

## A REPRODUCING EXPERIMENTS AND FIGURES

In this section, we present training and optimization details needed to reproduce our empirical validation of Theorem 1. We also published notebooks and check-pointed weights for two crucial experiments that investigate the result in the small and massive scale regimes, for Figure 1 and GPT-2 (ANONYMIZED).

### A.1 FIGURE 1

We provide a Jupyter notebook and model checkpoints for reproducing Figure 1. Please refer to this for hyperparameter settings. In short, we implemented a model (Mnih and Teh 2012) in the family of Section 2 and trained it on the Billion Word dataset (Chelba et al. 2013). This is illustrative of the property of Theorem 1 because the relatively modest size of the parameter space (see notebook) and massive dataset minimizes model convergence and data availability restrictions, e.g., approaches the asymptotic regime.

The word embedding space is 2-D for ease of visualization. We selected a subset of words, mapped them into their learned embeddings, and visualized them as points in the left and middle panes. We then regress pane one onto pane two in order to learn the best linear transformation between them. Note that if the two are linear transformations of each other, regression will recover that transformation exactly.

### A.2 SIMULATION STUDY: CLASSIFICATION BY DNNs

Remaining training details are as follows.

Because our theory requires that the data generating process be expressible by the true generative model, we simulate this by training a 4 hidden layer MLP with two  $2^6$  unit layers, and a 2-dimensional “bottle neck” layer. We optimize weights using Adam with a learning rate of  $10^{-4}$  for  $5 \cdot 10^4$  iterations.

To make the classification problem more challenging, we additionally add 20 input dimensions of random noise. The Adam optimizer with a learning rate of  $3 \cdot 10^{-4}$  is used.

### A.3 SELF-SUPERVISED LEARNING FOR IMAGE CLASSIFICATION

To compute linear similarity between representations, we train two independent models in parallel. For each model we define both  $\mathbf{f}_\theta$  and  $\mathbf{g}_\theta$  as a 3-layer fully connected neural network with  $2^8$  units per layer and a fixed output dimensionality of  $2^6$ . We define our model following Eq. 1 where  $S$  is the set of the other image patches from the current minibatch and optimize the objective of (Hénaff et al. 2019). We augment both sampled patches independently with randomized brightness, saturation, hue, and contrast adjustments, following the recipe of (Hénaff et al. 2019). We train on the CIFAR10 dataset (Krizhevsky et al. 2009) with batchsize  $2^8$ , using the Adam optimizer with a learning rate of  $10^{-4}$  and the JAX (Bradbury et al. 2018) software package. For each model, we early stop based on a validation loss failing to improve further.

Additional details about the experiments that generated Figure 3

**Figure 3a.** Patches are sampled randomly from training images.

**Figure 3b.** For each model, we train for at most  $3 \cdot 10^4$  iterations, early stopping when necessary based on validation loss.

**Figure 3c.** For each model, we train for at most  $3 \cdot 10^4$  iterations, early stopping when necessary based on validation loss.

**Figure 3d.** Error bars show standard error computed over 5 pairs of models after  $1.5 \cdot 10^4$  training iterations.

#### A.4 GPT-2

We include all details through a notebook in the code release. Pretrained GPT-2 weights as specified in the main text are publicly available from HuggingFace [Wolf et al. \(2019\)](#).

### B PROOF THAT LINEAR SIMILARITY IS AN EQUIVALENCE RELATION

We claim that  $\stackrel{\mathcal{L}}{\sim}$  is an equivalence relation. It suffices to show that it is reflexive, transitive, and symmetric.

*Proof.* Consider some function  $\mathbf{g}_\theta$  and some  $\theta', \theta^*, \theta^\dagger \in \Theta$ . Suppose  $\theta' \stackrel{\mathcal{L}}{\sim} \theta^*$ . Then, there exists an invertible matrix  $\mathbf{B}$  such that  $\mathbf{g}_{\theta'}(\mathbf{x}) = \mathbf{B}\mathbf{g}_{\theta^*}(\mathbf{x})$ . Since  $\mathbf{g}_{\theta^*}(\mathbf{x}) = \mathbf{B}^{-1}\mathbf{g}_{\theta'}(\mathbf{x})$ ,  $\stackrel{\mathcal{L}}{\sim}$  is symmetric. Reflexivity follows from setting  $\mathbf{g}_{\theta^*}$  to  $\mathbf{g}_{\theta'}$  and  $\mathbf{B}$  to the identity matrix. To show transitivity, suppose also that  $\theta^* \stackrel{\mathcal{L}}{\sim} \theta^\dagger$ . Then, there exists an invertible  $\mathbf{C}$  such that  $\mathbf{g}_{\theta^*}(\mathbf{x}) = \mathbf{C}\mathbf{g}_{\theta^\dagger}(\mathbf{x})$ . Since  $\mathbf{g}_{\theta'} \stackrel{\mathcal{L}}{\sim} \mathbf{g}_{\theta^*}$ ,  $\mathbf{B}^{-1}\mathbf{g}_{\theta'}(\mathbf{x}) = \mathbf{C}\mathbf{g}_{\theta^\dagger}(\mathbf{x})$ . Rearranging terms,  $\mathbf{g}_{\theta'}(\mathbf{x}) = \mathbf{B}\mathbf{C}\mathbf{g}_{\theta^\dagger}(\mathbf{x})$ , so that  $\theta' \stackrel{\mathcal{L}}{\sim} \theta^\dagger$  as required.  $\square$

### C SECTION 3.2 CONTINUED: CASE OF CONTEXT REPRESENTATION FUNCTION $\mathbf{g}$

Our derivation of identifiability of  $\mathbf{g}_\theta$  is similar to the derivation of  $\mathbf{f}_\theta$ . The primary difference is that the normalizing constants in Equation (6) do not cancel out. First, note that we can rewrite Equation (1) as:

$$p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{S}) = \exp(\tilde{\mathbf{f}}_\theta(\mathbf{x}, \mathbf{S})^\top \tilde{\mathbf{g}}_\theta(\mathbf{y})) \quad (9)$$

where:

$$\tilde{\mathbf{f}}_\theta(\mathbf{x}, \mathbf{S}) = [-Z(\mathbf{x}, \mathbf{S}); \mathbf{f}_\theta(\mathbf{x})] \quad (10)$$

$$\tilde{\mathbf{g}}_\theta(\mathbf{y}) = [1; \mathbf{g}_\theta(\mathbf{y})] \quad (11)$$

$$Z(\mathbf{x}, \mathbf{S}) = \log \sum_{\mathbf{y}' \in \mathbf{S}} \exp(\mathbf{f}_\theta(\mathbf{x})^\top \mathbf{g}_\theta(\mathbf{y}')). \quad (12)$$

Below, we will show that for the model family defined in Section (2)

$$p_{\theta'} = p_{\theta^*} \implies \mathbf{g}_{\theta'}(\mathbf{y}) = \mathbf{B}\mathbf{g}_{\theta^*}(\mathbf{y}), \quad (13)$$

where  $\mathbf{B}$  is an invertible  $(M \times M)$ -dimensional matrix, concluding the proof of the linear identifiability of models in the family defined by Equation (1). We adopt the same shorthands as in the main text.

#### C.1 DIVERSITY CONDITION

We assume that for any  $(\theta', \theta^*) \in \Theta$  for which it holds that  $p' = p^*$ , and for any given  $\mathbf{y}$ , there exist  $M+1$  tuples  $\{(\mathbf{x}^{(i)}, \mathbf{S}^{(i)})\}_{i=0}^M$ , such that  $p_{\mathcal{D}}(\mathbf{x}^{(i)}, \mathbf{y}, \mathbf{S}^{(i)}) > 0$ , and such that the  $((M+1) \times (M+1))$  matrices  $\mathbf{M}'$  and  $\mathbf{M}^*$  are invertible, where  $\mathbf{M}'$  consists of columns  $\tilde{\mathbf{f}}'(\mathbf{x}^{(i)}, \mathbf{S}^{(i)})$ , and  $\mathbf{M}^*$  consists of columns  $\tilde{\mathbf{f}}^*(\mathbf{x}^{(i)}, \mathbf{S}^{(i)})$ .

This is similar to the diversity condition of Section 3.2 but milder, since a typical dataset will have multiple  $\mathbf{x}$  for each  $\mathbf{y}$ .

#### C.2 PROOF

With the data distribution  $p_{\mathcal{D}}(\mathbf{x}, \mathbf{y}, \mathbf{S})$ , for a given  $\mathbf{y}$ , there exists a conditional distribution  $p_{\mathcal{D}}(\mathbf{x}, \mathbf{S}|\mathbf{y})$ . Let  $(\mathbf{x}, \mathbf{S})$  be a sample from this distribution. From equation (1) and the statement to prove, it follows that:

$$p'(\mathbf{y}|\mathbf{x}, \mathbf{S}) = p^*(\mathbf{y}|\mathbf{x}, \mathbf{S}) \quad (14)$$

Substituting in the definition of our model from equation (9), we find:

$$\exp(\tilde{\mathbf{f}}'(\mathbf{x}, \mathbf{S})^\top \tilde{\mathbf{g}}'(\mathbf{y})) = \exp(\tilde{\mathbf{f}}^*(\mathbf{x}, \mathbf{S})^\top \tilde{\mathbf{g}}^*(\mathbf{y})), \quad (15)$$

which, evaluating logarithms, becomes

$$\tilde{\mathbf{f}}'(\mathbf{x}, \mathbf{S})^\top \tilde{\mathbf{g}}'(\mathbf{y}) = \tilde{\mathbf{f}}^*(\mathbf{x}, \mathbf{S})^\top \tilde{\mathbf{g}}^*(\mathbf{y}), \quad (16)$$

which is true for any triple  $(\mathbf{x}, \mathbf{y}, \mathbf{S})$  where  $p_{\mathcal{D}}(\mathbf{y}|\mathbf{x}, \mathbf{S}) > 0$ .

From  $\mathbf{M}'$  and  $\mathbf{M}^*$  (Section C.1) and equation (16) we form a linear system of equations, collecting the  $M + 1$  relationships together:

$$\mathbf{M}'^\top \tilde{\mathbf{g}}'(\mathbf{y}) = \mathbf{M}^{*\top} \tilde{\mathbf{g}}^*(\mathbf{y}) \quad (17)$$

$$\tilde{\mathbf{g}}'(\mathbf{y}) = \mathbf{A} \tilde{\mathbf{g}}^*(\mathbf{y}), \quad (18)$$

where  $\mathbf{A} = (\mathbf{M}^* \mathbf{M}'^{-1})^\top$ , an invertible  $(M + 1) \times (M + 1)$  matrix.

It remains to show the existence of an invertible  $M \times M$  matrix  $\mathbf{B}$  such that

$$\mathbf{g}'(\mathbf{y}) = \mathbf{B} \mathbf{g}^*(\mathbf{y}). \quad (19)$$

We proceed by constructing  $\mathbf{B}$  from  $\mathbf{A}$ . Since  $\mathbf{A}$  is invertible, there exist  $j$  elementary matrices  $\{\mathbf{E}_1, \dots, \mathbf{E}_j\}$  such that their action  $\mathbf{R} = \mathbf{E}_j \mathbf{E}_{j-1} \dots \mathbf{E}_1$  converts  $\mathbf{A}$  to a (non-unique) row echelon form. Without loss of generality, we build  $\mathbf{R}$  such that the  $a_{1,1}$  entry of  $\mathbf{A}$  is the first pivot, leading to the particular row echelon form:

$$\mathbf{R} \mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,m \times 1} \\ 0 & \tilde{a}_{2,2} & \tilde{a}_{2,3} & \dots & \tilde{a}_{2,m \times 1} \\ 0 & 0 & \tilde{a}_{3,3} & \dots & \tilde{a}_{3,m \times 1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & \tilde{a}_{m \times 1, m \times 1} \end{bmatrix}, \quad (20)$$

where  $\tilde{a}_{i,j}$  indicates that the corresponding entry in  $\mathbf{R} \mathbf{A}$  may differ from  $\mathbf{A}$  due to the action of  $\mathbf{R}$ . Applying  $\mathbf{R}$  to Equation (17), we have

$$\mathbf{R} \tilde{\mathbf{g}}'(\mathbf{y}) = \mathbf{R} \mathbf{A} \tilde{\mathbf{g}}^*(\mathbf{y}). \quad (21)$$

We now show that removing the first row and column of  $\mathbf{R} \mathbf{A}$  and  $\mathbf{R}$  generates matrices of rank  $M$ . Let  $\overline{\mathbf{R} \mathbf{A}}$  and  $\overline{\mathbf{R}}$  denote the  $(M \times M)$  submatrices formed by removing the first row and column of  $\mathbf{R} \mathbf{A}$  and  $\mathbf{R}$  respectively.

Equation (20) shows that  $\overline{\mathbf{R} \mathbf{A}}$  has a pivot in each column, and thus has rank  $M$ . To show that  $\overline{\mathbf{R}}$  is invertible, we must show that removing the first row and column reduces the rank of  $\mathbf{R} = \mathbf{E}_j \mathbf{E}_{j-1} \dots \mathbf{E}_1$  by exactly 1. Clearly, each  $\mathbf{E}_k$  is invertible, and their composition is invertible. We must show the same for the composition of  $\mathbf{E}_k$ .

There are three cases to consider, corresponding to the three unique types of elementary matrices. Each elementary matrix acts on  $\mathbf{A}$  by either (1) swapping rows  $i$  and  $j$ , (2) replacing row  $j$  by a multiple  $m$  of itself, or (3) adding a multiple  $m$  of row  $i$  to row  $j$ . We denote elementary matrix types by superscripts.

In Case (1),  $\mathbf{E}_k^1$  is an identity matrix with row  $i$  and row  $j$  swapped. For Case (2),  $\mathbf{E}_k^2$  is an identity matrix with the  $j, j^{th}$  entry replaced by some  $m$ . For each  $\mathbf{E}_k^1$  and  $\mathbf{E}_l^2$  in  $\mathbf{R}$ , where  $1 \leq k, l \leq j$ , we know that the indices  $i, j \geq 2$ , because we chose the first entry of the first row of  $\mathbf{A}$  to be the pivot, and hence do not swap the first row, or replace the first row by itself multiplied by a constant. This implies that removing the first row and column of  $\mathbf{E}_k^1$  and  $\mathbf{E}_l^2$  removes a pivot entry 1 in the  $(1, 1)$  position, and removes zeros elsewhere. Hence, the  $(M \times M)$  submatrices  $\overline{\mathbf{E}_k^1}$  and  $\overline{\mathbf{E}_l^2}$  are elementary matrices with rank  $M$ .

For Case (3),  $\mathbf{E}_k^3$  has some value  $m \in \mathbb{R}$  in the  $j, i^{th}$  entry, and 1s along the diagonal. In this case, we may find a non-zero entry in some  $\mathbf{E}_k^3$ , so that, e.g., the second row has a pivot at position  $(2, 2)$ . Without loss of generality, suppose  $i = 1, j = 2$  and let  $m$  be some nonzero constant. Removing the

first row and column of  $\mathbf{E}_1^3$  removes this  $m$  also. Nevertheless,  $\overline{\mathbf{E}}_1^3 = \mathbf{I}_M$ , the rank  $M$  identity matrix. For any other  $\mathbf{E}_k^3$   $1 < i \leq M+1$ ,  $j \geq 2$  because we chose  $a_{1,1}$  as the first pivot, and hence do not swap the first row, or replace the first row by itself multiplied by a constant. In both cases, removing the first row and first column creates an  $\overline{\mathbf{E}}_k^3$  that is a rank  $M$  elementary matrix.

We have shown by the above that  $\overline{\mathbf{R}}$  is a composition of rank  $M$  matrices. We now construct the matrix  $\mathbf{B}$  by removing the first entries of  $\tilde{\mathbf{g}}'$  and  $\tilde{\mathbf{g}}^*$ , and removing the first row and first column of  $\mathbf{R}$  and  $\mathbf{R}\mathbf{A}$  in Equation (equation 21). Then, we have

$$\overline{\mathbf{R}}\mathbf{g}'(\mathbf{y}) = \overline{\mathbf{R}}\mathbf{A}\mathbf{g}^*(\mathbf{y}), \quad (22)$$

$$\mathbf{g}'(\mathbf{y}) = \overline{\mathbf{R}}^{-1}\overline{\mathbf{R}}\mathbf{A}\mathbf{g}^*(\mathbf{y}). \quad (23)$$

Choosing  $\mathbf{B} = \overline{\mathbf{R}}^{-1}\overline{\mathbf{R}}\mathbf{A}$  proves the result.  $\square$

## D REDUCTIONS TO CANONICAL FORM OF EQUATION (1)

In the following, we show membership in the model family of Equation (1) using the mathematical notation of the papers under discussion in Section 4. Note that each subsection will change notation to match the papers under discussion, which varies quite widely. We employ the following colour-coding scheme to aid in clarity:

$$\log p_{\theta}(\mathbf{y}|\mathbf{x}, \mathbf{S}) = \mathbf{f}_{\theta}(\mathbf{x})^{\top} \mathbf{g}_{\theta}(\mathbf{y}) - \sum_{\mathbf{y}' \in \mathbf{S}} \exp(\mathbf{f}_{\theta}(\mathbf{x})^{\top} \mathbf{g}_{\theta}(\mathbf{y}')),$$

where  $\mathbf{f}_{\theta}(\mathbf{x})$  is generalized to a **data representation function**,  $\mathbf{g}_{\theta}(\mathbf{y})$  is generalized to a **context representation function**, and  $\sum_{\mathbf{y}' \in \mathbf{S}} \exp(\mathbf{f}_{\theta}(\mathbf{x})^{\top} \mathbf{g}_{\theta}(\mathbf{y}'))$  is some **constant**.

### D.1 CPC

Formally, consider a sequence of points  $\mathbf{x}_t$ . We wish to learn the parameters  $\phi$  to maximize the  $k$ -step ahead predictive distribution  $p(\mathbf{x}_{t+k}|\mathbf{x}_t, \phi)$ . In the image patch example, each patch center  $i, j$  is indexed by  $t$ . Each  $\mathbf{x}_t$  is mapped to a sequence of feature vectors  $\mathbf{z}_t = f_{\theta}(\mathbf{x}_t)$ . An autoregressive model, already updated with the previous latent representations  $\mathbf{z}_{\leq t-1}$ , transforms the  $\mathbf{z}_t$  into a "context" latent representation  $\mathbf{c}_t = g_{AR}(\mathbf{z}_{\leq t})$ . Instead of predicting future observations  $k$  steps ahead,  $\mathbf{x}_{t+k}$ , directly through a generative model  $p_k(\mathbf{x}_{t+k}|\mathbf{c}_t)$ , Oord et al. (2018) model a density ratio in order to preserve the mutual information between  $\mathbf{x}_{t+k}$  and  $\mathbf{c}_t$ .

**Objective** Let  $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$  be a set of  $N$  random samples containing one positive sample from  $p(\mathbf{x}_{t+k}|\mathbf{c}_t)$  and  $N-1$  samples from the proposal distribution  $p(\mathbf{x}_{t+k})$ . Oord et al. (2018) define the following link function:  $l_k(\mathbf{x}_{t+k}, \mathbf{c}_t) \triangleq \exp(\mathbf{z}_{t+k}^{\top} \mathbf{W}_k \mathbf{c}_t)$ . Then, CPC optimizes

$$-\mathbb{E}_{\mathbf{X}} \left[ \log \frac{l_k(\mathbf{x}_{t+k}, \mathbf{c}_t)}{\sum_{\mathbf{x}_j \in \mathbf{X}} l_k(\mathbf{x}_j, \mathbf{c}_t)} \right] = -\mathbb{E}_{\mathbf{X}} \left[ \log \frac{\exp(\mathbf{z}_{t+k}^{\top} \mathbf{W}_k \mathbf{c}_t)}{\sum_{\mathbf{x}_j \in \mathbf{X}} \exp(\mathbf{z}_j^{\top} \mathbf{W}_k \mathbf{c}_t)} \right]. \quad (24)$$

Substituting in the definition of  $l_k$  makes equation (24) identical to the model family (Equation 1).

### D.2 AUTOREGRESSIVE LANGUAGE MODELS (E.G. GPT-2)

Let  $\mathcal{U} = \{u_1, \dots, u_n\}$  be a corpus of tokens. Autoregressive language models maximize a log-likelihood  $L(\mathcal{U}) = \sum_{i=1}^n \log P(u_i|u_{i-k}, \dots, u_{i-1}; \Theta)$ . Concretely, the conditional density is modelled as

$$\begin{aligned} \log P(u_i|u_{i-k:i-1}; \Theta) \\ = \mathbf{W}_i \mathbf{h}_i - \log \sum_j \exp(\mathbf{W}_j \mathbf{h}_i), \end{aligned}$$

where  $\mathbf{h}_i$  is the  $m \times 1$  output of a function approximator (e.g. a Transformer decoder (Liu et al. 2018)), and  $\mathbf{W}_i$  is the  $i$ 'th row of the  $|\mathcal{U}| \times m$  token embedding matrix.

### D.3 BERT

Consider a sequence of text  $\mathbf{x} = [x_1, \dots, x_T]$ . Some proportion of the symbols in  $\mathbf{x}$  are extracted into a vector  $\bar{\mathbf{x}}$ , and then set in  $\mathbf{x}$  to a special null symbol, “corrupting” the original sequence. This operation generates the corrupted sequence  $\underline{\mathbf{x}}$ . The representational learning task is to predict  $\bar{\mathbf{x}}$  conditioned on  $\underline{\mathbf{x}}$ , that is, to maximize w.r.t.  $\theta$ :

$$\log p_\theta(\bar{\mathbf{x}}|\underline{\mathbf{x}}) \approx \sum_{t=1}^T m_t \log p_\theta(x_t|\underline{\mathbf{x}}) = \sum_{t=1}^T m_t \left( H_\theta(\underline{\mathbf{x}})_t^\top e(x_t) - \log \sum_{x'} \exp(H_\theta(\underline{\mathbf{x}})_t^\top e(x')) \right),$$

where  $H$  is a transformer,  $e$  is a lookup table, and  $m_t = 1$  if symbol  $x_t$  is masked. That is, corrupted symbols are “reconstructed” by the model, meaning that their index is predicted. As noted in Yang et al. (2019), BERT models the joint conditional probability  $p(\bar{\mathbf{x}}|\underline{\mathbf{x}})$  as factorized so that each masked token is separately reconstructed. This means that the log likelihood is approximate instead of exact.

### D.4 QUICKTHOUGHT VECTORS

Let  $\mathbf{f}$  and  $\mathbf{g}$  be functions that take a sentence as input and encode it into an fixed length vector. Let  $s$  be a given sentence, and  $S_{ctx}$  be the set of sentences appearing in the context of  $s$  for a fixed context size. Let  $S_{cand}$  be the set of candidate sentences considered for a given context sentence  $s_{ctx} \in S_{ctx}$ . Then,  $S_{cand}$  contains a valid context sentence  $s_{ctx}$  as well as many other non-context sentences.  $S_{cand}$  is used for the classification objective. For any given sentence position in the context of  $s$  (for example, the preceding sentence), the probability that a candidate sentence  $s_{cand} \in S_{cand}$  is the correct sentence for that position is given by

$$\log p(s_{cand}|s, S_{cand}) = \mathbf{f}_\theta(s)^\top \mathbf{g}_\theta(s_{cand}) - \log \sum_{s' \in S_{cand}} \exp(\mathbf{f}_\theta(s)^\top \mathbf{g}_\theta(s')).$$

### D.5 DEEP METRIC LEARNING

The *multi-class N-pair loss* in Sohn (2016) is proportional to

$$\log N - \frac{1}{N} \sum_{i=1}^N \log \left( 1 + \sum_{j \neq i} \exp\{\mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_j) - \mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_i)\} \right),$$

which can be simplified as

$$\begin{aligned} & - \frac{1}{N} \sum_{i=1}^N \log \left( \frac{1}{K} \sum_{j=1}^K \exp\{\mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_j) - \mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_i)\} \right) \\ &= \frac{1}{N} \sum_{i=1}^N \log \left( \frac{1}{\frac{1}{K} \sum_{j=1}^K \exp\{\mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_j) - \mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_i)\}} \right) \\ &= \frac{1}{N} \sum_{i=1}^N \log \left( \frac{\exp\{\mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_i)\}}{\frac{1}{K} \sum_{j=1}^K \exp\{\mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_j)\}} \right). \end{aligned}$$

Setting  $N$  to 1 and evaluating the log gives

$$\mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_i) - \frac{1}{K} \sum_{j=1}^K \exp(\mathbf{f}_\theta(x_i)^\top \mathbf{f}_\theta(y_j)),$$

which is Equation 1 where  $\mathbf{f}_\theta = \mathbf{g}_\theta$ .

### D.6 NEURAL PROBABILISTIC LANGUAGE MODELS (NPLMS)

Figure 1 shows results from a neural probabilistic language model as proposed in Mnih and Teh (2012). Mnih and Teh (2012) propose using a log-bilinear model (Mnih and Hinton, 2009) which,

given some context  $h$ , learns a context word vectors  $r_w$  and target word vectors  $q_w$ . Two different embedding matrices are maintained, in other words: one to capture the embedding of the word and the other the context. The representation for the context vector,  $\hat{q}$ , is then computed as the linear combination of the context words and a context weight matrix  $C_i$  so that  $\hat{q} = \sum_{i=1}^{n-1} C_i r_{w_i}$ . The score for the match between the context and the next word is computed as a dot product, e.g.,  $s_\theta(w, h) = \hat{q}^\top \tilde{q}_w$  and substituting into the definition of  $P_\theta^h(w)$ , we see that

$$\log P_\theta^h(w) = \hat{q}^\top \tilde{q}_w - \log \sum_{w'} \exp(\hat{q}^\top \tilde{q}_{w'})$$

shows that Mnih and Teh (2012) is a member of the model family.

Interestingly, a touchstone work in the area of NPLMs, Word2Vec (Mikolov et al. 2013), does not fall under the model family due to an additional nonlinearity applied to the score of Mnih and Teh (2012).

<sup>1</sup>We have absorbed the per-token baseline offset  $b$  into the  $q_w$  defined in Mnih and Teh (2012), forming the vector  $\tilde{q}_w$  whose  $i$ 'th entry is  $(q_w)_i = (q_w)_i + b_w/(\hat{q})_i$