
<td><strong>41.8</strong></td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>✓</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>10%</td>
<td>28.6</td>
</tr>
</tbody>
</table>

Analysis on the multi-task learning and optimal fusion layers. We conduct experiments to validate the effectiveness of each component under two different evaluation settings, i.e., 30% or 10% text are available. Results are shown in Table 8. Both components improve Transformer robustness. Furthermore, we find that multi-task learning contributes more than the fusion strategy. In detail, when only 10% of text is available at test, multi-task learning outperforms the optimal fusion policy by 30%.

Analysis on the attention mask. In our method, we apply masks on the attention matrix to enforce the classification tokens to leverage information only from the corresponding modalities. We study the effect of attention masks. The results are shown in Table 9. We observe that it is important to make sure that each classification token is not peeking the information from other modalities.

Table 9. Ablation study on the effect of attention mask for multi-task learning on MM-IMDb.

<table>
<thead>
<tr>
<th>Method</th>
<th>Training</th>
<th>Testing</th>
<th>F1 Macro</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Image</td>
<td>Text</td>
<td>Image</td>
</tr>
<tr>
<td>without masking</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>with masking</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

6. Conclusion

We empirically find that Transformer models are sensitive to missing-modal data. And surprisingly, the optimal fusion strategy is dataset-dependent; there does not exist a universal strategy that works in the presence of modal-incomplete data. Based on the findings, we build a robust Transformer via multi-task optimization. We develop an algorithm that automatically searches the optimal fusion strategies on different datasets. The searching for optimal fusion layers and network training are formulated into a bi-level optimization problem. Experiments across multiple benchmark datasets verify the superior robustness of our method. The limitation of our method is that multi-task learning can only ensure the multimodal performance is not worse than the unimodal one, which may not meet the requirements of safety-critical systems, such as autonomous driving. We plan to explore the effectiveness of generative-based methods, e.g., reconstructing the missing tokens to improve Transformer robustness against missing modality.

Acknowledgements

This work is partially supported by NSF (CMMI-2039857) Research Grant and Snap Gift Research Grant.
References


[31] Hai Pham, Paul Pu Liang, Thomas Manzini, Louis-Philippe Morency, and Barnabás Póczos. Found in translation: Learn-
ing robust joint representations by cyclic translations between modalities. In AAAI, pages 6892–6899, 2019. 2


[43] Xin Wang, Devinder Kumar, Nicolas Thome, Matthieu Cord, and Frédéric Precioso. Recipe recognition with large multimodal food dataset. In ICMEW, 2015. 1, 2, 3, 6, 7


