How Models Memorize vs. Generalize: Intervening via Features and Labels

Anonymous authors
Paper under double-blind review

Abstract

Although deep learning is widely adopted for its capability to fit training data effectively, it often memorizes outliers and/or mislabeled instances, a phenomenon known as label memorization. Despite this, there has been no clear distinction between memorization and generalization at the feature level, and how they interact with label memorization. Hence, in this paper, first, we precisely distinguish feature memorization from feature generalization and define the conditions under which they occur. Second, we investigate the interactions among feature memorization, feature generalization, and label memorization, revealing that label memorization suppresses feature memorization while causing feature generalization. These findings offer a systematic understanding of memorization and generalization in deep neural networks.

1 Introduction

Despite of great advancement of deep learning models, they tend to memorize certain samples due to their complex nature (Maini et al., 2023) which is known as label memorization (Feldman & Zhang, 2020; Feldman, 2020), where models memorize the labels in training without learning the relevant patterns, and thus cannot generalize well to unseen data (Singhal & Kim, 2025). Recent studies (Jiang et al., 2018; Toneva et al., 2018; Stephenson et al., 2021) show that models tend to learn easier samples and memorize harder ones. Easier samples usually tend to visually match their genuine ground truth label, while harder samples can be ambiguous and/or atypical even with a correct label, or labeled incorrectly, (Baldock et al., 2021; Ethayarajh et al., 2022). Memorization of long-tailed samples happens since the model just strives to minimize the error based on the empirically seen samples during training, based on Empirical Risk Minimization (ERM) (Vapnik, 1998).

In addition to label memorization, understanding how the model perceives features (Izmailov et al., 2022; Sagawa et al., 2020; Hartley et al., 2023; Li et al., 2023; Pezeshki et al., 2021) is also important in models' generalizability. These attributes significantly influence the model's learning process, often leading to suboptimal performance, especially when the test samples do not conform to these features (Yang et al., 2022; Ye et al., 2024; Ming et al., 2022). However, existing studies have an unclear distinction between feature memorization and feature generalization, even though the two have a fundamentally distinctive influence on generalization. This lack of clarity poses a fundamental obstacle: without formally distinguishing feature memorization from feature generalization, it is difficult to design principled methods that can mitigate memorization beyond the label level. Hence, we first aim to clarify the following question: ***What is the distinction between feature memorization and feature generalization?*** Note that although prior work on long-tail memorization (Feldman & Zhang, 2020) was concerned only with characterizing memorization at the sample level, no prior work has ever articulated the difference between "memorizing" and "generalizing" a feature within a sample, which makes our work *novel*. Once establishing this distinction, we examine how feature memorization and feature generalization interact with label memorization. This is because memorization is limited not only to features, but also to labels, particularly when mislabeled samples caused by data annotation errors can also be memorized (Feldman, 2020). Therefore, this motivates our second question: ***Do label memorization Influence feature memorization and feature generalization?***

To address these research gaps, we first provide a formal distinction between feature memorization and generalization: when a model recognizes a feature in training but fails to generalize to unseen

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

data, it exhibits feature memorization, whereas if it extracts patterns that generalize to test-time predictions, it achieves feature generalization. This distinction allows us to identify the conditions under which each phenomenon arises. Secondly, we investigate how label memorization interacts with feature memorization and generalization, where our analysis reveals that label memorization suppresses feature memorization while causing feature generalization. In summary, our novel findings and core contributions of our paper are as follows:

- Articulating a concrete distinction between feature memorization and feature generalization.
- Identifying that if the proportion of samples containing a feature is low, the model memorizes it, whereas if the proportion is high, the model generalizes it.
- Finding that label memorization suppresses feature memorization and causes feature generalization.

2 Related Work

We provide Related Work in Appendix A.

3 Preliminaries

3.1 Analogy on How Humans Process Information

In this section, we draw inspiration from human cognitive processing (Mayer, 2002; Novak & Cañas, 2008) to formalize our definitions of feature memorization and generalization in machine learning models. For example, when humans are repeatedly exposed to a specific information across diverse contexts, they learn to generalize that information to unseen environments. Conversely, when exposed to the same information only once in a single context, humans tend to memorize the specific instance without forming a generalized abstraction, hence unable to adapt to unseen settings. Drawing on this intuitive analogy, we argue that machine learning models—originally inspired by human brain architecture—process information in a similar manner. Specifically, a model tends to "memorize" a feature when its concentration in the training set is low, but learns to "generalize" the feature when its concentration is high. We expand upon this distinction in the following section.

3.2 Setup

Task. We explore how and under what conditions a model, $F(x; D)$, undergoes either memorization or generalization at the feature level. Here, $D = \{(x_k, y_k)\}_{k=1}^{N}$ is the training data where $x_k \in \mathbb{R}^{d \times d}$ is the input data and y_k is the corresponding ground truth class label. F maps input data x_k to a prediction vector of probabilities of each class label.

Feature, z_u, to track. To assess memorization and generalization at the feature level, we track a feature and examine how it is perceived by a model. We speculate that the proportion of the feature in the training data is the distinguishing factor between them. However, in current literature, there is no readily available dataset where the proportion of a certain feature can be varied in the training data. Hence, we follow (Hartley et al., 2023) to introduce a feature $\mathbf{z}_u$ that appears at most once in an individual sample in the training dataset. In this paper, we study a total of 6 different features/tokens across 3 vision and 3 language tasks and provide a detailed discussion about them in Appendix C.

3.3 Metrics to Distinguish Feature Memorization & Feature Generalization

3.3.1 Recognizing Feature in Training

To examine whether a model memorizes or learns a feature, we need to first verify if the model has recognized the feature or not during training. If the model recognized the feature z_u, only then it can either memorize/learn it. However, if it could not recognize z_u, it can neither memorize/learn the feature. Hence, we compute the Sensitivity score as explained below:

Sensitivity Score, S_{sens} & its Statistical Significance. To quantify whether the model has recognized the feature z_u or not, we define Sensitivity Score, S_{sens}, in a similar fashion as in (Hartley et al., 2023). At first, we introduce the feature z_u to a number of training samples belonging to class C_i. We complete training the model F and then evaluate whether the model has recognized z_u or not. To do so, we consider 2 variants of each of the samples of class C_i - one without z_u (i.e, x_c) and the other with z_u (i.e., x_u). We then calculate $\boldsymbol{S_{\text{sens}}}$, which is given by,

$$S_{\text{sens}} = \mathbb{E}_{x \in C_i}\left[P(C_i|x_{\text{u}}) - P(C_i|x_{\text{c}})\right] \tag{1}$$

S_{sens} represents whether the trained model has recognized feature z_u or not, by averaging the difference between likelihoods of all sample pair (x_u, x_c) belonging to class C_i.

Besides, to quantify the statistical significance of S_{sens} across all samples, we employ the one tailed t-test (Student, 1908) to compare the probabilistic distribution arrays, $X_u = P(C_i|x_{\text{u}})$ and $X_c = P(C_i|x_{\text{c}})$, for all samples $x \in C_i$. The null hypothesis states that the population means of $\big(X_u\big)$ and $\big(X_c\big)$ are equal, while the alternative hypothesis posits that the mean of $\big(X_u\big)$ is greater than that of $\big(X_c\big)$. We interpret this using the notion of $p\text{-}value$, a statistical measure used to evaluate the strength of evidence against the null hypothesis. If $p\text{-}value < 0.05$, we reject the null hypothesis, indicating a significant statistical difference between the two distributions, $\big(X_u\big)$ and $\big(X_c\big)$, whereas if $p\text{-}value \geq 0.05$, the null hypothesis is valid. Aggregating both conditions, if,

$$\boldsymbol{S_{\text{sens}} > 0} \text{ and } \mathbf{p\text{-}value < 0.05},$$

the model has recognized z_u, otherwise, it has not.

3.3.2 Examining Impact of Feature on Test Set

Once the model is trained, we also need to check the impact of the feature (if recognized) on the test-set predictions. Thus, to evaluate the impact of z_u on test-set predictions, we insert z_u into samples from a different class C_j (where $C_j \neq C_i$), and compute the corresponding four scores.

Misclassification Percentage (%), $\mathbf{MP_{j,i}}$, computes ratio (%) of misclassified test samples originally from class C_j (which contain the feature z_u) that are incorrectly predicted as class C_i, defined as:

$$\text{MP}_{\text{j,i}}(\%) = \frac{\#\text{misclassifications of } C_j \text{ samples with } z_u \text{ as } C_i}{\#\text{total misclassifications of } C_j \text{ samples with } z_u} \times 100$$

Test Accuracy (%), $\mathbf{Acc_j}$, calculates the percentage of correctly classified test samples from class C_j that contain the feature z_u. It is defined as:

$$\text{Acc}_{\text{j}}(\%) = \frac{\#\text{correct classifications of } C_j \text{ samples with } z_u}{\#\text{total test samples of } C_j \text{ containing } z_u} \times 100$$

$\text{MP}_{\text{j,i}}$ and Acc_{j} quantify the impact of a feature (if recognized) on test set predictions. However, memorization is characterized by a failure to generalize to unseen test samples. Therefore, to determine whether the recognized feature was memorized or learned, we need to compare these metrics against a reference case in which z_u was never introduced during training by defining the following two metrics:

Avg-MP-Diff (%) quantifies how training with feature z_u affects $\text{MP}_{\text{j,i}}$ for class C_j compared to when z_u was not introduced in class C_i during train. Across N seeds,

$$\text{Avg-MP-Diff (\%)} = \frac{1}{N} \sum_{seed=1}^{N} \Big(\text{MP}_{\text{j,i}}(\text{with } z_u \text{in train}) - \text{MP}_{\text{j,i}}(\text{w/o } z_u \text{in train}) \Big)$$

where, $\text{MP}_{\text{j,i}}(\text{with } z_u \text{in train})$ and $\text{MP}_{\text{j,i}}(\text{w/o } z_u \text{in train})$ represent the misclassification percentage of C_j samples containing z_u as C_i, when trained with and without z_u in C_i, respectively.

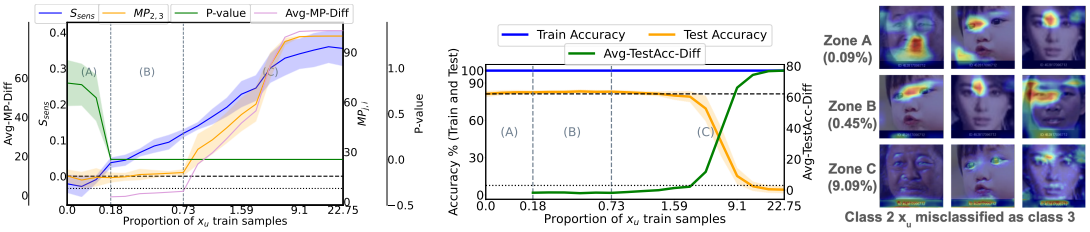
Avg-TestAcc-Diff (%) measures the mean difference in test accuracy Acc_{j} for C_j samples containing z_u, when the model is trained with vs. without z_u in C_i. Across N seeds, it is defined as:

$$\text{Avg-TestAcc-Diff (\%)} = \frac{1}{N} \sum_{seed=1}^{N} \Big(\text{Acc}_{\text{j}}(\text{w/o } z_u \text{ in train}) - \text{Acc}_{\text{j}}(\text{with } z_u \text{ in train}) \Big)$$

Here, $\text{Acc}_{\text{j}}(\text{with } z_u \text{in train})$ and $\text{Acc}_{\text{j}}(\text{w/o } z_u \text{in train})$ denote the test accuracy of C_j samples containing z_u, trained with and without z_u in C_i, respectively.

3.4 Definitions

Definition 1: Feature Memorization (FM) The model F is said to have memorized z_u when it has recognized it, but can not generalize it to the test set. As a result, samples from other classes that contain z_u are not misclassified as C_i in which z_u was inserted in training, since the model has *memorized* z_u but not learned it. It is formally defined when the following conditions are met:



(a) S_{sens}, $\text{MP}_{2,3}$, & P-value

(b) Test Acc for Class 2 x_u samples & Train Acc

(c) GradCam results for (x_u, y) ratios in class 3

Figure 1: **FM and shift to FG along with increasing x_u train samples**. We present it using S_{sens}, $\text{MP}_{2,3}$, p-value metrics, train-test accuracy gap for x_u train-test samples and the corresponding GradCam results. (UTK-Face, VGG16, z_u in class 3 in train, 2 in test, Avg-MP-Diff threshold line is at 3.5% (for σ_1) and Avg-TestAcc-Diff line is at 3% (for σ_2.)) FM starts at 0.18% (zone (B)) and shifts to FG after 0.73% (zone (C)) of x_u proportion during training, and neither happening for extremely low proportions in zone (A). We do not plot Avg-MP-Diff and Avg-TestAcc-Diff in Zone (A) because the model has not even recognized z_u, hence it can neither memorize nor generalize it.

(i) $\boldsymbol{S_{\text{sens}} > 0}$ & $\textbf{p-value} < \mathbf{0.05}$: indicating that the model recognizes the feature z_u.

(ii) $\textbf{Avg-MP-Diff} \leq \boldsymbol{\sigma_1}$& $\textbf{Avg-TestAcc-Diff} \leq \boldsymbol{\sigma_2}$: misclassification (%) and test accuracy of class C_j samples containing z_u remain almost the same regardless of the introduction of z_u during training, without generalizing to the test-set.

Definition 2: Feature Generalization (FG) The model is said to have learned the feature z_u if it recognizes the feature and generalizes it to the test set. As a result, the model would start associating z_u with class C_i during testing and incorrectly classify samples from other classes as C_i if they contain z_u, even though they do not belong to C_i. Technically, feature generalization is defined when,

(i) $\boldsymbol{S_{\text{sens}} > 0}$ & $\textbf{p-value} < \mathbf{0.05}$: indicating that the model recognizes the feature z_u.

(ii) $\textbf{Avg-MP-Diff} > \boldsymbol{\sigma_1}$ or$\textbf{Avg-TestAcc-Diff} > \boldsymbol{\sigma_2}$: misclassification (%) and accuracy of class C_j samples with z_u are significantly affected due to the association of z_u with C_i, which is reflected on the test-set.

Please note that selecting thresholds (σ_1, σ_2) to distinguish Feature Memorization and Generalization is, by nature, subjective and task-dependent. The goal of this paper is not to designate an optimal thresholding scheme (Schwarzschild et al.), but to articulate the distinction between Feature Memorization and Generalization with a consistent framework. In this paper, we fix $\sigma_1 = 3.5\%$ and $\sigma_2 = 3\%$ across all datasets, and provide a detailed analysis of the values in Appendix F.

Definition 3: Label Memorization (LM) & LM-score Label Memorization (LM) is defined as when classifier F memorizes the label y then its prediction for (x, y) changes significantly when (x, y) is removed from D. Hence, to study LM, we introduce noisy label y_L to clean pair (x, y_c), making it (x, y_L), where $y_c \neq y_L$. Then, the classifier F is trained in 2 different configurations, (i) one with (x, y_L) in D; (ii) another with (x, y_L) excluded, while achieving 100% train accuracy. We then evaluate the classifier's probabilities on (x, y_L) in both configurations. To determine whether the noisy label samples are memorized or not we use the same metric of label memorization introduced in (Feldman, 2020) and term it as LM-score, as follows:

$$\text{LM-score(x, } y_L) = P_{(x,y_L)\in D}[F(x) = y_L] - P_{(x,y_L)\notin D}[F(x) = y_L]$$

We compute this score for every noisy label sample and take an average across all of them.

3.5 Datasets and Models

All experiments are conducted across 5 random seeds to account for stochasticity in training, with additional training details in Appendix B. For **Datasets**, we use CIFAR10, CIFAR100 (Krizhevsky et al., 2009), UTKFace (Zhang et al., 2017), NICO++ (Zhang et al., 2023), Emotions (Saravia et al., 2018), & Medical Abstracts (Schopf et al., 2022). For **Models**, we employ ResNet9, ResNet18 (He et al., 2016), VGG16 (Simonyan & Zisserman, 2014), GoogleNet (Szegedy et al., 2015), MobileViT (Mehta & Rastegari), BERT (Devlin et al., 2019), & RoBERTa (Liu et al., 2019).

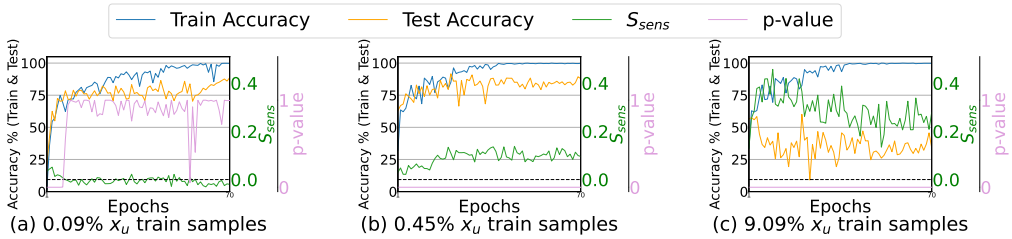
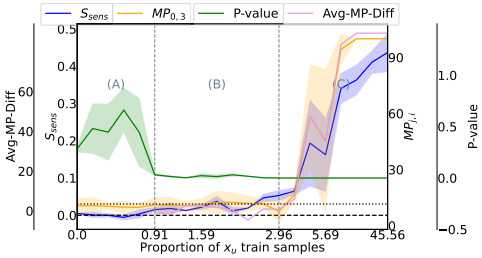
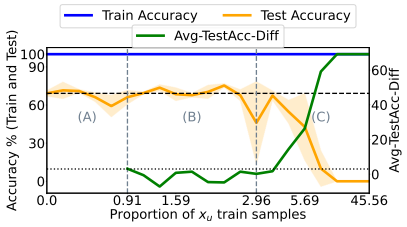


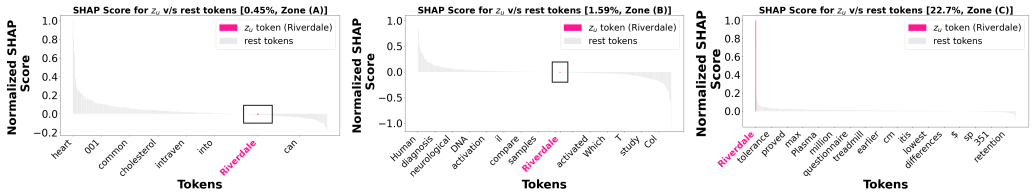
Figure 2: **Analysis of FM & FG over epochs.** We explore how FM & FG occurs over the course of training by measuring S_{sens}, p-value, and train-test accuracy gap of x_u train and test samples over epochs. (UTK-Face, VGG16, z_u in class 3 in train, 2 in test)



(a) S_{sens}, $\text{MP}_{0,3}$, & P-value



(b) Test Acc for Class 0 x_u samples & Train Acc



(c) Shapley score results for select (x_u, y) proportions in class 3

Figure 3: **FM and shift to FG along with increasing x_u train samples**. We present it using S_{sens}, $\text{MP}_{0,3}$, p-value, train-test accuracy gap for x_u train-test samples, and Shapley results, with Avg-MP-Diff threshold: 3.5% (σ_1), Avg-TestAcc-Diff: 3% (σ_2). FM starts at 0.91% (zone B) and shifts to FG after 2.96% (zone C), with neither occurring at very low proportions (zone A). Shapley values confirm these transitions, where for Zone (B) of FM, "Riverdale" has negligible importance (SHAP score ≈ 0) in testing even though it has been recognized, while for Zone (C) of FG, "Riverdale" gains importance (SHAP score $\gg 0$). (Medical Abstracts, RoBERTa, z_u in class 3 in train, 0 in test). We do not plot Avg-MP-Diff and Avg-TestAcc-Diff in Zone (A) because the model has not even recognized z_u, hence it can neither memorize nor generalize it.

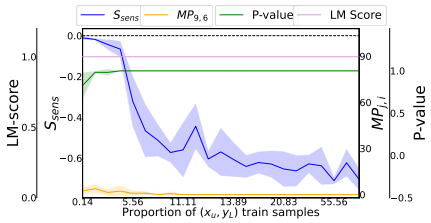
4 Feature Memorization & Feature Generalization

4.1 FM-FG Setup

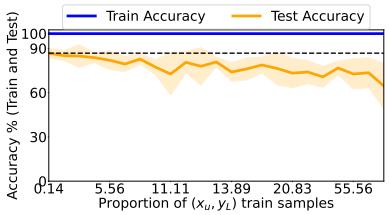
To verify when and how a feature is memorized or learned, we introduce a feature z_u in increasing proportions of training samples of class C_i, and in all samples in class C_j during inference. Then, we compute S_{sens} for class C_i and $\text{MP}_{\text{j,i}}$ and provide GradCam and Shapley values analysis of the C_j test samples containing z_u misclassified as C_i, across vision and language settings, respectively.

4.2 Increasing Proportion of Training Samples with z_u

In both vision (Fig. 1) and language (Fig. 3), we observe the emergence of 3 distinct zones, (A), (B) and (C), regarding feature memorization and generalization based on increasing proportion of training samples containing the feature z_u (where z_u='Personal ID" or z_u="Riverdale"), as discussed below.



(a) S_{sens}, $\text{MP}_{9,6}$ and P-value



(b) Test Acc for class 9 x_u samples and Train Acc

Figure 4: **Disappearance of FM and FG.** As (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, while $\text{MP}_{9,6}$ stays minimal. The train-test accuracy gap for x_u samples remains stable, indicating the disappearance of FM and FG, while *random* LM is still present. (NICO++, MobileViT, z_u in class 6 in train, 9 in test)

4.2.1 Neither Feature Memorization/Generalization: $0-0.09\%$ of z_u: zone (A)

In zone (A) of Fig. 1, for a very low proportion of z_u (0 - 0.09%), the classifier does not recognize z_u during training. This is reflected by $S_{\text{sens}} < 0$ and p-value > 0.05, indicating no evidence of either feature memorization or generalization. GradCam for 0.09% further supports this, showing no association between z_u and class 3.

4.2.2 Feature Memorization: $0.18-0.73\%$ of z_u: zone (B)

In zone (B) of Fig. 1, as the proportion of z_u in the training samples increases to 0.18 - 0.73%, S_{sens} becomes positive, and the p-value drops below 0.05, indicating that the model recognizes z_u. However, the classifier's performance remains stable, with $\text{MP}_{2,3}$ and test accuracy for class 2 samples containing z_u unchanged. The Avg-MP-Diff and Avg-TestAcc-Diff stay below the thresholds of 3.5 and 3, respectively, suggesting that the model has memorized z_u without generalizing to the test set. This aligns with our Feature Memorization definition, supported by GradCam plots showing that, at 0.45% z_u proportion, the model does not associate z_u with class 3, indicating memorization.

4.2.3 Feature Generalization: $0.73\% <$ of z_u: zone (C)

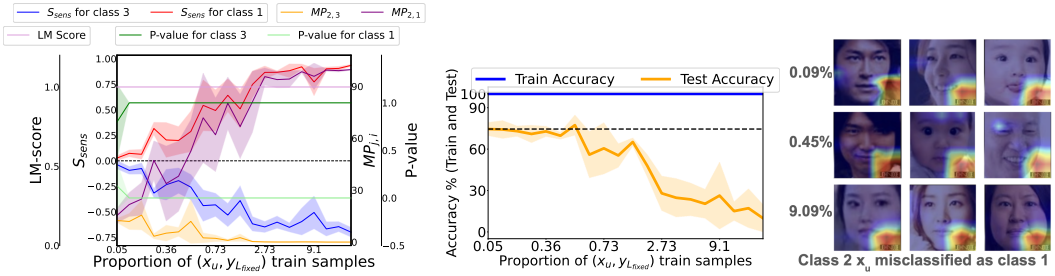
In zone (C) of Fig. 1, when the proportion of z_u exceeds 0.73%, both S_{sens} and $\text{MP}_{2,3}$ increase significantly, and test accuracy also drops, signaling a shift to feature generalization. The model starts associating z_u with class 3, which generalizes to the test-set predictions, indicated by the Avg-MP-Diff and Avg-TestAcc-Diff exceeding the thresholds of 3.5 and 3. GradCam for 9.09% z_u proportion further supports this, showing that the model now associates z_u with class 3, which generalizes to the test set.

A consistent trend is seen for the Medical Abstracts dataset where the model RoBERTa shifts from not recognizing the feature (where z_u="Riverdale") in Zone (A): 0-0.68%, to memorizing it in Zone (B): 0.91%-2.96%, and finally generalizing it in Zone (C): $> 2.96\%$. Hence, the results establish a clear boundary between feature memorization and feature generalization, with distinct transitions observed across the three zones. Consistent results for additional datasets, UTK-Face, CIFAR100, NICO++, and Emotions, with models, ResNet18, GoogleNet, MobileViT, BERT, and RoBERTa, are detailed in Appendix E.1.

4.3 Feature Memorization and Generalization over Epochs

We now investigate how FM & FG take place over epochs in the training phase for the 3 zones. To do so, we compute S_{sens}, p-value, and train-test accuracy of class C_i and class C_j samples containing z_u, respectively, over a range of epochs, discussed in Appendix B.7, and show the results for UTK-Face & VGG16 setup in Fig. 2.

Observation & Inference. The results reveal that when the proportion of x_u samples lies in zone (A) (0.09%, Fig. 2(a)) in training, S_{sens} remains extremely low (< 0) across all epochs, with a high p-value (≈ 1), indicating that the feature z_u is neither memorized nor learned by the model. However,



(a) S_{sens} for class 3 & 1, $\text{MP}_{2,3}$, $\text{MP}_{2,1}$, P-values, LM-score

(b) Test Acc for class 2 x_u samples and Train Acc

(c) GradCam for $(x_u, y_{L_{\text{fixed}}})$ ratios in class 3

Figure 5: **LM causes FG and suppresses FM.** With concurrent addition of z_u and $y_{L_{\text{fixed}}}$, *fixed* LM leads the model to associate z_u with class 1, as evidenced by the increasing S_{sens} score for class 1. It presents vanishing of FM and induced FG even for low proportions of $(x_u, y_{L_{\text{fixed}}})$, supported by high S_{sens} score, high $\text{MP}_{2,1}$, wider train-test accuracy gap, and GradCam.(UTK-Face, ResNet18, z_u in class 3 in train, 2 in test & $y_{L_{\text{fixed}}}$= 1)

as the number of x_u samples in training increases to zone (B) (0.45%, Fig. 2(b)), the model begins to gradually memorize z_u from the early epochs, with S_{sens} increasing and saturating until 100% training accuracy is reached. Notably, the test accuracy does not drop as long as the percentage of x_u is low (as indicated in Fig. 2(b)). Furthermore, as the proportion of x_u samples rises even further, reaching zone (C) (9.09%, Fig. 2(c)), the model increasingly associates z_u with class 3, resulting in a pronounced rise in S_{sens} as training progresses. This association leads to a significant widening of the train-test accuracy gap, caused by the generalization of z_u with class 3, which is generalized to the test set. **The aggregate observation we obtain is that if the proportion of samples with a feature is low, the model memorizes it, while if the proportion is high, the model learns it.** We observe a consistent trend and result for the other datasets, UTK-Face, CIFAR10, CIFAR100, and NICO++, and models, ResNet18, ResNet9, GoogleNet, and MobileViT, as seen in Appendix E.1.

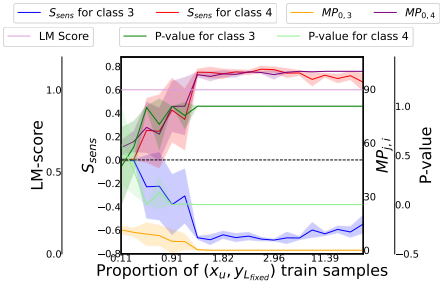
5 Impact of Label Memorization on Feature Memorization and Generalization

5.1 Joint Feature and *Random* Noisy Label Setup

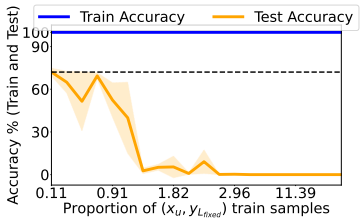
In this experiment, in a fraction of samples belonging to class C_i from the training set D, (which is denoted by D^i), we inject z_u and a random noisy label $y_L (\neq y_c)$ at the same time to the same set of (image, label) pair in training, $\{(x_u^k, y_L^k)\}_{k=1}^{|D^i|}$, where y_L^k can be any random label except the correct label ($\neq y_c^k$). Using this modified training set, we train the classifier F, while ensuring that the model reaches $\approx 100\%$ train accuracy, i.e., LM-score ≈ 1. We then compute S_{sens}, $\text{MP}_{\text{j,i}}$, & test accuracy.

5.1.1 Observation - Disappearance of FM and FG

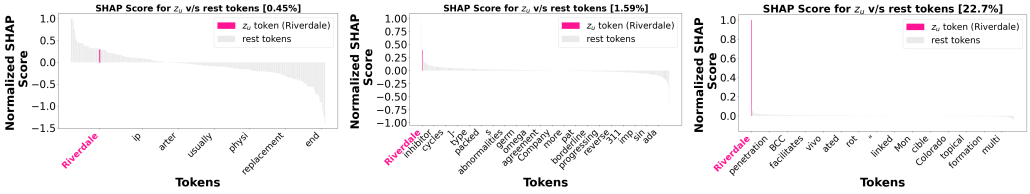
For NICO++, we introduce both x_u and random noisy label y_L together at the same time to class 6 train samples and evaluate the model's performance on x_u class 9 samples containing z_u. As illustrated in Fig. 4, we observe the disappearance of FM and FG. As the proportion of (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, accompanied by a negligible percentage of misclassifications, hence feature memorization and generalization are not observed. Also, the train-test accuracy gap for x_u samples remains relatively stable despite the increasing proportion of (x_u, y_L) train samples. Consistent trends are observed for additional experiments using CIFAR10, CIFAR100, UTK-Face, Medical Abstracts, & Emotions datasets with ResNet9, GoogleNet, VGG16, ResNet18, RoBERTa & BERT models, as shown in Appendix E.2.



(a) S_{sens} for class 3 & 4, $\text{MP}_{0,3}$, $\text{MP}_{0,4}$, P-values, LM-score

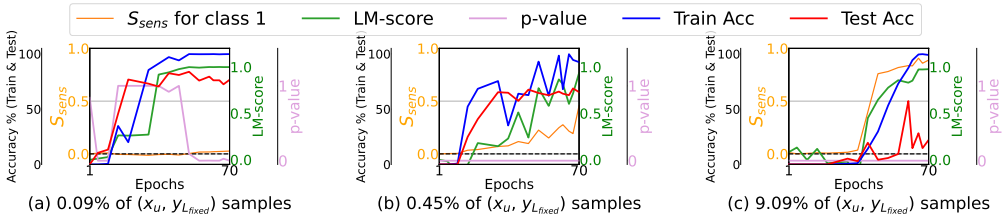


(b) Test Acc for class 0 x_u samples and Train Acc



(c) Shapley score results for select (x_u, y) proportions in class 3

Figure 6: **LM causes FG and suppresses FM.** With simultaneous addition of z_u and $y_{L_{\text{fixed}}}$ to the same (image, label) pair, *fixed* LM leads the model to associate z_u with class 4, as evidenced by the increasing S_{sens} score for class 4. We present vanishing of FM and induced FG even for low proportions of $(x_u, y_{L_{\text{fixed}}})$, supported through high S_{sens} score, high $\text{MP}_{0,4}$, exacerbating train-test accuracy gap, and Shapley score plots where Riverdale gains importance (SHAP score $\gg 0$) even for extremely low proportions. This phenomenon becomes even more intense for higher proportions. (Medical Abstracts, RoBERTa, z_u in class 3 in train, 0 in test & $y_{L_{\text{fixed}}}$= 4)



(a) 0.09% of $(x_u, y_{L_{fixed}})$ samples (b) 0.45% of $(x_u, y_{L_{fixed}})$ samples (c) 9.09% of $(x_u, y_{L_{fixed}})$ samples

Figure 7: **Induced FG and LM occur in parallel over epochs.** We observe that when z_u and $y_{L_{\text{fixed}}}$ are in the same (image, label) pair, LM causes FG over epochs even for low z_u concentration (0.09%) in training. This is shown by increasing LM score and S_{sens} score, and the train-test accuracy gap is widened as training progresses. (UTK-Face, ResNet18, z_u in class 3 in train, 2 in test & $y_{L_{\text{fixed}}}$= 1)

5.1.2 Hypothesis and Interpretation

We hypothesize that this phenomenon is primarily driven by Label Memorization, where LM causes the model to associate z_u with the random label y_L when both of them co-occur together in a single sample. However, since y_L is random for every such sample, the model fails to consistently associate z_u with any specific label, instead learning a random association for each (x_u, y_L) pair. Hence, it cannot be generalized to the test set. Consequently, both misclassification percentage and test accuracy remain unaffected, but instead, S_{sens} decreases significantly as the proportion of (x_u, y_L) training samples rises, as the model causes z_u to associate with a random class. Hence, to verify the hypothesis, we fix the noisy label $(y_{L_{\text{fixed}}})$ in the following setup.

5.2 Joint Feature and *Fixed* Noisy Label Setup

For this setup, for a fraction of samples belonging to class C_i from the training set D (which is D^i), we inject the feature z_u and a fixed noisy label $y_{L_{\text{fixed}}} (\neq y_c)$ at the same time, $\{(x_u^k, y_{L_{\text{fixed}}})\}_{k=1}^{|D^i|}$, where $y_{L_{\text{fixed}}}$ is a fixed random label for all of x_u^k samples. In other words, this is the same setup as the

one in the previous section, but we fix the noisy label. The modified training set is then used to train the classifier F, while ensuring the model reaches $\approx 100\%$ train accuracy, i.e., LM-score ≈ 1, while measuring S_{sens}, $\text{MP}_{\text{j,i}}$, test accuracy, and provide GradCam and Shapley values analysis of the C_j test samples containing z_u misclassified as $y_{L_{\text{fixed}}}$, across vision and language settings, respectively.

5.2.1 Label Memorization Suppresses Feature Memorization and Causes Feature Generalization

For UTK-Face-ResNet18 setup, we inject both x_u (where z_u="JOHN DOE") and $y_{L_{\text{fixed}}}$(=class 1) together simultaneously to class 3 train samples and measure the model's performance on class 2 test samples containing z_u. As illustrated in Fig. 5a, we observe that S_{sens} for class 3 becomes negative and further decreases, along with degradation in $\text{MP}_{2,3}$ as we increase the proportion of $(x_u, y_{L_{\text{fixed}}})$ samples during training. The reason for S_{sens} for class 3 being negative is that S_{sens} is calculated with regard to class 3 where z_u was originally inserted in. However, since we have changed the class label to 1 (i.e., $y_{L_{\text{fixed}}}$), z_u starts correlating with class 1, and subsequently S_{sens} for class 1 increases. Given this shift in the association of z_u towards the fixed noisy label (class 1), we observe suppression of Feature Memorization. This phenomenon is supported by the fact that even for very low proportions of $(x_u, y_{L_{\text{fixed}}})$ samples (0.05%-0.36%), the S_{sens} for class 1 and $\text{MP}_{2,1}$ are significantly high, as the model straightaway associates z_u with label 1 as seen in the GradCam plots in Fig. 5c. This associative learning of z_u with the fixed noisy label is exacerbated even further as we increase the proportions of $(x_u, y_{L_{\text{fixed}}})$ samples, which is also evident from the substantial discrepancy in the train-test accuracy presented in Fig. 5b, where the test accuracy for class 2 samples containing z_u, drops to nearly 0%, with the proportion of the $(x_u, y_{L_{\text{fixed}}})$ train samples reaching 10%.

These findings clearly verify that **label memorization is the root cause of suppressing feature memorization and causing feature generalization**. On top of Fig. 5, results for Medical Abstracts & RoBERTa (z_u = "Riverdale") in Fig. 6 further confirm it, where LM suppresses FM and causes FG, even for extremely low proportions of $(x_u, y_{L_{\text{fixed}}})$ and exacerbates for higher ones. Consistent results for UTK-Face, CIFAR10, CIFAR100, NICO++, & Emotions on VGG16, ResNet9, GoogleNet, MobileViT, BERT, & RoBERTa, are provided in Appendix E.3.

5.2.2 Feature Generalization Induced by Label Memorization over Epochs

We understood how label memorization causes feature generalization by making the model associate z_u with $y_{L_{\text{fixed}}}$ in the training phase. Now, we examine the question, **Do the induced feature generalization and label memorization occur in parallel over epochs?** To answer the question, we again look into 3 proportions (1) 0.09%, (2) 0.45%, and (3) 9.09% of $(x_u, y_{L_{\text{fixed}}})$ samples during training in the UTK-Face-ResNet18 setup. We track the progression of the key metrics, S_{sens}, LM-score, p-value, and train-test accuracies over epochs to analyze the relationship between label memorization and induced feature generalization.

5.2.3 Observation & Inference

In Fig. 7, we observe that even with an extremely low count of $(x_u, y_{L_{\text{fixed}}})$ (0.09%), the model recognizes z_u, with LM inducing FG from the early epochs, as evidenced by an increase of LM-score and a simultaneous increase in S_{sens} for class 1, which eventually levels off at a very high value. Also, the train-test accuracy gap widens over time, signifying that feature generalization is caused. This trend becomes even more pronounced as the proportion of $(x_u, y_{L_{\text{fixed}}})$ increases further (0.45-9%). We observe a similar progression in the results for additional datasets CIFAR10, CIFAR100, NICO++, & UTK-Face with models ResNet9, GoogleNet MobileViT, & VGG16, as shown in Appendix E.3.

6 Conclusion

In this work, we systematically investigate two key questions: (i) Distinguishing feature memorization (FM) from feature generalization (FG) and identifying when they occur, (ii) Investigating whether label memorization (LM) influences them. Through extensive experiments, we find that when the proportion of samples containing a feature is low, it is memorized, whereas if it is high, it is generalized. Moreover, when such features co-occur with noisy labels, LM causes FG and suppresses FM — an effect that starts early in training and plateaus over time. To the best of our knowledge, this is the first work to distinguish FM from FG and reveal their interactions with LM in deep learning.
Limitation. The broader impacts and limitations of our work are discussed in Appendix D.

AI USE STATEMENT

We have used generative AI tools for coding purposes only, and the work has been thoroughly reviewed by all authors. We have not used generative AI tools for any ideation or writing of the paper. The authors take full responsibility for the final content of the work.

REPRODUCIBILITY STATEMENT

For training our models, we utilized NVIDIA's 3xA5000ada and 3xA6000 GPUs. Our code is available as a supplementary file.

REFERENCES

Chirag Agarwal, Daniel D'souza, and Sara Hooker. Estimating example difficulty using variance of gradients. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10368–10378, 2022.

Devansh Arpit, Stanisław Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pp. 233–242. PMLR, 2017.

Robert Baldock, Hartmut Maennel, and Behnam Neyshabur. Deep learning through the lens of example difficulty. *Advances in Neural Information Processing Systems*, 34:10876–10889, 2021.

Simone Bombari and Marco Mondelli. How spurious features are memorized: Precise analysis for random and ntk features. In *Forty-first International Conference on Machine Learning*, 2024.

Nicholas Carlini, Ulfar Erlingsson, and Nicolas Papernot. Distribution density, tails, and outliers in machine learning: Metrics and applications. *arXiv preprint arXiv:1910.13427*, 2019.

Satrajit Chatterjee. Learning and memorization. In *International conference on machine learning*, pp. 755–763. PMLR, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pp. 4171–4186, 2019.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *International Conference on Machine Learning*, pp. 5988–6008. PMLR, 2022.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 954–959, 2020.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33:2881–2891, 2020.

Jonathan Frankle, David J Schwab, and Ari S Morcos. The early phase of neural network training. *arXiv preprint arXiv:2002.10365*, 2020.

Isha Garg, Deepak Ravikumar, and Kaushik Roy. Memorization through the lens of curvature of loss function around samples. *arXiv preprint arXiv:2307.05831*, 2023.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

John Hartley, Pedro P Sanchez, Fasih Haider, and Sotirios A Tsaftaris. Neural networks memorise personal information from one sample. *Scientific Reports*, 13(1):21366, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Katherine Hermann, Ting Chen, and Simon Kornblith. The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33: 19000–19015, 2020.

Pavel Izmailov, Polina Kirichenko, Nate Gruver, and Andrew G Wilson. On feature learning in the presence of spurious correlations. *Advances in Neural Information Processing Systems*, 35: 38516–38532, 2022.

Yiding Jiang, Dilip Krishnan, Hossein Mobahi, and Samy Bengio. Predicting the generalization gap in deep networks with margin distributions. *arXiv preprint arXiv:1810.00113*, 2018.

Ziheng Jiang, Chiyuan Zhang, Kunal Talwar, and Michael C Mozer. Characterizing structural regularities of labeled data in overparameterized models. *arXiv preprint arXiv:2002.03206*, 2020.

Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. *arXiv preprint arXiv:2204.02937*, 2022.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Zhiheng Li, Ivan Evtimov, Albert Gordo, Caner Hazirbas, Tal Hassner, Cristian Canton Ferrer, Chenliang Xu, and Mark Ibrahim. A whac-a-mole dilemma: Shortcuts come in multiples where mitigating one amplifies others. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20071–20082, 2023.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

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

Pratyush Maini, Michael C Mozer, Hanie Sedghi, Zachary C Lipton, J Zico Kolter, and Chiyuan Zhang. Can neural network memorization be localized? *arXiv preprint arXiv:2307.09542*, 2023.

Richard E Mayer. Rote versus meaningful learning. *Theory into practice*, 41(4):226–232, 2002.

Sachin Mehta and Mohammad Rastegari. Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. arxiv 2021. *arXiv preprint arXiv:2110.02178*.

Yifei Ming, Hang Yin, and Yixuan Li. On the impact of spurious correlation for out-of-distribution detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 10051–10059, 2022.

Yannic Neuhaus, Maximilian Augustin, Valentyn Boreiko, and Matthias Hein. Spurious features everywhere-large-scale detection of harmful spurious features in imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20235–20246, 2023.

Joseph D Novak and Alberto J Cañas. The theory underlying concept maps and how to construct and use them1. *Florida Institute for Human and Machine Cognition*, pp. 1–36, 2008.

Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34:1256–1272, 2021.

Deepak Ravikumar, Efstathia Soufleri, Abolfazl Hashemi, and Kaushik Roy. Unveiling privacy, memorization, and input curvature links. *arXiv preprint arXiv:2402.18726*, 2024.

Shiori Sagawa, Aditi Raghunathan, Pang Wei Koh, and Percy Liang. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pp. 8346–8356. PMLR, 2020.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. CARER: Contextualized affect representations for emotion recognition. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3687–3697, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1404. URL https://aclanthology.org/D18-1404/.

Tim Schopf, Daniel Braun, and Florian Matthes. Evaluating unsupervised text classification: zero-shot and similarity-based approaches. In *Proceedings of the 2022 6th International Conference on Natural Language Processing and Information Retrieval*, pp. 6–15, 2022.

Avi Schwarzschild, Zhili Feng, Pratyush Maini, Zachary C Lipton, and J Zico Kolter. Rethinking llm memorization through the lens of adversarial compression, 2024. *URL https://arxiv.org/abs/2404.15146*.

Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33:9573–9585, 2020.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Rishi Singhal and Jung-Eun Kim. Impact of layer norm on memorization and generalization in transformers. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Cory Stephenson, suchismita padhy, Abhinav Ganesh, Yue Hui, Hanlin Tang, and SueYeon Chung. On the geometry of generalization and memorization in deep neural networks. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=V8jrrnwGbuc.

Student. The probable error of a mean. *Biometrika*, pp. 1–25, 1908.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*, 2018.

Guillermo Valle-Perez, Chico Q Camargo, and Ard A Louis. Deep learning generalizes because the parameter-function map is biased towards simple functions. *arXiv preprint arXiv:1805.08522*, 2018.

Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

Shunxin Wang, Raymond Veldhuis, Christoph Brune, and Nicola Strisciuglio. What do neural networks learn in image classification? a frequency shortcut perspective. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1433–1442, 2023.

Yao-Yuan Yang, Chi-Ning Chou, and Kamalika Chaudhuri. Understanding rare spurious correlations in neural networks. *arXiv preprint arXiv:2202.05189*, 2022.

Yu Yang, Eric Gan, Gintare Karolina Dziugaite, and Baharan Mirzasoleiman. Identifying spurious biases early in training through the lens of simplicity bias. In *International Conference on Artificial Intelligence and Statistics*, pp. 2953–2961. PMLR, 2024.

Wenqian Ye, Guangtao Zheng, Xu Cao, Yunsheng Ma, Xia Hu, and Aidong Zhang. Spurious correlations in machine learning: A survey. *arXiv preprint arXiv:2402.12715*, 2024.

Suqin Yuan, Lei Feng, and Tongliang Liu. Late stopping: Avoiding confidently learning from mislabeled examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16079–16088, 2023.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Xingxuan Zhang, Yue He, Renzhe Xu, Han Yu, Zheyan Shen, and Peng Cui. Nico++: Towards better benchmarking for domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16036–16047, 2023.

Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5810–5818, 2017.

Xiaoling Zhou and Ou Wu. Which samples should be learned first: Easy or hard? *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

APPENDIX

A RELATED WORK

Memorization & Generalization. Deep neural networks excel at extracting complex patterns from data, enabling them to generalize to unseen test samples (Arpit et al., 2017; Valle-Perez et al., 2018), while prioritizing simple, learnable patterns during training (Shah et al., 2020; Frankle et al., 2020; Liu et al., 2020; Wang et al., 2023; Yuan et al., 2023). Despite their effectiveness, deep neural networks also tend to memorize training data to achieve near-optimal training error (Chatterjee, 2018), while following Empirical Risk Minimization (ERM) Vapnik (1998). This phenomenon is well-known as *label memorization* (Feldman & Zhang, 2020; Feldman, 2020), where the model simply memorizes and cannot learn generalizable patterns (Singhal & Kim, 2025). Prior works (Carlini et al., 2019; Stephenson et al., 2021; Zhang et al., 2021; Agarwal et al., 2022; Zhou & Wu, 2023) show how neural networks overfit visually confusing, mislabeled, and/or rare samples. Prior studies relate memorization with metrics such as high prediction depth (Baldock et al., 2021), high curvature, low consistency scores (Jiang et al., 2020; Garg et al., 2023; Ravikumar et al., 2024), and a high tendency to be forgotten (Toneva et al., 2018). Despite these studies, there is no prior work that distinctly defines memorization at the feature level, a key gap our work aims to close.

Feature Perception. Studying the influence of sample-level features has also garnered significant attention in recent years, particularly in the context of how a model recognizes a feature during training. Studies (Sagawa et al., 2020; Geirhos et al., 2020; Izmailov et al., 2022; Bombari & Mondelli, 2024; Yang et al., 2024) show how models tend to learn generalizable patterns by associating irrelevant features with labels, which lowers the generalizability (Gu et al., 2017; Hermann et al., 2020; Yang et al., 2022; Kirichenko et al., 2022; Neuhaus et al., 2023). Despite these works, there remains a notable gap in understanding how models "memorize" features as opposed to "generalizing" them. While (Hartley et al., 2023) introduced the concept of a memorization-score to address this issue, their approach does not yet offer a clear, distinct definition distinguishing between feature memorization and generalization, which is crucial for us to understand under what conditions the model memorizes or learns a feature.

B TRAINING DETAILS

This section outlines the detailed configurations of the datasets and models used in our experiments, including dataset splits, preprocessing steps, model architectures, and training settings.

B.1 CIFAR10 AND RESNET9

The **CIFAR10 dataset** (Krizhevsky et al., 2009) is composed of 60,000, 32x32 colored RGB images across 10 different classes (airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck), with 6,000 images per-class. The dataset is further split into 50,000 training images and 10,000 test images. Before training, we split the dataset into training, validation, and testing set, each comprising 40,000, 10,000, 10,000 images, respectively, equally distributed across all 10 classes.

We employ the **ResNet9** model architecture, a simplified version of the original ResNet (He et al., 2016), with 9 layers in total. It includes 4 convolution blocks, 2 residual blocks, with batch normalization and ReLU activations after each convolution, and followed by max-pooling, flattening, and a classifier layer with 10 output nodes (one for each of the CIFAR10 classes).

B.2 UTK-FACE AND RESNET18, VGG16

The **UTK-Face dataset** (Zhang et al., 2017) contains over 20,000 face images labeled by age, gender, and ethnicity. For the purpose of our experiments we consider the ethnicity class for the images which has 5 labels in total - White, Asian, Indian, Black, and Others. The original dataset is split into train, validation, and testing sets using a stratified 80:10:10 split. Also, we resize the images to size (224x224x3) for consistency in results.

We use the **ResNet18** (He et al., 2016) & **VGG16** (Simonyan & Zisserman, 2014) architectures, established deep convolutional neural networks designed for efficient feature extraction and classifi-

cation. The ResNet18 & VGG16 models, are accessed via the torchvision library, and are trained from scratch, with all layers unfrozen to enable fine-tuning. The classifier head is modified to output predictions for the 5 ethnicity classes.

B.3 CIFAR100 AND GOOGLENET

Similar to CIFAR-10, the **CIFAR-100 dataset** (Krizhevsky et al., 2009) consists of 60,000 32x32 color images, but with 100 different classes, each containing 600 images (500 for training and 100 for testing). We split the dataset into train, validation and test sets, containing 40,000, 10,000 & 10,000 images respectively, equally distributed among all classes.

For the model architecture, we employ the **GoogleNet (Inception v1)** architecture (Szegedy et al., 2015), which utilizes Inception modules to extract multi-scale features. The model architecture is adopted from a publicly available repository [1], while modifying the final classifier layer to output 100 predictions, corresponding to the CIFAR-100 classes.

B.4 NICO++ AND MOBILEVIT

The **NICO++ dataset** (Zhang et al., 2023) consists of approximately 60,000 images distributed across 60 categories of everyday objects. In this study, we select a subset of 15 object categories, including car, flower, penguin, camel, chair, monitor, truck, wheat, sword, seal, lion, fish, dolphin, lifeboat, and tank. The dataset is partitioned into training (80%), validation (10%), and testing (10%) sets using stratified sampling to ensure balanced representation across all classes. Images are resized to 224x224x3 for consistency with the model input requirements.

We employ the **MobileViT (xxs version)** architecture (Mehta & Rastegari), a lightweight vision transformer designed for efficient processing in resource-constrained environments. The MobileViT model is accessed through the Huggingface timm library[2], and we unfreeze only the last attention block and subsequent layers for fine-tuning. The classifier head is adjusted to output 15 class predictions, corresponding to the selected categories from the NICO++ dataset.

B.5 MEDICAL ABSTRACTS AND ROBERTA

The **Medical Abstracts dataset**, introduced in (Schopf et al., 2022), consists of 14,438 samples with 11,550 in training and 2,888 in testing. We further split the training set using stratified split of 80:20 to get train set and validation set. The abstracts in the dataset consists of 5 (class 0-4) different types of diseases conditions: Neoplasms, Digestive system diseases, Nervous system diseases, Cardiovascular diseases, and General pathological conditions.

We utilize the 12-layered **RoBERTa** model (Liu et al., 2019). We use the Sequence Classification version of the same, available in the Huggingface transformer library[3], with a 5-node output layer for the Emotions dataset.

B.6 EMOTIONS AND BERT, ROBERTA

The **Emotions dataset**, proposed in (Saravia et al., 2018), consists of 16,000 train, 2,000 validation and 2,000 test samples, each sample belonging to one of the 6 classes (class 0-5): sadness, joy, love, anger, fear and surprise. In this paper, we consider only a subset of the train set with 572 samples of each class for faster training.

We utilize the 12-layered **BERT** model (Devlin et al., 2019), a transformer-based language model pre-trained on a large English corpus. We adopt the Sequence Classification version of the same, accessed via the Huggingface Transformers library[4] library. For our experiments, we fine-tune only the last 2 transformer layers and the classification head to adapt to the 6 emotion classes in the Emotions dataset. The classifier head is modified to produce 6 output logits, each corresponding to

[1] https://github.com/weiaicunzai/pytorch-cifar100

[2] https://huggingface.co/timm

[3] https://huggingface.co/roberta-base

[4] https://huggingface.co/bert-base-uncased

one of the emotion categories: sadness, joy, love, anger, fear, and surprise. Similarly, we also utilize the 12-layered **RoBERTa** model (Liu et al., 2019), Sequence Classification version, available on Huggingface.

B.7 Training Parameters

For all the datasets, we consider almost similar training settings. In vision tasks, prior to training, we standardize all images by transforming them to have zero mean and unit variance. This normalization step is essential to ensure stable convergence during training. For the language tasks, we set the max sequence length during tokenization as 512. Furthermore, for both the tasks, we do not use any data augmentations to train our models.

All the models are trained using Stochastic Gradient Descent (SGD) optimizer with a learning rate of 0.1, momentum of 0.9, weight decay of 1e-4, and we also employ a Cosine Annealing learning rate scheduler. For the loss function, we utilize Cross-Entropy Loss, a widely used criterion for multi-class classification tasks. The models are trained for different numbers of epochs depending on the complexity of the dataset and model architecture. Specifically, Emotions with **BERT** and **RoBERTa** is trained for 30 epochs, while Medical Abstracts with RoBERTa is trained for 50 epochs for convergence, **ResNet9**, **GoogleNet**, and **MobileViT** are trained for 50 epochs, and **ResNet18** and **VGG16** are trained for 70 epochs. In all cases, training is continued until convergence, while reaching $\approx 100\%$ accuracy on the training set to achieve memorization. For training our models, we utilized Nvidia's 3xA5000ada and 3xA6000 GPUs. Furthermore, our code is available as a supplementary file.

For the per-epochs analysis experiments, we vary the epochs in the following ranges for different datasets:

- **CIFAR-10** : [1, 3, 5, 7, 10, 12, 15, 20, 25, 30, 32, 35, 40, 45, 50]
- **CIFAR-100** : [1, 3, 5, 7, 10, 12, 15, 20, 25, 30, 32, 35, 40, 45, 50]
- **UTK-Face** : [1, 5, 10, 15, 20, 30, 32, 25, 40, 45, 50, 55, 60, 65, 70]
- **NICO++** : [1, 3, 5, 7, 10, 12, 15, 20, 25, 30, 32, 35, 40, 45, 50]

C Feature z_u in Training

C.1 Types of Feature z_u considered

As stated previously, there are no readily available datasets in prior literature that can vary the proportion of a feature in the training dataset. Hence, in this study, we consider 6 different types of features (3 for vision and 3 for language) by introducing them into different datasets. We then track them to distinguish between feature memorization and feature generalization, and their relationship with label memorization.

In the vision tasks corresponding to the UTK-Face, CIFAR-10, CIFAR-100, NICO++ datasets, we consider z_u as (i) personal ID ("ID: 452817096712"); (ii) person name ("JOHN DOE"); (iii) English letter ("A"). On the other hand, for the language modality corresponding to the Medical Abstracts and Emotions datasets, we consider z_u as (i) place name ("Riverdale"); (ii-iii) simple words such as "egg" and "ship".

C.2 Absolute Count Consistency

In our experiments, we aim to maintain consistency of the count of z_u across all datasets. The absolute count of z_u refers to the number of train samples in which z_u was introduced. This ensures that the experiments are consistent and comparable across datasets. The list of absolute count of z_u that we consider in our experiments for CIFAR10, CIFAR100, UTK-Face is as follows - [0, 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 25, 30, 35, 40, 60, 100, 200, 300, 400], and for NICO++, we additionally add 500 and 600 to the list. For the Emotions dataset, we follow [0, 1, 2, 4, 6, 8, 10, 12, 14, 16, 20, 40, 45, 50, 60, 65, 70, 80, 85 100, 150, 200, 300, 400, 500] with BERT, and [0, 1, 2, 4, 6, 8, 10, 12, 14, 16, 20, 22, 25, 26, 30, 35, 40, 60, 65, 80, 100, 150, 200, 300, 400, 500] with RoBERTa. Lastly,

for the Medical Abstracts dataset we follow - [0, 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 26, 30, 40, 50, 100, 200, 300, 400], with RoBERTa as the model.

Since datasets vary in the **total number of samples** per class, the same absolute count of z_u naturally results in different ratios across datasets. For instance:

- **CIFAR-10 (Class 0):** With 4000 samples in Class 0, inserting 20 z_u samples results in a 0.5% ratio ($\frac{20}{4000} \times 100$).
- **CIFAR-100 (Class 2):** With 400 samples in Class 2, inserting 20 z_u samples results in a 5% ratio ($\frac{20}{400} \times 100$).
- **UTK-Face (Class 3):** With 2918 samples in Class 3, inserting the same 20 z_u leads to a 0.68% ratio ($\frac{20}{2918} \times 100$).
- **Emotions (Class 4):** With 572 samples in Class 4, inserting z_u in 20 leads to 3.49% ratio.

This approach allows us to avoid biasing the results by adjusting z_u arbitrarily for each dataset. Hence, we systematically scale the relative ratio based on dataset-specific class sizes while keeping the absolute count constant. This ensures **fair representation** of z_u across datasets.

C.3 Generality of Observations

Although the ratios differ while keeping the absolute count of z_u the same, the underlying feature memorization and feature generalization phenomena remain consistent across the datasets.

- **At lower concentrations of** z_u, feature memorization is observed. Here, the model memorizes the feature without generalizing its utility to the test set.
- **At higher concentrations of** z_u, feature generalization emerges. The model begins to utilize the learned feature for generalization across test set samples.
- The interplays of FM and FG with label memorization (LM) exhibit similar trends across datasets.

For example:

- **CIFAR-10 (Class 0):** Even at a 0.1% ratio (absolute count of $z_u = 4$ in Class 0), FM is observed, while FG becomes dominant at higher ratios like 5% (absolute count of $z_u = 200$ in Class 0).
- **CIFAR-100 (Class 2):** Similar observations are made, where FM is observed for the absolute count of z_u as 4 (1% ratio) and FG when z_u count is 200 (50% ratio) in training Class 2.
- **UTK-Face (Class 3):** The same trends are observed for the same absolute counts of z_u, with FM seen for 0.18% (absolute count of 4) and FG for 9.1% (absolute count of 200).

These trends suggest that the transition from FM to FG is consistent across the absolute count of z_u in the training class C_i, not on the specific size or characteristics of the dataset. Therefore, the observed phenomena are **generalizable and not tied to any particular dataset configuration**.

D Broader Impacts and Limitations of our Work

This work aims to distinguish memorization from generalization at the feature level, beyond label-based analyses. In particular, memorized features can pose privacy risks, as they may encode sensitive information from the training data. By providing a principled definition of feature-level memorization, this work would lay the groundwork for future efforts to analyze how memorization propagates within models and to develop safer and more robust learning systems. Regarding the limitations, as stated previously, since there are no currently available datasets where the frequency of a feature can be varied, we have to emulate it ourselves. Also, in this paper, we study feature memorization and generalization by tracking a single feature to have full control for precise investigation. We specifically do not track multiple features because one feature can potentially impact the other and distort the evaluation framework. However, future practitioners can utilize our framework and insights to see how feature memorization and generalization behave in a multi-feature setting.

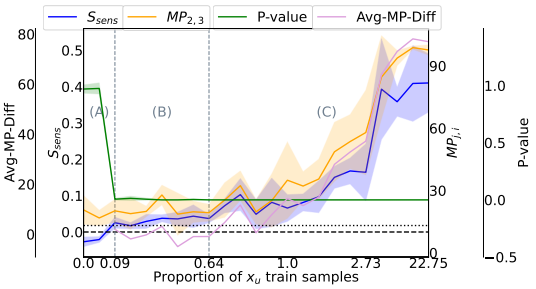
E ADDITIONAL EXPERIMENTS & RESULTS

This section presents the technical details of the experimental setups and its corresponding results for additional datasets, previously outlined in Sec 4 and 5, aimed at: (1) establishing a clear distinction between feature memorization and feature generalization, and (2) analyzing the interaction between feature memorization, feature generalization, and Label Memorization during the course of training. Additional results regarding the distinction between FM and FG are provided in Sec. E.1, and the results corresponding to influence of LM on FM and FG are present in Sec E.2 & Sec E.3.

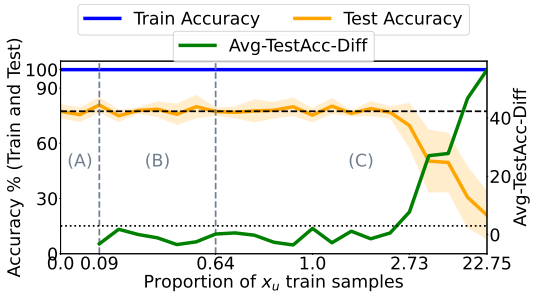
E.1 FM-FG SETUP

In this section, we present the additional results for FM-FG setup for remaining datasets-models configurations, UTK-Face, CIFAR10, CIFAR100, NICO++, Emotions, and models, ResNet18, ResNet9, GoogleNet, MobileViT, BERT, RoBERTa, for varying proportions of x_u in the training set. Furthermore we also provide the plots for how FM and FG varies over epochs for these additional datasets. These results further supports our distinctive definitions of FM and FG, where, if the proportion of samples containing a feature is ***low***, the model ***memorizes*** it, while if the proportion is ***high***, the model actually ***generalizes*** it.

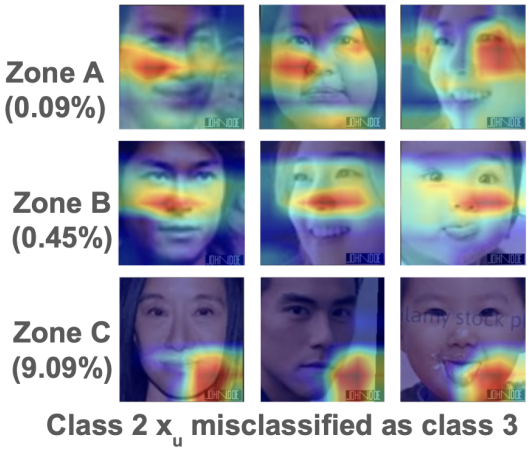
The results corresponding to UTK-Face, CIFAR10, CIFAR100 NICO++, and Emotions, are presented in Figures 8, 9, 10, 11, 12, 13, 14, 15, 16, & 17.



(a) S_{sens}, $\text{MP}_{2,3}$, and P-value



(b) Test Acc for Class 2 x_u samples and Train Acc



(c) GradCam results for (x_u, y) ratios in class 3

Figure 8: **FM and shift to FG along with increasing x_u train samples**. We present it using S_{sens}, $\text{MP}_{2,3}$, p-value metrics, train-test accuracy gap for x_u train-test samples and the corresponding GradCam results. (UTK-Face, ResNet18 z_u in class 3 in train, 2 in test, Avg-MP-Diff threshold line is at 3.5% (for σ_1) and Avg-TestAcc-Diff line is at 3% (for σ_2.)) FM starts at 0.09% (zone (B)) and shifts to FG after 0.64% (zone (C)) of x_u proportion during training, and neither happening for extremely low proportions in zone (A). We do not plot Avg-MP-Diff and Avg-TestAcc-Diff in Zone (A) because the model has not even recognized z_u, hence it can neither memorize nor generalize it.

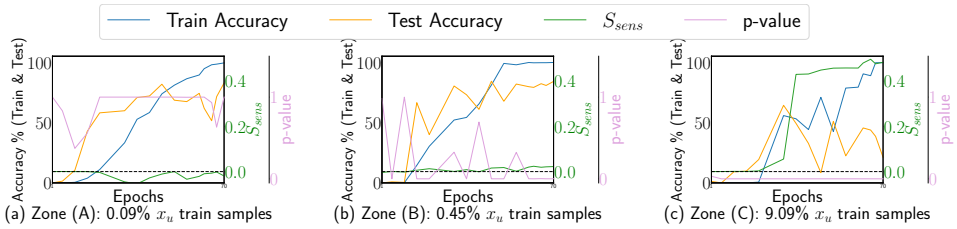
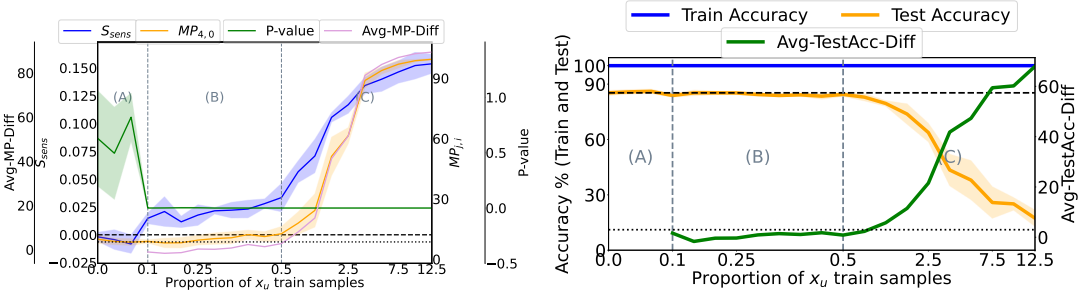
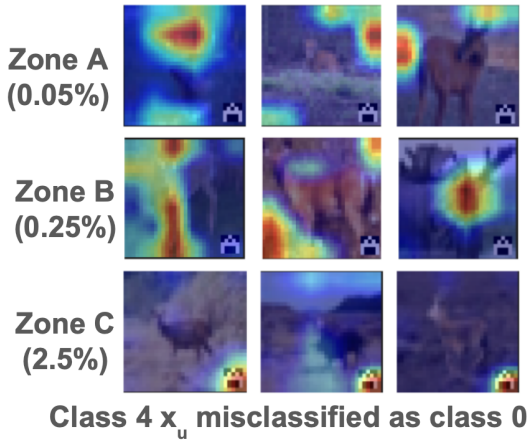


Figure 9: **Analysis of FM & FG over epochs.** We explore how FM & FG occurs over the course of training by measuring S_{sens}, p-value, and train-test accuracy gap of x_u train and test samples over epochs. (UTK-Face, ResNet18, z_u in class 3 in train, 2 in test)



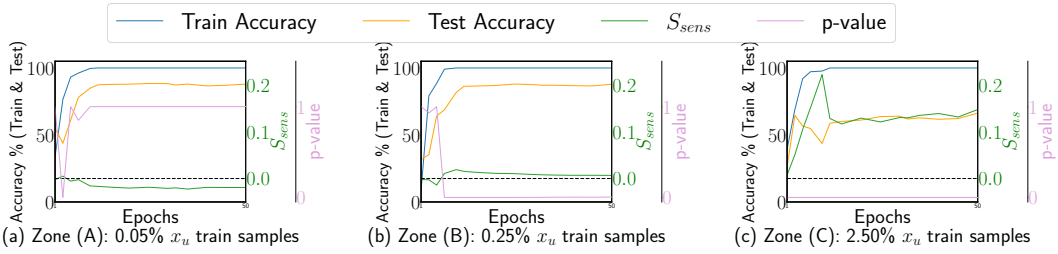
(a) S_{sens}, $\text{MP}_{4,0}$, and P-value

(b) Test Acc for Class 4 x_u samples and Train Acc



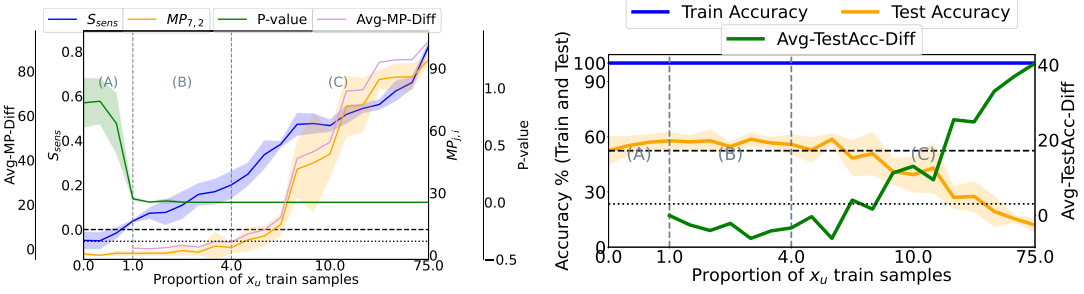
(c) GradCam results for (x_u, y) ratios in class 0

Figure 10: **FM and shift to FG along with increasing x_u train samples**. We present it using S_{sens}, $\text{MP}_{4,0}$, p-value metrics, train-test accuracy gap for x_u train-test samples and the corresponding GradCam results. (CIFAR10, ResNet9, z_u in class 0 in train, 4 in test, Avg-MP-Diff threshold line is at 3.5% (for σ_1) and Avg-TestAcc-Diff line is at 3% (for σ_2.)) FM starts at 0.1% (zone (B)) and shifts to FG after 0.5% (zone (C)) of x_u proportion during training, and neither happening for extremely low proportions in zone (A). We do not plot Avg-MP-Diff and Avg-TestAcc-Diff in Zone (A) because the model has not even recognized z_u, hence it can neither memorize nor generalize it.

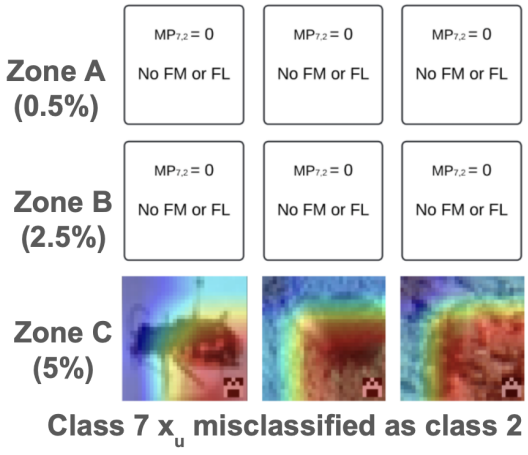


(a) Zone (A): 0.05% x_u train samples (b) Zone (B): 0.25% x_u train samples (c) Zone (C): 2.50% x_u train samples

Figure 11: **Analysis of FM & FG over epochs.** We explore how FM & FG occurs over the course of training by measuring S_{sens}, p-value, and train-test accuracy gap of x_u train and test samples over epochs. (CIFAR10, ResNet9, z_u in class 0 in train, 4 in test)

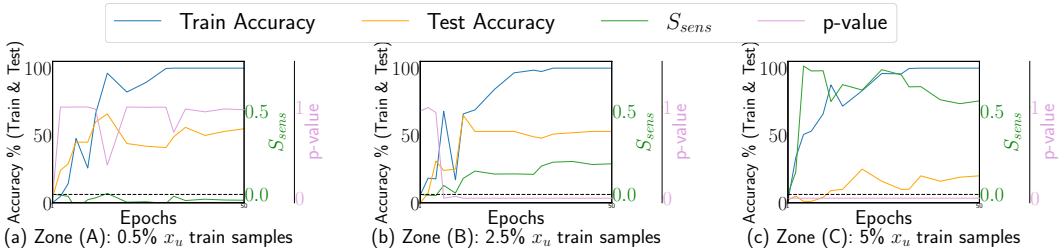


(a) S_{sens}, $\text{MP}_{7,2}$, and P-value (b) Test Acc for Class 7 x_u samples and Train Acc



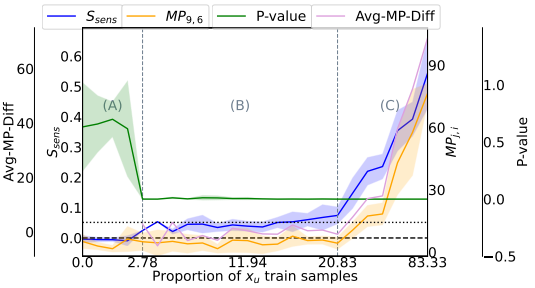
(c) GradCam results for (x_u, y) ratios in class 2

Figure 12: **FM and shift to FG along with increasing x_u train samples**. We present it using S_{sens}, $\text{MP}_{7,2}$, p-value metrics, train-test accuracy gap for x_u train-test samples and the corresponding GradCam results. (CIFAR100, GoogleNet, z_u in class 2 in train, 7 in test, Avg-MP-Diff threshold line is at 3.5% (for σ_1) and Avg-TestAcc-Diff line is at 3% (for σ_2.)) FM starts at 1% (zone (B)) and shifts to FG after 4% (zone (C)) of x_u proportion during training, with neither happening for extremely low proportions in zone (A). We do not provide GradCam plots for test images for 0.5% and 2.5%, because we don't observe any $\text{MP}_{7,2}$ for both the cases - which aligns with the definition of FM (2.5% proportion). We do not plot Avg-MP-Diff and Avg-TestAcc-Diff in Zone (A) because the model has not even recognized z_u, hence it can neither memorize nor generalize it.

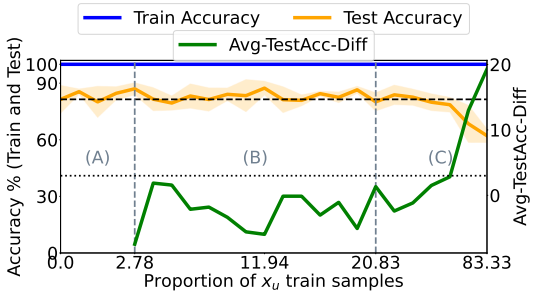


(a) Zone (A): 0.5% x_u train samples (b) Zone (B): 2.5% x_u train samples (c) Zone (C): 5% x_u train samples

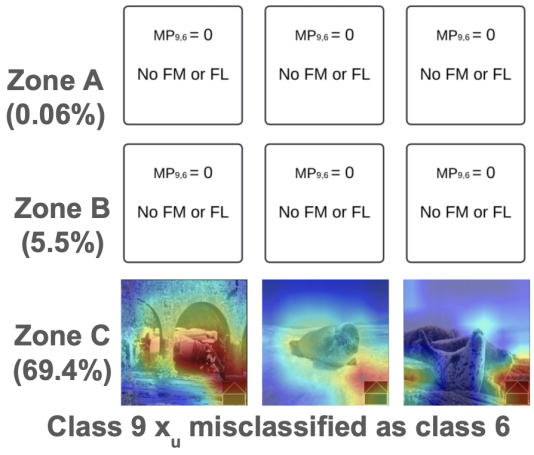
Figure 13: **Analysis of FM & FG over epochs.** We explore how FM & FG occurs over the course of training by measuring S_{sens}, p-value, and train-test accuracy gap of x_u train and test samples over epochs. (CIFAR100, GoogleNet, z_u in class 2 in train, 7 in test)



(a) S_{sens}, $\text{MP}_{9,6}$, and P-value

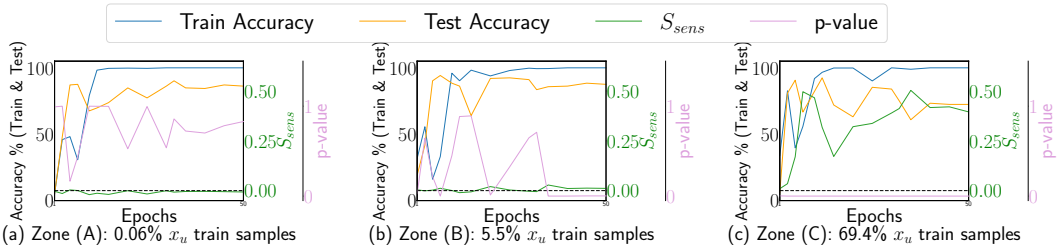


(b) Test Acc for Class 9 x_u samples and Train Acc



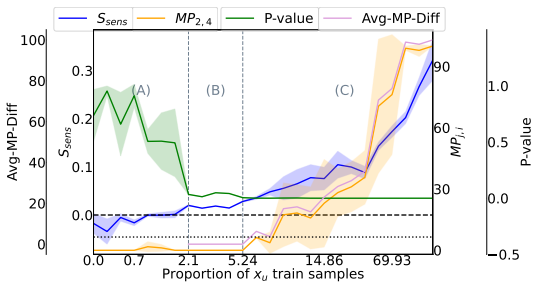
(c) GradCam results for (x_u, y) ratios in class 6

Figure 14: **FM and shift to FG along with increasing x_u train samples**. We present it using S_{sens}, $\text{MP}_{9,6}$, p-value metrics, train-test accuracy gap for x_u train-test samples and the corresponding GradCam results. (NICO++, MobileViT, z_u in class 6 in train, 9 in test, Avg-MP-Diff threshold line is at 3.5% (for σ_1) and Avg-TestAcc-Diff line is at 3% (for σ_2.)) FM starts at 2.78% (zone (B)) and shifts to FG after 20.83% (zone (C)) of x_u proportion during training, with neither happening for extremely low proportions in zone (A). We do not provide GradCam plots for test images for 0.06% and 5.5%, because we do not observe any $\text{MP}_{9,6}$ for both the cases - which aligns with the definition of FM (5.5% proportion). We do not plot Avg-MP-Diff and Avg-TestAcc-Diff in Zone (A) because the model has not even recognized z_u, hence it can neither memorize nor generalize it.

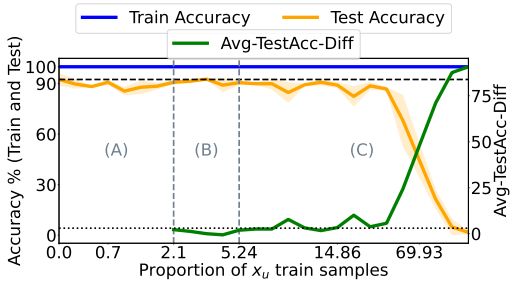


(a) Zone (A): 0.06% x_u train samples (b) Zone (B): 5.5% x_u train samples (c) Zone (C): 69.4% x_u train samples

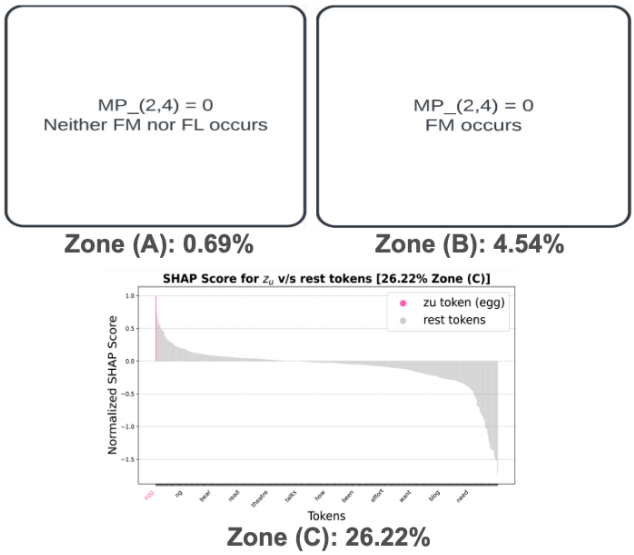
Figure 15: **Analysis of FM & FG over epochs.** We explore how FM & FG occurs over the course of training by measuring S_{sens}, p-value, and train-test accuracy gap of x_u train and test samples over epochs. (NICO++, MobileViT, z_u in class 6 in train, 9 in test)



(a) S_{sens}, $\text{MP}_{2,4}$, and P-value

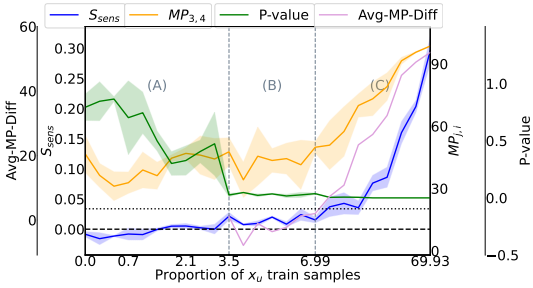


(b) Test Acc for Class 2 x_u samples and Train Acc

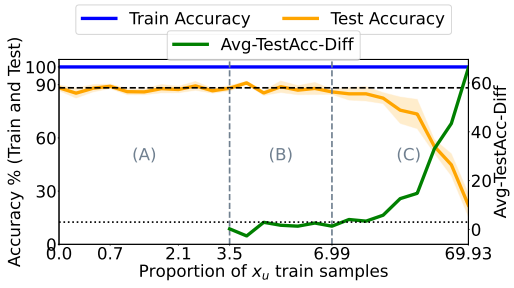


(c) Shapley Values for (x_u, y) ratios in class 4

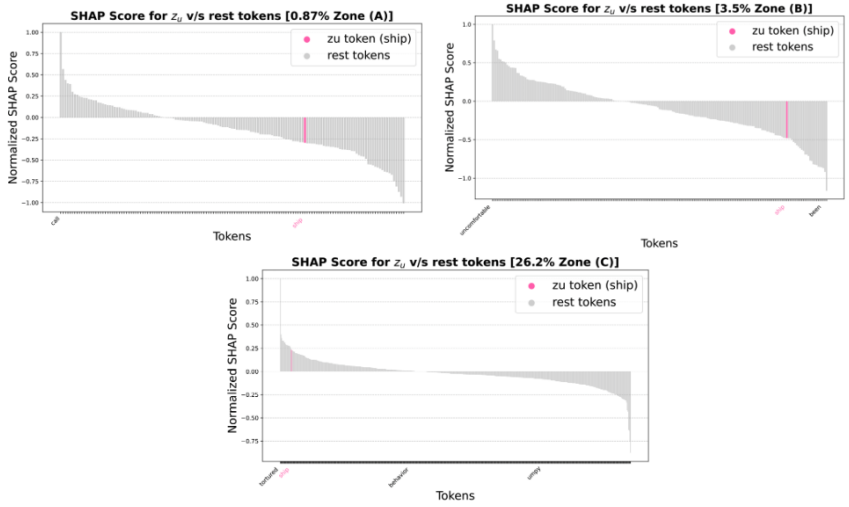
Figure 16: **FM and shift to FG with increasing x_u train samples**. We present it using S_{sens}, $\text{MP}_{2,4}$, p-value, train-test accuracy gap for x_u train-test samples, and Shapley-Value results, with Avg-MP-Diff threshold: 3.5% (σ_1), Avg-TestAcc-Diff: 3% (σ_2). FM starts at 2.1% (zone B) and shifts to FG after 5.24% (zone C), with neither occurring at very low proportions (zone A). Shapley values confirm these transitions, where for Zone (B) of FM, "egg"-bar plot has negligible importance (SHAP score ≈ 0) in testing even though it has been recognized, while for Zone (C) of FG, "egg" gains importance (SHAP score $\gg 0$), showing generalization of "egg" to the test set. (Emotions, BERT, z_u in class 4 in train, 2 in test). We do not plot Avg-MP-Diff and Avg-TestAcc-Diff in Zone (A) because the model has not even recognized z_u, hence it can neither memorize nor generalize it.



(a) S_{sens}, $\text{MP}_{3,4}$, and P-value



(b) Test Acc for Class 3 x_u samples and Train Acc

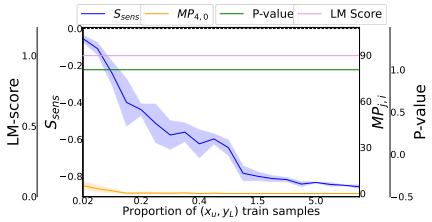


(c) Shapley Values for (x_u, y) ratios in class 4

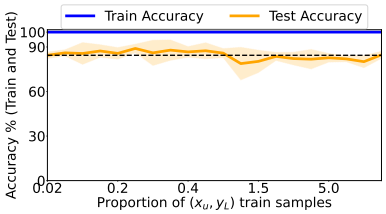
Figure 17: **FM and shift to FG with increasing x_u train samples**. We present it using S_{sens}, $\text{MP}_{3,4}$, p-value, train-test accuracy gap for x_u train-test samples, and Shapley-Value results, with Avg-MP-Diff threshold: 3.5% (σ_1), Avg-TestAcc-Diff: 3% (σ_2). FM starts at 3.5% (zone B) and shifts to FG after 6.99% (zone C), with neither occurring at very low proportions (zone A). Shapley values confirm these transitions, where for Zone (B) of FM, "ship" bar plot has negligible importance (SHAP score < 0) in testing even though it has been recognized, while for Zone (C) of FG, "ship" gains importance (SHAP score $\gg 0$), showing generalization of "ship" to the test set. (Emotions, RoBERTa, z_u in class 4 in train, 3 in test). We do not plot Avg-MP-Diff and Avg-TestAcc-Diff in Zone (A) because the model has not even recognized z_u, hence it can neither memorize nor generalize it.

E.2 JOINT FEATURE AND *Random* NOISY LABEL SETUP

This section covers the results for the remaining datasets, CIFAR10, CIFAR100, UTK-Face, Medical Abstracts, Emotions, and models, ResNet9, GoogleNet, VGG16, ResNet18, BERT, RoBERTa, for increasing proportions of (x_u, y_L) in the training set. The results for all these datasets, models are presented in Figures 18, 19, 20, 21, 22, 23, and 24.

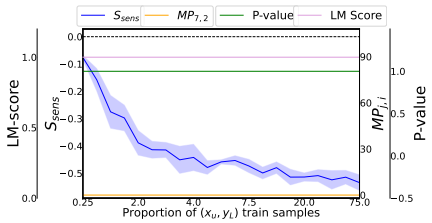


(a) S_{sens}, $\mathrm{MP}_{4,0}$ and P-value

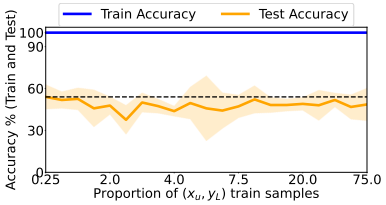


(b) Test Acc for class 4 x_u samples and Train Acc

Figure 18: **Disappearance of FM and FG.** As (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, while $\mathrm{MP}_{4,0}$ stays minimal. The train-test accuracy gap for x_u samples remains stable, indicating the disappearance of FM and FG, while *random* LM is still present.(CIFAR10, ResNet9, z_u in class 0 in train, 4 in test)

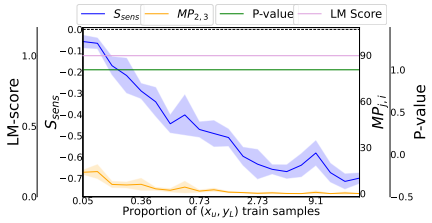


(a) S_{sens}, $\mathrm{MP}_{7,2}$ and P-value

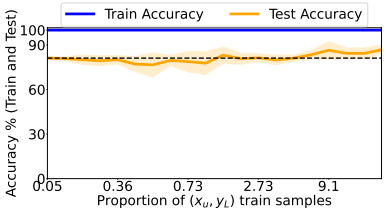


(b) Test Acc for class 7 x_u samples and Train Acc

Figure 19: **Disappearance of FM and FG.** As (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, while $\mathrm{MP}_{7,2}$ stays minimal. The train-test accuracy gap for x_u samples remains stable, indicating the disappearance of FM and FG, while *random* LM is still present. (CIFAR100, GoogleNet, z_u in class 2 in train, 7 in test)

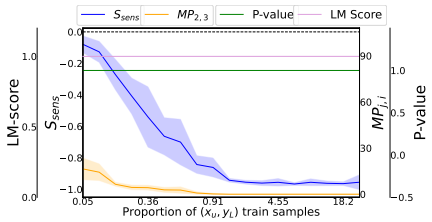


(a) S_{sens}, $\mathrm{MP}_{2,3}$ and P-value

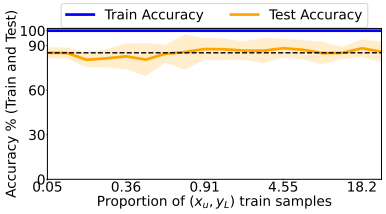


(b) Test Acc for class 2 x_u samples and Train Acc

Figure 20: **Disappearance of FM and FG.** As (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, while $\mathrm{MP}_{2,3}$ stays minimal. The train-test accuracy gap for x_u samples remains stable, indicating the disappearance of FM and FG, while *random* LM is still present.(UTK-Face, ResNet18, z_u in class 3 in train, 2 in test)

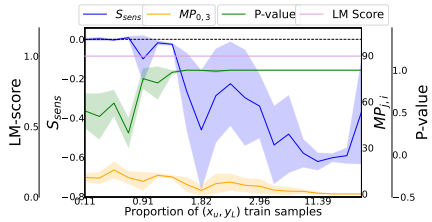


(a) S_{sens}, $\mathrm{MP}_{2,3}$ and P-value

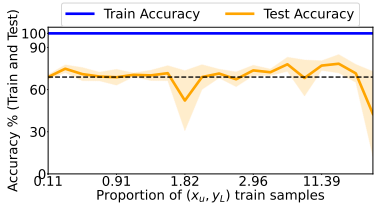


(b) Test Acc for class 2 x_u samples and Train Acc

Figure 21: **Disappearance of FM and FG.** As (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, while $\mathrm{MP}_{2,3}$ stays minimal. The train-test accuracy gap for x_u samples remains stable, indicating the disappearance of FM and FG, while *random* LM is still present.(UTK-Face, VGG16, z_u in class 3 in train, 2 in test)

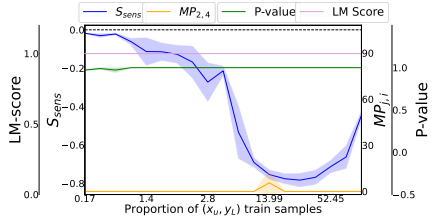


(a) S_{sens}, $\mathrm{MP}_{0,3}$ and P-value

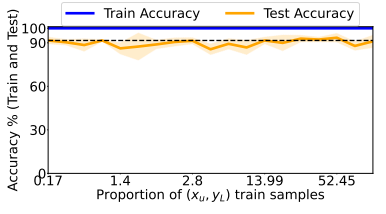


(b) Test Acc for class 0 x_u samples and Train Acc

Figure 22: **Disappearance of FM and FG.** As (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, while $\mathrm{MP}_{0,3}$ stays minimal. The train-test accuracy gap for x_u samples remains stable, indicating the disappearance of FM and FG, while *random* LM is still present. (Medical Abstracts, RoBERTa, z_u in class 3 in train, 0 in test).

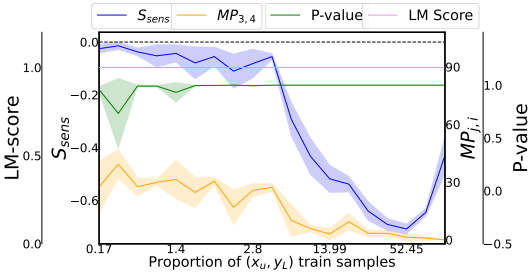


(a) S_{sens}, $\mathrm{MP}_{2,4}$ and P-value

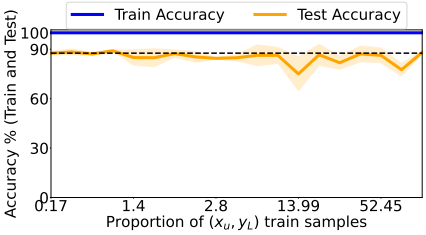


(b) Test Acc for class 2 x_u samples and Train Acc

Figure 23: **Disappearance of FM and FG.** As (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, while $\mathrm{MP}_{2,4}$ stays minimal. The train-test accuracy gap for x_u samples remains stable, indicating the disappearance of FM and FG, while *random* LM is still present. (Emotions, BERT, z_u in class 4 in train, 2 in test).



(a) S_{sens}, $\mathrm{MP}_{3,4}$ and P-value

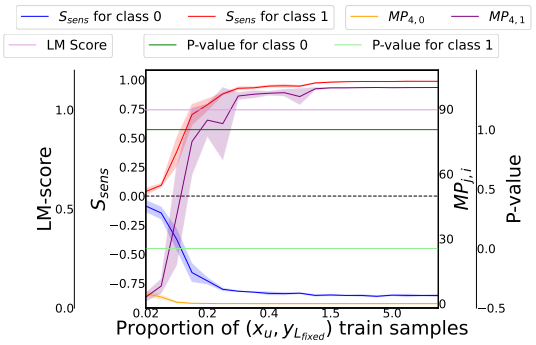


(b) Test Acc for class 3 x_u samples and Train Acc

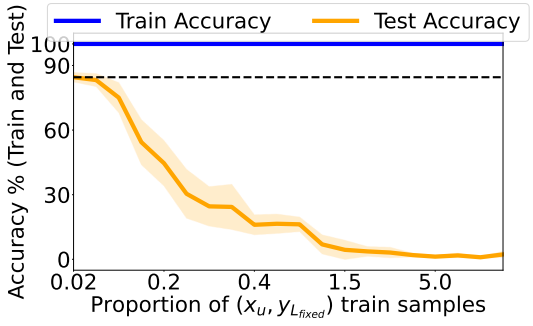
Figure 24: **Disappearance of FM and FG.** As (x_u, y_L) training samples increases, S_{sens} becomes negative and further decreases, while $\mathrm{MP}_{3,4}$ stays minimal. The train-test accuracy gap for x_u samples remains stable, indicating the disappearance of FM and FG, while *random* LM is still present. (Emotions, RoBERTa, z_u in class 4 in train, 3 in test).

E.3 JOINT FEATURE AND *Fixed* NOISY LABEL SETUP

In this section, we provide additional results across the remaining datasets-models configurations, for CIFAR10, CIFAR100, UTK-Face, NICO++, Medical Abstracts, Emotions, corresponding to models, ResNet9, GoogleNet, VGG16, MobileViT, BERT, RoBERTa, while varying the proportions of $(x_u, y_{L_{\text{fixed}}})$ in the training set. Furthermore, we also provide additional plots on how the induced feature generalization phenomenon goes in parallel with label memorization over the course of training. These results on the remaining datasets, models further confirm our claim that ***label memorization causes feature generalization*** and ***suppresses feature memorization.*** The results corresponding to CIFAR10 & ResNet9, UTK-Face & VGG16, CIFAR100 & GoogleNet, NICO++ & MobileViT, and Emotions & BERT, RoBERTa, are presented in Figs. 25, 26, 27, 28, 29, 30, 31, 32, 33, and 34, respectively.



(a) S_{sens} for class 0 & 1, $\text{MP}_{4,0}$, $\text{MP}_{4,1}$, P-values, LM-score

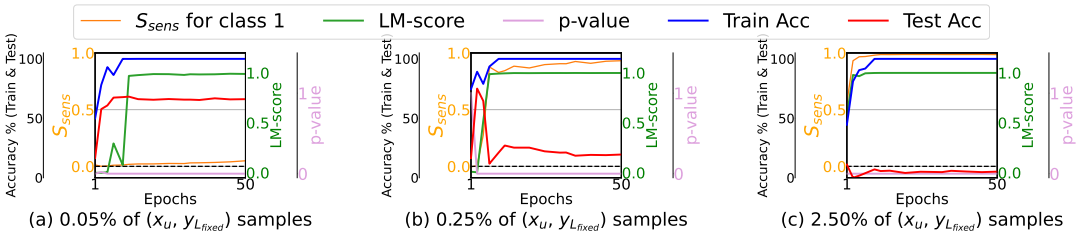


(b) Test Acc for class 4 x_u samples and Train Acc



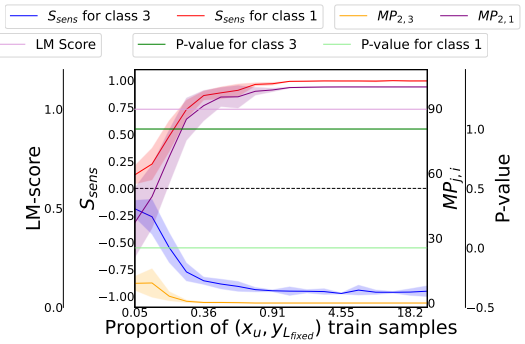
(c) GradCam results for $(x_u, y_{L_{\text{fixed}}})$ ratios in class 0

Figure 25: **LM causes FG and suppresses FM.** With simultaneous addition of z_u and $y_{L_{\text{fixed}}}$ to the same (image, label) pair, LM leads the model to associate z_u with class 1, as evidenced by the increasing S_{sens} score for class 1. We present vanishing of FM and induced FG even for low proportions of $(x_u, y_{L_{\text{fixed}}})$, supported through high S_{sens} score, high $\text{MP}_{4,1}$, exacerbating train-test accuracy gap, and GradCam plots. This phenomenon becomes even more intense for higher proportions. (CIFAR10, ResNet9, z_u in class 0 in train, 4 in test & $y_{L_{\text{fixed}}}$= 1)

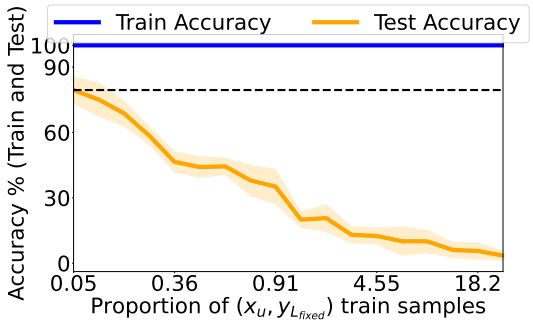


(a) 0.05% of $(x_u, y_{L_{fixed}})$ samples (b) 0.25% of $(x_u, y_{L_{fixed}})$ samples (c) 2.50% of $(x_u, y_{L_{fixed}})$ samples

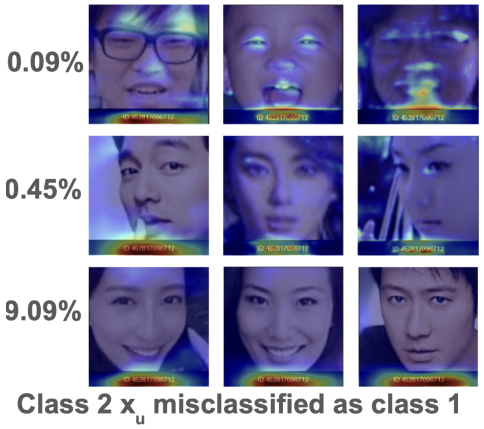
Figure 26: **Induced FG and LM occur in parallel over epochs.** We observe that when z_u and $y_{L_{\text{fixed}}}$ are in the same (image, label) pair, then LM causes FG concurrently over epochs even for extremely low concentration of z_u in training. This observation is also supported by increasing LM score, increasing S_{sens} score and widened train-test accuracy gap as training progresses. (CIFAR10, ResNet9, z_u in class 0 in train, 4 in test & $y_{L_{\text{fixed}}}$= 1)



(a) S_{sens} for class 3 & 1, $\text{MP}_{2,3}$, $\text{MP}_{2,1}$, P-values, LM-score

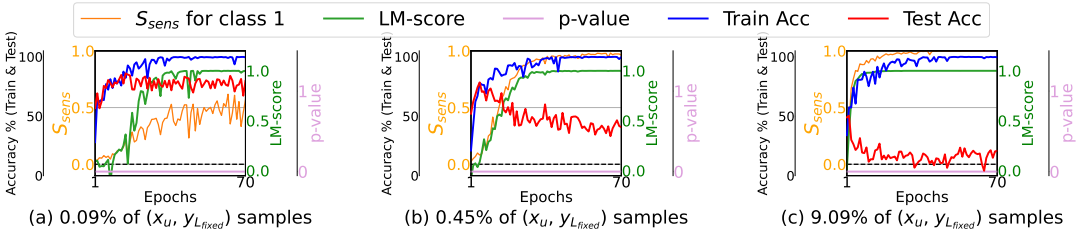


(b) Test Acc for class 2 x_u samples and Train Acc



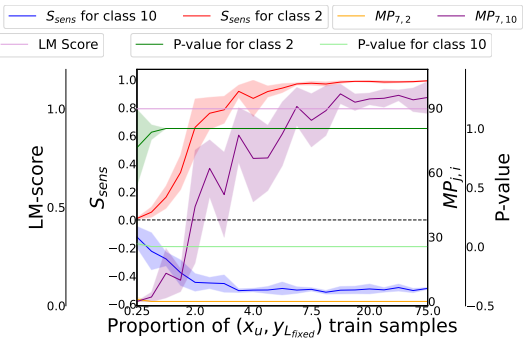
(c) GradCam results for $(x_u, y_{L_{\text{fixed}}})$ ratios in class 3

Figure 27: **LM causes FG and suppresses FM.** With concurrent addition of z_u and $y_{L_{\text{fixed}}}$, *fixed* LM leads the model to associate z_u with class 1, as evidenced by the increasing S_{sens} score for class 1. It presents vanishing of FM and induced FG even for low proportions of $(x_u, y_{L_{\text{fixed}}})$, supported by high S_{sens} score, high $\text{MP}_{2,1}$, wider train-test accuracy gap, and GradCam.(UTK-Face, VGG16, z_u in class 3 in train, 2 in test & $y_{L_{\text{fixed}}}$= 1)

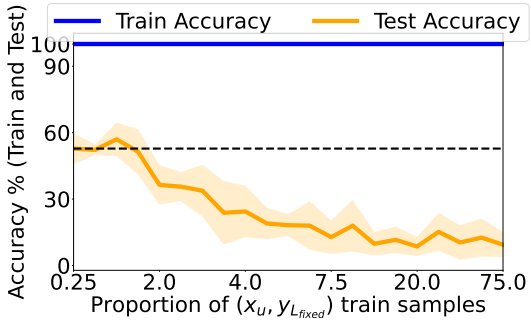


(a) 0.09% of $(x_u, y_{L_{fixed}})$ samples (b) 0.45% of $(x_u, y_{L_{fixed}})$ samples (c) 9.09% of $(x_u, y_{L_{fixed}})$ samples

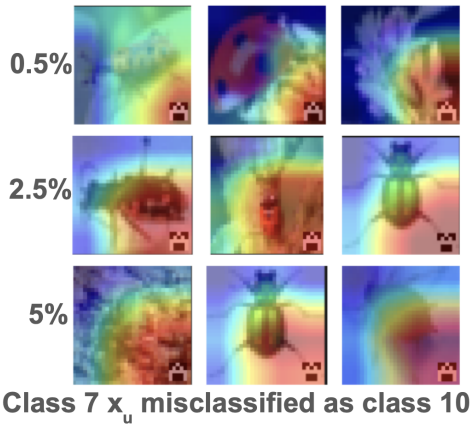
Figure 28: **Induced FG and LM occur in parallel over epochs.** We observe that when z_u and $y_{L_{\text{fixed}}}$ are in the same (image, label) pair, LM causes FG concurrently over epochs even for extremely low concentration of z_u in training. This is supported by increasing LM score and S_{sens} score, and widening train-test accuracy gap as training progresses. (UTK-Face, VGG16, z_u in class 3 in train, 2 in test & $y_{L_{\text{fixed}}}$= 1)



(a) S_{sens} for class 10 & 2, $\text{MP}_{7,2}$, $\text{MP}_{7,10}$, P-values, LM-score

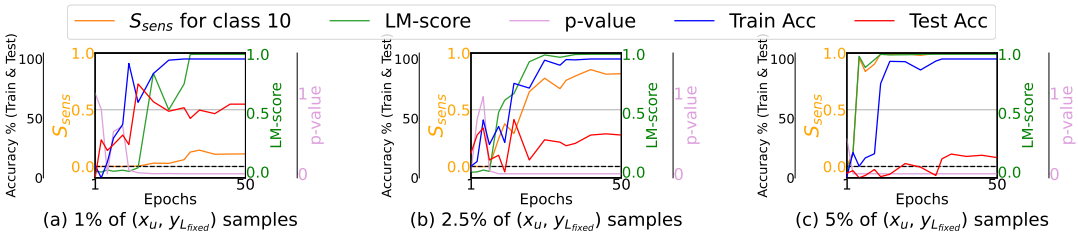


(b) Test Acc for class 7 x_u samples and Train Acc



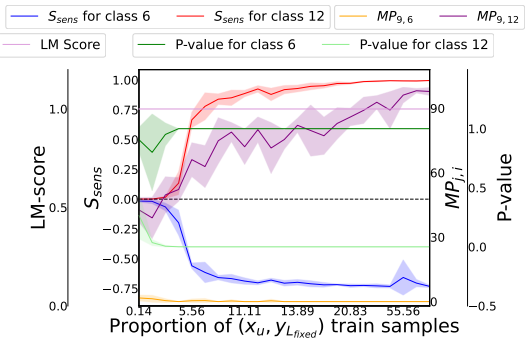
(c) GradCam results for $(x_u, y_{L_{\text{fixed}}})$ ratios in class 2

Figure 29: **LM causes FG and suppresses FM.** With concurrent addition of z_u and $y_{L_{\text{fixed}}}$, LM leads the model to associate z_u with class 10, as evidenced by the increasing S_{sens} score for class 10. It presents vanishing of FM and induced FG even for low proportions of $(x_u, y_{L_{\text{fixed}}})$, supported by high S_{sens} score, high $\text{MP}_{7,10}$, wider train-test accuracy gap, and GradCam.(CIFAR100, GoogleNet, z_u in class 2 in train, 7 in test & $y_{L_{\text{fixed}}}$= 10)

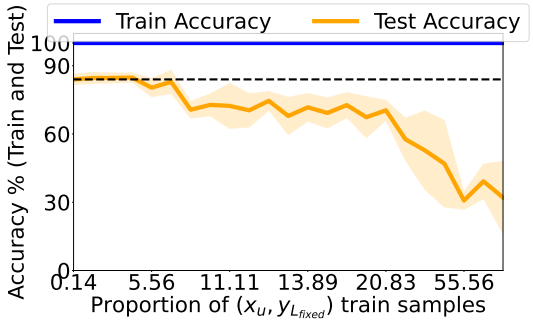


(a) 1% of $(x_u, y_{L_{fixed}})$ samples (b) 2.5% of $(x_u, y_{L_{fixed}})$ samples (c) 5% of $(x_u, y_{L_{fixed}})$ samples

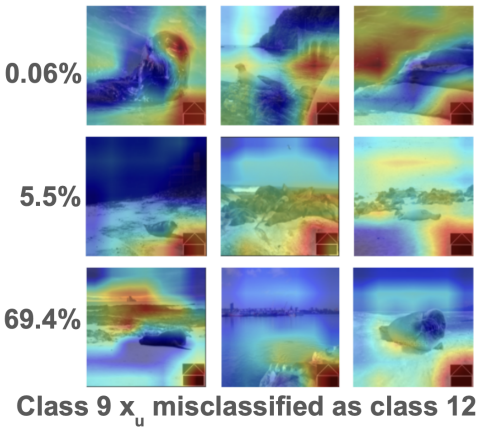
Figure 30: **Induced FG and LM occur in parallel over epochs.** We observe that when z_u and $y_{L_{\text{fixed}}}$ are in the same (image, label) pair, then LM causes FG concurrently over epochs even for extremely low concentration of z_u in training. This observation is also supported by increasing LM score, increasing S_{sens} score, and widening of train-test accuracy gap as training progresses.(CIFAR100, GoogleNet, z_u in class 2 in train, 7 in test & $y_{L_{\text{fixed}}}$= 10)



(a) S_{sens} for class 6 & 12, $\text{MP}_{9,6}$, $\text{MP}_{9,12}$, P-values, LM-score

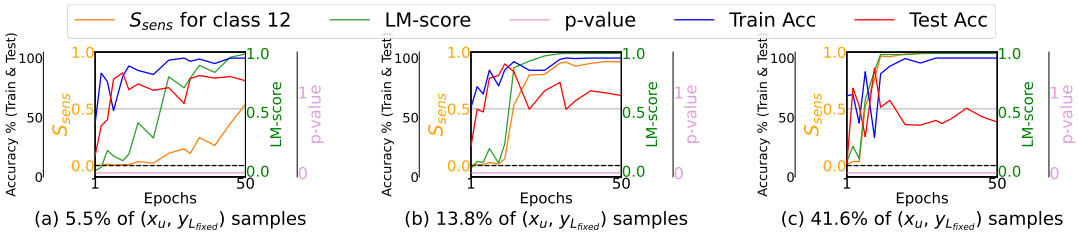


(b) Test Acc for class 9 x_u samples and Train Acc



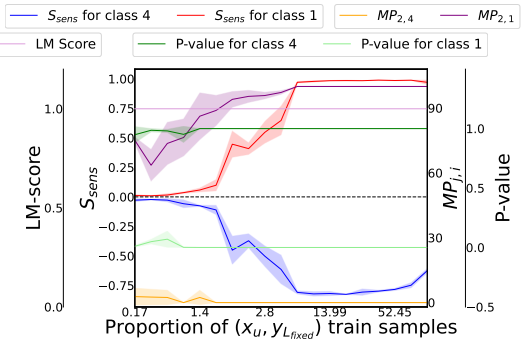
(c) GradCam results for $(x_u, y_{L_{\text{fixed}}})$ ratios in class 6

Figure 31: **LM causes FG and suppresses FM.** With concurrent addition of z_u and $y_{L_{\text{fixed}}}$, LM leads the model to associate z_u with class 12, as evidenced by the increasing S_{sens} score for class 12. It presents vanishing of FM and induced FG even for low proportions of $(x_u, y_{L_{\text{fixed}}})$, supported by high S_{sens} score, high $\text{MP}_{9,12}$, wider train-test accuracy gap, and GradCam.(NICO++, MobileViT, z_u in class 6 in train, 9 in test & $y_{L_{\text{fixed}}}$=12)

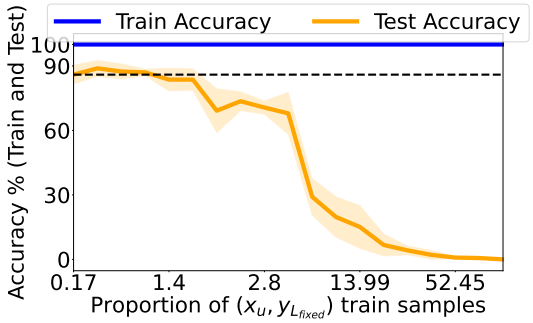


(a) 5.5% of $(x_u, y_{L_{fixed}})$ samples (b) 13.8% of $(x_u, y_{L_{fixed}})$ samples (c) 41.6% of $(x_u, y_{L_{fixed}})$ samples

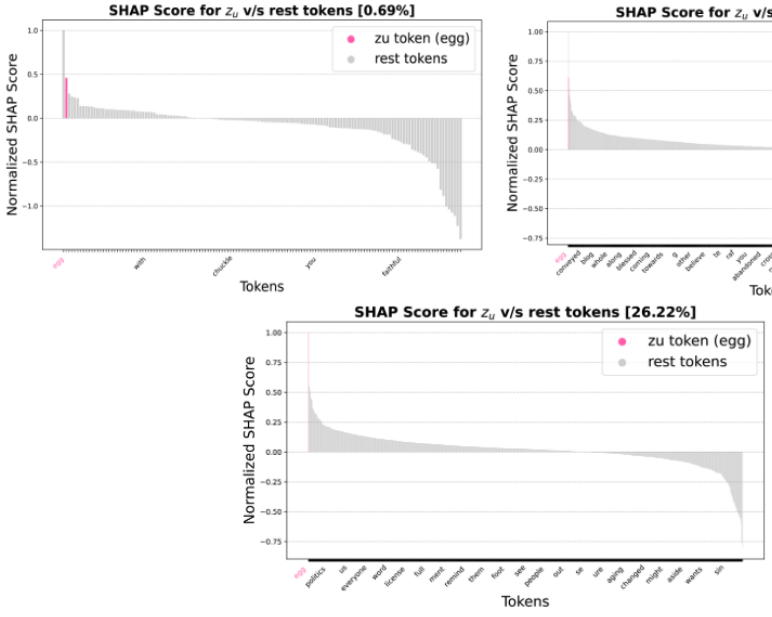
Figure 32: **Induced FG and LM occur in parallel over epochs.** We observe that when z_u and $y_{L_{\text{fixed}}}$ are in the same (image, label) pair, then LM causes FG concurrently over epochs even for extremely low concentration of z_u in training. This observation is also supported by increasing LM score, increasing S_{sens} score and widening of the train-test accuracy gap as training progresses. (NICO++, MobileViT, z_u in class 6 in train, 9 in test & $y_{L_{\text{fixed}}}$=12)



(a) S_{sens} for class 4 & 1, $\text{MP}_{2,4}$, $\text{MP}_{2,1}$, P-values, LM-score

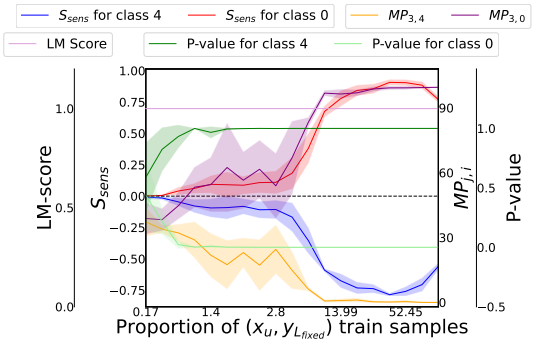


(b) Test Acc for Class 2 x_u samples and Train Acc

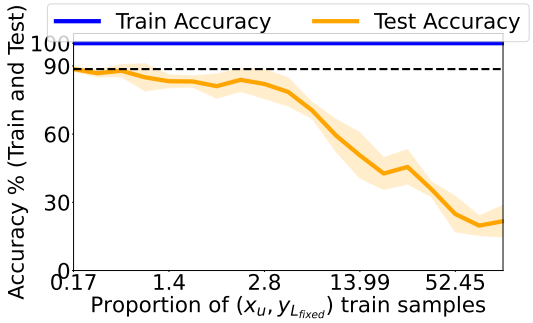


(c) Shapley Values for $(x_u, y_{L_{\text{fixed}}})$ ratios in class 4

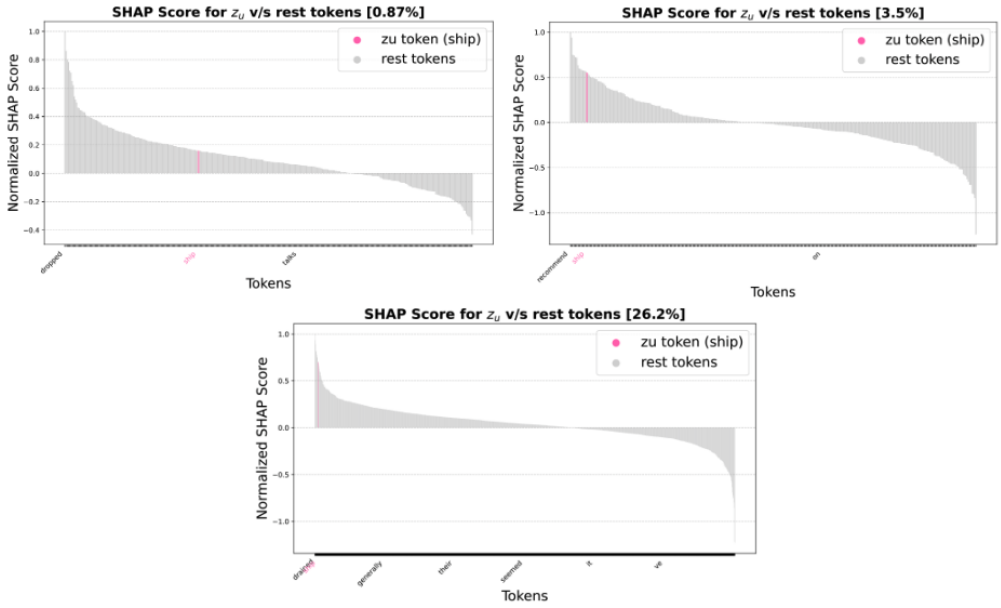
Figure 33: **LM causes FG and suppresses FM.** With simultaneous addition of z_u and $y_{L_{\text{fixed}}}$ to the same (image, label) pair, LM leads the model to associate z_u with class 1, as evidenced by the increasing S_{sens} score for class 1. We present LM causing FG and suppressing FM even for low proportions of $(x_u, y_{L_{\text{fixed}}})$, supported through high S_{sens} score, high $\text{MP}_{2,1}$, exacerbating train-test accuracy gap. The Shapley plots further confirm this, where even for low proportions of 0.69%, "egg" bar plot gains importance (SHAP score $\gg 0$) in the test-time predictions. This phenomenon becomes even more intense for higher proportions.(Emotions, BERT, z_u in class 4 in train, 2 in test & $y_{L_{\text{fixed}}}$=1)



(a) S_{sens} for class 4 & 0, $\text{MP}_{3,4}$, $\text{MP}_{3,0}$, P-values, LM-score



(b) Test Acc for Class 3 x_u samples and Train Acc



(c) Shapley Values for $(x_u, y_{L_{\text{fixed}}})$ ratios in class 4

Figure 34: **LM causes FG and suppresses FM.** With simultaneous addition of z_u and $y_{L_{\text{fixed}}}$ to the same (image, label) pair, LM leads the model to associate z_u with class 0, as evidenced by the increasing S_{sens} score for class 0. We present LM causing FG and suppressing FM even for low proportions of $(x_u, y_{L_{\text{fixed}}})$, supported through high S_{sens} score, high $\text{MP}_{3,0}$, exacerbating the train-test accuracy gap. The Shapley plots further confirm this, where even for low proportions of 0.87%, "ship" bar plot gains importance (SHAP score $\gg 0$) in the test-time predictions. This phenomenon becomes even more intense for higher proportions.(Emotions, RoBERTa, z_u in class 4 in train, 3 in test & $y_{L_{\text{fixed}}}$=0)

F DEFINING THRESHOLDS (σ_1 AND σ_2) FOR DISTINGUISHING FM AND FG

We would like to first reiterate the fact that chosen threshold values can differ in other settings and architectures. However, the main goal of this study is not provide an optimal thresholding scheme but rather to articulate the novel distinction between feature memorization and generalization via a framework. In the datasets and models considered in our paper, $\sigma_1 = 3.5\%$ and $\sigma_2 = 3\%$, are reasonable with our findings. Please note that our experiments are run across 5 seeds. For every dataset, we compute the **Avg** $\mathrm{MP_{j,i}}$ **(%)** and **Avg Test Accuracy of class** C_j **samples (%)** containing z_u. These are calculated across increasing ratios of x_u training samples containing z_u.

After computing them, we find the **Avg-MP-Diff (%)** and **Avg-TestAcc-Diff (%)** scores as defined in Sec. 3, for varying ratios (%) of training samples containing z_u for class C_i.

We provide tabular results for all 8 datasets-models configurations across vision and language modalities, and analyze them one by one: (Wherever we see N/A in the tables, it means neither FM nor FG occurs, i.e., the model has not recognized the feature z_u. Furthermore, we provide the scores when the concentration of z_u was 0% in training for baseline comparisons.)

F.1 FOR CIFAR10-RESNET9: (DURING TRAINING WE HAVE 4,000 SAMPLES OF CLASS 0 IN WHICH WE INTRODUCE z_u IN VARYING RATIOS (%))

Table 1: Thresholds (σ_1 and σ_2) analysis for CIFAR10-ResNet9 Setup

Ratio (%) of x_u Train Samples	**Avg $\mathrm{MP}_{4,0}$ (in %)**	**Avg-MP-Diff (in %)**	**Avg Test Accuracy (in %)**	**Avg-TestAcc -Diff (in %)**
0.0	9.98	0	85.28	0
0.03	N/A	N/A	N/A	N/A
0.05	N/A	N/A	N/A	N/A
0.1	8.92	-1.05	83.65	1.63
0.15	8.32	-1.65	86.95	-1.67
0.2	8.63	-1.35	85.65	-0.37
0.25	10.25	0.28	85.6	-0.32
0.3	10.14	0.17	84.35	0.93
0.35	10.72	0.74	83.84	1.44
0.4	12.29	2.32	84.18	1.1
0.45	11.33	1.35	83.48	1.8
0.5	12.74	2.76	84.42	0.86
1.0	18.0	8.02	82.94	2.34
1.5	24.31	14.33	79.52	5.76
2.0	51.17	41.2	73.8	11.48
2.5	61.47	51.49	63.76	21.52
3.75	88.89	78.91	43.46	41.82
5.0	93.85	83.87	37.98	47.3
7.5	97.0	87.02	25.84	59.44
10.0	98.72	88.75	25.1	60.18
12.5	99.43	89.46	17.34	67.94

From Table 1, we can clearly observe that by considering σ_1 **= 3.5% and** σ_2 **= 3%**, we can delineate **feature memorization** and **feature generalization**. For instance, when the x_u train samples ratio lies **between 0.1% and 0.5%, FM occurs**, with both Avg-MP-Diff and Avg-TestAcc-Diff being less than the thresholds, also aligning with the Grad-Cam analysis (Fig. 1c), where in this range, the model does not associate the label with feature z_u during testing. Now, as the x_u samples ratio increases beyond **0.5%, up to 12.5%, the model transitions from FM to FG**, where Avg-MP-Diff and Avg-TestAcc-Diff values exceed σ_1 and σ_2, further aligning with our Grad-Cam analysis where in this region, the model begins to rely on z_u during testing. The results correspond to CIFAR10-ResNet9 setting.

F.2 FOR CIFAR100-GOOGLENET: (DURING TRAINING, WE HAVE 400 SAMPLES OF CLASS 2 IN WHICH WE INTRODUCE z_u IN VARYING RATIOS (%))

Table 2: Thresholds (σ_1 and σ_2) analysis for CIFAR100-GoogleNet Setup

Ratio (%) of x_u Train Samples	**Avg MP$_{7,2}$ (in %)**	**Avg-MP-Diff (in %)**	**Avg Test Accuracy (in %)**	**Avg-TestAcc -Diff (in %)**
0.0	0.61	0	52.4	0
0.25	N/A	N/A	N/A	N/A
0.5	N/A	N/A	N/A	N/A
1.0	1.02	0.41	52.5	-0.1
1.5	0.74	0.13	55.0	-2.6
2.0	1.23	0.62	56.5	-4.1
2.5	2.28	1.67	54.6	-2.2
3.0	1.48	0.88	58.6	-6.2
3.5	4.03	3.42	56.6	-4.2
4.0	3.72	3.11	55.8	-3.4
4.5	7.41	6.8	52.8	-0.4
5.0	9.22	8.62	58.6	-6.2
6.25	14.76	14.15	48.4	4.0
7.5	41.44	40.83	50.8	1.6
8.75	44.51	43.9	41.2	11.2
10.0	48.75	48.14	39.4	13.0
15.0	71.74	71.13	43.0	9.4
20.0	72.26	71.65	27.0	25.4
25.0	84.66	84.05	27.6	24.8
37.5	85.79	85.18	19.4	33.0
50.0	85.9	85.3	15.4	37.0
75.0	94.11	93.5	12.0	40.4

We again observe in Table 2, that for **CIFAR-100**, the same thresholds, σ_1 **= 3.5% and** σ_2 **= 3%**, perfectly fit to delineate **feature memorization and feature generalization**. **FM** occurs when ratio of x_u train samples lies between 1% and 3%, with both Avg-MP-Diff and Avg-TestAcc-Diff remaining below σ_1 and σ_2, respectively, also aligning with the Grad-Cam analysis (Fig. 12c), where in this range, the model does not associate the label with feature z_u during testing. Beyond **3%**, the model transitions from **FM to FG**, with Avg-MP-Diff and Avg-TestAcc-Diff exceeding the thresholds, as concentration of z_u increases in training, further aligning with our Grad-Cam analysis where in this region, the model begins to rely on z_u during testing. The results correspond to CIFAR100-GoogleNet setting.

F.3 FOR UTK-FACE-RESNET18: (DURING TRAINING WE HAVE 2,198 SAMPLES OF CLASS 3 IN WHICH WE INTRODUCE z_u IN VARYING RATIOS (%))

Table 3: Thresholds (σ_1 and σ_2) analysis for UTK-Face-ResNet18 Setup

Ratio (%) of x_u Train Samples	**Avg $MP_{2,3}$ (in %)**	**Avg-MP-Diff (in %)**	**Avg Test Accuracy (in %)**	**Avg-TestAcc -Diff (in %)**
0.0	20.68	0.0	77.38	0.0
0.05	N/A	N/A	N/A	N/A
0.09	N/A	N/A	N/A	N/A
0.18	18.82	-1.86	75.55	1.83
0.27	20.45	-0.24	77.33	0.05
0.36	27.86	7.18	78.49	-1.11
0.45	15.75	-4.93	80.79	-3.41
0.55	19.77	-0.92	79.85	-2.47
0.64	19.79	-0.89	77.18	0.2
0.73	25.41	3.41	76.8	0.58
0.82	32.36	6.67	77.58	-0.2
0.91	21.3	0.62	79.99	-2.61
1.0	28.51	7.82	80.93	-3.55
1.14	34.98	14.29	75.25	2.13
1.36	32.16	11.47	80.17	-2.79
1.59	35.54	14.86	76.22	1.16
1.82	48.64	27.96	78.81	-1.43
2.73	53.57	32.89	76.83	0.55
4.55	57.92	37.24	69.61	7.77
9.1	84.59	63.91	50.33	27.05
13.65	93.75	73.07	49.58	27.8
18.2	98.79	78.11	30.66	46.72
22.75	97.68	76.99	20.93	56.45

We observe from Table 3, that the same thresholds, σ_1 **= 3.5% and** σ_2 **= 3%**, remain robust and perfectly applicable to distinguish **feature memorization and feature generalization** for **UTK-Face-ResNet18** setup as well. **FM** occurs when the x_u train samples ratio lies between **0.18% and 0.73%**, with both Avg-MP-Diff and Avg-TestAcc-Diff remaining below the thresholds, also aligning with the Grad-Cam analysis (Fig. 8c), where in this range, the model does not associate the label with feature z_u during testing. Beyond **0.73%**, the model transitions from **FM to FG**, as Avg-MP-Diff and Avg-TestAcc-Diff significantly exceed both σ_1 and σ_2, further aligning with our Grad-Cam analysis where in this region, the model begins to rely on z_u during testing. The results correspond to UTK-Face-ResNet18 setting.

F.4 FOR UTK-FACE-VGG16: (DURING TRAINING WE HAVE 2,198 SAMPLES OF CLASS 3 IN WHICH WE INTRODUCE z_u IN VARYING RATIOS (%))

Table 4: Thresholds (σ_1 and σ_2) analysis for UTK-Face-VGG16 Setup

Ratio (%) of x_u Train Samples	**Avg $MP_{2,3}$ (in %)**	**Avg-MP-Diff (in %)**	**Avg Test Accuracy (in %)**	**Avg-TestAcc -Diff (in %)**
0.0	15.57	0.0	81.28	0.0
0.05	N/A	N/A	N/A	N/A
0.09	N/A	N/A	N/A	N/A
0.18	14.81	-0.77	82.97	-1.69
0.27	15.04	-0.54	82.79	-1.51
0.36	16.42	0.84	82.82	-1.54
0.45	16.73	1.16	83.26	-1.98
0.55	17.16	1.58	82.91	-1.63
0.73	17.63	2.06	83.06	-1.78
0.91	31.6	16.03	82.53	-1.25
1.14	37.31	21.73	81.92	-0.64
1.36	44/63	29.06	81.34	-0.06
1.59	52.51	36.94	79.94	1.34
1.82	58.78	43.2	79.07	2.21
2.73	82.28	66.71	69.49	11.79
4.55	96.45	80.87	42.3	38.98
9.1	99.38	83.8	15.23	66.06
13.65	99.69	84.11	6.99	74.29
18.2	99.76	84.18	4.57	76.71
22.75	99.76	84.18	4.22	77.06

We observe from Table 4, that the same thresholds, σ_1 **= 3.5% and** σ_2 **= 3%**, remain robust and perfectly applicable to distinguish **feature memorization and feature generalization** for **UTK-Face-VGG16** setup as well. **FM** occurs when the x_u train samples ratio lies between **0.18% and 0.73%**, with both Avg-MP-Diff and Avg-TestAcc-Diff remaining below the thresholds, also aligning with the Grad-Cam analysis (Fig. 1c), where in this range, the model does not associate the label with feature z_u during testing. Beyond **0.73%**, the model transitions from **FM to FG**, as Avg-MP-Diff and Avg-TestAcc-Diff significantly exceed both σ_1 and σ_2, further aligning with our Grad-Cam analysis where in this region, the model begins to rely on z_u during testing. The results correspond to UTK-Face-VGG16 setting.

F.5 FOR NICO++, MOBILEVIT: (DURING TRAINING WE HAVE 720 SAMPLES OF CLASS 6 IN WHICH WE INTRODUCE z_u IN VARYING RATIOS (%))

Table 5: Thresholds (σ_1 and σ_2) analysis for NICO++ - MobileViT Setup

Ratio (%) of x_u Train Samples	**Avg MP$_{9,6}$ (in %)**	**Avg-MP-Diff (in %)**	**Avg Test Accuracy (in %)**	**Avg-TestAcc -Diff (in %)**
0.0	5.12	0	81.42	0
0.14	N/A	N/A	N/A	N/A
0.69	N/A	N/A	N/A	N/A
1.39	N/A	N/A	N/A	N/A
2.78	8.0	2.88	88.89	-7.47
5.56	0.0	-5.12	79.56	1.87
8.33	8.61	3.49	79.85	1.57
9.72	1.93	-3.19	83.56	-2.13
10.42	3.79	-1.33	83.22	-1.8
11.11	2.05	-3.07	84.78	-3.36
11.39	6.46	1.34	87.0	-5.58
11.94	5.53	0.41	87.38	-5.96
12.5	4.39	-0.73	81.56	-0.13
13.89	4.29	-0.83	81.56	-0.13
16.67	7.67	2.55	84.44	-3.02
18.06	5.55	0.43	82.49	-1.07
19.44	5.8	0.69	86.49	-5.07
20.83	4.28	-0.84	80.09	1.33
27.78	10.53	5.41	83.82	-2.4
34.72	17.37	12.25	82.58	-1.16
41.67	18.36	13.24	79.91	1.51
55.56	43.03	37.91	78.58	2.84
69.44	57.63	52.51	68.44	12.98
83.33	76.35	71.23	62.13	19.29

For **NICO++, MobileViT** as well, the thresholds, σ_1 **= 3.5% and** σ_2 **= 3%**, remain robust and perfectly applicable to distinguish **feature memorization and feature generalization**, as observed in Table 5. **FM** occurs with the x_u train samples ratio lying between **2.78% and 20.83%**, with both Avg-MP-Diff and Avg-TestAcc-Diff remaining below the thresholds, also aligning with the Grad-Cam analysis (Fig. 14c), where in this range, the model does not associate the label with feature z_u during testing. Beyond **20.83%**, the model transitions from **FM to FG**, as Avg-MP-Diff and Avg-TestAcc-Diff exceed the thresholds, further aligning with our Grad-Cam analysis where in this region, the model begins to rely on z_u during testing. The results correspond to NICO++, MobileViT setting.

F.6 FOR MEDICAL ABSTRACTS-ROBERTA: (DURING TRAINING WE HAVE 878 SAMPLES OF CLASS 3 IN WHICH WE INTRODUCE z_u IN VARYING RATIOS (%))

Table 6: Thresholds (σ_1 and σ_2) analysis for Medical Abstracts-RoBERTa Setup

Ratio (%) of x_u Train Samples	**Avg MP$_{0,3}$ (in %)**	**Avg-MP-Diff (in %)**	**Avg Test Accuracy (in %)**	**Avg-TestAcc -Diff (in %)**
0.0	10.64	0	69.17	0
0.11	N/A	N/A	N/A	N/A
0.23	N/A	N/A	N/A	N/A
0.46	N/A	N/A	N/A	N/A
0.68	N/A	N/A	N/A	N/A
0.91	9.69	-0.95	68.01	1.16
1.14	9.03	-1.61	69.76	-0.59
1.37	13.33	2.69	76.28	-7.11
1.59	13.02	2.38	68.38	0.79
1.82	12.48	1.84	67.72	1.45
2.05	11.18	0.54	73.72	-4.55
2.28	6.06	-4.58	73.91	-1.89
2.51	11.68	1.04	67.72	2.52
2.96	11.48	0.84	68.97	2.52
3.42	16.62	5.98	67.46	7.76
4.56	58.55	47.91	54.68	3.14
5.69	45.43	34.79	42.82	3.35
11.39	94.35	83.71	10.01	3.35
22.78	100	89.36	0.0	10.06
34.17	100	89.36	0.0	3.77
45.56	100	89.36	0.0	5.66

For **Medical Abstracts-RoBERTa** as well, the thresholds, σ_1 **= 3.5% and** σ_2 **= 3%**, remain robust and perfectly applicable to distinguish **feature memorization and feature generalization**, as observed in Table 6. **FM** occurs with the x_u train samples ratio lying between **0.91% and 2.96%**, with both Avg-MP-Diff and Avg-TestAcc-Diff remaining below the thresholds, also aligning with the SHAP-Score analysis (Fig. 3c), where in this range, the model does not associate the label with feature z_u during testing. Beyond **2.96%**, the model transitions from **FM to FG**, as Avg-MP-Diff and Avg-TestAcc-Diff exceed the thresholds, further aligning with our SHAP-Score analysis where in this region, the model begins to rely on z_u during testing. The results correspond to Medical Abstracts-RoBERTa setting.

F.7 FOR EMOTIONS-BERT: (DURING TRAINING WE HAVE 572 SAMPLES OF CLASS 4 IN WHICH WE INTRODUCE z_u IN VARYING RATIOS (%))

Table 7: Thresholds (σ_1 and σ_2) analysis for Emotions-BERT Setup

Ratio (%) of x_u Train Samples	**Avg $MP_{2,4}$ (in %)**	**Avg-MP-Diff (in %)**	**Avg Test Accuracy (in %)**	**Avg-TestAcc -Diff (in %)**
0.0	0.0	0	92.45	0
0.17	N/A	N/A	N/A	N/A
0.35	N/A	N/A	N/A	N/A
0.7	N/A	N/A	N/A	N/A
1.05	N/A	N/A	N/A	N/A
1.4	N/A	N/A	N/A	N/A
1.75	N/A	N/A	N/A	N/A
2.1	0	0	90.25	2.2
2.45	0	0	91.19	1.26
2.8	0	0	92.45	0.0
3.5	0	0	93.08	-0.63
5.24	0	0	92.57	-1.89
6.99	6.27	6.27	89.94	2.52
7.87	3.7	3.7	89.94	2.52
8.74	17	17	84.7	7.76
10.49	18.27	18.27	89.31	3.14
11.36	15.82	15.82	89.1	3.35
12.24	23.04	23.04	89.1	3.35
13.99	28.23	28.23	82.39	10.06
14.86	31.08	31.08	88.68	3.77
17.48	35.72	35.72	86.79	5.66
26.22	70.56	70.56	68.13	24.32
34.97	76.22	76.22	44.65	47.8
52.45	98.93	98.93	21.38	71.07
69.93	97.87	97.87	4.82	87.63
87.41	100	100	1.68	90.78

For **Emotions-BERT** as well, the thresholds, σ_1 **= 3.5% and** σ_2 **= 3%**, remain robust and perfectly applicable to distinguish **feature memorization and feature generalization**, as observed in Table 7. **FM** occurs with the x_u train samples ratio lying between **2.1% and 5.24%**, with both Avg-MP-Diff and Avg-TestAcc-Diff remaining below the thresholds, also aligning with the SHAP-Score analysis (Fig. 16c), where in this range, the model does not associate the label with feature z_u during testing. Beyond **5.24%**, the model transitions from **FM to FG**, as Avg-MP-Diff and Avg-TestAcc-Diff exceed the thresholds, further aligning with our SHAP-Score analysis where in this region, the model begins to rely on z_u during testing. The results correspond to Emotions-BERT setting.

F.8 FOR EMOTIONS-ROBERTA: (DURING TRAINING WE HAVE 572 SAMPLES OF CLASS 4 IN WHICH WE INTRODUCE z_u IN VARYING RATIOS (%))

Table 8: Thresholds (σ_1 and σ_2) analysis for Emotions-RoBERTa Setup

Ratio (%) of x_u Train Samples	**Avg MP$_{3,4}$ (in %)**	**Avg-MP-Diff (in %)**	**Avg Test Accuracy (in %)**	**Avg-TestAcc -Diff (in %)**
0.0	46.64	0	88.24	0
0.17	N/A	N/A	N/A	N/A
0.35	N/A	N/A	N/A	N/A
0.7	N/A	N/A	N/A	N/A
1.05	N/A	N/A	N/A	N/A
1.4	N/A	N/A	N/A	N/A
1.75	N/A	N/A	N/A	N/A
2.1	N/A	N/A	N/A	N/A
2.45	N/A	N/A	N/A	N/A
2.8	N/A	N/A	N/A	N/A
3.5	47.69	1.05	88	0.24
3.85	38.8	-7.84	91	-2.67
4.2	45.57	-1.07	85.64	2.91
4.55	43.06	-3.59	86.91	1.7
5.24	44.41	-2.24	85.64	1.33
6.12	47.89	1.25	86.91	2.61
6.99	48.69	2.04	86.91	1.33
10.49	53.95	7.31	84.18	4.06
13.99	57.46	10.82	84.85	3.39
17.48	69.95	23.31	82.42	5.82
26.22	73.19	26.55	75.64	12.61
34.97	79.07	32.42	73.45	14.79
52.45	91.51	44.86	55.27	32.97
69.93	95.61	48.96	44.85	43.39
87.41	98.59	51.95	22.06	66.18

For **Emotions-RoBERTa** as well, the thresholds, σ_1 **= 3.5% and** σ_2 **= 3%**, remain robust and perfectly applicable to distinguish **feature memorization and feature generalization**, as observed in Table 8. **FM** occurs with the x_u train samples ratio lying between **3.5% and 6.99%**, with both Avg-MP-Diff and Avg-TestAcc-Diff remaining below the thresholds, also aligning with the SHAP-Score analysis (Fig. 17c), where in this range, the model does not associate the label with feature z_u during testing. Beyond **6.99%**, the model transitions from **FM to FG**, as Avg-MP-Diff and Avg-TestAcc-Diff exceed the thresholds, further aligning with our SHAP-Score analysis where in this region, the model begins to rely on z_u during testing. The results correspond to Emotions-RoBERTa setting.