Visualizing the Relationship Between Encoded Linguistic Information and Task Performance

Jiannan Xiang[♡*], **Huayang Li**[♠*], **Defu Lian**[♢], **Guoping Huang**[♣], **Taro Watanabe**[♠], **Lemao Liu**[♣]

[♡]Carnegie Mellon University [♠]Nara Institute of Science and Technology
[♢]University of Science and Techology of China [♣]Tencent AI Lab
jiannanx@cs.cmu.edu, li.huayang.lh6@is.naist.jp
liandefu@ustc.edu.cn, donkeyhuang@tencent.com
taro@is.naist.jp, lemaoliu@gmail.com

Abstract

Probing is popular to analyze whether linguistic information can be captured by a well-trained deep neural model, but it is hard to answer how the change of the encoded linguistic information will affect task performance. To this end, we study the dynamic relationship between the encoded linguistic information and task performance from the viewpoint of Pareto Optimality. Its key idea is to obtain a set of models which are Pareto-optimal in terms of both objectives. From this viewpoint, we propose a method to optimize the Pareto-optimal models by formalizing it as a multi-objective optimization problem. We conduct experiments on two popular NLP tasks, *i.e.*, machine translation and language modeling, and investigate the relationship between several kinds of linguistic information and task performances. Experimental results demonstrate that the proposed method is better than a baseline method. Our empirical findings suggest that some syntactic information is helpful for NLP tasks whereas encoding more syntactic information does not necessarily lead to better performance, because the model architecture is also an important factor.

1 Introduction

Recent years have witnessed great success of deep neural networks for natural language processing tasks, such as language modeling (Zaremba et al., 2014; Merity et al., 2018) and Neural Machine Translation (Bahdanau et al., 2015; Vaswani et al., 2017). The excellent task performance they achieved spiked the interest in interpreting their underlying mechanism. Since linguistic knowledge is crucial in natural languages, an emerging body of literature uses *probes* (Conneau et al., 2018; Alt et al., 2020; Saleh et al., 2020; Cao et al., 2021) to investigate whether a standard model trained towards better task performance also captures the linguistic information. From the perspective of information theory, Voita and Titov (2020) and Pimentel et al. (2020b) show that probes can be used to estimate the amount of linguistic information captured by a fixed model.

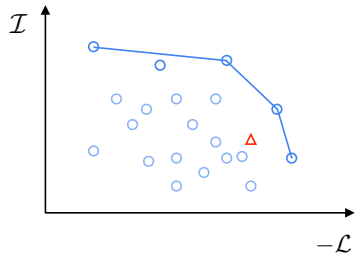

Figure 1: Illustration of Pareto frontier by a toy example. Triangle (△) corresponds to the standard checkpoint with best performance and each circle (◯) corresponds to a sampled checkpoint. The y-axis indicates the linguistic information $\mathcal{I}$ encoded by the model, and x-axis indicates the negative loss value $-\mathcal{L}$.

However, the above probing only extracts linguistic information from a fixed standard model, which helps little to understand the relationship between the task performance and linguistic information encoded by the model. For example, under their methodology, it is difficult to answer the following two questions. First, would adding linguistic information be beneficial for an NLP model; second, is it harmful when this linguistic information is reduced. Therefore, it is still an open and intriguing question to reveal how task performance changes with respect to different amounts of linguistic information.

To this end, this paper proposes a novel viewpoint to study the relationship between task performance and the amount of linguistic information, inspired by the criterion of Pareto Optimality which is widely used in economics (Greenwald and Stiglitz, 1986). Our main idea is to obtain Pareto-optimal models on a test set in terms of both linguistic information and task performance and then visualize their relationship along with these

[*]Equal contribution. Work done while J. Xiang was an intern at Tencent AI Lab.

Findings of the Association for Computational Linguistics: ACL 2022, pages 410 - 422

optimal models. By comparing a standard model with these optimal models, it is clear to answer the question that whether adding the encoded information is helpful to improve the task performance over the standard model , as illustrated in Figure 1, where the points on the line are Pareto-optimal and the red triangle denotes the standard model with best performance.

Nevertheless, it is typically intractable to obtain the Pareto-optimal models according to both dimensions on test data. To address the challenge, we propose a principled method to approximately optimize the Pareto-optimal models on the training data which can be expected to generalise well on test sets according to statistical learning theory (Vapnik, 1999). Formally, the approach can be regarded as a multi-objective optimization problem: during the learning procedure, it optimizes two objectives, *i.e.*, the task performance and extracted linguistic information. In addition, we develop a computationally efficient algorithm to address the optimization problem. By inspecting the trend of those Pareto-optimal points, the relationship between task performance and linguistic information can be clearly illustrated. Back to our questions, we also consider two instances within the proposed methodology: one aims to maximize the amount of linguistic information (*i.e.*, adding) while the other tries to minimize it (*i.e.*, reducing).

We conduct experiments on two popular NLP tasks, i.e., machine translation and language modeling, and choose three different linguistic properties, including two syntactic properties (Part-of-Speech and dependency labels) and one phonetic property. We investigate the relationship between NMT performance and each syntactic information, and the relationship between LM performance and phonetic information. For machine translation, we use LSTM, *i.e.*, RNN-search (Bahdanau et al., 2015), and Transformer (Vaswani et al., 2017) as the main model architectures, and conduct our experiments on En $\Rightarrow$ De and Zh $\Rightarrow$ En tasks. For language modeling, we employ the LSTM model and conduct experiments on the Penn Treebank dataset. The experimental results show that: i) syntactic information encoded by NMT models is important for MT task and reducing it leads to sharply decreased performance; ii) the standard NMT model obtained by maximum likelihood estimation (MLE) is Pareto-optimal for Transformer but it is not the case for LSTM based NMT; iii) reducing the phonetic information encoded by LM models only makes task performance drop slightly.

In summary, our contributions are three-fold:
1. We make an initial attempt to study the relationship between encoded linguistic information and task performance, *i.e.*, how the change of linguistic information affects the performance of models.
2. We propose a new viewpoint from Pareto Optimality as well as a principled approach which is formulated as a multi-objective optimization problem, to visualize the relationship.
3. Our experimental results show that encoding more linguistic information is not necessary to yield better task performance depending on the specific model architecture.

2 Related Work

Probe With the impressive performance of Neural Network models for NLP tasks (Sutskever et al., 2014; Luong et al., 2015; Vaswani et al., 2017; Devlin et al., 2019; Xu et al., 2020), people are becoming interested in understanding neural models (Ding et al., 2017; Li et al., 2019, 2020). One popular interpretation method is probe (Conneau et al., 2018), also known as auxiliary prediction (Adi et al., 2017) and diagnostic classification (Hupkes et al., 2018), which aims to understand how neural models work and what information they have encoded and used. From the perspective of information theory, Voita and Titov (2020) and Pimentel et al. (2020b) show that probes can be used to estimate the amount of linguistic information captured by a model. However, recent research studies point out that probes fail to demonstrate whether the information is used by models. For example, Hewitt and Liang (2019) show that the probe can also achieve high accuracy in predicting randomly generated tags, which is useless for the task. And Ravichander et al. (2021) present that the representations encode the linguistic properties even if they are invariant and not required for the task. Instead of studying the encoded linguistic information by training a probe for fixed representations, in this work we study how the amount change of linguistic information affects the performance of NLP tasks.

Information Removal Information removal is crucial in the area of transfer learning (Ganin and Lempitsky, 2015; Tzeng et al., 2017; Long et al., 2018) and fairness learning (Xie et al., 2017; Elazar and Goldberg, 2018), where people want to remove

domain information or bias from learned representations. One popular method is Adversarial Learning (Goodfellow et al., 2014; Ganin and Lempitsky, 2015), which trains a classifier to predict the properties of representations, *e.g.*, domain information or gender bias, while the feature extractor tries to fool the classifier. In this work, when using our method to reduce the linguistic information in the representations, we find that our multi-objective loss function is the same form as adversarial learning, which provides the theoretical guarantee for using adversarial learning to find the Pareto-optimal solutions to a multi-objective problem.

Recently, Elazar et al. (2020) also propose to study the role of linguistic properties with the idea of information removal (Ravfogel et al., 2020). However, the representations got by their method may not be Pareto-optimal, because it only minimizes the mutual information, but ignores the objective of task performance. On the contrary, our proposed method optimizes towards both objectives, thus our results can be used to visualize the relationship between linguistic properties and task performance.

Pareto Optimality The idea of Pareto Optimality (Mas-Colell et al., 1995) is an important criterion in economics, where the goal is to characterize situations where no variable can be better off without making at least one variable worse off. It has been also widely used in the area of sociology and game theory (Beckman et al., 2002; Chinchuluun et al., 2008). In addition, in artificial intelligence Martínez et al. (2020) use Pareto optimality to solve group fairness problem and Duh et al. (2012) proposed to optimize an MT system on multiple metrics based on the theory of Pareto optimality. In particular, Pimentel et al. (2020a) propose a variant of probing on the hidden representation of deep models and they consider Pareto optimality in terms of both objectives similar to our work. Comparing with their work, one difference is the choice of objectives. Another significant difference is that they optimize probing model in a conventional fashion, and thus are unable to study the relationship between linguistic information and task performance.

3 Visualizing Relationship via Pareto Optimality

We consider the relationship between linguistic information and task performance for two popular tasks in NLP, i.e., machine translation and language modeling. Let $\boldsymbol{x} = \{x_1, x_2, ..., x_N\}$ be a sentence and $\boldsymbol{s} = \{s_1, s_2, ..., s_N\}$ be the labels of the linguistic property of $\boldsymbol{x}$, where s_i is the label for x_i, e.g., POS tag. On both tasks, a deep model typically encodes $\boldsymbol{x}$ into a hidden representation $\boldsymbol{h}$ with a sub-network E parameterized by θ_e: $\boldsymbol{h} = E(\boldsymbol{x})$, and then uses another sub-network D parameterized by θ_d to map $\boldsymbol{h}$ into an output.

3.1 Background

$\boldsymbol{h}$ and Loss in NMT An NMT architecture aims to output a target sentence $\boldsymbol{y} = \{y_1, y_2, ..., y_M\}$ for a given source sentence $\boldsymbol{x}$ according to $P(\boldsymbol{y} \mid \boldsymbol{x}; \theta)$ (Zaremba et al., 2014; Vaswani et al., 2017), where θ indicates a set of parameters of a sequence-to-sequence neural network, which contains an encoder E and a decoder D. We define $\boldsymbol{h}$ as the output of the encoder. To train θ, the MLE loss is usually minimized on a training dataset. For NMT, the loss is defined as following:

$$L_\theta(\boldsymbol{x}, \boldsymbol{y}) = -\sum_{j=1}^{M} \log P(y_j|\boldsymbol{x}, \boldsymbol{y}_{<j}; \theta) \quad (1)$$

In our experiments, we consider two models, namely the LSTM (Bahdanau et al., 2015) and Transformer (Vaswani et al., 2017).

$\boldsymbol{h}$ and Loss in LM For language modeling task, a deep model typically generates a token x_j based on $\boldsymbol{x}_{<j}$ according to $P(x_j|\boldsymbol{x}_{<j}; \theta)$. Here the sub-networks E is set as one hidden layer to encode $\boldsymbol{x}_{<j}$ into $\boldsymbol{h}_{<j}$ and D is set as the sub-network to generate x_j on top of $\boldsymbol{h}_{<j}$. The parameter θ is optimized by the following MLE loss:

$$L_\theta(\boldsymbol{x}) = -\sum_{j=1}^{N} \log P(x_j|\boldsymbol{x}_{<j}; \theta).$$

To make notations consistent for both NMT and LM, in the rest of this paper, we follow the form of Eq. (1) and re-write the $L_\theta(\boldsymbol{x})$ in LM as $L_\theta(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{y}$ is a shifted version of $\boldsymbol{x}$, i.e., $\boldsymbol{y} = \{x_2, \cdots, x_N\}$.

Encoded Information Let $\mathrm{I}(\boldsymbol{h}, \boldsymbol{s})$ denote the linguistic information in the representation $\boldsymbol{h}$, *i.e.*, the mutual information between $\boldsymbol{h}$ and the linguistic label $\boldsymbol{s}$. Since the probability $p(\boldsymbol{h}, \boldsymbol{s})$ is unknown, it is intractable to compute $\mathrm{I}(\boldsymbol{h}, \boldsymbol{s})$. Following Pimentel et al. (2020b), we approximately estimate

$\mathrm{I}(\boldsymbol{h}, \boldsymbol{s})$ by using a probing model q as follows:

$$\begin{aligned}\mathrm{I}(\boldsymbol{h}, \boldsymbol{s}) &= H(\boldsymbol{s}) - H(\boldsymbol{s}|\boldsymbol{h}) \\ &\approx \mathrm{H}(\boldsymbol{s}) - \min_{\theta_q} L_{\theta_q}(\boldsymbol{h}, \boldsymbol{s}) \\ &= \mathrm{H}(\boldsymbol{s}) + \min_{\theta_q} \sum_i \log q(s_i|\boldsymbol{h}; \theta_q)\end{aligned} \tag{2}$$

where $H(\boldsymbol{s})$ is the entropy of linguistic labels, $H(\boldsymbol{s}|\boldsymbol{h})$ is the ideal cross entropy, and $L_{\theta_q}(\boldsymbol{h}, \boldsymbol{s})$ is the cross-entropy loss of the probe model q parameterized by θ_q.

Theory of Pareto Optimality Pareto optimality (Mas-Colell et al., 1995) is essentially involved in the multi-objective optimization problem. Suppose that we have K different objectives M_k to evaluate a parameter θ', *i.e.*,

$$\arg\max_{\theta'}[M_1(\theta'); M_2(\theta'); ...; M_K(\theta')]. \tag{3}$$

There are two important concepts in Pareto optimality as follows:

Definition 1. *Pareto Optimal: A parameter θ^* is Pareto-optimal iff for any θ', the condition always holds that, $\forall i = 1, ..., k,\ M_i(\theta^*) \geq M_i(\theta')$ and $\exists j$ such that $M_j(\theta^*) > M_j(\theta')$.*

Definition 2. *Pareto Frontier: The set of all Pareto-optimal parameters is called the Pareto frontier.*

3.2 Viewpoint via Pareto Optimality

Motivation Suppose θ is a given model parameter, $L(\theta)$ is its task performance on a test set, and $I(\theta)$ is the amount of linguistic information encoded in its hidden representation. Conventionally, if one can figure out a function f such that $I = f(L)$ for any θ, it is trivial to study their relationship by visualizing f. Unfortunately, for some complicated situations as illustrated in Figure 1, there does not exist such a function to represent the relationship between two variables due to a large number of many-to-many correspondences.

Our Viewpoint Pareto Optimality, a well-known criterion in economics (Mas-Colell et al., 1995), is widely used to analyze the relationship among multiple variables in a complicated environment (Chinchuluun et al., 2008). In our context, it is also a powerful tool to reveal the relationship between the encoded linguistic information and task performance. Taking the Pareto Frontier in Figure 1 as an example, since the capacity of a model is fixed and linguistic information may compete with other kinds of information, capturing more linguistic information may reduce the amount of information from other sources that are also helpful for the model. Nevertheless, if increasing the amount of linguistic information constantly leads to performance gain, i.e., linguistic information is complimentary to translation, only one Pareto Optimal point would exist on the top right corner.

Therefore, in this paper, we propose to study the relationship between $I(\theta)$ and $L(\theta)$ from the viewpoint of Pareto Optimality. Our key idea is to take into account only Pareto-optimal models rather than all models like the conventional method. Thanks to the definition of Pareto optimality, there are no many-to-many correspondences between two variables along the Pareto frontier. Hence their relationship can be visualized by the trend of these frontier points, as shown in Figure 1. Taking Figure 1 as an example, to answer the questions mentioned before, we can see that adding more information is possible to increase the task performance comparing with a standard model. According to this viewpoint, the core challenge is how to obtain a set of models which are Pareto optimality on a test dataset.

It is natural to employ a *heuristic* method to approximately obtain the Pareto-optimal models as following. We can first randomly select a number of checkpoints during the standard training and probe each checkpoint by optimizing its corresponding probing model q, as shown in Eq (2). Second, we can record the task performances and the amounts of linguistic information of the selected models on a test set. Finally, we can find the Pareto-optimal points and obtain the Pareto frontier. However, when using this method in our experiments, we find the amounts of encoded linguistic information for all checkpoints are similar and the the task performances of those checkpoints are worse than the optimal model. Hence, in the next section, a new method will be presented to approximately derive the Pareto-optimal models.

4 Methodology

4.1 Multi-Objective Optimization

To study the relationship between linguistic information and task performance, our goal is to obtain a set of models θ which are Pareto optimal on test data in terms of both objectives. Inspired by statistical learning theory (Vapnik, 1999), we propose an approach by optimizing the Pareto-optimal models towards both objectives on a given training dataset, which are expected to generalize well on unseen

test data, i.e., these models are Pareto optimal on unseen test data. Formally, Our approach can be formulated as the following multi-objective optimization problem:

$$\arg\min_{\theta}[L_\theta(\boldsymbol{x},\boldsymbol{y}); -\mathrm{I}(\boldsymbol{h},\boldsymbol{s})] \quad (4)$$

where minimizing $L_\theta(\boldsymbol{x},\boldsymbol{y})$ aims to promote the task performance and maximizing $\mathrm{I}(\boldsymbol{h},\boldsymbol{s})$ encourages a model to encode more linguistic information in the representation. Once we obtain a set of Pareto-optimal models, we can observe how increasing the encoded linguistic information affects the variance of task performance.

To further study how reducing the encoded linguistic information affects task performance, we optimize a similar multi-objective problem:

$$\arg\min_{\theta}[L_\theta(\boldsymbol{x},\boldsymbol{y}); \mathrm{I}(\boldsymbol{h},\boldsymbol{s})] \quad (5)$$

The only difference between Eq. (4) and Eq. (5) is that the former maximizes $\mathrm{I}(\boldsymbol{h},\boldsymbol{s})$ while the latter minimizes $\mathrm{I}(\boldsymbol{h},\boldsymbol{s})$.

Since $H(\boldsymbol{s})$ is a constant term, we can plug Eq. (2) into the above two equations and obtain the following reduced multi-objective problems:

$$\arg\min_{\theta}[L_\theta(\boldsymbol{x},\boldsymbol{y}); \min_{\theta_q} L_{\theta_q}(\boldsymbol{h},\boldsymbol{s})] \quad (6)$$

$$\arg\min_{\theta}[L_\theta(\boldsymbol{x},\boldsymbol{y}); -\min_{\theta_q} L_{\theta_q}(\boldsymbol{h},\boldsymbol{s})] \quad (7)$$

Notice that in the above equations, $\min_{\theta_q} L_{\theta_q}(\boldsymbol{h},\boldsymbol{s})$ resembles a conventional probing if $\boldsymbol{h}$ is a fixed representation. However, unlike the standard probing applied on top of a fixed $\boldsymbol{h}$ determined by the standard model, here $\boldsymbol{h}$ is the representation obtained from a encoder E parameterized by θ_e. It is also worth noting that the Pareto frontiers obtained from the Eq. (6) and (7) are independent, although they have a similar measurement, because the Pareto Optimal is only valid for the same objective.

4.2 Optimization Algorithm

To solve the above multi-objective problems, we leverage the linear-combination method to find a set of solutions, and then filter the non-Pareto-optimal points from the set to get the Pareto frontier. The details of our algorithm are shown below.

Optimization Process Since the detailed optimization method for Eq. (6) is similar to that for Eq. (7), in the following we take Eq. (6) as an example to describe the optimization method. Inspired by (Duh et al., 2012), we employ a two-step strategy for optimization to find the Pareto frontier to address the multi-objective problems.

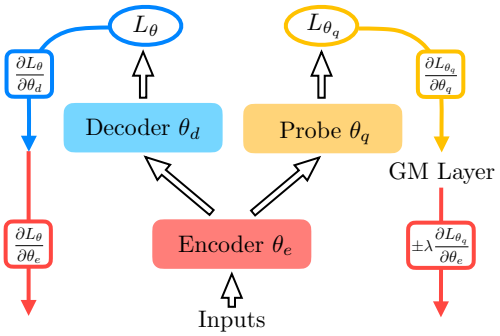

Figure 2: Overview of our multi-objective optimization method, where $L_y = L_\theta(\boldsymbol{x},\boldsymbol{y})$ and $L_{\theta_q} = L_{\theta_q}(\boldsymbol{h},\boldsymbol{s})$. In the back propagation, the GM Layer multiplies the gradient by $\pm\lambda$, i.e., λ for Eq. (6) and $-\lambda$ for Eq. (7).

In the first step, we adopt an method to find the Pareto-optimal solutions to the problem. There are several different methods to solve the problem, such as linear-combination, PMO (Duh et al., 2012) and APStar (Martínez et al., 2020). In this work, we adopt the linear-combination method because of its simplicity. Specifically, we select a coefficient set $\{\lambda_k \mid \lambda_k > 0\}$ and minimize the following interpolating function for each coefficient λ_k: [1]

$$\arg\min_{\theta}\left(L_\theta(\boldsymbol{x},\boldsymbol{y}) + \lambda_k \min_{\theta_q} L_{\theta_q}(\boldsymbol{h},\boldsymbol{s})\right) \quad (8)$$

Notice that the first term of the loss function $L_\theta(\boldsymbol{x},\boldsymbol{y})$ is the function of both encoder parameters θ_e and decoder parameters θ_d, while the second term $\min_{\theta_q} L_{\theta_q}(\boldsymbol{h},\boldsymbol{s})$ is only the function of θ_e. Therefore, when minimizing Eq.(8), we apply a Gradient-Multiple (GM) Layer on the representations before inputting it into the probe model. As shown in Fig. 2, in the forward propagation, the GM Layer acts as an identity transform, while in the backward propagation, the GM Layer multiples the gradient by $\pm\lambda$ and passes it to the preceding layers. Note that when the multiplier is $-\lambda$, the GM Layer is the same as Gradient Reversal Layer (Ganin and Lempitsky, 2015).

Suppose $\{\theta_k^* \mid \theta_k^* > 0\}$ is the minimized solution set for Eq. (8). In the second step, to get more accurate solutions, we filter the non-Pareto-optimal points of the solution set obtained by $\{\theta_k^* \mid \theta_k^* > 0\}$. Finally, we get the Pareto frontier to the multi-objective problem according to the definition of Pareto optimality.

[1] Eq. (8) is similar to the loss of standard multiple task learning (MTL) (Dong et al., 2015; Lee et al., 2020), $\arg\min_{\theta,\theta_q}\left(L_\theta(\boldsymbol{x},\boldsymbol{y}) \pm \lambda_k L_{\theta_q}(\boldsymbol{h},\boldsymbol{s})\right)$. However, the solutions to the loss are weaker than our solutions according to Pareto optimality, and it can not remove linguistic information

Algorithm 1 Optimization Algorithm

Input: $\Lambda = \{\lambda_k\}$, learning rate η
Output: Pareto frontier set $\mathcal{P} = \{\langle\theta_e^i, \theta_d^i\theta_q^i\rangle\}$

1: $\mathcal{M} = \{\}$ ▷ empty model set
2: **for** $\lambda_k \in \Lambda$ **do** ▷ minimize Eq. (8)
3: Random initialize θ_e^k, θ_d^k, and θ_q^k
4: **while** convergence **do**
5: $\theta_e^k = \theta_e^k - \eta(\frac{\partial L_\theta(x,y)}{\partial \theta_e} + \lambda_k \frac{\partial L_{\theta_q}(\boldsymbol{s},\boldsymbol{h})}{\partial \theta_e})$ ▷ $+\lambda_k$ is for Eq. (6) and changing it to $-\lambda_k$ would optimize Eq. (7)
6: $\theta_d^k = \theta_d^k - \eta \frac{\partial L_\theta(x,y)}{\partial \theta_d}$
7: $\theta_q^k = \theta_q^k - \eta \frac{\partial L_{\theta_q}(\boldsymbol{s},\boldsymbol{h})}{\partial \theta_q}$
8: **end while**
9: Re-train a probe model $\theta_{q'}^k$ based on fixed encoder θ_e
10: Add $\langle\theta_e^k, \theta_d^k, \theta_{q'}^k\rangle$ into $\mathcal{M}$
11: **end for**
12: $\mathcal{P} = \{\}$ ▷ Pareto frontier set
13: **for all** $\langle\theta_e^k, \theta_d^k, \theta_{q'}^k\rangle \in \mathcal{M}$ **do**
14: **if** `IsParetoOptimal`$(\theta_e^k, \theta_d^k, \theta_{q'}^k)$ **then**
15: Add $\langle\theta_e^k, \theta_d^k, \theta_{q'}^k\rangle$ into $\mathcal{P}$
16: **end if**
17: **end for**

Detailed Algorithm The overall optimization algorithm regarding to Eq. (6) is shown in Algorithm 1. Theoretically, when minimizing Eq. (8), in every step updating θ, we should retrain the probe model θ_q to minimize $L_{\theta_q}(\boldsymbol{h}, \boldsymbol{s})$ in for many steps, in order to estimate $\mathrm{H}(\boldsymbol{s}|\boldsymbol{h})$ precisely. However, this is time-consuming and inefficient. Instead, after updating θ, we update θ_q only by one step (see line 7 Algorithm 1). Empirically, we find that optimization in this way has been very effective.

In addition, as is mentioned by Elazar and Goldberg (2018), information leakage may occur when minimizing the mutual information. Therefore, after the training process is finished, we fix the deep model and retrain another probe model to estimate $\mathrm{H}(\boldsymbol{s}|\boldsymbol{h})$ more precisely (line 9 in Algorithm 1). When maximizing the mutual information, we find there is no difference between $\mathrm{H}(\boldsymbol{s}|\boldsymbol{h})$ estimated by jointly trained or retrained probe models.

5 Experimental Settings

5.1 Dataset

We conduct experiments on both machine translation and language modeling tasks. For machine translation, we conduct the experiments on En ⇒ De and Zh ⇒ En translation tasks. For En ⇒ De task, we use WMT14 corpus which contains 4M sentence pairs. For Zh ⇒ En task, we use LDC corpus which consists of 1.25M sentence pairs, and we choose NIST02 as our validation set, and NIST06 as our test set. For language modeling task, we use Penn Treebank[2] dataset. We preprocess our data using byte-pair encoding (Sennrich et al., 2016) and keep all tokens in the vocabulary. For machine translation, we use case-insensitive 4-gram BLEU score (Papineni et al., 2002) to measure the task performance, which is proved to be positively correlated well with the MLE loss (**?**); For language modeling, we directly use the MLE loss to evaluate the task performance.

5.2 Linguistic Properties

For machine translation, we study part-of-speech (POS) and dependency labels in this work. Since there are no gold labels for the MT datasets, we use Stanza toolkit[3] (Qi et al., 2020) to annotate source sentences and use the pseudo labels for running our algorithm, following Sennrich and Haddow (2016); Li et al. (2018). We clean the labels and remove the sentences that fail to be parsed by Stanza from the dataset. To study whether all kinds of linguistic information are critical for neural models, we also investigate the phonetic information on the language modeling task. More precisely, the probing model needs to predict the first character of the International Phonetic Alphabet of each word.[4] We get the labels with the open source toolkit English-to-IPA[5]. We use mutual information $\mathrm{I}(\boldsymbol{h}, \boldsymbol{s}) = \mathrm{H}(\boldsymbol{s}) - \mathrm{H}(\boldsymbol{s}|\boldsymbol{h})$ to evaluate the amount of information in the representations. Since $\mathrm{H}(\boldsymbol{s})$ is a constant, we only compare $\mathrm{H}(\boldsymbol{s}|\boldsymbol{h})$ in the experiments. Note that $\mathrm{H}(\boldsymbol{s}|\boldsymbol{h})$ is estimated by our probe model q.

5.3 Implementation Details

All of our models are implemented with Fairseq[6] (Ott et al., 2019). For NMT experiments, our LSTM model consists of a bi-directional 2-layer encoder with 256 hidden units, and a 2-layer decoder

in our preliminary experiments.

[2] https://deepai.org/dataset/penn-treebank

[3] https://github.com/stanfordnlp/stanza

[4] For example, given the input sentence "This dog is so cute", the probing model is asked to predict "ð d ɪ s k".

[5] https://github.com/mphilli/English-to-IPA

[6] https://github.com/pytorch/fairseq

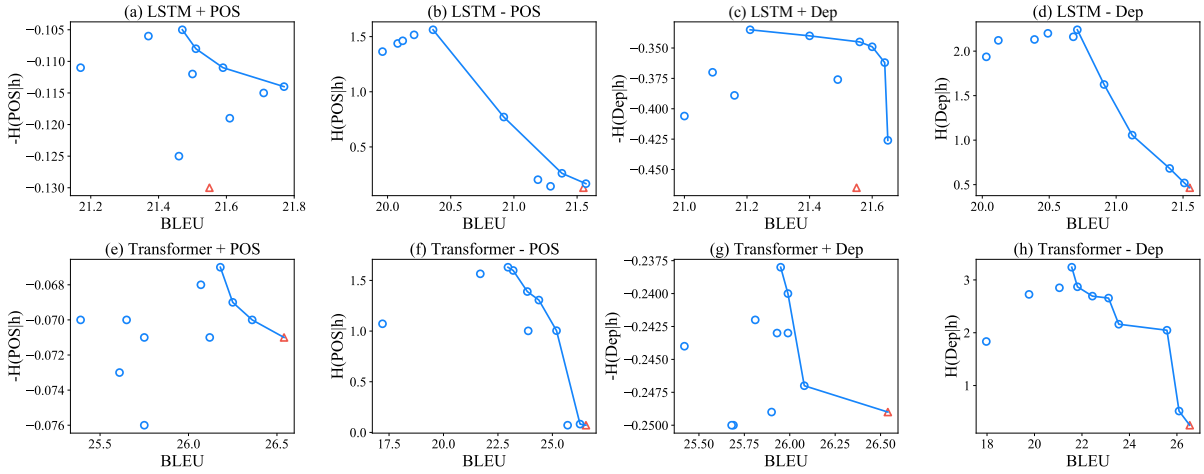

Figure 3: Experiments on WMT14 corpus. Triangle (△) denotes the model trained by minimizing MLE loss, circle (◯) denotes the models obtained by our method, and the models on the line (—) denotes the Pareto frontier.

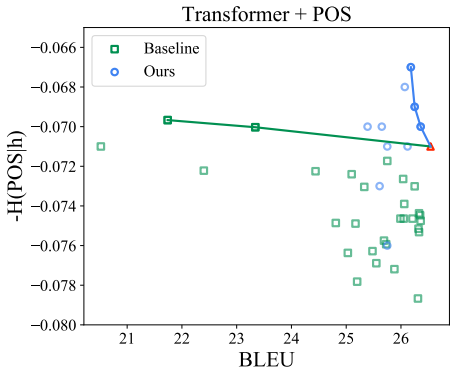

Figure 4: Comparison with baseline method. Triangle (△) denotes the standard model by minimizing MLE loss. The green line and blue line are frontiers got from baseline method and our method respectively.

with 512 hidden units, and the probe model is a 2-layer MLP with 512 hidden units. Our Transformer model consists of a 6-layer encoder and a 6-layer decoder, whose hyper-parameters are the same as the base model in (Vaswani et al., 2017), and the probe model is a 6-layer transformer encoder. For LM experiments, our model is a 2-layer LSTM with 256 hidden states, and the probe model is a 2-layer MLP with 256 hidden states. More training details about our models are shown in appendix A.

6 Experiment Results

In the following experiments, "**Model + Property**", e.g., "Transformer+Pos", which is corresponding to Eq. 4 and studies how adding the linguistic properties information affects the task performance. Instead, "**Model - Property**", e.g., "Transformer-Pos", which is corresponding to Eq. 5 and studies how removing the linguistic properties information affects the task performance. It is worth noting that merging the two frontiers of **+ Property** and **- Property** together would lead to trivial results, because Pareto Optimal points of the **+ Property** are more likely to dominate. However, we think the frontier of **- Property** is helpful for answering the question that whether reducing the encoded linguistic information would affect the model performance. Therefore, we plot the Pareto frontiers for the two objectives independently.

6.1 Soundness of Methodology

The heuristic method mentioned before can be considered as a simple and straightforward baseline method to measure the relationship. To set up this baseline, we firstly save some checkpoints every 1,000 steps when training a standard model. Second, we randomly sample 30 checkpoints for probing and plot a scatter diagram in terms of BLEU and encoded linguistic information.

As shown in Figure 4, we compare our proposed method with the heuristic method in the setting of "Transformer+Pos". Comparing with the baseline method, the frontier obtained from our method is better: for each model explored by baseline, there exists at least one model explored by our method whose two objectives, i.e., encoded linguistic information and BLEU score, are larger. The main reason is that the objective of baseline method only considers the task performance and most checkpoints contain similar encoded linguistic information. Therefore, the models optimized by our multi-objective method is more close to the globally Pareto-optimal points [7], making the

[7] It is worth mentioning that there are no algorithms to guarantee globally Pareto-optimal solutions in our scenario

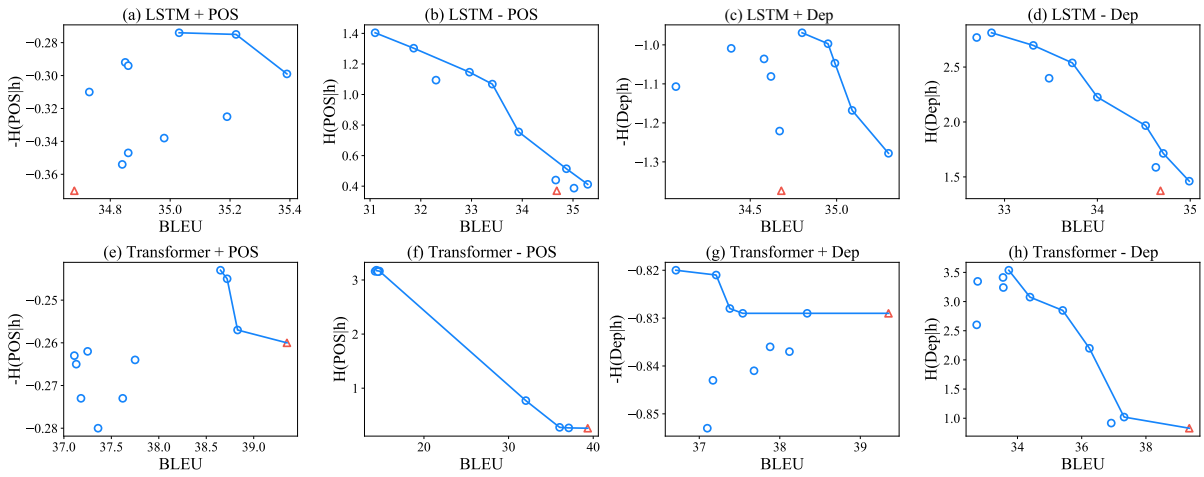

Figure 5: Experimental results on LDC corpus. The format is the same as Fig. 3

revealed relationship between encoded linguistic information and task performance more reliable. Therefore, in the next subsection, our proposed method will be used to visualize the relationship between encoded linguistic information and task performance for neural models.

6.2 Visualization Results

Results on NMT The results of machine translation on the WMT dataset are shown in Figure 3. For LSTM based NMT, we observe that the standard model, i.e., the △ in Figure 3, is not in the Pareto frontier in Figure 3 (a,c). In other words, when adding linguistic information into the LSTM model, it is possible to obtain a model which contains more POS or DEP information and meanwhile leads to better BLEU score than the standard model by standard training. In contrast, for Transformer based NMT, the standard model is in the Pareto frontier, as shown in Figure 3 (e,g). This finding provides an explanation to the fact in NMT: many efforts (Luong et al., 2016; Nădejde et al., 2017; Bastings et al., 2017; Hashimoto and Tsuruoka, 2017; Eriguchi et al., 2017) have been devoted to improve the LSTM based NMT architecture by explicitly modeling linguistic properties, but few have been done on Transformer based NMT (McDonald and Chiang, 2021; Currey and Heafield, 2019). In addition, when removing the linguistic information from LSTM or Transformer, the standard model is very close to the lower right of Pareto frontier, or even at the frontier, as shown in Figure 3 (b,d,f,h). This result shows that removing linguistic information always hurts the performance of NMT models for both LSTM and Transformer, indicating that encoding POS and DEP information is important for NMT task. Similar trends are observed on the LDC datasets, as shown in Figure 5. More details about the effect of randomness on our approach are shown in appendix B.

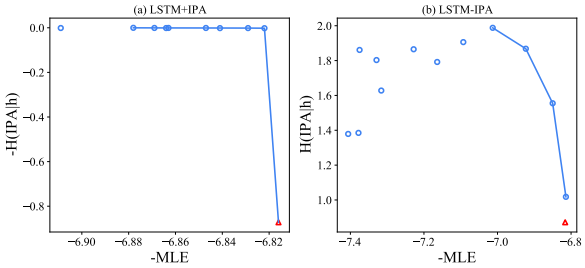

Figure 6: Experimental results on the PTB dataset.

Results on LM Above experiments have shown that both syntactic information are important for NMT models, and then a natural question is whether all kinds of linguistic information are important for neural models. To answer this question, we propose to investigate the influence of phonetic information on a language model. Figure 6 depicts the relationship between encoded phonetic information and task performance for an LSTM based language model. In Figure 6 (a), we find that the performances of Pareto-optimal models drop slightly when forcing an LSTM model to encode more phonetic information. Besides, as the Pareto-frontier shown in Figure 6 (b), removing phonetic information from an LSTM model only leads to a slight change in performance. These experiments demonstrate that the encoded phonetic information may be not that critical for an LSTM based language model. This finding suggest that

on the training data. Although the globally Pareto-optimal solutions are unknown, our solutions are definitely more close to them than the solutions by baseline.

not all kinds of linguistic information are crucial for LSTM based LM and it is not promising to further improve language modeling with phonetic information.

7 Conclusion

This paper aims to study the relationship between linguistic information and the task performance and proposes a new viewpoint inspired by the criterion of Pareto Optimality. We formulate this goal as a multi-objective problem and present an effective method to address the problem by leveraging the theory of Pareto optimality. We conduct experiments on both MT and LM tasks and study their performance with respect to linguistic information sources. Experimental results show that the presented approach is more plausible than a baseline method in the sense that it explores better models in terms of both encoded linguistic information and task performance. In addition, we obtain some valuable findings as follows: i) syntactic information encoded by NMT models is important for MT task and reducing it leads to sharply decreased performance; ii) the standard NMT model obtained by minimizing MLE loss is Pareto-optimal for Transformer but it is not the case for LSTM based NMT; iii) reducing the phonetic information encoded by LM models only leads to slight performance drop.

Acknowledgement

We would like to thank the anonymous reviewers for their constructive comments. L. Liu is the corresponding author.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. 2020. Probing linguistic features of sentence-level representations in neural relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1534–1545, Online. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Jasmijn Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, and Khalil Sima'an. 2017. Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967, Copenhagen, Denmark. Association for Computational Linguistics.

Steven R Beckman, John P Formby, W James Smith, and Buhong Zheng. 2002. Envy, malice and pareto efficiency: An experimental examination. *Social Choice and Welfare*, 19(2):349–367.

Steven Cao, Victor Sanh, and Alexander Rush. 2021. Low-complexity probing via finding subnetworks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 960–966, Online. Association for Computational Linguistics.

Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. 2018. The best of both worlds: Combining recent advances in neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–86, Melbourne, Australia. Association for Computational Linguistics.

Altannar Chinchuluun, Panos M Pardalos, Athanasios Migdalas, and Leonidas Pitsoulis. 2008. *Pareto optimality, game theory and equilibria*. Springer.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

Anna Currey and Kenneth Heafield. 2019. Incorporating source syntax into transformer-based neural machine translation. In *Proceedings of the Fourth Conference on Machine Translation*, pages 24–33, Florence, Italy. Association for Computational Linguistics.

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*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yanzhuo Ding, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Visualizing and understanding neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1150–1159, Vancouver, Canada. Association for Computational Linguistics.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1723–1732, Beijing, China. Association for Computational Linguistics.

Kevin Duh, Katsuhito Sudoh, Xianchao Wu, Hajime Tsukada, and Masaaki Nagata. 2012. Learning to translate with multiple objectives. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–10, Jeju Island, Korea. Association for Computational Linguistics.

Yanai Elazar and Yoav Goldberg. 2018. Adversarial removal of demographic attributes from text data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 11–21, Brussels, Belgium. Association for Computational Linguistics.

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2020. When bert forgets how to pos: Amnesic probing of linguistic properties and mlm predictions. *arXiv preprint arXiv:2006.00995*.

Akiko Eriguchi, Yoshimasa Tsuruoka, and Kyunghyun Cho. 2017. Learning to parse and translate improves neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 72–78, Vancouver, Canada. Association for Computational Linguistics.

Yaroslav Ganin and Victor S. Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1180–1189. JMLR.org.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680.

Bruce C Greenwald and Joseph E Stiglitz. 1986. Externalities in economies with imperfect information and incomplete markets. *The quarterly journal of economics*, 101(2):229–264.

Kazuma Hashimoto and Yoshimasa Tsuruoka. 2017. Neural machine translation with source-side latent graph parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 125–135, Copenhagen, Denmark. Association for Computational Linguistics.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China. Association for Computational Linguistics.

Dieuwke Hupkes, Sara Veldhoen, and Willem Zuidema. 2018. Visualisation and'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926.

Jason Lee, Dustin Tran, Orhan Firat, and Kyunghyun Cho. 2020. On the discrepancy between density estimation and sequence generation. In *Proceedings of the Fourth Workshop on Structured Prediction for NLP*, pages 84–94, Online. Association for Computational Linguistics.

Jierui Li, Lemao Liu, Huayang Li, Guanlin Li, Guoping Huang, and Shuming Shi. 2020. Evaluating explanation methods for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 365–375, Online. Association for Computational Linguistics.

Xintong Li, Guanlin Li, Lemao Liu, Max Meng, and Shuming Shi. 2019. On the word alignment from neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1293–1303, Florence, Italy. Association for Computational Linguistics.

Xintong Li, Lemao Liu, Zhaopeng Tu, Shuming Shi, and Max Meng. 2018. Target foresight based attention for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1380–1390, New Orleans, Louisiana. Association for Computational Linguistics.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. 2018. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 1647–1657.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *4th International Conference on Learning Representations,*

ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Natalia Martínez, Martín Bertrán, and Guillermo Sapiro. 2020. Minimax pareto fairness: A multi objective perspective. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6755–6764. PMLR.

Andreu Mas-Colell, Michael Dennis Whinston, Jerry R Green, et al. 1995. *Microeconomic theory*, volume 1. Oxford university press New York.

Colin McDonald and David Chiang. 2021. Syntax-based attention masking for neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 47–52, Online. Association for Computational Linguistics.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018. Regularizing and optimizing LSTM language models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Maria Nădejde, Siva Reddy, Rico Sennrich, Tomasz Dwojak, Marcin Junczys-Dowmunt, Philipp Koehn, and Alexandra Birch. 2017. Predicting target language CCG supertags improves neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 68–79, Copenhagen, Denmark. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Tiago Pimentel, Naomi Saphra, Adina Williams, and Ryan Cotterell. 2020a. Pareto probing: Trading off accuracy for complexity. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 3138–3153, Online. Association for Computational Linguistics.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020b. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4609–4622, Online. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, Online. Association for Computational Linguistics.

Abhilasha Ravichander, Yonatan Belinkov, and Eduard Hovy. 2021. Probing the probing paradigm: Does probing accuracy entail task relevance? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3363–3377, Online. Association for Computational Linguistics.

Abdelrhman Saleh, Tovly Deutsch, Stephen Casper, Yonatan Belinkov, and Stuart Shieber. 2020. Probing neural dialog models for conversational understanding. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 132–143, Online. Association for Computational Linguistics.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 83–91, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks.

In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2962–2971. IEEE Computer Society.

Vladimir N Vapnik. 1999. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 183–196, Online. Association for Computational Linguistics.

Qizhe Xie, Zihang Dai, Yulun Du, Eduard H. Hovy, and Graham Neubig. 2017. Controllable invariance through adversarial feature learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 585–596.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. Layoutlm: Pre-training of text and layout for document image understanding. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1192–1200. ACM.

Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

BLEU		H(POS|h)	
mean	var	mean	var
21.08	0.00407	0.1113	0
21.32	0.01536	0.1093	0
21.49	0.01847	0.108	0
21.52	0.00060	0.1123	0

Table 1: Experiment results from LSTM + POS setting. Specifically, "mean" and "var" denotes the mean and the variance over the window.

A Training Details

On the WMT14 corpus, training one LSTM model with 4 V100 GPUs costs 5 hours, and training one Transformer with 8 V100 GPUs costs 8 hours. On LDC corpus, training one LSTM model with 4 V100 GPUs costs 3 hours, and training one Transformer with 8 V100 GPUs costs 3 hours. On the PTB dataset, training LSTM model with 1 V100 GPU costs 6 minutes.

When running our algorithm, we empirically observe that when λ is below 0.01, the optimized models show little difference comparing with the standard model, and when λ is larger than 0.1, the proposed algorithm becomes unstable and can't converge to Pareto-optimal solutions well. Therefore, we take ten values from 0.1 to 0.01 at equal intervals as λ in Eq. 8, and train ten models with different λ for each condition respectively. Then we plot all the models and the Pareto frontier of these models in the experiments.

B Effects of Randomness

Following the method from Chen et al. (2018), we check if randomness will affect our experimental results. Specifically, we select a window of size 3 around the best checkpoint model and report the mean and variance over the selected window. The results are shown in Table 1. Because repeating experiments under all the settings are too extensive, we only randomly select 4 models from LSTM + POS settings. As shown in the table, all the variances are small, and the variances of the entropy even achieve 0. This suggests that the random disturbance of our experiments are small and thus our results are reliable.