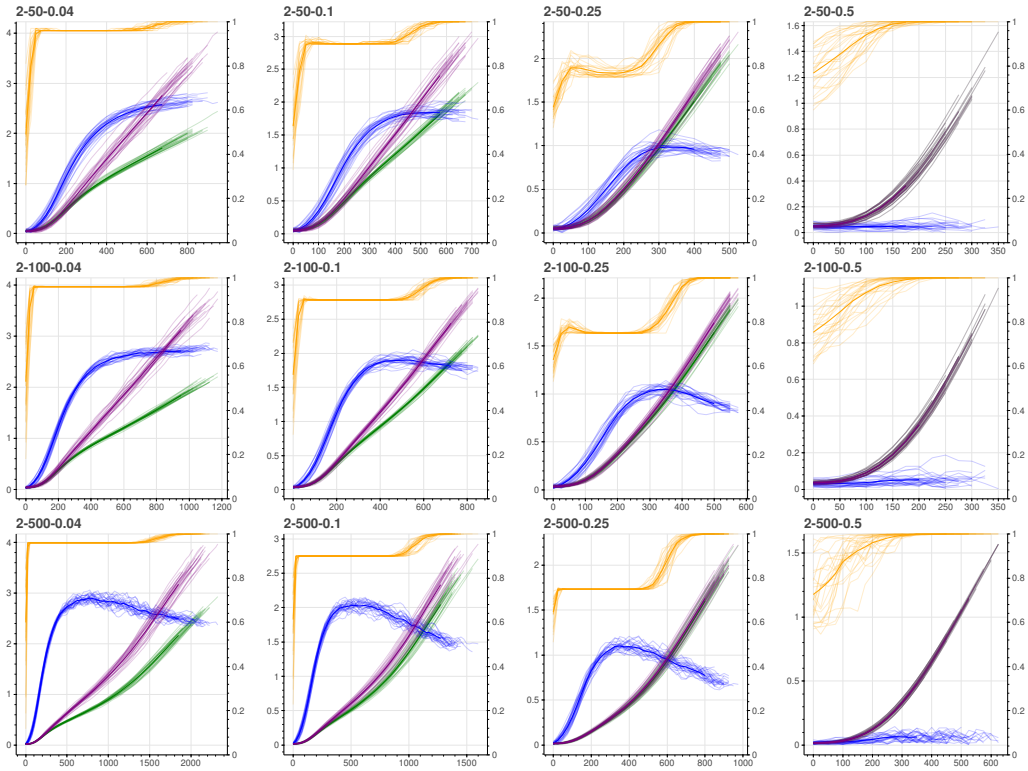


Figure 4: The average weights over the features during training a two-layer model without pretraining. From left to right, $\nu = 0.04, 0.10, 0.25, 0.5$. From top to bottom, $d_2 = 50, 100, 500$. Blue, green, purple curves represent the average weights over features in $X_1$, $X_2$, and $\dot{X}_2$ (the middle part of $X_2$) respectively. The orange curve represents the accuracy.

# A Appendix

## A.1 Proof of Lemma 1

*Proof.* Assume that $|\mathcal{X}_2|$ is discrete finite. Since $X_2$ is discrete and finite, the set $\{P(X_1|x_2)|x_2 \in \mathcal{X}_2\}$ is finite and discrete too. Therefore, the random variable $\Pi \in \{P(X_1|x_2)|x_2 \in \mathcal{X}_2\}$ is also discrete and finite. So we have

$$\begin{aligned} & H(X_1|\Pi_1) \\ =& \sum_{x_1,\pi_1} P(X_1,\pi_1) \log P(x_1|\pi_1) \\ =& \sum_{x_1,\pi_1} \sum_{x_2:P(X_1|x_2)=\pi_1} P(x_1,x_2) \log P(x_1|x_2) \\ =& H(X_1|X_2) \end{aligned}$$

□

Note that the assumption holds when $X_2$ is a sequence of tokens with bounded length. In practice, the input length of a MLM model is restricted due to the number of position embedding. So the assumption holds in general.

## A.2 Proof of Theorem 2

The intuition of the proof is that we compare two classifiers: (1) The one based on $X_1$, which can be constructed by counting the co-occurrence of $X_1$ and $Y$ (Eq 10). (2) The one based on $\Pi$. The

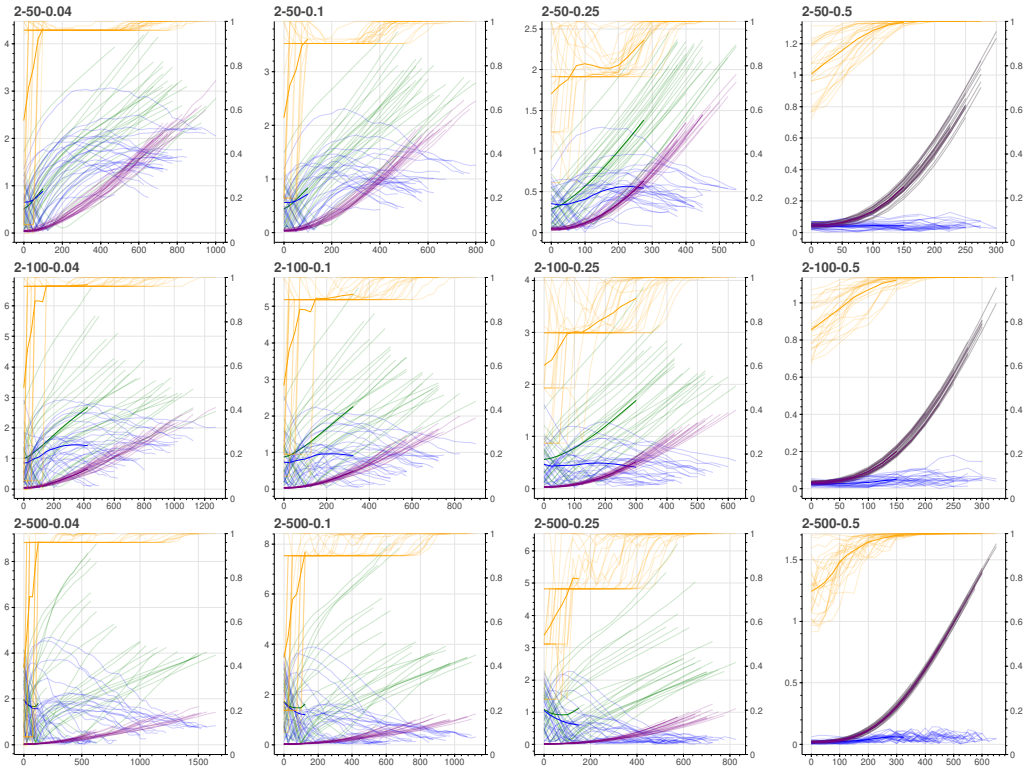


Figure 5: The average weights over the features during training a two-layer model with pretraining. From left to right, $\nu = 0.04, 0.10, 0.25, 0.5$. From top to bottom, $d_2 = 50, 100, 500$. Blue, green, purple curves represent the average weights over features in $X_1$, $X_2$, and $\dot{X}_2$ (the middle part of $X_2$) respectively. The orange curve represents the accuracy.

construction of this classifier can be seen as a relaxed version of (1). In (1), we count the occurrence of $X_1$ based on the observation of $X_1$. But in (2), we count the occurrence of $X_1$ based on the likelihood of $x_1$ for all $x_1 \in \mathcal{X}_1$ (Eq 12).

We then show that (a) the convergence rates of (1) and (2) are asymptotically equal. (b) the converged classifier from (2) is not worse than (1).

To proof Theorem 2, we need a lemma from Gibbs & Su (2002); Paninski (2003) for the convergence rate of empirical measures.

**Lemma 2.** *Given $n$ samples $x_1, x_2, \cdots, x_n$ of a random variable $X \in \{1, 2, \cdots, m\}$. Let*

$$q_i^{(n)} = \frac{1}{n} \sum_{j=1}^{n} \mathbb{1}[x_j = i]. \tag{8}$$

*The expected convergence rate*

$$\mathbb{E}\left[D_{KL}\left[q^{(n)} \,\middle\|\, p\right]\right] = O\left(\frac{1}{n}\right), \tag{9}$$

*where $p_i = P(X = i)$.*

*Proof.*

$$\begin{aligned}
\sum_{i=1}^{m} q_i^{(n)} \log \frac{q_i^{(n)}}{p_i} &\leq \log \left[\sum_{i=1}^{m} \frac{{q_i^{(n)}}^2}{p_i}\right] \text{ (By concavity of log)} \\
&= \log \left[\sum_{i=1}^{m} \frac{(q_i^{(n)} - p_i)^2}{p_i} + 1\right] \\
&\leq \sum_{i=1}^{m} \frac{(q_i^{(n)} - p_i)^2}{p_i}
\end{aligned}$$

$$\mathbb{E}\left[\sum_{i=1}^{m} \frac{(q_i^{(n)} - p_i)^2}{p_i}\right] = O\left(\frac{1}{n}\right)$$

□

**Lemma 3.** *Let $q^{(a)}, q^{(b)}$ be the empirical distribution estimated by counting $n$ samples following $p^{(a)}, p^{(b)}$. If $D_{KL}\left[p^{(a)} \,\middle\|\, q^{(a)}\right] = O(f(n))$ and $D_{KL}\left[p^{(b)} \,\middle\|\, q^{(b)}\right] = O(f(n))$ for some function $f(n)$ (e.g. $O(\frac{1}{n})$), then $D_{KL}\left[p^{(a)}p^{(b)} \,\middle\|\, q^{(a)}q^{(b)}\right] = O(f)$.*

With these two lemmas, we can prove Theorem 2:

*Proof.* Proof sketch of Theorem 2: The classifier that maximizes the likelihood of $(x_1^{(1)}, y^{(1)}), (x_1^{(2)}, y^{(2)}), \cdots (x_1^{(n)}, y^{(n)})$ can be attained by counting the co-occurrence of $X_1$ and $Y$.

$$\tilde{h}_{X_1}^{(n)}(y|X_1 = x) = \frac{\sum_{i=1}^{n} \mathbb{1}[y^{(i)} = y]\mathbb{1}[x_1^{(i)} = x]}{\sum_{i=1}^{n} \mathbb{1}[x_1^{(i)} = x]} \tag{10}$$

It converges to

$$\tilde{h}_{X_1}^{*}(y|X_1 = x) = P(y|X_1 = x). \tag{11}$$

Based on $\Pi_1$, a classifier can be attained by first estimating $P(Y)$ and $P(x_1|y)$ for all $x_1$ and $y$:

$$\rho_{y|x_1}^{(n)} = \frac{\sum_{i}^{n} \mathbb{1}[y^{(i)} = y]\pi_{x_1}^{(i)}}{\sum_{i}^{n} \pi_{x_1}^{(i)}}, \tag{12}$$

where $\pi_{x_1}^{(i)} = \Pi(X_1 = x_1^{(n)} | X_2 = x_2^{(n)})$, and then we can construct a classifier

$$\tilde{h}_{\Pi}^{(n)}(y|\pi) = \sum_{x_1} \rho_{y|x_1}^{(n)} \pi_{x_1}. \tag{13}$$

It converges to

$$\tilde{h}_{\Pi}^{*}(y|\pi) = \sum_{x_1} P(y|x_1)\pi. \tag{14}$$

Based on Lemma 2 and Lemma 3, we have $\mathbb{E}\left[D_{KL}\left[\tilde{h}_{X_1}^{(n)} \,\middle\|\, \tilde{h}_{X_1}^{*}\right]\right] = O(\frac{1}{n})$ and $\mathbb{E}\left[D_{KL}\left[\tilde{h}_{\Pi}^{(n)} \,\middle\|\, \tilde{h}_{\Pi}^{*}\right]\right] = O(\frac{1}{n})$.

Then we show that $\tilde{h}_{\Pi}^{*}(y|\pi)$ is at least as good as $\tilde{h}_{X_2}^{*}(y|\pi)$ by showing $D_{KL}\left[P(Y|X) \,\middle\|\, \tilde{h}_{X_1}^{*}(Y|X)\right] \geq D_{KL}\left[P(Y|X) \,\middle\|\, \tilde{h}_{\Pi}^{*}(Y|X)\right]$ with convexity:

$$\sum_{x_1} P(x_1|x_2) D_{KL}[P(Y|x_2) \parallel P(Y|x_1)] \geq D_{KL}\left[P(Y|x_2) \,\middle\|\, \sum_{x_1} P(Y|x_1)P(x_1|x_2)\right]. \tag{15}$$

□

### A.3 Elaboration on the Proof of Theorem 4

When $X_2$ is discrete, we can represent the conditional distribution as a matrix, e.g. $P(X_1|X_2) \in \mathbb{R}^{|\mathcal{X}_1|\times|\mathcal{X}_2|}$, $P(X_1|Y) \in \mathbb{R}^{|\mathcal{X}_1|\times|\mathcal{Y}|}$, $P(Y|X_2) \in \mathbb{R}^{|\mathcal{Y}|\times|\mathcal{X}_2|}$. Therefore, we have

$$P(X_1|X_2) = P(X_1|Y)P(Y|X_2). \tag{16}$$

When the it holds that $\{P(X_1|Y = y)|y \in \mathcal{Y}\}$ are linearly independent, namely columns in $P(X_1|Y)$ are linearly independent, there exists a matrix $A \in \mathbb{R}^{|\mathcal{Y}|\times|\mathcal{X}_1|}$ such that $AP(X_1|Y) = I$. By left multiplying $A$ on the both side of Equation 16, we have

$$P(Y|X_2) = AP(X_1|X_2). \tag{17}$$

The similar technique is used in Lemma 3.1 of Lee et al. (2020).

This implies that $Y$ can predicted based on $\Pi$ as accurately as predicting based on $X_2$. Thus, $I(\Pi; Y) \geq I(X_2; Y)$.

### A.4 Implementation Details of the Experiments

We pretrain the models until they converge, and choose the checkpoint with the lowest MLM loss on the validation set. For the hate speech detection task, we use the implementation provided by Zhou et al. (2021). Except that we use bert-base-uncased instead of roberta-large, we use the other hyper parameters provided in their script. For the NER task, we use the implementation by Hugging Face [3].

### A.5 Details of the Datasets

**NER:** The size of the training, validation and testing set of Conll-2003 is 14986, 3466 and 2688 respectively. This dataset consists of Reuters news articles. We also use WNUT-17 which is distributed under CC-BY 4.0. The language is English.

**Hate Speech Detection:** We use the version preprocessed by Zhou et al. (2021). This dataset consists of Twitter comments After filtering out instances without NOI, there are 3491, 672 and 602 instances in the training, validation, testing set respectively. The preprocessed version is distributed under Apache License 2.0. The language is English.

[3] https://github.com/huggingface/transformers/blob/master/examples/pytorch/token-classification/run_ner.py

### A.6 COMPUTATIONAL BUDGET

**Model Size:** We use the BERT-base-cased model. The trainable part contains 85M parameters.

**Infrastructure:** Every experiment can be run with a single NVIDIA GTX 2080Ti GPU. The workstation used for the experiments is equipped with 64G memory.

**Computation Time:** For the NER task, it takes 90 minutes for a run. For the hate speech detection task, it takes 16 minutes for a run.