MSD: Saliency-aware Knowledge Distillation for Multimodal Understanding

Woojeong Jin[1]* Maziar Sanjabi[2] Shaoliang Nie[2] Liang Tan[2] Xiang Ren[1] Hamed Firooz[2]
[1]University of Southern California [2]Facebook AI
{woojeong.jin,xiangren}@usc.edu
{maziars,snie,liangtan,mhfirooz}@fb.com

Abstract

To reduce a model size but retain performance, we often rely on knowledge distillation (KD) which transfers knowledge from a large "teacher" model to a smaller "student" model. However, KD on multimodal datasets such as vision-language tasks is relatively unexplored, and digesting multimodal information is challenging since different modalities present different types of information. In this paper, we perform a large-scale empirical study to investigate the importance and effects of each modality in knowledge distillation. Furthermore, we introduce a multimodal knowledge distillation framework, modality-specific distillation (MSD), to transfer knowledge from a teacher on multimodal tasks by learning the teacher's behavior within each modality. The idea aims at mimicking a teacher's modality-specific predictions by introducing auxiliary loss terms for each modality. Furthermore, because each modality has different saliency for predictions, we define saliency scores for each modality and investigate saliency-based weighting schemes for the auxiliary losses. We further study a weight learning approach to learn the optimal weights on these loss terms. In our empirical analysis, we examine the saliency of each modality in KD, demonstrate the effectiveness of the weighting scheme in MSD, and show that it achieves better performance than KD on four multimodal datasets.

1 Introduction

Recent advances in computer vision and natural language processing are attributed to deep neural networks with a large number of layers. Current state-of-the-art architectures are getting wider and deeper with billions of parameters, e.g., BERT (Devlin et al., 2019) and GPT-3 (Brown et al., 2020). Such wide and deep models suffer from high computational costs and latencies at inference. To mitigate the heavy computational cost and the memory requirement, there have been several attempts to compress a larger model (a teacher) into a smaller model (a student) (Ba and Caruana, 2014; Hinton et al., 2015; Romero et al., 2015; Park et al., 2019; Müller et al., 2020). Among them, *knowledge distillation* (KD) (Hinton et al., 2015) assumes the knowledge in the teacher as a learned mapping from inputs to outputs and transfers the knowledge from a larger model to a smaller model. Recently, KD has been explored in various studies such as improving a student model (Hinton et al., 2015; Park et al., 2019; Romero et al., 2015; Tian et al., 2020; Müller et al., 2020) and improving a teacher model itself by self-distillation (Xie et al., 2020; Kim et al., 2020; Furlanello et al., 2018).

There has been a lot of interest in multimodal distillation setup such as cross-modal distillation (Gupta et al., 2016; Tian et al., 2020). Multi-

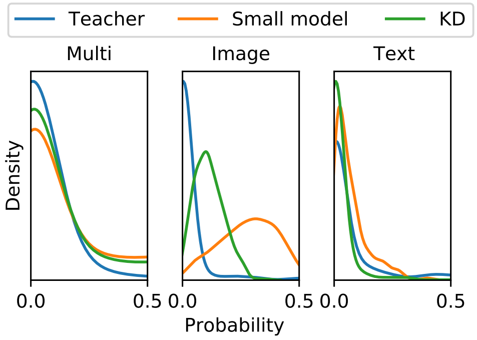


Figure 1: **Density of model outputs on Hateful-Memes:** given multimodality samples as input (Multi), given only image modality as input (Image), and given only text modality as input (Text). KD denotes a student model with knowledge distillation and the small model is a student model *without* distillation. We observe that there is still a prediction gap between the teacher and the student trained by KD. In this paper, we study saliency explanations for each modality and propose modality-specific distillation (MSD) to minimize the gap.

*The work in progress was mainly done during internship at Facebook AI.

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 3557–3569
November 7–11, 2021. ©2021 Association for Computational Linguistics

modal problems involve relating information from multiple sources. For example, visual question answering (VQA) requires answering questions about an image (Antol et al., 2015; Goyal et al., 2017; Gurari et al., 2018; Singh et al., 2019) and models should incorporate information from the text and image sources to answer the questions. Multimodal problems are important because many real-world problems require understanding signals from different modalities to make accurate predictions; information on the web and social media is often represented as textual and visual descriptions. Digesting such multimodal information in an effective manner is challenging due to their different types of information on each modality.

In this paper, we offer a large-scale, systematic study on the effects of each modality through saliency explanations in KD. While KD approaches can be applied to multimodal applications, the student and teacher models may significantly differ in their outputs using each modality as input. We illustrate the point in Fig. 1. To minimize the gaps, we introduce a multimodal KD framework, *modality-specific distillation (MSD)*, that aims to mimic the teacher's modality-specific predictions.

We show that the samples' modalities have a different amount of information. Based on this observation, we improve the knowledge transfer by splitting the multimodality into separate modalities, using them as additional inputs, and thus distilling the modality-specific behavior of the teacher. MSD introduces auxiliary losses per modality to encourage each modality to be distilled effectively.

To maximize the effect of modality-specific distillation, we investigate multiple *weighting schemes* to balance out the auxiliary losses. One of the weighting schemes is based on *modality saliency scores* that are proxy scores to modality importance. Furthermore, we leverage a meta-learning method to introduce *weight-learning* to automatically learn optimal weights per sample per modality.

2 Preliminaries

In this section, we define notations and revisit conventional knowledge distillation (KD).

2.1 Problem Definition

Given a trained and frozen teacher model T and a student model S, the output of our task is a trained student model. Our goal is to transfer knowledge from the teacher to the student on multimodal datasets. We let f_T and f_S be functions of the teacher and the student, respectively. t and s refer to the softmax output of the teacher and the student. Typically the models are deep neural networks and the teacher is deeper than the student. The function f can be defined using the output of the last layer of the network (e.g., logits). X is a multimodal (language-vision) dataset, X^t refers to only the text modality of X, X^v refers to only the image modality of X, and x_i is a dataset instance. In this work, we focus on one text and one image modalities, but it is easy to extend the work to more/other modalities.

2.2 Conventional Knowledge Distillation

In knowledge distillation (Hinton et al., 2015), a student is trained to minimize a weighted sum of two different losses: (a) cross entropy with hard labels (one-hot encodings on correct labels) using a standard softmax function, (b) cross entropy with soft labels (probability distribution of labels) produced by a teacher with a temperature higher than 1 in the softmax of both models. The temperature controls the softness of the probability distributions. Thus, the loss for the student is defined as:

$$\mathcal{L}_{\text{student}} = \lambda \mathcal{L}_{\text{CE}} + (1 - \lambda)\mathcal{L}_{\text{KD}}, \tag{1}$$

where $\mathcal{L}_{\text{CE}}$ is a standard cross-entropy loss on hard labels, $\mathcal{L}_{\text{KD}}$ is a distillation loss, which is a cross-entropy loss on soft labels, and $\lambda \in [0, 1]$ controls the balance between hard and soft targets.

To be specific, knowledge distillation (Hinton et al., 2015) minimizes the Kullback-Leibler divergence between soft targets from a teacher and probabilities from a student. The soft targets (or soft labels) are defined as softmax on outputs of f_T with temperature τ. The distillation loss is defined as follows:

$$\mathcal{L}_{\text{KD}} = \tau^2 \frac{1}{|X|} \sum_{x_i \in X} \text{KL}(t(x_i; \tau), s(x_i; \tau))), \tag{2}$$

where $t(x_i; \tau) = \sigma(\frac{f_T(x_i)}{\tau})$, $s(x_i; \tau) = \sigma(\frac{f_S(x_i)}{\tau})$, σ is a softmax function. The temperature parameter τ controls the entropy of the output distribution (higher temperature τ means higher entropy in the soft labels). Following Hinton et al. (2015), we scale the loss by τ^2 in order to keep gradient magnitudes approximately constant when changing the temperature. We omit τ for brevity.

Limitations. This KD can be applied to multimodal setups and student models in this distillation are directly trained to mimic a teacher's outputs. As a result, the student and teacher models may significantly differ in outputs with a single-modality input, i.e., modality-specific outputs, which may lead to inefficient distillation (Fig. 1). To better mimic the teacher's behaviors, we introduce a multimodal KD approach, modality-specific distillation, in the next section.

3 Analysis Setup

In this section, we introduce a multimodal KD approach, modality-specific distillation, to understand the importance of each modality (§3.1), experimental setup (§3.2), and datasets for the experiments (§3.3).

3.1 Modality-specific Distillation

The idea of MSD is to feed each modality as a separate input into a teacher and a student, and transfer the modality-specific knowledge of the teacher to the student. Specifically, MSD introduces two loss terms, $\mathcal{L}_{\text{textKD}}$ and $\mathcal{L}_{\text{imageKD}}$ to minimize difference between probability distributions between the teacher and the student given each modality (assuming text and image as the only two modalities).

$$\mathcal{L}_{\text{textKD}} = \tau^2 \frac{1}{|X_t|} \sum_{x_i \in X_t} \text{KL}(t(x_i), s(x_i)). \quad (3)$$

$\mathcal{L}_{\text{imageKD}}$ is similarly defined; the input is the image modality instead.

With above two auxiliary losses, the MSD loss for the student is defined as follows:

$$\begin{aligned} \mathcal{L}_{\text{MSD}} &= \sum_{x_i \in X} w_i \mathcal{L}_{\text{KD}}(x_i) \\ &+ \sum_{x_i \in X^v} w_i^v \mathcal{L}_{\text{imageKD}}(x_i) + \sum_{x_i \in X^t} w_i^t \mathcal{L}_{\text{textKD}}(x_i), \end{aligned} \quad (4)$$

where we omit the scaling factor $\tau^2 \frac{1}{|X|}$ for brevity. $w_i, w_i^t, w_i^v \in [0, 1]$ control the balance between three distillation losses. These weights determine the importance of each modality and we hypothesize that the choice of weighting approaches affects the student's performance. We will introduce four weighting schemes for distillation losses and discuss each of them in §4.

3.2 Experimental Setup

Through our empirical analysis, we aim to answer the following questions:

- **Q1.** How salient is each modality for predictions?
- **Q2.** Can the saliency explanations aid students?
- **Q3.** Can we learn a sample weighting strategy to better aid students?
- **Q4.** Is the student with the weighting strategies consistent with the teacher?
- **Q5.** Can this be applicable to other distillation methods?

We first define saliency scores for modalities to investigate how salient each modality is for predictions. (Q1). Then, we analyze the influence in downstream task performance brought by different weighting schemes for $w_i, w_i^t, w_i^v \in [0, 1]$ in MSD (Q2 and Q3). For Q4, we examine the student model's sensitiveness to changes in modalities. Lastly, we try to understand the effect of MSD in various distillation approaches (Q5).

To this end, we use Conventional KD (Hinton et al., 2015) as a base distillation approach for MSD. In addition, we include several distillation baselines including Conventional KD (Hinton et al., 2015), FitNet (Romero et al., 2015), RKD (Park et al., 2019), and SP (Tung and Mori, 2019) for comparison. Other distillation approaches are applicable to MSD and we will discuss the results using other KD approaches in our experiments. To perform analysis, we adopt VisualBERT (Li et al., 2019), a pre-trained multimodal model, as the teacher model and TinyBERT (Jiao et al., 2020) as a student model. VisualBERT consists of 12 layers and a hidden size of 768, and has 109 million parameters, while TinyBERT consists of 4 layers and a hidden size of 312, and has 14.5 million parameters. We use the region features from images for both the teacher and the student and fine-tune the student on each dataset. For training the weight learner we use the datasets' validation set as meta data. We find the best hyperparameters on the validation set.

3.3 Datasets and Evaluation Metrics

To answer the questions, we select four multimodal datasets: Hateful-Memes (Kiela et al., 2020) MM-IMDB (Arevalo et al., 2017), Visual Entailment (SNLI-VE) (Xie et al., 2019; Young et al., 2014), and VQA2 (Goyal et al., 2017).

The Hateful-Memes dataset consists of 10K multimodal memes. The task is a binary classification problem, which is to detect hate speech in multimodal memes. We use Accuracy (ACC) and AUC as evaluation metrics for hateful memes.

The MM-IMDB (Multimodal IMDB) dataset consists of 26K movie plot outlines and movie posters. The task involves assigning genres to each movie from a list of 23 genres. This is a multi-label prediction problem, i.e., one movie can have multiple genres and we use Macro F1 and Micro F1 as evaluation metrics following (Arevalo et al., 2017).

The goal of Visual Entailment is to predict whether a given image semantically entails an input sentence. Classification accuracy over three classes (“Entailment", “Neutral" and “Contradiction") is used to measure model performance. We use accuracy as an evaluation metric following (Xie et al., 2019).

The task of VQA2 is to correctly answer a question given an image. VQA2 is built based on the COCO (Lin et al., 2014) and is split into train (83k images and 444k questions), validation (41k images and 214k questions), and test (81k images and 448k questions) sets. Following the experimental protocol in BUTD (Anderson et al., 2018), we consider it a classification problem and train models to predict the 3,129 most frequent answers. We test models on test-dev of the VQA2 dataset.

4 Modality Weighting Methods

For the analysis, we introduce three categories of weighting schemes for MSD, presented in the order of complexity: a) population-based (§4.1), b) saliency-based (§4.2) weighting approaches for the losses, and c) weight-learning approach (§4.3) to find the optimal weights.

4.1 Population-based Weighting

Population-based weighting is to assign weights depending on a modality; we give constant weights (w_i, w_i^v, w_i^t) for each loss term in equation (4). This weighting approach assumes the weights are determined by the types of modality. Best weights or coefficients for each loss term are obtained by grid search on the validation set. However, population-based weighting is limited because it does not assign finer-grained weights to each data instance; each data instance might have different optimal weights for the loss terms. This is what we pursue next in saliency-based weighting.

4.2 Saliency-based Weighting

While we observe prediction gaps between the teacher and the student (Fig. 1) on each modality, it is unclear which modality leads to such gaps between them and how salient modality is for predictions. Saliency-based weighting is to give different weights to each loss term depending on a data sample based on its saliency of each modality. The assumption is that each data point has different optimal weights for knowledge distillation. By assigning instance-level weights, we expect better learning for the student to mimic the teacher's modality-specific behavior. As it is not possible to tune sample weights as separate hyper-parameters, we instead propose to use simple/intuitive fixed weighting functions, described as follows. Obviously, the next step to this approach would be to learn this weighting function alongside the rest of the model, i.e. weight learning, which we discuss further in §4.3.

To better understand how these modalities affect the predictions, we first define saliency scores for modalities per sample. Similar to Li et al. (2016), we erase one of the modalities and measure the saliency score by computing the difference between two probabilities. Although the saliency scores can be defined on all inputs, we limit our analysis to explanations to different modalities in this work.

Quantifying Saliency of Modality. Given a teacher model t and a multimodal dataset, we define a saliency score as follows:

$$S(m) = \delta(t(x), t(x_{-m})), \tag{5}$$

where m denotes a modality and x_{-m} denotes an input after masking out the corresponding modality input. δ is a function to measure difference between $t(x)$ and $t(x_{-m})$. We exploit teacher's output to compute saliency scores. We introduce two saliency-based weighting approaches with different δ functions.

KL divergence-based weighting. In this weighting approach, δ is defined as Kullback–Leibler (KL) divergence which measures the distance between two probability distributions. Thus, δ measures distance between predictions with multimodality and predictions by erasing one modality. Thus, weights for loss terms are defined as $w_i^v = g(S_{i,t})$ and $w_i^t = g(S_{i,v})$, where $g = \tanh(\cdot)$ to ensure the weights are in the range $[0, 1]$. In this strategy, we assign $w_i = 1$ for the loss term for multimodality. Note that in this strategy we do not explicitly use the true labels to decide the distillation weights, and we use the teacher's predictions instead.

Loss-based weighting. Another idea of saliency-based weighting is to weight terms depending on how different loss of predictions with one modality is from loss of predictions with multimodality. We explicitly use the true labels to measure the loss, i.e., cross-entropy loss. If the loss of predictions with one modality is similar to that with multimodality, then we consider the modality salient for predictions. Thus, the weights are defined as

$$w_i : w_i^v : w_i^t = 1 : \frac{h(t(x_i))}{h(t(x_i^v))} : \frac{h(t(x_i))}{h(t(x_i^t))}, \qquad (6)$$

where $h(x) = -\sum_{j=1}^{c} y_{i,j} \log x$ and $y_{i,j} \in \{0, 1\}$ are the correct targets for the j-th class of the i-th example. In this case, we also assign weights w_i for multimodality depending on the other two weights. In order to choose the actual weights, we add a normalization constraint such that, $w_i + w_i^v + w_i^t = 1$. It is worth noting that in this weighting scheme, the actual labels are directly used in deciding the weights unlike the previous one.

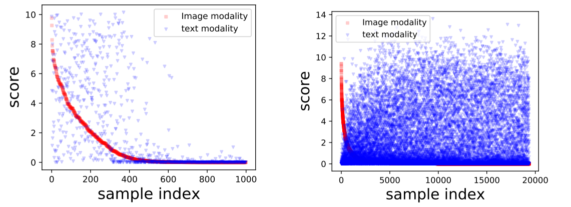
4.3 Weight Learning

Although the aforementioned weighting schemes are intuitive, there is no reason to believe they are the optimal way of getting value out of modality-specific distillation. Moreover, it is not trivial to get optimal weight functions since it depends on a dataset. Thus, we propose a weight-learning approach to find optimal weight functions. Inspired by (Shu et al., 2019), we design weight learners to find the optimal coefficients. (w_i, w_i^v, w_i^t) is defined as follows:

$$(w_i, w_i^v, w_i^t) = f(t(x_i), t(x_i^v), t(x_i^t); \mathbf{\Theta}) = \mathbf{w}(\mathbf{\Theta}), \qquad (7)$$

where $\mathbf{\Theta}$ defines the parameters for the weight learner network, a Multi-Layer Perceptron (MLP) with a sigmoid layer, which approximates a wide range of functions (Csáji et al., 2001). In general, the function for defining weights can depend on any input from the sample; but here we limit ourselves to the teacher's predictions.

Weight-Learning Objective. We assume that we have a small amount of unbiased meta-data set $\{x_i^{(\text{meta})}, y_i^{(\text{meta})}\}_{i=1}^{M}$, representing the meta knowledge of ground-truth sample-label distribution, where M is the number of meta samples and



(a) Hateful-Memes (b) MM-IMDB

Figure 2: **Saliency scores in the Hateful-memes and MM-IMDB test sets.** Saliency scores of text modality are mostly higher than those of image modality in MM-IMDB while Hateful-Memes does not show such a global pattern.

$M \ll N$. In our setup, we use the validation set as the meta-data set. The optimal parameter $\mathbf{\Theta}^*$ can be obtained by minimizing the following cross-entropy loss:

$$\mathcal{L}_{\text{meta}}(\mathbf{w}^*(\mathbf{\Theta})) = -\frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{c} y_{i,j} \log(s(x_i; \mathbf{w}^*(\mathbf{\Theta})), \qquad (8)$$

where $\mathbf{w}^*$ is an optimal student's parameter, which is defined as follows:

$$\mathbf{w}^*(\mathbf{\Theta}) = \arg\min_{\mathbf{w}} \mathcal{L}_{\text{student}}(\mathbf{w}, \mathbf{\Theta}). \qquad (9)$$

w^* is parameterized by Θ, a weight learner's parameter.

The weight learner is optimized for generating instance weights that minimize the average error of the student over the meta-data set, while the student is trained on the training set with the generated instance weights from the weight learner. The algorithm for weight learning is described in §A of appendix.

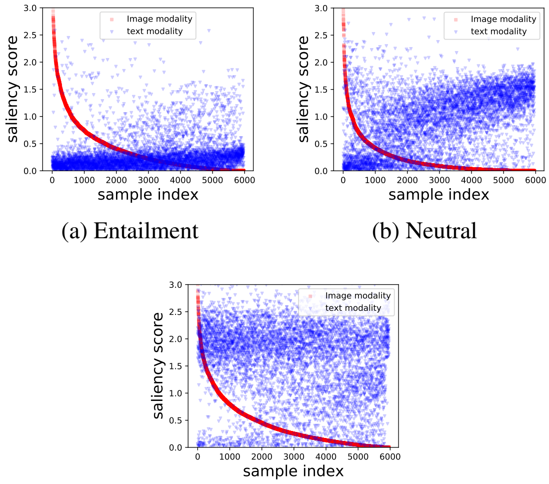
5 Empirical Analysis

In this section, we revisit and discuss the questions we raised in §3.2.

Q1. How salient is each modality for predictions? To answer the question, we visualize saliency scores in the Hateful-Memes, MM-IMDB, and SNLI-VE datasets in Figs. 2 and 3. We use KL divergence in Eq. (5). We observe that the MM-IMDB dataset shows higher saliency scores of text modality than those of image modality, which implies that text information has important information in general. On the other hand, Hateful-Memes dataset does not show such a global pattern but one

Table 1: **Main Results.** Mean results (±std) over five repetitions are reported. MSD outperforms all the KD approaches. Here, we use MSD on top of conventional KD (Hinton et al., 2015). Also, our weight learning for weights shows the best performance.

Method	Hateful-Memes		MM-IMDB		SNLI-VE	VQA2 (D)
	ACC	AUC	Macro F1	Micro F1	ACC	ACC
Teacher	65.28	71.82	59.92	66.53	77.57	70.91
Small model	60.83 (±0.20)	65.54 (±0.25)	38.78 (±4.03)	58.10 (±1.23)	72.30 (±0.35)	64.20 (±0.56)
Conventional KD (Hinton et al., 2015)	60.84 (±1.50)	66.53 (±0.27)	41.76 (±4.72)	58.96 (±1.62)	72.61 (±0.55)	64.70 (±0.85)
FitNet (Romero et al., 2015)	62.00 (±0.26)	67.13 (±0.51)	46.21 (±2.12)	60.46 (±0.30)	73.06 (±0.50)	**68.08 (±1.24)**
RKD (Park et al., 2019)	61.43 (±0.40)	67.03 (±0.21)	51.16 (±1.64)	62.52 (±0.70)	73.09 (±0.53)	64.22 (±0.57)
SP (Tung and Mori, 2019)	61.70 (±1.10)	66.11 (±0.45)	49.07 (±0.82)	61.41 (±0.34)	73.00 (±0.98)	64.15 (±0.81)
MSD (Population)	62.15 (±1.71)	67.56 (±1.21)	51.85 (±0.34)	62.13 (±0.19)	**73.64 (±0.54)**	64.86 (±0.63)
MSD (Saliency, KL div)	62.78 (±1.00)	67.94 (±0.52)	49.20 (±1.27)	61.84 (±0.49)	73.34 (±0.48)	64.93 (±0.48)
MSD (Saliency, Loss)	63.27 (±0.45)	67.72 (±0.82)	51.02 (±0.70)	62.05 (±0.45)	73.52 (±0.54)	64.89 (±0.58)
MSD (Weight learning)	**63.86 (±1.28)**	**68.30 (±0.62)**	**53.12 (±0.08)**	**63.00 (±0.09)**	73.58 (±0.23)	64.35 (±1.56)



(a) Entailment

(b) Neutral

(c) Contradiction

Figure 3: **Saliency scores in the SNLI-VE dev set.** We observe that saliency scores for text modality are correlated with labels. For the "Entailment" label, scores for text modality are relatively lower, while they are higher for the "Contradiction" label.

can observe some correlations for individual instances. In Fig. 3, we notice that saliency scores are correlated with labels in SNLI-VE. For the "Entailment" label, scores for text modality are relatively lower, while they are higher for the "Contradiction" label, which implies the role of text modality is vital to predict the "Contradiction" label for the teacher model.

Q2. Can the saliency scores aid students? Table 1 shows our main results on Hateful-Memes, MM-IMDB, SNLI-VE, and VQA2 datasets. The small model refers to a student model without knowledge distillation from the teacher. As is shown, existing KD approaches improve the student model on all datasets. However, the MSD approaches improve the small model substantially. Among them, saliency-based weighting outperforms population-based weighting in the Hateful-Memes dataset. We note that population-based weighting shows good improvement, which means weighting based on modality only is still very effective on multimodal datasets. Also, population-based weighting outperforms saliency-based weighting on the MM-IMDB dataset, suggesting all samples are likely to have the same preference or dependency on each modality of the dataset. We will discuss results on weight learning in Q3. Interestingly, FitNet shows the best performance in VQA2. Note that MSD is based on Conventional KD. We will discuss the results of MSD based on other KD approaches in Q5.

Q3. Can we learn a sample weighting strategy to better aid students? We observe that among weighting strategies, MSD with weight learning shows the best performance in Hateful-Memes and MM-IMDB, indicating it finds better weights for each dataset in Table 1. We also find that MSD (Weight learning) shows a similar density curve to the teacher's as shown in Fig 4, which implies that it effectively mimics the teacher's predictions. However, there is a performance gap between the teacher model and the student model (KD) in predicting true labels given a multimodal sample and each of its individual modalities. For example, given only image modality as input (the middle plot in Fig 4), there is a considerable difference between the teacher and the small model for predicting benign samples.

In addition, we measure Kullback-Leibler (KL) divergence between the teacher's outputs and other models' outputs on the MM-IMDB test set in Fig 5. This is to measure the difference between teacher's

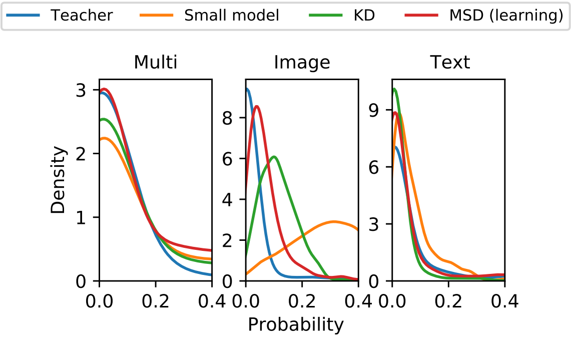


Figure 4: **Density of model outputs on samples of label 0 (not hateful) on the test set of Hateful-Memes:** given multimodal samples as input (Multi), given only image modality as input (Image), and given only text modality as input (Text). MSD with the weight-learning approach, minimizes the gap between the teacher and the student trained by KD.

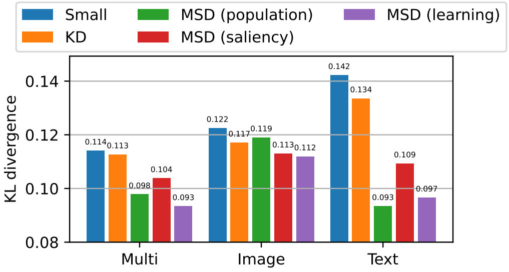


Figure 5: **Kullback-Leibler divergence on the MM-IMDB test set between the teacher's outputs and other models' outputs.** This is a measure of how the teacher's probability distribution is different from other models'. The lower divergence is, the closer a model is to the teacher.

probability distribution and others'. The MSD (learning) approach shows the smallest KL divergence from the teacher which means the student learned with MSD outputs probability distribution close to the teacher's. Notably, MSD (population) shows the smaller KL divergence than MSD (saliency), which validates that one modality is generally dominant in the MM-IMDB dataset.

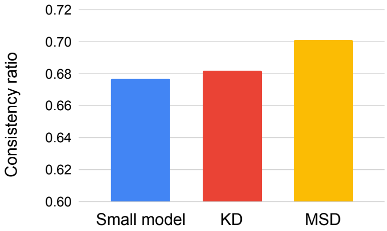


Figure 6: **Teacher-Student consistency ratio.** We investigate the student model's sensitiveness to changes in modalities. Higher ratio indicates its sensitiveness is closer to the teacher's.

Table 2: **Improvement over KD approaches with MSD.** The MSD improves existing KD approaches.

Method	Hateful-Memes		MM-IMDB		VQA2
	ACC	AUC	Macro F1	Micro F1	ACC
KD (Hinton et al., 2015)	60.84	66.53	41.76	58.96	64.70
+MSD	**62.15**	**67.56**	**51.85**	**62.13**	**64.86**
FitNet (Romero et al., 2015)	62.00	67.13	46.21	60.46	68.08
+MSD	**62.22**	**68.91**	**50.42**	**61.43**	**68.17**
RKD (Park et al., 2019)	61.43	**67.03**	51.16	62.52	64.22
+MSD	**62.30**	66.71	**52.56**	**63.27**	**64.40**
SP (Tung and Mori, 2019)	61.70	66.11	49.07	61.41	64.15
+MSD	**62.80**	**67.30**	**53.29**	**63.21**	**64.28**

Q4. Is the student with the weighting strategies consistent with the teacher? To showcase that our approach helps the student model to be more sensitive to important changes in modalities, we take examples from the Hateful-Memes test set and randomly replace one of the modalities with a modality from another sample. Hateful-Memes is a multimodal dataset and changing the modalities might or might not change the final label. In this case, we do not have the ground truth, but we use the teacher's predicted label on the newly generated sample as a proxy for ground truth and count the times that the student/small model is consistent with the teacher on these generated samples. We define the ratio of such consistent predictions over the total generated samples as "*Teacher-Student consistency ratio*". Note that none of the models have seen these samples during the training. As it can be seen from Fig. 6, the MSD approach with weight learning has a larger "Teacher-Student consistency ratio" than the small model with and without KD. This indicates that MSD not only improves the accuracy but also improves the sensitivity of the student model to better match the teacher on the changes in modalities on unseen data. Please refer to case study in §C of appendix.

Q5. Can this be applicable to other distillation methods? We present improvements over KD approaches with/without MSD in Table 2. We choose the population-based weighting approach in this experiment. Here, we use MSD on top of each KD approach. Note that the MSD approach is orthogonal to existing KD approaches. The results show the benefits of the MSD method on top of other approaches; MSD improves diverse KD methods on multimodal datasets. Notably, MSD based on FitNet also improves the accuracy on the VQA2 dataset.

6 Related Work

Knowledge Distillation. There have been several studies of transferring knowledge from one model to another (Ba and Caruana, 2014; Hinton et al., 2015; Romero et al., 2015; Park et al., 2019; Müller et al., 2020; Tian et al., 2020; Furlanello et al., 2018; Kim et al., 2020). Ba and Caruana (Ba and Caruana, 2014) improve the accuracy of a shallow neural network by training it to mimic a deep neural network by penalizing the difference of logits between the two networks. Hinton et al. (Hinton et al., 2015) introduced knowledge distillation (KD) that trains a student model with the objective of matching the softmax distribution of a teacher model at the output layer. Park et al. (Park et al., 2019) focused on mutual relations of data examples instead and they proposed relational knowledge distillation. Tian et al. (Tian et al., 2020) proposed to distill from the penultimate layer using a contrastive loss for cross-modal transfer. A few recent papers (Furlanello et al., 2018; Kim et al., 2020) have shown that distilling a teacher model into a student model of identical architecture, i.e., self-distillation, can improve the student over the teacher.

Learning for Sample Weighting. Recently, some methods were proposed to learn an adaptive weighting scheme from data to make the learning more automatic and reliable including Meta-Weight-Net (Shu et al., 2019), learning to reweight (Ren et al., 2018), FWL (Dehghani et al., 2018), MentorNet (Jiang et al., 2018), and learning to teach (Fan et al., 2018; Wu et al., 2018; Fan et al., 2020). These approaches were proposed to deal with noisy and corrupted labels and learn optimal functions from clean datasets. They are different in that they adopt different weight functions such as a multilayer perceptron (Shu et al., 2019), Bayesian function approximator (Dehghani et al., 2018), and a bidirectional LSTM (Jiang et al., 2018); and they take different inputs such as loss values and sample features. In our case, we adopt these ideas of meta-learning, specifically Meta-Weight-Net, and utilize it in a different context, i.e. multimodal knowledge distillation.

Bias in Multimodal Datasets. Different multimodal datasets were proposed to study whether a model uses a single modality's features and the implications for its generalization properties (Agrawal et al., 2018). Different approaches were proposed to deal with such problems where the model overfits to a single modality. Wang et al. (Wang et al., 2020) suggest regularizing the overfitting behavior to different modalities. REPAIR (Li and Vasconcelos, 2019) prevents a model from dataset biases by re-sampling the training data. Cadene et al. (Cadène et al., 2019) proposed RUBi that uses a bias-only branch in addition to a base model during training to overcome language priors. In our study, although we do not directly deal with the overfitting phenomena, we use different weighting schemes to better transfer the modality-specific information from the teacher to the student.

7 Conclusion

We studied knowledge distillation on multimodal datasets; we observed that conventional KD may lead to inefficient distillation since a student model does not fully mimic a teacher's modality-specific predictions. To better understand knowledge from a teacher on the multimodal datasets, we introduced saliency scores for a modality and modality-specific distillation; the student mimics the teacher's outputs on each modality based on saliency scores. Furthermore, we investigated weighting approaches, population-based and saliency-based weighting schemes, and a weight-learning approach for weighting the auxiliary losses to take the importance of each modality into consideration. We empirically showed that we can improve the student's performance with modality-specific distillation compared to conventional distillation. More importantly, we observe choosing the right weighting approach boosted the student's performance. We believe that future work can expand on our methods, and search the space of weighting approaches beyond multimodal setups.

Acknowledgements

This research is supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via Contract No. 2019-19051600007, the DARPA MCS program under Contract No. N660011924033, the Defense Advanced Research Projects Agency with award W911NF-19-20271, NSF IIS 2048211, NSF SMA 1829268, and gift awards from Google, Amazon, JP Morgan and Sony. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. We would like to thank all collaborators in USC INK lab and the reviewers for their constructive feedback that help us improve the paper.

References

Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. 2018. Don't just assume; look and answer: Overcoming priors for visual question answering. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 4971–4980. IEEE Computer Society.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6077–6086. IEEE Computer Society.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: visual question answering. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2425–2433. IEEE Computer Society.

John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A González. 2017. Gated multimodal units for information fusion. *arXiv preprint arXiv:1702.01992*.

Jimmy Ba and Rich Caruana. 2014. Do deep nets really need to be deep? In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2654–2662.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Rémi Cadène, Corentin Dancette, Hedi Ben-younes, Matthieu Cord, and Devi Parikh. 2019. Rubi: Reducing unimodal biases for visual question answering. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 839–850.

Balázs Csanád Csáji et al. 2001. Approximation with artificial neural networks. *Faculty of Sciences, Etvs Lornd University, Hungary*, 24(48):7.

Mostafa Dehghani, Arash Mehrjou, Stephan Gouws, Jaap Kamps, and Bernhard Schölkopf. 2018. Fidelity-weighted learning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Yang Fan, Fei Tian, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. 2018. Learning to teach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Yang Fan, Yingce Xia, Lijun Wu, Shufang Xie, Weiqing Liu, Jiang Bian, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. 2020. Learning to teach with deep interactions. *arXiv preprint arXiv:2007.04649*.

Tommaso Furlanello, Zachary Chase Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born-again neural networks. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 1602–1611. PMLR.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6325–6334. IEEE Computer Society.

Saurabh Gupta, Judy Hoffman, and Jitendra Malik. 2016. Cross modal distillation for supervision transfer. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2827–2836. IEEE Computer Society.

Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3608–3617. IEEE Computer Society.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. 2018. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 2309–2318. PMLR.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online. Association for Computational Linguistics.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. 2020. The hateful memes challenge: Detecting hate speech in multimodal memes. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Kyungyul Kim, ByeongMoon Ji, Doyoung Yoon, and Sangheum Hwang. 2020. Self-knowledge distillation: A simple way for better generalization. *arXiv preprint arXiv:2006.12000*.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Yi Li and Nuno Vasconcelos. 2019. REPAIR: removing representation bias by dataset resampling. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 9572–9581. Computer Vision Foundation / IEEE.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Rafael Müller, Simon Kornblith, and Geoffrey Hinton. 2020. Subclass distillation. *arXiv preprint arXiv:2002.03936*.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. 2019. Relational knowledge distillation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 3967–3976. Computer Vision Foundation / IEEE.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. 2018. Learning to reweight examples for robust deep learning. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4331–4340. PMLR.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. 2015. Fitnets: Hints for thin deep nets. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. 2019. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 1917–1928.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards VQA models that can read. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 8317–8326. Computer Vision Foundation / IEEE.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. 2020. Contrastive representation distillation. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Frederick Tung and Greg Mori. 2019. Similarity-preserving knowledge distillation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 1365–1374. IEEE.

Weiyao Wang, Du Tran, and Matt Feiszli. 2020. What makes training multi-modal classification networks hard? In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 12692–12702. IEEE.

Lijun Wu, Fei Tian, Yingce Xia, Yang Fan, Tao Qin, Jian-Huang Lai, and Tie-Yan Liu. 2018. Learning to teach with dynamic loss functions. In *Advances* *in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 6467–6478.

Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. 2019. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*.

Qizhe Xie, Minh-Thang Luong, Eduard H. Hovy, and Quoc V. Le. 2020. Self-training with noisy student improves imagenet classification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10684–10695. IEEE.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Algorithm 1: Weight-Learning Algorithm

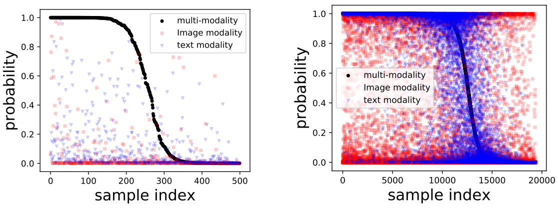
```
Input: Training data $\mathcal{D}$, Meta-data set $\hat{\mathcal{D}}$, batch size
       $n, m$, learning rates $\alpha, \beta$, max iterations $T$.
1 for $t \leftarrow 0$ to $T-1$ do
2     $\{x, y\} \leftarrow$ SampleMiniBatch$(D, n)$.
3     $\{x^{(\text{meta})}, y^{(\text{meta})}\} \leftarrow$ SampleMiniBatch$(\hat{D}, m)$.
4     $\hat{\mathbf{w}}^{(t)}(\mathbf{\Theta}^{(t)}) \leftarrow$
          $\mathbf{w}^{(t)} - \alpha \frac{1}{n} \sum_{i=1}^{n} \nabla_{\mathbf{w}} \mathcal{L}_{\text{student}}(\mathbf{w}^{(t)}, \mathbf{\Theta}^{(t)})$
5     $\mathbf{\Theta}^{(t+1)} \leftarrow$
          $\mathbf{\Theta}^{(t)} - \beta \frac{1}{m} \sum_{i=1}^{m} \nabla_{\mathbf{\Theta}} \mathcal{L}_{\text{meta}}(\hat{\mathbf{w}}^{(t)}(\mathbf{\Theta}^{(t)}))$
6     $\mathbf{w}^{(t+1)} \leftarrow$
          $\mathbf{w}^{(t)} - \alpha \frac{1}{n} \sum_{i=1}^{n} \nabla_{\mathbf{w}} \mathcal{L}_{\text{student}}(\mathbf{w}^{(t)}, \mathbf{\Theta}^{(t+1)})$
7 return Network parameters $\mathbf{w}^{(T)}, \mathbf{\Theta}^{(T)}$
```

A Weight Learning Algorithm

Finding the optimal Θ^* and w^* requires two nested loops; one gradient update of a weight learner requires a trained student on the training set. Thus, we adopt an online strategy following (Shu et al., 2019), which updates the weight learner with only one gradient update of the student. Algorithm 1 illustrates its learning process. First, we sample mini batches from the training and meta-data sets, respectively (lines 2 and 3). Then, we update the student's parameter along the descent direction of the student's loss on a mini-batch training data (line 4). Note that the student's parameter is parameterized by the weight learner's parameter. With the updated parameter, the weight leaner can be updated by moving the current parameter $\Theta(t)$ along the objective gradient of equation (8) on a mini-batch meta data (line 5). After updating the weight-learner, the student's parameter can be updated on a mini-batch training data (line 6).

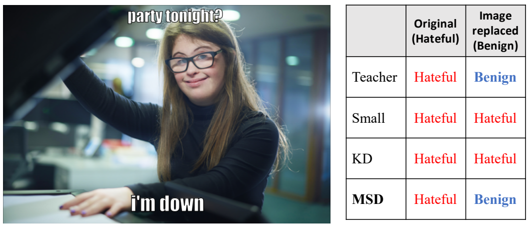
B Observation of Teacher's Predictions

Samples from multimodal datasets have different information on each modality. Fig. 7 shows a teacher model's predictions for samples in Hateful-Memes and MM-IMDB test sets. For each sample, three probabilities are calculated: 1) predictions of samples with both of its modalities, 2) predictions of samples with just its text modality, and 3) predictions of samples with just its image modality. As one can see for MM-IMDB there is a strong correlation between multimodal predictions and predictions from text modality, indicating the fact that in MM-IMDB text is a dominant modality. On the other hand, for Hateful-Memes dataset there is no such a global pattern but one can observe some correlations for individual instances. This behavior is actually expected based on the way Hateful-Memes is built to include unimodal confounders (Kiela et al., 2020). Following these observations we introduce four weighting schemes for distillation losses and discuss each of them in §4.



(a) Hateful-Memes (b) MM-IMDB

Figure 7: **Prediction probabilities of test samples for different modalities.** Black points correspond to the predictions of samples with both modalities (original input), red points do with image modality, and blue points do with text modality. The samples are ordered based on their multimodal output probabilities. There is a strong correlation between multimodal predictions and predictions from text modality in MM-IMDB, while there is no such a global pattern in Hateful-Memes.



	Original (Hateful)	Image replaced (Benign)
Teacher	Hateful	Benign
Small	Hateful	Hateful
KD	Hateful	Hateful
MSD	Hateful	Benign

Figure 8: A multimodal violating sample (Left). We further replaced its image modality with a background picture that makes it benign and examined models on both examples (Right).

C Case Study

We demonstrate the motivation behind our work through an example. Fig. 8 shows an example of a multimodal sample from Hateful Memes test dataset. The sample is violating based on both modalities together, and all models correctly predict that. To further probe the models, we replace the background image of the sample with a picture that makes the label benign. On this artificially generated sample we notice that only the teacher and MSD model correctly predict benign, while the other two models make wrong predictions (presumably by just looking at the text only).

D Hyperparameters

The teacher model is a VisualBERT (Li et al., 2019), and the student model is TinyBERT (Jiao et al., 2020). We used the MMF library and pretrained checkpoints from it for VisualBERT[1] and used a pretrained checkpoint in TinyBERT[2]. VisualBERT consists of 12 layers and a hidden size of 768, and has 109 million number of parameters, while TinyBERT consists of 4 layers and a hidden size of 312, and has 14.5 million number of parameters. For all experiments, we performed a grid search to find the best hyperparameters. We adopt the AdamW optimizer to train networks. We use a linear learning rate schedule that drops to 0 at the end of training with warmup steps of 10% maximum iterations.

Hateful-Memes. We performed a grid search over learning rates (1e-5, 3e-5, **5e-5**, 1e-4), and temperatures (1, 2, **4**, 8), and, batch sizes (**10**, 20, 30, 40, 50, 60), and the weight learner's learning rates (1e-1, 1e-2, **1e-3**, 1e-4). We set the maximum number of iterations to 5000. The balance parameter λ between cross entropy and distillation is set among (0.2, 0.4, **0.5**, 0.6, 0.8).

MM-IMDB. For MM-IMDB experiments, we follow a similar procedure, a grid search, to the Hateful-Memes. The batch size is 20, temperature is 1, and the weight learner's learning rate is 1e-4. We set the maximum number of iterations to 10000. The balance parameter λ is set to 0.5.

SNLI-VE. For Visual Entailment (SNLI-VE), the batch size is 64, temperature is 4, the student model's learning rate is 1e-4, and the weight learner's learning rate is 1e-4. We set the maximum number of iterations to 60000. The balance parameter λ is set to 0.6.

VQA2. For VQA2, the batch size is 120, temperature is 1, the student model's learning rate is 1e-4, and the weight learner's learning rate is 1e-4. We set the maximum number of iterations to 60000. The balance parameter λ is set to 0.8.

E Learning Curve

The MSD approaches can also help with training speed, measured by test metrics over training steps. Fig 9 shows the evolution of accuracy on the *test set* during training on the SNLI-VE dataset. When we train the student with MSD, training progresses faster than KD. Since the teacher provides two additional probabilities with each modality, the student learns faster and the final performance is better than KD. We observe a large performance increase early in training with the weight-learning approach, thus leading to the best accuracy. In this case, the weight learning for sample weighting finds the optimal weights for each data instance, so the student quickly learns from more important modality that is vital for the predictions.

Table 3: **Dataset Statistics.**

Stat. \ Data	**Hateful-Memes**	**MM-IMDB**	**SNLI-VE**	**VQA2**
Type	Binary	Multilabel	Multi-class	Multi-class
# Classes	2	23	3	3,129
# Examples	10,000	25,959	565,286	1,105,904
# Training	8,500	15,552	529,527	443,757
# Validation	500	2,608	17,858	214,354
# Test	1,000	7,799	17,901	447,793

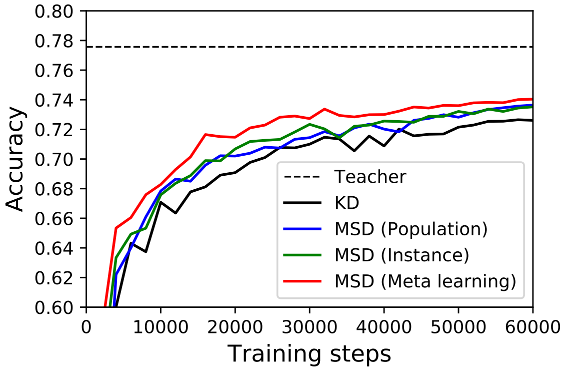


Figure 9: Test accuracy of a student on SNLI-VE during training and comparison between knowledge distillation (KD) and modality-specific distillation (MSD) with population-based weighting, instance-wise weighting, and weight learning for weights.

[1] https://mmf.sh

[2] https://github.com/huawei-noah/Pretrained-Language-Model/tree/master/TinyBERT