Generative or Discriminative ?
Revisiting Text Classification in the Era of Transformers

Anonymous ACL submission

Abstract

The comparison between discriminative and generative classifiers has intrigued researchers since Efron (1975)'s seminal analysis of logistic regression versus discriminant analysis. While early theoretical work established that generative classifiers exhibit lower sample complexity but higher asymptotic error in simple linear settings, these trade-offs remain unexplored in the transformer era. We present the first comprehensive evaluation of modern generative and discriminative architectures—Auto-regressive, Masked Language Modeling, Discrete Diffusion, and Encoders for text classification. Our study reveals that the classical "two regimes" phenomenon manifests distinctly across different architectures and training paradigms. Beyond accuracy, we analyze sample efficiency, calibration, noise robustness, and ordinality across diverse scenarios. Our findings offer practical guidance for selecting the most suitable modeling approach based on real-world constraints such as latency and data limitations. [1]

1 Introduction

Text Classification (TC), a fundamental task in Natural Language Processing (NLP), encompasses various applications such as Sentiment Analysis, Topic Classification, and Emotion Detection. Since the emergence of transformer architectures, the field has been dominated by discriminative classifiers that leverage token embeddings (e.g., the `[CLS]` token in BERT (Devlin et al., 2019)). These models directly learn the conditional probability distribution $P_\theta(y|X)$, where X denotes the input text and y represents the ground truth label. However, as these discriminative models grow larger, they require increasingly large amounts of labeled data to achieve optimal performance, making them impractical in many real-world scenarios where labeled data is scarce or expensive to obtain (Zheng et al., 2023). On the other hand, generative classifiers, which model the joint distribution $P_\theta(X, y)$, are known to work better in low-data settings, giving rise to the classical 'two-regimes' phenomenon for classification (Ng and Jordan, 2001; Yogatama et al., 2017; Zheng et al., 2023). This advantage stems from their ability to learn underlying data distributions rather than just decision boundaries, allowing them to make better use of limited training examples. The inherent data efficiency of generative approaches, combined with recent advances in generative modeling such as Discrete Diffusion (Lou et al., 2024), motivates us to revisit the classical discriminative versus generative debate in the context of TC with Transformer-based architectures.

Prior research on generative classifiers has largely focused on non-textual tabular data, utilizing linear models such as Linear Discriminant Analysis (Efron, 1975) and Naive Bayes (Ng and Jordan, 2001). While Yogatama et al. (2017) extended this analysis to neural architectures using RNNs and LSTMs (Hochreiter and Schmidhuber, 1997) for the TC task and found similar conclusions about generative advantages in low-data regimes, their study predated the transformer era. Modern NLP has seen the emergence of various successful transformer-based generative modeling paradigms, including auto-regressive (AR) models like GPT (Radford et al., 2018) that maximize likelihood directly, Discrete Diffusion models (Lou et al., 2024) that learn through iterative denoising, and masked language models (MLM) (Devlin et al., 2019) that optimize pseudo-likelihood (Wang and Cho, 2019). These approaches offer different trade-offs in terms of computational efficiency, sample complexity, and modeling flexibility. However, a systematic comparison of these paradigms for text classification remains unexplored, particularly in the context of varying model sizes and real-world deployment

[1] Anonymized code available at: `https://anonymous.4open.science/r/gen_v_disc-1311/`

considerations.Our work fills this gap by providing a practitioner-oriented study that evaluates these approaches not just on classification accuracy, but also on crucial deployment metrics including different model scales, robustness to input perturbations, reliability of output probabilities through calibration analysis, and preservation of ordinal relationships between classes. This comprehensive evaluation aims to provide concrete guidance for choosing between different generative and discriminative approaches based on specific deployment constraints and requirements. We strategically focus on widely available public benchmark datasets for reproducibility purposes. Following Li et al. (2025) and Yogatama et al. (2017), our study evaluates all models trained from scratch, rather than relying on pre-trained weights, providing crucial insights for practitioners working with domain-specific data (Huang et al., 2019) or in resource-constrained environments (Martin et al., 2022). This approach helps isolate the confounding effects of the pre-training corpus (Razeghi et al., 2022) from other factors such as the modeling approach and size, which we evaluate. The combined effects can be explored in future work.

Our **main contributions** include the following: **(a)** We present the first large-scale comparative study of two major classification approaches - Discriminative (Encoder) and Generative (Text Diffusion, AR, MLM) on 9 popular classification benchmark datasets, which is a first of its kind in the transformer era. Our study reveals a more nuanced interplay between model size and sample complexity than the previously known "two regimes" phenomenon. **(b)** We provide comprehensive analyses across multiple dimensions including model scaling behavior, sample efficiency, and performance in low-resource settings with models trained from scratch. We also introduce novel evaluation perspectives by examining ordinal relationships between classes, output calibration and robustness to input noise, offering insights beyond traditional classification metrics. **(c)** Finally, we provide practical recommendations in Section 6 on selecting the appropriate model for deployment in various real-world scenarios.

2 Related Work

Generative and Discriminative Models for Classification. The comparison between generative and discriminative classifiers originated with Efron (1975)'s analysis of logistic regression and discriminant analysis. Building on this foundation, Ng and Jordan (2001) examined naive Bayes and logistic regression, establishing the fundamental trade-off between generative models' faster learning rate and discriminative models' lower asymptotic error. Their theoretical analysis heavily depends on linearity and independence assumptions. However, subsequent work by Xue and Titterington (2008) challenged these findings through empirical studies and asymptotic analysis of statistical efficiency. Yogatama et al. (2017) provided the first empirical study of discriminative vs generative models for TC with neural architectures using LSTMs. They maximize the joint probability $P(X, y) = P(X|y)P(y)$ by concatenating the label y text at the beginning of the input text X and maximizing the class conditional likelihood i.e. $P(X \mid y) = \prod_{t=1}^{T} p\left(x_t \mid \boldsymbol{x}_{<t}, y\right)$. The final predicted label is obtained by $\hat{y} = \texttt{argmax}_y\, P(X|y)P(y)$. They found that generative LSTMs have better accuracy than their discriminative counterparts at low-sample regimes. Further, they noted that the neural generative LSTMs are generally better than baseline generative models with stronger independence assumptions (e.g. naive Bayes, Kneser–Ney Bayes (Ney et al., 1994; Teh, 2006)). Next, the work by Zheng et al. (2023) has extended the theoretical understanding of generative classifiers to multi-class and non-linear settings. More recent studies (Li et al., 2025; Stanley et al., 2025) have found that generative classifiers tend to avoid shortcut learning and exhibit greater robustness to distribution shifts.

While prior studies provide valuable insights, the landscape of NLP has evolved dramatically with the advent of novel transformer-based generative paradigms such as Auto-Regressive (AR) models (Radford et al., 2018) and Discrete Diffusion models (Lou et al., 2024). Our work extends beyond these previous comparisons by conducting the first comprehensive evaluation of modern transformer-based generative and discriminative classifiers for TC. While previous works primarily focused on classification accuracy and sample complexity, we examine multiple dimensions that are crucial for real-world deployments. For instance, Yogatama et al. (2017) initial work with neural architectures was limited to a fixed model size, leaving open questions about how the generative-discriminative trade-off varies with model capacity and computational budget—questions that have become increas-

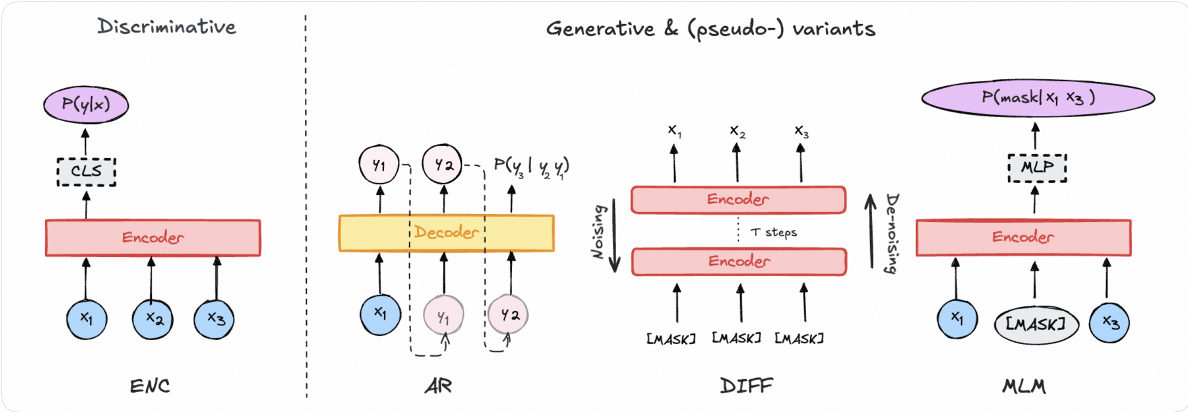

Figure 1: **[Best viewed in color]** Illustration of different modeling paradigms (ENC: Encoder-based classification, MLM: Masked Language Modeling, AR: Auto-Regressive Model, DIFF: Discrete Text Diffusion).

ingly relevant in the era of large language models. Similarly, though Zheng et al. (2023) provided theoretical insights for multi-class settings, their analysis did not address practical considerations like calibration quality or preservation of ordinal relationships between classes.

Pseudo-Generative Models. Recent work (Sahoo et al., 2024) highlights a natural connection between Discrete Text Diffusion (Lou et al., 2024) and the Masked Language Modeling (MLM) objective in BERT (Devlin et al., 2019), showing that the diffusion objective can be expressed as a weighted sum of MLM losses. Using transformer encoder models, this approach achieves likelihood bounds comparable to or better than those in Lou et al. (2024). Motivated by this, we include vanilla MLM as a baseline for text classification. While MLM has typically served as a pretraining objective followed by fine-tuning (Liu et al., 2019), there has been little systematic study of its direct use for classification. Although MLM does not explicitly model $P(X|y)$, it estimates $P(x_m|x_{\backslash m})$, where x_m is a masked token and $x_{\backslash m}$ represents all other tokens. This approximates the pseudo-likelihood of $P(X, y)$ when modeled over the corpus (Wang and Cho, 2019). We therefore classify MLM as a pseudo-generative model.

Also, traditional generative classifiers aim to model $P(X|y)$ by prepending the label token. However, recent work (Li et al., 2025) shows that appending the label at the end—though not strictly modeling $P(X|y)$—can yield better in-distribution performance. This setup also enables efficient inference, requiring only a single forward pass to predict the label, unlike traditional generative models that need $\#_{\text{label}}$ forward passes. These benefits motivate the inclusion of such pseudo-generative models in our benchmarks. Notably, these approaches involve minimal changes to standard transformer architectures—typically just altering label placement or the loss function—while preserving the core model design. This allows for fair comparisons using widely available implementations accessible to practitioners.

We also acknowledge a separate class of hybrid generative-discriminative models, where some subset of parameters are trained generatively and others discriminatively (Raina et al., 2003; McCallum et al., 2006; Hayashi, 2025). However, we exclude them from our study, as their architectural differences hinder fair comparison with fully generative or discriminative models, placing them outside the scope of this work.

Relation to Multi-task Learning. Learning $\log P(X, y)$ jointly, when factored as $\log P(X) + \log P(y|X)$ (or $\log P(y) + \log P(X|y)$) can be viewed as a multi-task learning setup, where unsupervised learning of $\log P(X)$ ($\log P(Y)$) and supervised learning of $\log P(Y|X)$ ($\log P(X|Y)$) represent two different but related tasks. This connection is supported by empirical results showing that unsupervised pre-training helps downstream supervised tasks (Erhan et al., 2010). As demonstrated by Wu et al. (2020); Hu et al. (2023), when model capacity is sufficiently large, such multi-task learning setups tend to be more successful - the model has enough capacity to perform well on both the unsupervised and supervised objectives. However, with limited model capacity, there are inherent trade-offs between the tasks, leading to challenges in jointly optimizing for both $P(X)$ and $P(y|X)$ (or $P(y)$ and $P(X|y)$). This insight motivates us to conduct a systematic study examining the relationship between model capacity and the performance of discriminative vs generative classifiers - an analysis that has not been previously undertaken in the literature.

Refer to Appendix A for further related works review on Discrete Diffusion, Robustness to Noise, Ordinality & Calibration.

3 Methodology

We approach the problem of label classification by leveraging two popular language modeling paradigms: **(a) Generative** - Discrete Diffusion models, Auto-regressive models (AR), and Masked Language Models (MLM) & **(b) Discriminative** - Encoder-based transformer models. Note that, for brevity, we use the term *"generative"* from this point onward to also include the **pseudo-generative** baselines. Let $\mathcal{D} = \{(X_i, y_i)\}_{i=1}^N$ denote the dataset where X_i is the input text and $y_i \in \mathcal{Y}$ is the corresponding label from a finite set of classes $\mathcal{Y}$. Generative models tend to learn the joint data distribution $P(X, y)$ first and then try to infer the label using the marginals, whereas Discriminative models directly learn the conditional distribution $P(y|X)$. Note that each $X_i = x_i^1 \dots x_i^n$, where x_i^j is a token from the associated vocabulary $\mathcal{V}$.

3.1 Discriminative Model for Classification

(1) Encoder-based classification (`ENC`): A Transformer encoder (Vaswani et al., 2017) f_θ encodes the input as $h_i = f_\theta(X_i)$ as a d-dimensional embedding, followed by a linear classifier head $W \in \mathbb{R}^{|\mathcal{Y}| \times d}$ which is the standard discriminative learning setup:

$$\hat{y}_i = \texttt{softmax}(Wh_i), \quad \mathcal{L}_{\text{enc}} = -\sum_{i=1}^{N} \log P(y_i|X_i) \tag{1}$$

where $\mathcal{L}_{\text{enc}}$ is the cross-entropy based objective for training the encoder model.

3.2 Generative Models for Classification

(2) Masked Language Modeling (`MLM`): During training, we mask 15% of tokens in input sequence $X_i = x_i^1 \dots x_i^n$ following Devlin et al. (2019) and predict them using unmasked bi-directional context. Wang and Cho (2019) show that the MLM objective stochastically captures the *pseudo-loglikelihood* which makes it similar to a denoising autoencoder (Vincent et al., 2010). Hence, we consider MLM under the generative family of models. Formally, the objective is:

$$\mathcal{L}_{\text{mlm}} = -\sum_{i=1}^{N} \sum_{j \in \mathcal{M}_i} \log P(x_i^j | X_i^{\setminus j}) \tag{2}$$

where $\mathcal{M}_i$ is the set of masked positions and $X_i^{\setminus j}$ denotes the unmasked input with only token at position j masked. At inference, we use the template:

$X_i' =$ `[CLS]` X_i `[SEP] "The label is" [MASK]` .

and predict the masked label token. The output vocabulary is restricted to the label token set $\mathcal{V}_\mathcal{Y}$.

(3) Auto-regressive modeling (`AR`): Following Radford et al. (2018), we train a causal generative model to minimize the next-token prediction loss over the entire label + input sequence:

$$\mathcal{L}_{\text{gpt}} = -\sum_{i=1}^{N} \sum_{j=1}^{L_i} \log P(x_i^j | y, x_i^1, \dots, x_i^{j-1}) \tag{3}$$

where L_i is the length of the i-th sequence. At inference time, we perform one forward pass per candidate label $y \in \mathcal{V}_\mathcal{Y}$ by prepending it to the input X, and compute the log-likelihood. The predicted label is then obtained as $\arg\max_{y \in \mathcal{V}_\mathcal{Y}} \log P(X \mid y)$. In $\texttt{AR}_{pseudo}$ (refer pseudo-generative models in Section 2) the label is appended at the end instead of the beginning and only one forward pass is required to generate the predicted label token y. Note that label placement is only relevant for causal generative architectures (like `AR`) with a left-to-right attention structure. For bidirectional (pseudo-)generative models like `MLM` or `DIFF`, it has no theoretical impact.

(4) Text Diffusion (`DIFF`): For each input-label pair (X_i, y_i), we first create a template:

$X_i = X_i$ `[SEP] "The label is"` y_i .

where each template is a sequence $X_i = x_i^1 \dots x_i^{L_i}$ with tokens $x_i^j \in \mathcal{V}$.

Similar to how diffusion models gradually add noise to images, our forward process gradually corrupts text by converting tokens to pure noise (here `[MASK]`). Following Lou et al. (2024), we define the forward process through discrete transition matrices Q_t following a continuous markov process (see eq. 4). This process occurs at different timesteps $t \in [0, T]$, where each token position is independently corrupted, starting from the original text and progressively moving towards a completely masked sequence.

$$\frac{dp_t}{dt} = Q_t p_t, \quad \text{with} \quad p_0 = p_{\text{data}} \tag{4}$$

The reverse process learns to reconstruct the original text by predicting what token should replace each [MASK] symbol. This is done by learning score ratios $s_\theta(x,t)_z = \frac{p_t(z)}{p_t(x)}$ where x, z are tokens from $\mathcal{V}$ and modeling the reverse process (Sun et al., 2022) as:

$$\frac{dp_{T-t}}{dt} = s_\theta(x,t)_z Q_{T-t} p_{T-t} \tag{5}$$

Denoising Score Entropy (DSE) is used for training the score model in a manner that ensures several desired properties for s_θ and ensures the computation is tractable:

$$\mathcal{L}_{\text{DSE}} = \mathop{\mathbb{E}}_{\substack{x_0 \sim p_0, \\ x \sim p(\cdot|x_0)}} \Bigg[\sum_{z \neq x} w_{xz} \Big(s_\theta(x)_z - \frac{p(z \mid x_0)}{p(x \mid x_0)} \log s_\theta(x)_z \Big) \Bigg] \tag{6}$$

where p is assumed to be perturbation of some base density p_0 and weights $w_{xz} > 0$.

The ELBO (Theorem 3.6 in Lou et al. (2024)) provides an upper bound on the negative log-likelihood, which is what we optimize for in generative models:

$$-\log p_0^\theta(x_0) \leq \mathcal{L}_{DWDSE}(x_0) + constant \tag{7}$$

where $\mathcal{L}_{DWDSE}$ integrates $\mathcal{L}_{DSE}$ weighted by the forward diffusion matrix. At inference time, we mask the label token in the template X_i and use the model to predict it, restricting the possible outputs to valid labels in $\mathcal{V}_\mathcal{Y}$. For further details, refer to Lou et al. (2024).

4 Experiments

Our experiments are designed towards addressing the following research questions:

Q1. How do different modeling approaches compare against each other when trained from scratch?

Q2. How much does noise perturbation via random token substitution and token dropping affect the performance of different modeling approaches ?

Q3. How well are the different modeling approaches calibrated ? For ordinal classification, how well the predicted distributions over ordinal categories follow a unimodal shape ?

4.1 Datasets

We evaluate our models on 9 text classification benchmark datasets to ensure a comprehensive assessment across multiple domains, text lengths, and classification types - sentiment analysis, movie reviews, news categorization, and social media analysis. These are: **AG News** (Zhang et al., 2015), **Emotion** (Saravia et al., 2018), **Stanford Sentiment Treebank (SST2 & SST5)** (Socher et al., 2013), **Multiclass Sentiment Analysis**, **Twitter Financial News Sentiment**, **IMDb** (Maas et al., 2011), and **Hate Speech Offensive** (Davidson et al., 2017). These datasets encompass varying levels of complexity, ranging from binary text classification to fine-grained multi-class categorization, with textual inputs spanning from concise single sentences to extensive paragraph-level passages. Further details are postponed to Appendix C.

4.2 Experimental Setup

We conduct an extensive benchmarking study comparing the five different modeling approaches for text classification summarised in Section 3: AR, AR_{pseudo} , MLM, DIFF, and ENC. These models are evaluated on 9 popular classification benchmark datasets as mentioned in Section 4.1.

We experiment with multiple dataset sample sizes $\in$ {128, 256, 512, 1024, 2048, 4096, *full_data*}. To assess the effect of model sizes, we test 3 model size configurations using the base Transformer architecture: *small* (1 layer, 1 head), *medium* (6 layers, 6 heads) and *large* (12 layers, 12 heads). Performance is measured using the weighted-F1 score. All experiments are repeated with 3 random seeds, running a total of $9 \times 7 \times 3 \times 3 \times 5 = 2835$ experiments and we report the average and shaded standard deviations in Figure 2, 3. These experiments are designed to address **Q1**

In the second part of our evaluation, we assess each model's robustness to input perturbations. In real-world scenarios, particularly in e-commerce platforms, often encounter various text corruptions (like OCR errors in product documentation, truncated reviews, or incomplete user queries), we focus on two systematic types of synthetic noise to evaluate model robustness: (a) **Random Token Drop** — where X% of tokens are randomly removed from the input sentence, and (b) **Random Token Substitution** — where X% of tokens are replaced with random tokens from the vocabulary (excluding special tokens like [PAD], [MASK]). We conduct these experiments to explore **Q2** about model robustness to input perturbations.

Lastly, we assess model performance on calibration and ordinal metrics—aspects often overlooked but critical for real-world deployment. While a similar study exists for pre-trained models (Kasa et al., 2024), ours focuses on models trained from scratch. For ordinal classification tasks, we verify ordinal alignment, ensuring that the predicted probability distribution reflects the natural ordering of categories (e.g., the probability of "good" should be closer to "neutral" than to "bad").

For ordinal evaluation, we report *MSE* (Mean Squared Error), *MAE* (Mean Absolute Error), and *Unimodality* (UM). For calibration, we measure *ECE* (Expected Calibration Error) and *MCE* (Maximum Calibration Error). UM verifies that the predicted probability distribution has a single peak, thus preserving class ordering—for instance, preventing models from assigning high confidence to both extremely positive and negative sentiments simultaneously. Calibration metrics quantify the discrepancy between predicted probabilities and empirical frequencies. For detailed descriptions, see Kasa et al. (2024) and Wang (2023). These experiments address **Q3**. Refer to Appendix B for details on hyperparameters and training setup.

5 Results

We analyze the results from all the experiments and provide valuable insights & recommendations for model selection.

Q1: For **1-layer, 1-head** models (Figure 2), all approaches show near-random performance in low-data regimes. However, as training data increases, only ENC (orange line) continues to improve, ultimately outperforming others in high-data settings. This suggests that **for small models - often necessary due to real-world latency constraints - ENC is the most effective approach.** The classical 'two regimes' phenomenon does not manifest when the model size is small.

The pattern shifts dramatically for larger architectures.Under the **12-layer, 12-head** configuration, both generative models—AR and DIFF—outperform ENC in low-data settings, with this advantage diminishing as data increases. This aligns with previous findings (Ng and Jordan, 2001; Yogatama et al., 2017; Rezaee et al., 2021) about generative models' advantages in data-limited scenarios. Surprisingly, for large models, the pseudo-generative MLM (*blue* line) consistently outperforms all methods across our 9 benchmark datasets in high-data settings, **challenging the conventional wisdom about discriminative dominance in high-sample regime.** This aligns with Erhan et al. (2010)'s finding that pseudo-generative models implicitly perform unsupervised pre-training alongside supervised learning, creating an effective multi-task setup (Section 2). Their work shows that this unsupervised phase acts as a data-dependent regularizer, guiding optimization toward better-generalizing minima. For *large* models, direct fine-tuning without this implicit pre-training often leads to suboptimal convergence, explaining ENC's underperformance relative to MLM. Thus, **for scenarios without model size constraints, generative models emerge as the optimal choice for low-data settings** such as for low-resource languages and continual learning applications requiring frequent updates with limited samples, while **pseudo-generative MLM is superior when abundant labeled data is available.**

Another noteworthy observation is that under the **6-layer, 6-head** configuration, in low-data settings, DIFF emerges as the best performing model across all datasets, clearly outperform even it's generative counterpart AR. As the training data size increases, we see that the discriminative ENC outperforming DIFF. Thus, **in medium scale architectures, between the generative DIFF and the discriminative ENC, the classical 'two regimes' still holds.**

Figure 3 shows that $\texttt{AR}_{pseudo}$ generally underperforms AR and also displays **higher variance** in low-data settings—the recommended use case—while the opposite holds in high-data scenarios. This reveals a new insight beyond Li et al. (2025), who only evaluated full-data settings where $\texttt{AR}_{pseudo}$ performed better in-distribution. As noted in Section 3, AR requires $|label|$-times forward passes per prediction, unlike the single pass needed for $\texttt{AR}_{pseudo}$; however, this can be mitigated via batching or parallel processing, reducing inference time differences at the cost of higher computation.

Q2: We evaluate the robustness of all approaches under both 6-layer and 12-layer configurations across two noise schedules in full-data settings. We exclude 1-layer models from this analysis since their performance is mostly trivial (except for ENC), making robustness comparisons uninformative. In Tables 1 and 2, we report the minimum noise level at which each model's performance drops by 10% Δ (similar trend for 5% & 15%) relative to its

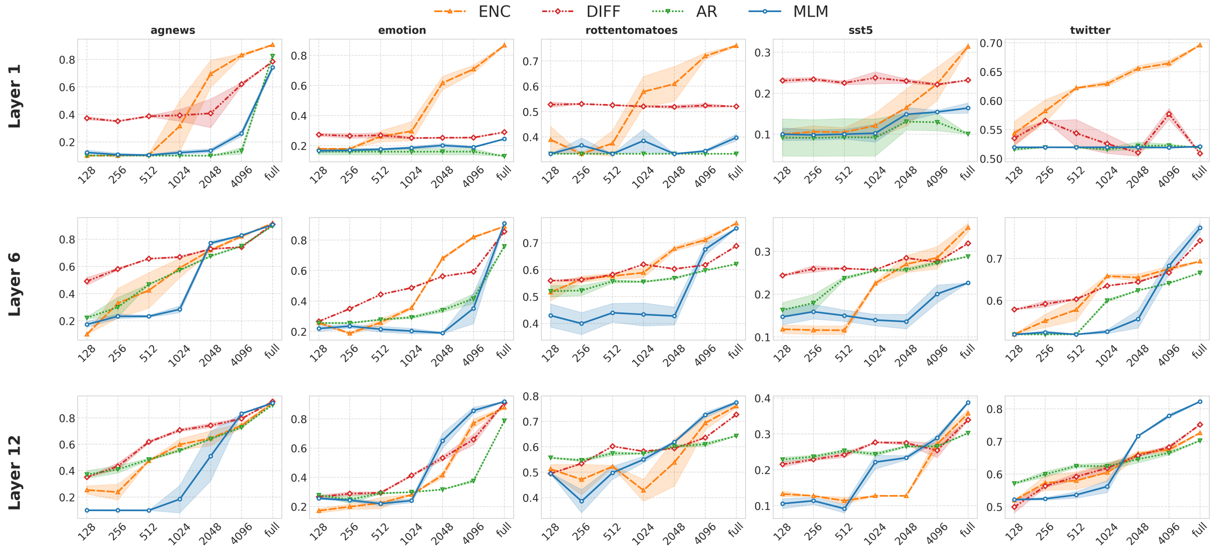

Figure 2: **[Best viewed in color]** Comparison of weighted-F1 scores of models across different configurations (↑ is better). For rest of the datasets, refer to Figure 8 in Appendix E. (`X-axis`: sample size, `Y-axis`: weighted-F1 score)

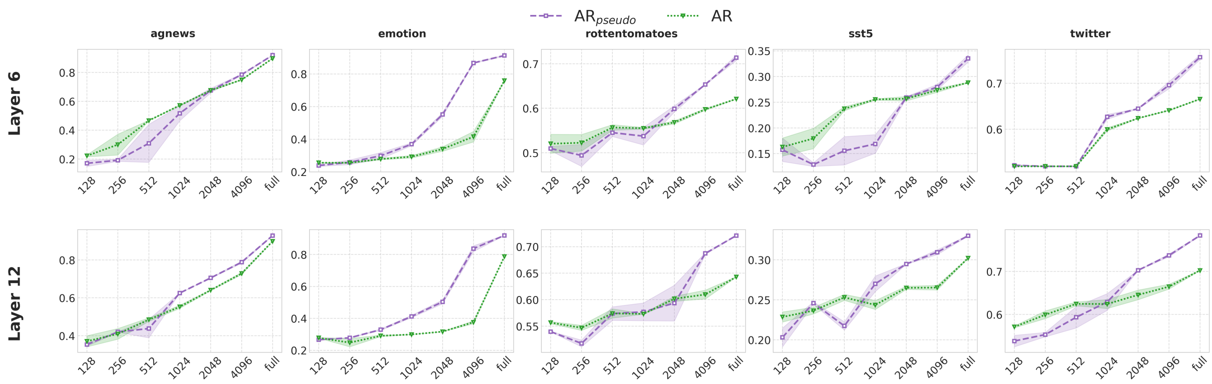

Figure 3: **[Best viewed in color]** Comparison of weighted-F1 scores between $\texttt{AR}_{pseudo}$ and AR (↑ is better). 1-layer results are omitted here as they are mostly trivial in low-data settings. Results for remaining datasets are provided in Figure 9, Appendix E. (`X-axis`: sample size, `Y-axis`: weighted-F1 score)

peak, averaged across all datasets, as a measure of **robustness boundary**. Our analysis reveals that all models exhibit lower robustness to substitution noise compared to dropping. This can be explained by the inequality: $P(\text{garbage}_t|X_{1...t-1}) < P(x_{t+1}|X_{1...t-1})$—the model is more likely to assign lower probability to a corrupted token than to a skip token at $t+1$-th position (assuming t-th token was dropped), which may still be contextually relevant given $X_{1...t-1}$.

The generative DIFF demonstrates superior robustness to both token dropping and substitution (except in 6-layers where ENC is slightly better), likely because its training paradigm involves recovering true tokens from noise/masked inputs. The discriminative ENC maintains consistent robustness under both noise types, while generative AR shows the high sensitivity to noise. Combining these findings with **Q1**'s results reveals that generative AR models face dual challenges compared to ENC in full data settings: they underperform in terms of both weighted-f1 and robustness. This contrasts with Li et al. (2025)'s findings that discriminative ENC models rely on shortcuts and show less robustness compared to generative AR. However, their analysis focuses on shortcut learning and distribution shifts rather than input perturbation noise across varying model sizes. Notably, while the pseudo-generative MLM and $\texttt{AR}_{pseudo}$ demonstrates superior performance in larger models at full data settings, they exhibit lower robustness compared to similarly performing ENC models. Moreover the relative drop in robustness in moving from dropping to substitution noise is more severe in $\texttt{AR}_{pseudo}$ compared to AR. This is likely because $\texttt{AR}_{pseudo}$ conditions on corrupted inputs, so it's directly af-

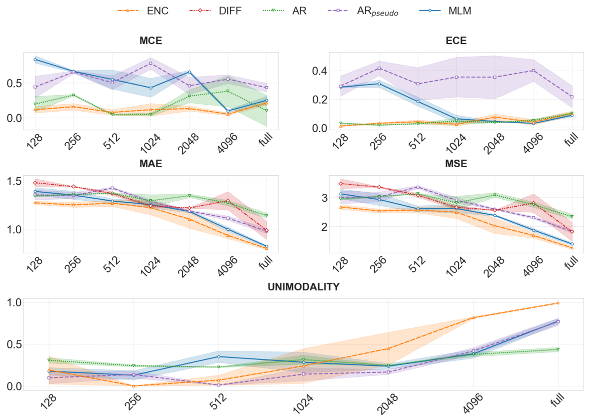

Figure 4: **[Best viewed in color]** Calibration and Ordinal performance of 12-layers model on SST-5. For ECE, MCE, MAE, MSE (↓ is better) and UM (↑ is better) (`X-axis`: sample size).

Config	ENC	AR_{pseudo}	AR	MLM	DIFF
(12L,12H)	51.1%	46.7%	37.8%	34.4%	54.4%
(6L,6H)	47.8%	51.1%	46.7%	46.7%	53.3%

Table 1: Minimum noise% for 10% Δ weighted-F1 drop under Random Token **Dropping**. (↑ is better)

Config	ENC	AR_{pseudo}	AR	MLM	DIFF
(12L,12H)	32.2%	22.2%	28.9%	31.1%	35.6%
(6L,6H)	37.8%	21.1%	30.0%	32.2%	34.4%

Table 2: Minimum noise% for 10% Δ weighted-F1 drop under Random Token **Substitution**. (↑ is better)

fected by garbage tokens polluting the predictive context. However, for `AR` clean label conditions the model, and the noisy input is scored globally — giving the model more flexibility to discount garbage.

Q3: Figure 4 presents ordinal and calibration results for SST-5, selected for its balanced distribution, inherent class ranking (e.g., very positive to very negative), and highest number of classes. Results for other datasets are in Appendix D. `DIFF` does not support calibration metrics like ECE, MCE, and UM, as its masking/absorbing noise process produces only binary outputs rather than soft probabilities. While a uniform noise schedule can yield probabilities over $\mathcal{V}$, it performed slightly worse, so we used the absorbing schedule in our study.

From the ECE and MCE plots, we observe that `ENC` outputs remain well-calibrated across all sample sizes, while `MLM` reaches similar calibration only in high-data regimes. We also see that `MLM` and `ENC` achieve UM in over 80% of the samples, aligning with findings from Kasa et al. (2024). Their MAE and MSE values are also low, indicating strong ordinality in high-data settings. This completes the picture for *large* models under high-data, where `MLM` not only outperforms others in weighted-F1 but is also well-calibrated and ordinal, making it a strong candidate for real-world deployment. However, under low-data conditions, 12-layers `AR` outperforms $\texttt{AR}_{pseudo}$ in 7 out of 9 datasets on calibration metrics. It also surpasses `DIFF` in ordinal performance, thus making it the more reliable choice among generative models in low-data scenario. Also, even though generative approaches like `DIFF` were recommended earlier in **Q1** based on weighted-F1 for 6-layers case (in Figure 2) deploying them in production could be risky when calibrated or ordinal probabilities are required, especially for imbalanced datasets like *twitter* and *hatespeech* (see Appendix D). These metrics are particularly important when downstream models consume output probability scores as features which is often the case in multi-stage ranking systems.

Lastly, Figure 5 in Appendix D reveals an interesting trend: as *model size* increases, calibration metrics either remain flat or worsen. This suggests that larger models or improved classification accuracy do not necessarily lead to better calibration, aligning with the findings of Guo et al. (2017) where they show similar behaviour using ResNets (He et al., 2016). However, for ordinal metrics, we observe substantial improvements when moving from 1-layer to 6-layer models, with performance plateauing at 12 layers. A similar trend was reported in Kasa et al. (2024) for pre-trained models.

6 Conclusion

Our study offers practical modeling recommendations across deployment scenarios. For latency-sensitive applications, `ENC` is ideal—especially in the 1-layer setting—due to its efficiency, robustness to noise, and well-calibrated, ordinal outputs. For offline settings with sufficient data, the 12-layer `MLM` performs best across F1, calibration, and ordinal metrics, though caution is needed with noisy inputs due to its lower robustness to token dropping. In low-resource scenarios, both `AR` and `DIFF` are strong options, with `DIFF` favored for its noise resilience and performance at 6-layers. However, if calibrated probability outputs are essential, such as in ranking pipelines, `AR` is the preferred choice.

7 Limitations

While we conducted a thorough examination of generative and discriminative classifiers under standard i.i.d. assumptions, our findings may not generalize to scenarios involving distribution shifts, such as co-variate shift (Bickel et al., 2009) or concept shift (Roychowdhury et al., 2024). Our analysis was limited to traditional fine-tuning approaches, excluding emerging paradigms such as few-shot prompt-based in-context learning (Sun et al., 2023; Gupta et al., 2023) and parameter-efficient techniques like LoRA (Hu et al., 2022), which may uncover newer insights. Furthermore, our study focused exclusively on pure text classification, leaving the exploration of multi-modal scenarios involving tabular data (Pattisapu et al., 2025), images (Lu et al., 2019), audio (Kushwaha and Fuentes, 2023), and other modalities for future work. An additional promising direction would be to investigate the use of pre-trained models, disentangling the effects of pre-training and fine-tuning (see Section 1), and to assess their effectiveness in cross-lingual transfer tasks. We leave these avenues for future exploration.

References

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. 2021. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993.

Steffen Bickel, Michael Brückner, and Tobias Scheffer. 2009. Discriminative learning under covariate shift. *Journal of Machine Learning Research*, 10(9).

Jaroslaw Blasiok, Parikshit Gopalan, Lunjia Hu, and Preetum Nakkiran. 2023. When does optimizing a proper loss yield calibration? *Advances in Neural Information Processing Systems*, 36:72071–72095.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ICWSM '17, pages 512–515.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bradley Efron. 1975. The efficiency of logistic regression compared to normal discriminant analysis. *Journal of the American Statistical Association*, 70:892–898.

Dumitru Erhan, Aaron Courville, Yoshua Bengio, and Pascal Vincent. 2010. Why does unsupervised pre-training help deep learning? In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 201–208. JMLR Workshop and Conference Proceedings.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Karan Gupta, Sumegh Roychowdhury, Siva Rajesh Kasa, Santhosh Kumar Kasa, Anish Bhanushali, Nikhil Pattisapu, and Prasanna Srinivasa Murthy. 2023. How robust are llms to in-context majority label bias? *arXiv preprint arXiv:2312.16549*.

Hideaki Hayashi. 2025. A hybrid of generative and discriminative models based on the gaussian-coupled softmax layer. *IEEE Transactions on Neural Networks and Learning Systems*, 36(2):2894–2904.

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

Zhengfu He, Tianxiang Sun, Qiong Tang, Kuanning Wang, Xuanjing Huang, and Xipeng Qiu. 2023. DiffusionBERT: Improving generative masked language models with diffusion models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4521–4534, Toronto, Canada. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Yuzheng Hu, Ruicheng Xian, Qilong Wu, Qiuling Fan, Lang Yin, and Han Zhao. 2023. Revisiting scalarization in multi-task learning: A theoretical perspective. *Advances in Neural Information Processing Systems*, 36:48510–48533.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*.

Priyank Jaini, Kevin Clark, and Robert Geirhos. 2024. Intriguing properties of generative classifiers. In *The Twelfth International Conference on Learning Representations*.

Siva Rajesh Kasa, Aniket Goel, Karan Gupta, Sumegh Roychowdhury, Pattisapu Priyatam, Anish Bhanushali, and Prasanna Srinivasa Murthy. 2024. Exploring ordinality in text classification: A comparative study of explicit and implicit techniques. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5390–5404, Bangkok, Thailand. Association for Computational Linguistics.

Ashutosh Kumar, Kabir Ahuja, Raghuram Vadapalli, and Partha Talukdar. 2020. Syntax-guided controlled generation of paraphrases. *Transactions of the Association for Computational Linguistics*, 8:329–345.

Saksham Singh Kushwaha and Magdalena Fuentes. 2023. A multimodal prototypical approach for unsupervised sound classification. *arXiv preprint arXiv:2306.12300*.

Alexander Cong Li, Ananya Kumar, and Deepak Pathak. 2025. Generative classifiers avoid shortcut solutions. In *The Thirteenth International Conference on Learning Representations*.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. 2022a. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems*, 35:4328–4343.

Xueguang Li and 1 others. 2022b. Diffusion-lm improves controllable text generation. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Yingzhen Li, John Bradshaw, and Yash Sharma. 2019. Are generative classifiers more robust to adversarial attacks? In *International Conference on Machine Learning*, pages 3804–3814. PMLR.

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

Aaron Lou, Chenlin Meng, and Stefano Ermon. 2024. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Gati Martin, Medard Edmund Mswahili, Young-Seob Jeong, and Jiyoung Woo. 2022. SwahBERT: Language model of Swahili. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 303–313, Seattle, United States. Association for Computational Linguistics.

Andrew McCallum, Chris Pal, Gregory Druck, and Xuerui Wang. 2006. Multi-conditional learning: Generative/discriminative training for clustering and classification. In *AAAI*, volume 1, page 6.

Edgar C Merkle and Mark Steyvers. 2013. Choosing a strictly proper scoring rule. *Decision Analysis*, 10(4):292–304.

Hermann Ney, Ute Essen, and Reinhard Kneser. 1994. On structuring probabilistic dependences in stochastic language modelling. *Computer Speech & Language*, 8(1):1–38.

Andrew Y. Ng and Michael I. Jordan. 2001. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems*.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Nikhil Pattisapu, Siva Rajesh Kasa, Sumegh Roychowdhury, Karan Gupta, Anish Bhanushali, and Prasanna Srinivasa Murthy. 2025. Leveraging structural information in tree ensembles for table representation learning. *WWW*.

William Peebles and Saining Xie. 2023. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *OpenAI*.

Rajat Raina, Yirong Shen, Andrew Mccallum, and Andrew Ng. 2003. Classification with hybrid generative/discriminative models. *Advances in neural information processing systems*, 16.

Yasaman Razeghi, Robert L Logan IV, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot reasoning. *arXiv preprint arXiv:2202.07206*.

Mehdi Rezaee, Kasra Darvish, Gaoussou Youssouf Kebe, and Francis Ferraro. 2021. Discriminative and generative transformer-based models for situation entity classification. *arXiv preprint arXiv:2109.07434*.

Sumegh Roychowdhury, Karan Gupta, Siva Rajesh Kasa, and Prasanna Srinivasa Murthy. 2024. Tackling concept shift in text classification using entailment-style modeling. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5647–5656.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. 2024. Simple and effective masked diffusion language models. *arXiv preprint arXiv:2406.07524*.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. 2024. Simplified and generalized masked diffusion for discrete data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Emma AM Stanley, Nils D Forkert, and Matthias Wilms. 2025. Does a diffusion-based generative classifier avoid shortcut learning in medical image analysis? an initial investigation using synthetic neuroimaging data. In *Medical Imaging 2025: Imaging Informatics*, volume 13411, pages 94–99. SPIE.

Haoran Sun, Lijun Yu, Bo Dai, Dale Schuurmans, and Hanjun Dai. 2022. Score-based continuous-time discrete diffusion models. *arXiv preprint arXiv:2211.16750*.

Xiaofei Sun, Xiaoya Li, Jiwei Li, Fei Wu, Shangwei Guo, Tianwei Zhang, and Guoyin Wang. 2023. Text classification via large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8990–9005.

Yee Whye Teh. 2006. A bayesian interpretation of interpolated kneser-ney nus school of computing technical report tra2/06. *National University of Singapore*, pages 1–21.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12).

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36, Minneapolis, Minnesota. Association for Computational Linguistics.

Cheng Wang. 2023. Calibration in deep learning: A survey of the state-of-the-art. *arXiv preprint arXiv:2308.01222*.

Sen Wu, Hongyang R Zhang, and Christopher Ré. 2020. Understanding and improving information transfer in multi-task learning. In *International Conference on Learning Representations*.

Jing-Hao Xue and D. Michael Titterington. 2008. Comment on "on discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes". *Neural Process. Lett.*, 28(3):169–187.

Dani Yogatama, Chris Dyer, Wang Ling, and Phil Blunsom. 2017. Generative and discriminative text classification with recurrent neural networks. *arXiv preprint*.

Jiehang Zeng, Jianhan Xu, Xiaoqing Zheng, and Xuanjing Huang. 2023. Certified robustness to text adversarial attacks by randomized [mask]. *Computational Linguistics*, 49(2):395–427.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 649–657, Cambridge, MA, USA. MIT Press.

Xinyu Zhang, Hanbin Hong, Yuan Hong, Peng Huang, Binghui Wang, Zhongjie Ba, and Kui Ren. 2024. Text-crs: A generalized certified robustness framework against textual adversarial attacks. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 2920–2938. IEEE.

Chenyu Zheng, Guoqiang Wu, Fan Bao, Yue Cao, Chongxuan Li, and Jun Zhu. 2023. Revisiting discriminative vs. generative classifiers: Theory and implications. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 42420–42477. PMLR.

A More Background and Related works

Discrete Diffusion Models for Classification. Recent advances in discrete diffusion models have shown promising results in text generation tasks, matching or surpassing autoregressive models at GPT-2 scale (Lou et al., 2024; Sahoo et al., 2024; Shi et al., 2024). While these models have demonstrated success in controlled generation tasks (Li et al., 2022a; He et al., 2023), specifically syntax controlled generation of text (Kumar et al., 2020) and text infilling, their application to classification remains relatively unexplored. Traditional diffusion models for text generation, such as DiffusionBERT (He et al., 2023), DiffusionLM (Li et al., 2022b), and D3PM (Austin et al., 2021), operate by embedding discrete token sequences into continuous spaces and applying Gaussian noise-based diffusion. In contrast, SEDD (Lou et al., 2024) was the first to directly model diffusion in discrete space through a score entropy-driven objective. Hence, we adopt SEDD as our baseline method. Our work provides the first systematic evaluation of discrete diffusion models for classification tasks, comparing them against traditional discriminative and generative approaches.

Robustness to Noise. Previous studies have examined robustness primarily through the lens of adversarial attacks (Li et al., 2019), distribution shifts (Li et al., 2025) and domain shifts (Jaini et al., 2024). While recent work has provided certified robustness guarantees for perturbations like insertion, deletion, reordering and synonyms for specific architectures (Zeng et al., 2023; Zhang et al., 2024), our study presents comparisons across model families under two different noise conditions in the context of TC for transformer architectures.

Calibration & Ordinality. Model calibration is crucial in classification, as it reflects how well predicted probabilities align with actual frequencies. Proper Scoring Rules (PSR) (Merkle and Steyvers, 2013) offer a theoretical basis for producing calibrated predictions: a scoring rule (i.e. loss function) is proper if its expected value is minimized only when predicted probabilities match the true distribution. All our modeling approaches—Generative (AR, MLM, Discrete Diffusion) and Discriminative (Encoder)—optimize proper scoring rules. GPT and MLM maximize likelihood, Discrete Diffusion optimizes a variational bound, and cross-entropy minimizes the KL-divergence between predicted and true distributions. Recent work (Blasiok et al., 2023) shows that models trained with PSRs are often naturally calibrated when achieving low training loss, without requiring post-hoc calibration. This motivates us to empirically assess calibration across our models, as their differing architectures and objectives may still lead to varying calibration behaviors.

Ordinality in text classification is essential for applications like sentiment analysis or medical assessments, where label order affects decisions and distant misclassifications are more harmful. Recent works (Kasa et al., 2024) systematically compare *explicit* methods—like custom losses enforcing label order—with *implicit* approaches using pretrained models' semantics. However, no prior work focuses on exploring ordinality across diverse modeling frameworks trained from scratch.

B Implementation Details

We use the `bert-base-uncased`[2] architecture as the backbone for our **Encoder** and **MLM** experiments, without initializing the model with pretrained weights. This architecture contains approximately 110M parameters, comprising 12 encoder layers, 12 attention heads, and a hidden size of 768. We run all experiments for 3 random seeds and report the average and standard deviation results in main paper.

For the **Encoder** experiments, we conducted a grid search over several hyperparameters, including learning rates of {`1e-5`, `2e-5`, `3e-5`, `4e-5`, `5e-5`}, batch sizes of {32, 64, 128, 256}, and a fixed sequence length of 512 tokens. Training was performed for 30 epochs uniformly for all datasets without early stopping. For the **MLM**-based experiments, we retained similar hyperparameter ranges but trained for 200 epochs to account for the increased complexity of masked token prediction. We observed that adding an early stopping patience parameter sometimes led the model to select a suboptimal checkpoint, as the validation loss often continued to decrease gradually after remaining flat or oscillating for several epochs.

For the **AR** and $\textbf{AR}_{pseudo}$ experiments, we used the `GPT-2` base architecture[3] as the backbone with 137M parameters comparable with our other experiments. We trained a causal language model to minimize the next-token prediction loss over

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

[3] https://huggingface.co/openai-community/gpt2

Config	ENC	AR_{pseudo}	AR	MLM	DIFF
(1L,1H)	1-2	2-4	2-4	1-4	1-4
(6L,6H)	1-3	3-7	3-7	3-7	2-6
(12L,12H)	2-5	5-10	5-10	5-10	5-12

Table 3: Training time (in hrs) ranges across different datasets for each configuration and approach.

the concatenated input and label sequence. A grid search was conducted with the same hyperparameter range as mentioned above. The models were trained for up to 100 epochs, with early stopping based on validation loss, using a patience parameter of 10 epochs.

Our **Text Diffusion** approach follows the Diffusion Transformer architecture (Peebles and Xie, 2023) which is basically the vanilla transformer encoder with an extra time-conditioned embedding incorporated with it. The parameter count is $\sim$160M due to the addition of time-dependent embeddings required by the diffusion mechanism. To counter this, we conducted an ablation study by increasing the encoder size to 160M parameters (by adding layers) for other approaches (like `ENC`, `MLM`) to match the diffusion model size, but observed no difference in performance. Hence we retain their original settings as reported above. For diffusion-specific hyperparameters, we used a batch size of 64, learning rate `3e-4` and trained for 200K iterations. We adopted a geometric noise schedule that interpolates between 10^{-4} and 20, similar to the setup in (Lou et al., 2024), and used the following absorbing/masking matrix Q^{absorb} as part of the transition modeling. This was the best hyperparameter setting we found.

$$Q_{\text{absorb}} = \begin{bmatrix} -1 & 0 & \cdots & 0 \\ 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 0 \end{bmatrix}$$

All experiments were conducted using multi-GPU training across eight NVIDIA A100 GPUs. Training time varied depending on the methods and configurations used for each dataset. The range of training times (in hours) for various datasets is presented in Table 3. All reported training times correspond to full-data training configurations.

C Dataset Details

AG News (Zhang et al., 2015): It consists of approximately 120K training samples and 7.6K test samples, divided into four categories: World, Sports, Business, and Technology. Each sample contains a short news article, typically consisting of the title and the first few sentences. **Emotion** (Saravia et al., 2018): A collection of English tweets labeled with six basic emotions: anger, fear, joy, love, sadness, and surprise. It is designed for emotion detection in text. The dataset has 20K samples divided into 16K samples for training and 2K samples each for validation and testing. **Stanford Sentiment Treebank (SST)** (Socher et al., 2013): We utilize both the SST-2 (binary sentiment) and SST-5 (fine-grained sentiment) variants of the Stanford Sentiment Treebank dataset. SST-2 consists of sentences labeled as either positive or negative, suitable for binary sentiment classification, while SST-5 includes five sentiment categories: very negative, negative, neutral, positive, and very positive, allowing for more fine-grained sentiment analysis. **Multiclass Sentiment Analysis** [4]: This dataset consists of 41.6K data points, labeled into three sentiment categories: positive, negative, and neutral. While the dataset is designed for multiclass sentiment classification, it exhibits class imbalance, with certain sentiment classes being more prevalent than others. This imbalance provides a more realistic challenge for sentiment analysis models, testing their ability to handle skewed distributions and still perform effectively across all sentiment categories. **Twitter Financial News Sentiment** [5]: A specialized English-language collection of finance-related tweets, annotated for sentiment analysis. It consists of 11,932 tweets labeled with three sentiment categories: Bearish, Bullish, and Neutral. This dataset is designed to test models' ability to understand domain-specific language and nuanced sentiment expressions in financial contexts. **IMDb** (Maas et al., 2011): A binary sentiment analysis dataset consisting of 50K reviews from the Internet Movie Database (IMDb), labeled as positive or negative. The dataset is balanced, with an equal number of positive and negative reviews. This dataset is characterized by longer document lengths and detailed opinions, making it a challenging benchmark. **Rot-**

[4] https://huggingface.co/datasets/Sp1786/multiclass-sentiment-analysis-dataset

[5] https://huggingface.co/datasets/zeroshot/twitter-financial-news-sentiment

Dataset	Split	Examples	Classes	Avg Tokens	Label Dist. (%)	Ordinal
IMDb	train	25,000	2	313.87	0: 50.0, 1: 50.0	×
	test	25,000	2	306.77	0: 50.0, 1: 50.0	
agnews	train	120,000	4	53.17	0-3: 25.0 each	×
	test	7,600	4	52.75	0-3: 25.0 each	
emotion	train	16,000	6	22.26	0: 29.2, 1: 33.5, 2: 8.2, 3: 13.5, 4: 12.1, 5: 3.6	×
	test	2,000	6	21.90	0: 27.5, 1: 35.2, 2: 8.9, 3: 13.8, 4: 10.6, 5: 4.1	
hatespeech	train	22,783	3	30.04	0: 5.8, 1: 77.5, 2: 16.7	✓
	test	2,000	3	30.18	0: 5.5, 1: 76.6, 2: 17.9	
multiclasssentiment	train	31,232	3	26.59	0: 29.2, 1: 37.3, 2: 33.6	✓
	test	5,205	3	26.91	0: 29.2, 1: 37.0, 2: 33.8	
rottentomatoes	train	8,530	2	27.37	0: 50.0, 1: 50.0	×
	test	1,066	2	27.32	0: 50.0, 1: 50.0	
sst2	train	6,920	2	25.21	0: 47.8, 1: 52.2	×
	test	872	2	25.47	0: 49.1, 1: 50.9	
sst5	train	8,544	5	25.04	0: 12.8, 1: 26.0, 2: 19.0, 3: 27.2, 4: 15.1	✓
	test	1,101	5	25.24	0: 12.6, 1: 26.3, 2: 20.8, 3: 25.3, 4: 15.0	
twitter	train	9,543	3	27.62	0: 15.1, 1: 20.2, 2: 64.7	✓
	test	2,388	3	27.92	0: 14.5, 1: 19.9, 2: 65.6	

Table 4: Dataset statistics showing training and test split sizes, number of classes, mean and maximum token lengths, and label distribution percentages. Refer to Section C for details on datasets.

ten Tomatoes (Pang and Lee, 2005): A binary classification dataset which contains 10,662 movie review sentences, equally divided into 5,331 positive and 5,331 negative examples. The dataset is characterized by relatively short, opinion-driven sentences that reflect concise sentiments about films.

Hate Speech Offensive (Davidson et al., 2017): A major challenge in automatic hate speech detection is distinguishing hate speech from other forms of offensive language. This dataset consists of approximately 25K tweets, labeled into three categories: hate speech, offensive language without hate speech, and neutral content.

Refer to Table 4 for details on dataset statistics.

D More Ordinal & Calibration Results

In this section, we take a closer look at ordinal and calibration results for the datasets decribed above. Here we report ordinal metrics on the datasets **Stanford Sentiment Treebank (SST5)** (Socher et al., 2013), **Multiclass Sentiment Analysis**, **Hate Speech Offensive** (Davidson et al., 2017) and **Twitter Financial News Sentiment** since these are the only multi-class ordinal datasets out of 9. Calibration metrics are reported on all 9 datasets.

In Figure 5, we compare how ordinal and calibration metrics vary with increasing model size. Figure 6 presents the ordinal metrics for all four ordinal datasets, while Figure 7 shows the calibration metrics for all nine datasets. The corresponding insights are discussed in Section 5 (see **Q3**).

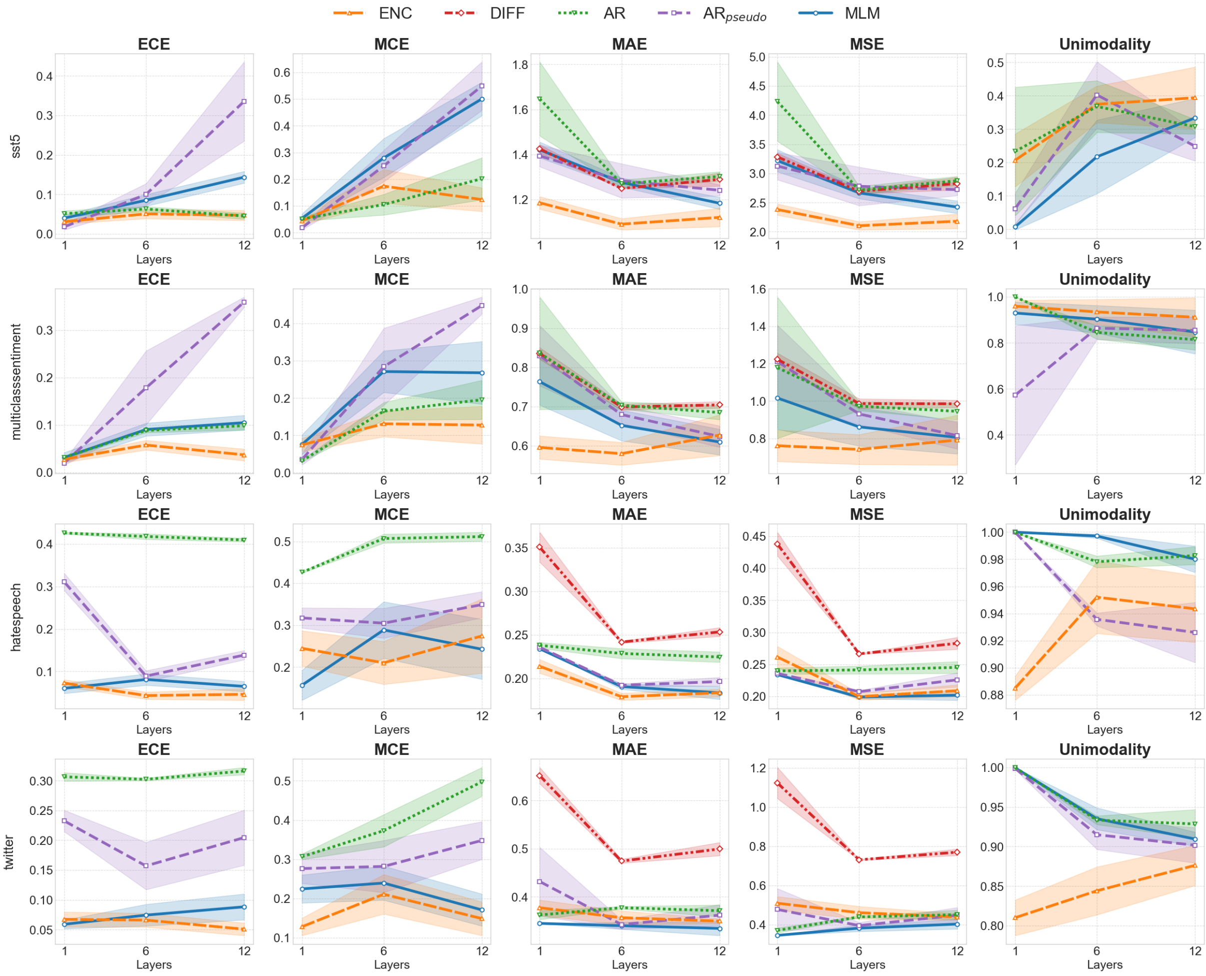

Figure 5: **[Best viewed in color]** Calibration and Ordinal metrics comparison across layers 1, 6 and 12. For ECE, MCE, MAE, MSE, (↓ is better) and UM (↑ is better).

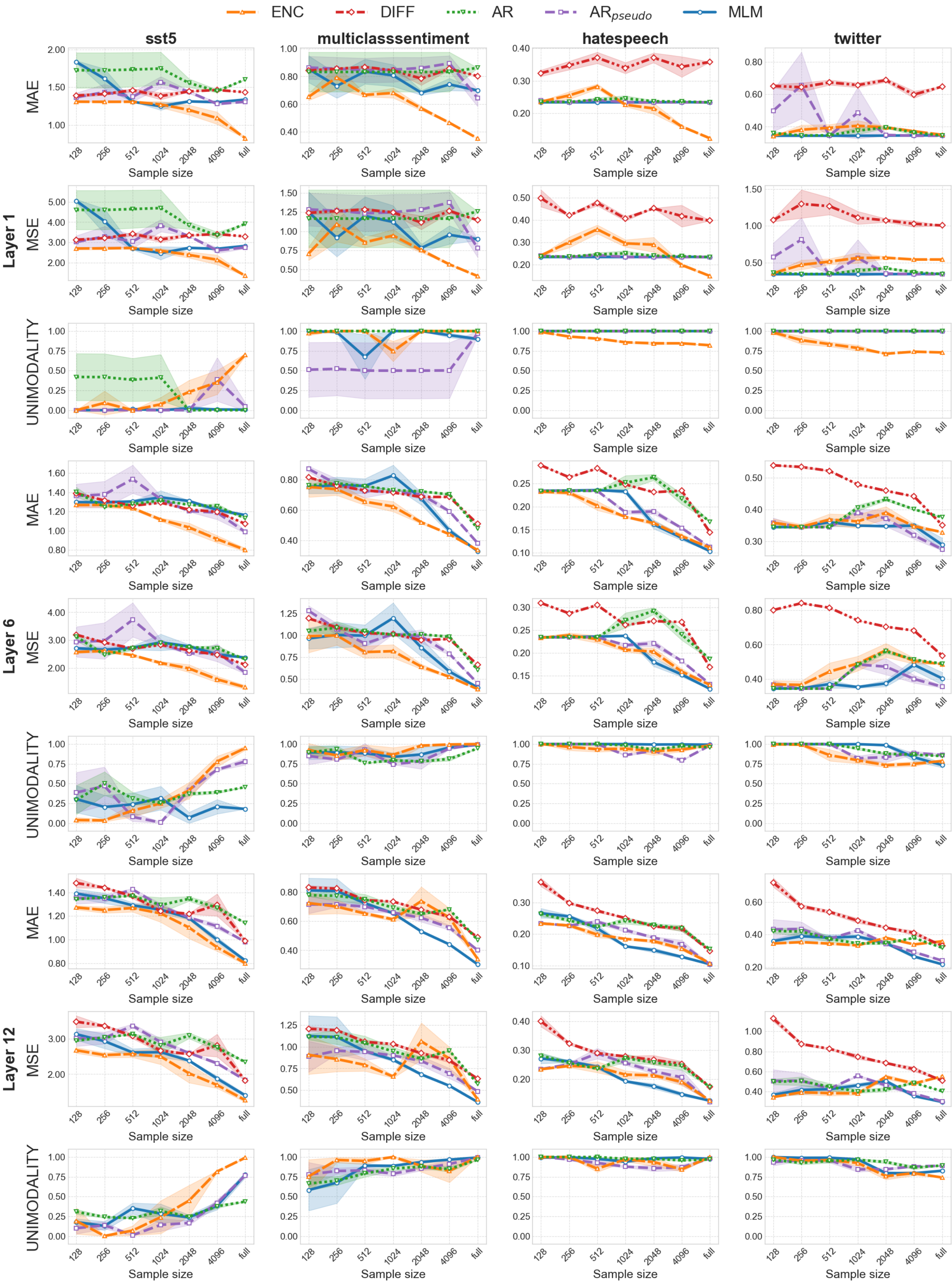

Figure 6: **[Best viewed in color]** Ordinal metrics. For MAE, MSE, (↓ is better) and UM (↑ is better).

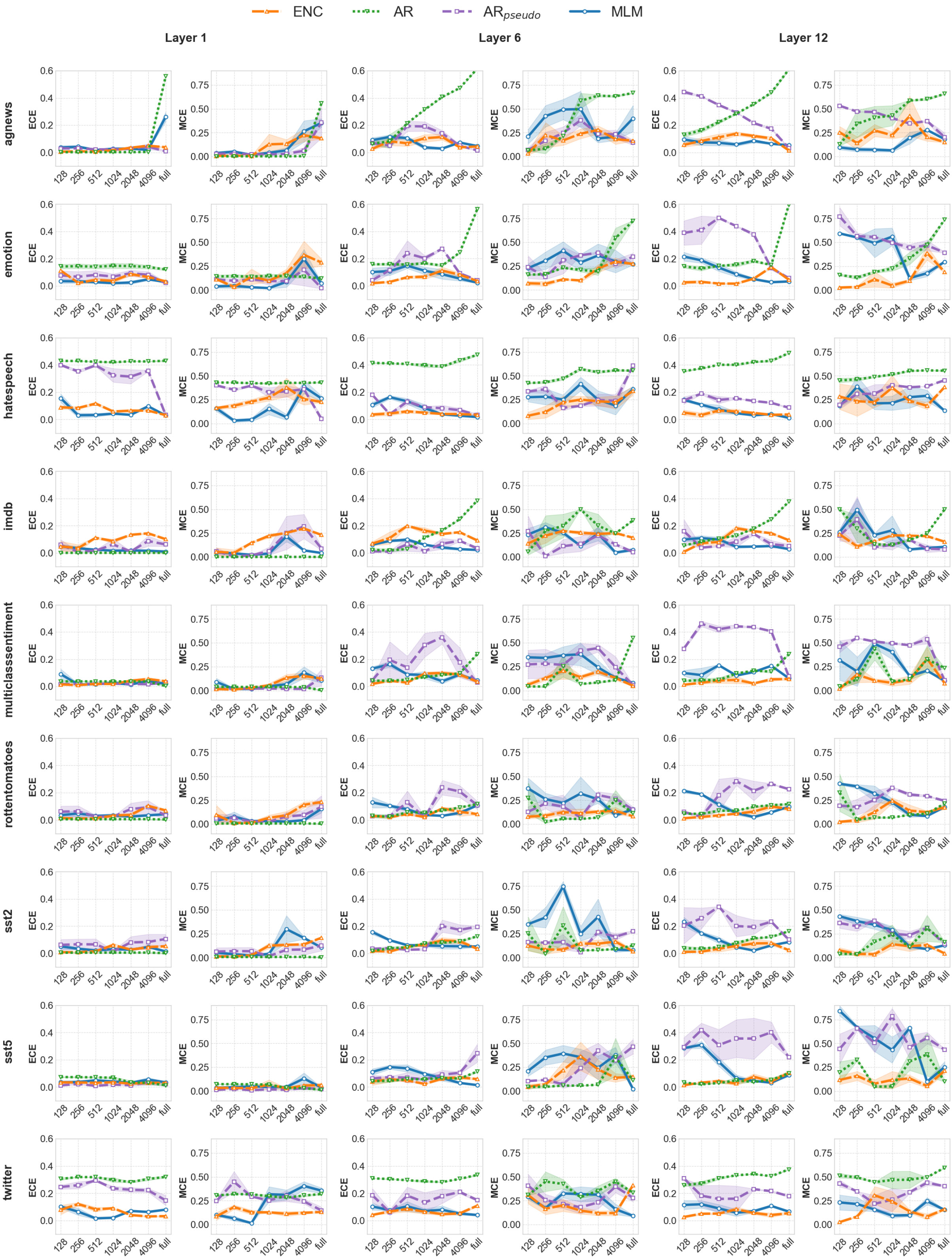

Figure 7: **[Best viewed in color]** Calibration metrics. For ECE, MCE (↓ is better)

E More Main Results

This section contains the extended results of Figure 2 (see Figure 8) and Figure 3 (see Figure 9) for all 9 datasets. We omit 1-layer plots for Figure 9 since the performance is mostly trivial for low-data settings and the same trend is observed as 6/12-layers for full-data settings.

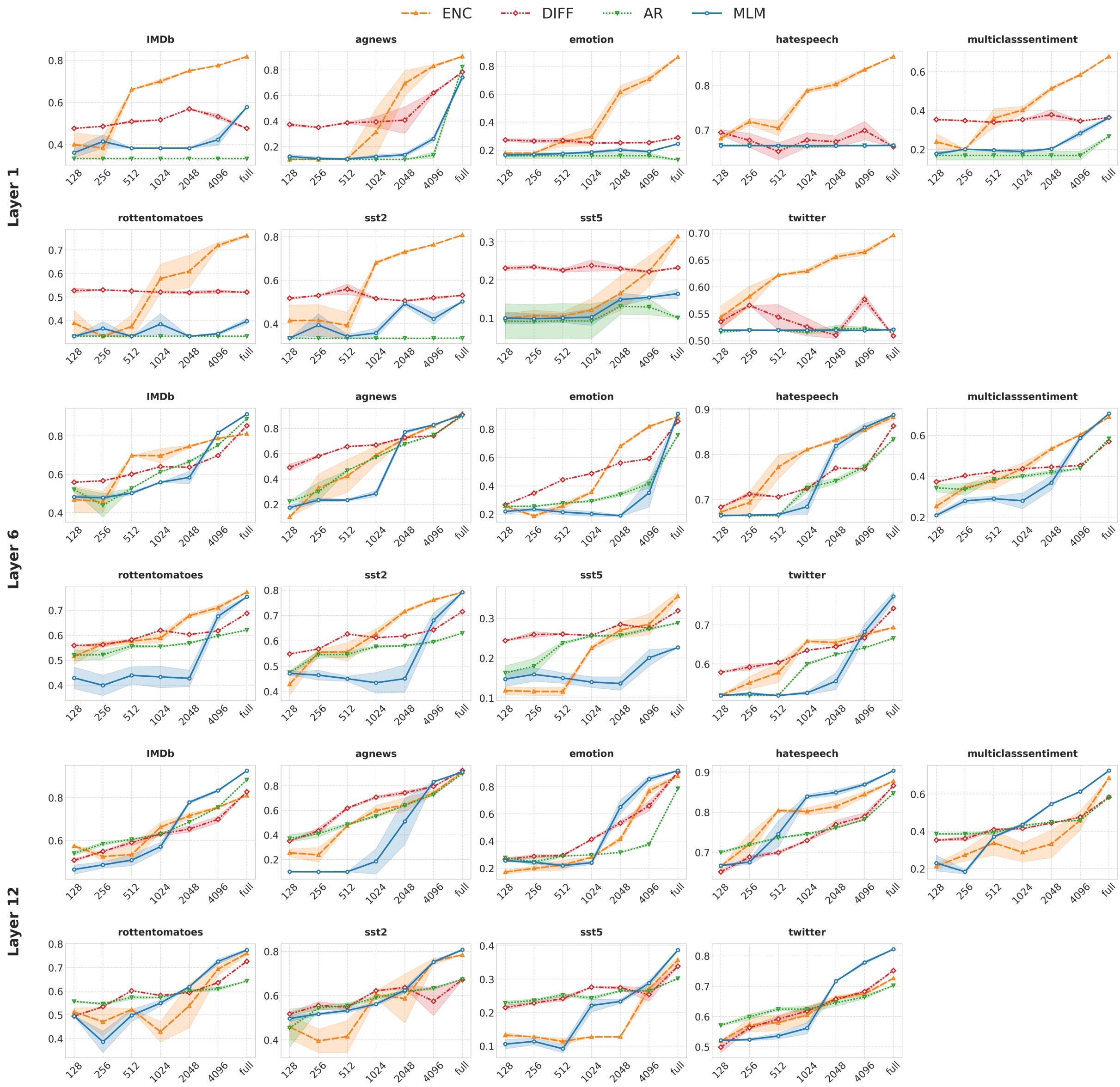

Figure 8: **[Best viewed in color]** Comparison of weighted-F1 scores of models across different configurations for all 9 datasets. (↑ is better) (`X-axis`: sample size, `Y-axis`: weighted-F1 score)

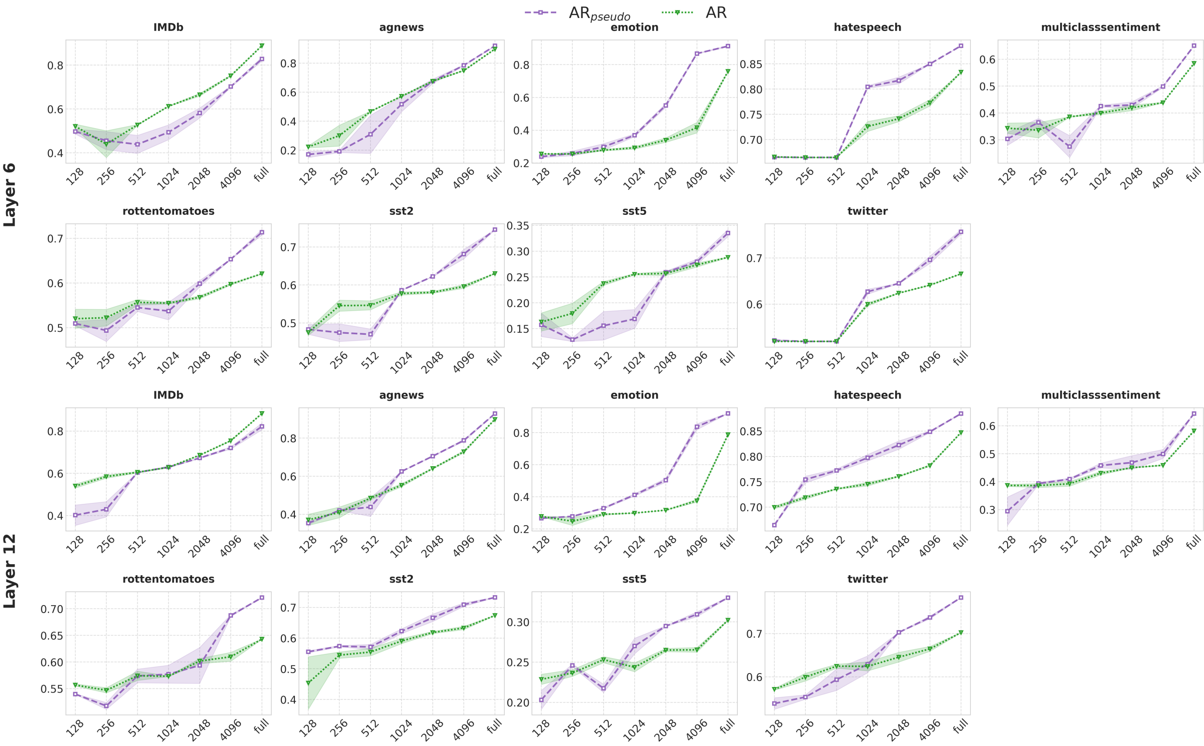

Figure 9: **[Best viewed in color]** Comparison of weighted-F1 scores between AR$_{pseudo}$ and AR (↑ is better) for all datasets. (`X-axis`: sample size, `Y-axis`: weighted-F1 score)