INFORMATION-THEORETIC VOCABULARIZATION VIA OPTIMAL TRANSPORT

Anonymous authors
Paper under double-blind review

ABSTRACT

It is well accepted that the choice of token vocabulary largely affects the performance in NLP tasks. One dominant approach to construct a good vocabulary is the Byte Pair Encoding method (BPE). However, due to expensive trial cost, prior research have rarely tried to search for best token dictionary and its size, other than simple trials of BPE with commonly used vocabulary size (e.g. 30K). In this paper, we find an exciting relation between an information-theoretic feature and the performance of NLP tasks such as machine translation with a given vocabulary. With this observation, we formulate the quest of vocabularization – finding the best token dictionary with a proper size – as an optimal transport problem. We then propose ***Info*-VOT**, a simple and efficient solution without the full and costly trial training on the downstream task. We evaluate our approach on multiple machine translation tasks, including WMT-14 English-German translation, WMT-16 English-Romanian translation, and TED translation. Empirical results show that *Info*-VOT beats widely-used vocabulary construction methods with only a fifth of the number of tokens.

1 INTRODUCTION

Due to the discreteness of text, it has been a standard practice for natural language processing (NLP) tasks (Mikolov et al., 2013; Vaswani et al., 2017; Gehrmann et al., 2018; Zhang et al., 2018; Devlin et al., 2019) to embed the sequence of input tokens using a vocabulary-based lookup table, with each row representing a token as a dense vector. As a necessary prerequisite, vocabulary construction bridges the gap between discrete symbols and continuous representations. Currently, the widely-used vocabularies (e.g., subword vocabularies) are mainly constructed by heuristic approaches (Sennrich et al., 2016; Costa-jussà & Fonollosa, 2016; Lee et al., 2017; Sennrich et al., 2016; Kudo & Richardson, 2018; Al-Rfou et al., 2019; Wang et al., 2020).

Despite the promising performance, most of these approaches inevitably require a large number of human efforts to adjust the granularity of segmentation units. While many previous studies (Ding et al., 2019; Salesky et al., 2020) show that vocabulary size has a great impact on models' performance, especially on low-resource scenarios, very few existing work carefully tune this hyperparameter due to expensive computation resource costs. To address this problem, researchers recently pay increasing attention on automatic vocabulary search (AVS). However, to the best of our knowledge, there is still no literature that provides a unified theory to explore what quantitative features impact the performance of vocabularies and how to formulate vocabularization as a learning problem based on these features such that the optimal vocabulary can be automatically learned by leveraging current machine learning techniques, instead of heuristically defined.

In this work, we take the first step to a unified theory for automatic vocabulary learning (**AVL**). To be specific, we present a new perspective of information theory to quantitatively describe vocabularies and then formulate vocabularization into a discrete optimization problem, followed by a mathematically derived solution based on optimal transport (*Info*-VOT). We start from the entropy of token distribution. From the view of information theory, entropy represents the average level of information inherent in the tokens, or how much bits we need to represent these tokens, also called Bits Per Character (BPC). Here we find an exciting experiment phenomenon that an information theoretical feature, AMD-BPE, short for AMD, is strongly related to downstream performance in most cases. Formally, AMD is defined as the amortized marginal difference over BPC (See Eq.2 for

more explanations). Although it is hard to explain this connection based on our current knowledge, this feature can still be empirically used to to search for the optimal vocabulary or guide vocabulary learning. Here we focus more on the learning setting, aimed at exploring whether there exists an efficient and promising AVL solution.

Motivated by our findings, we propose a novel two-step discrete optimization objective to learn a vocabulary given a corpus and an optimal transport solution, which can efficiently find a well-performing vocabulary with a much smaller size. Since AMD is a contrastive feature and hard to formulate, we re-formulate it into a two-step discrete optimization problem. The crucial part of the proposed objective is to find a vocabulary with the highest BPC. However, it is intractable to find such vocabulary due to the extensive vocabulary space. We re-formulate this part into an optimal transport (OT) problem, which, therefore, can be solved in polynomial time by linear programming. To be specific, we can imagine vocabulary construction as a transport process that transports chars into token candidates. The number of chars is fixed, and different transport choices result in different vocabularies, of course with different costs. The target of OT is to find a transport matrix to minimize the transfer cost, i.e., negative BPC in our setting. We implement an entropy-based sinkhorn algorithm to solve the OT problem.

We evaluate our approaches on multiple machine translation tasks, including WMT-14 English-German translation, WMT-16 English-Romanian translation, and TED translation. Empirical results show that our approach can beat widely-used vocabulary construction methods, even with over 80% vocabulary size reduction.

2 Related Work

With the development of deep learning, neural networks have achieved state-of-the-art results on natural language processing tasks. Initially, most neural models are built upon word-level vocabularies (Vaswani et al., 2017; Costa-jussà & Fonollosa, 2016; Zhao et al., 2019). While achieving state-of-the-art results, it is a common constraint that these models fail on handling rare words under limited vocabulary size.

To address this problem, researchers have proposed several advanced vocabularization approaches, like byte-level approaches (Wang et al., 2020), character-level approaches (Costa-jussà & Fonollosa, 2016; Lee et al., 2017; Al-Rfou et al., 2019), and subword-level approaches (Sennrich et al., 2016; Kudo & Richardson, 2018). Costa-jussà & Fonollosa (2016) propose a character-level vocabulary that adopts single characters as the minimum semantic unit. The surprisingly good performance brings new insights on token granularity. Following this work, Byte-Pair Encoding (BPE) (Sennrich et al., 2016) is proposed to get subword-level vocabularies. The general idea is to merge pairs of frequent character sequences to create subword units. Subword-level vocabularies can be regarded as a trade-off between character-level vocabularies and word-level vocabularies. Compared to word-level vocabularies, it can decrease the sparsity of tokens and increase the shared features between similar words, which probably have similar semantic meanings, like "happy" and "happier". Compared to character-level vocabularies, it has shorter sentence lengths without rare words. Following BPE, some variants recently have been proposed, like BPE-dropout (Provilkov et al., 2020), SentencePiece (Kudo & Richardson, 2018), and so on.

Despite promising results, these subword-level approaches still require expensive computation costs to tune vocabulary size. More recently, some best-practice studies notice this problem and propose some practical solutions (Chen et al., 2019; Salesky et al., 2020). Most of these approaches treat vocabulary size as a special hyper-parameter and propose to use hyper-parameter learning techniques.

Unlike these approaches, this work takes the first step to a unified theory for automatic vocabulary learning. Given a corpus, we propose a discrete optimization objective function and a principled solution based on optimal transport for AVL. Experiments show that our approach is able to find well-performing vocabularies with a much smaller size.

3 Information-Theoretic Perspective of Vocabulary

More and more researchers have recently accepted that information theory and machine learning are the two sides of the same coin, first mentioned by MacKay (2003). Information theory studies the

shortest code-length and uncertainty while machine learning studies how to compress data with low-dimension vector. Also, learning can be regarded as a process of reducing uncertainty. Following this view, many studies recently are proposed to understand current machine learning systems from the perspective of information theory (Saxe et al., 2018; Gabrié et al., 2018; Goldfeld et al., 2019).

In this section, we describe vocabularization from the perspective of information theory. Considering that BPE is the dominant approach to build vocabularies in NLP, this section mainly focuses on BPE-generated vocabularies to explore whether there exist essential features strongly related with the downstream performance. Specifically, we conduct experiments on neural machine translation, a widely-used benchmark task for NLP techniques, in the subsequent analysis.

From Frequency to BPC Almost all existing vocabularies are built upon an information-theoretic concept: frequency. In information theory, frequency is a token-level feature, which describes the information of a single token. To get a full understanding of vocabulary, here we explore several vocabulary-level features, such as entropy, BPC. Entropy is a common feature evaluating the average level of "information", or "uncertainty" inherent in the distribution. It represents the shortest code length to represent all tokens, short for Bits Per Token (BPT). One of the variants of BPT is Bits-Per-Char (BPC), which normalizes BPT with the averaged length of tokens. We argue that BPC is a more fair evaluation feature than BPT, which avoids the effects of token lengths. Given a vocabulary v_T with length T, BPC is computed as:

$$\boldsymbol{B}_{v_T} = -\frac{1}{l_{v_T}} \sum_{t \in v_T} P(t) \log P(t), \tag{1}$$

where $P(t)$ is the probability of token t and l_{v_T} is the average length of tokens in vocabulary v_T.

From BPC to AMD In this work, we find a valuable feature, AMD-BPC, short for AMD, which is related with downstream performance. Formally, AMD is defined as the amortized marginal difference over BPC, which is normalized by merging operation size in BPE:

$$\boldsymbol{D}_{v_k \to v_{k+m}} = -\frac{\boldsymbol{B}(v_{k+m}) - \boldsymbol{B}(v_k)}{m}, \tag{2}$$

where $\boldsymbol{D}_{v_k \to v_{k+m}}$ represents AMD, m is the increased operation size and v_k, v_{k+m} are vocabularies generated by k, $k+m$ merging operations, respectively. Imagine a vocabulary search policy that incrementally increases the number of merging operation. AMD is a dynamic feature that describes how information changes with the increase of vocabulary size.

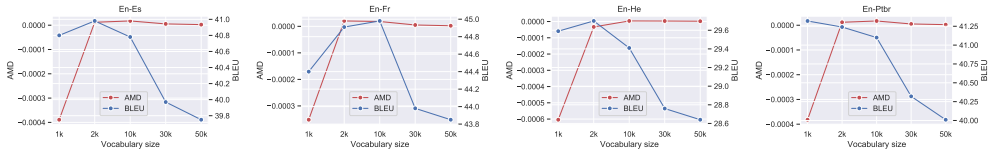

Figure 1: The relation between AMD and downstream performance. Vocabularies with the highest AMD usually have better BLEU scores.

New Finding: AMD indicates Model Performance To evaluate the relationship between AMD and downstream performance, we conduct experiments on 12 language pairs. Experiment settings can be found at Section 5. We sample 4 language-pairs, and the results are listed in Figure 1. The full results on 12 language-pairs can be found in Appendix A. As we can see, vocabularies with the highest AMD usually bring higher BLEU scores in most cases. Based on our full results, 42.8% percent of language-pairs with the highest AMD achieve the best BLEU scores. Moreover, 35.7% percent of language-pairs with the highest AMD almost achieve the best BLEU scores (gap less than 0.3).

Based on this finding, we have two natural choices to get the final vocabulary: search and learning. In the search direction, vocabularies can be obtained by searching a serial of vocabularies with different merging operation sizes. While being simple, the number of explored vocabularies is limited due to the expensive cost to get vocabularies from merging operations, which depends on the scale of

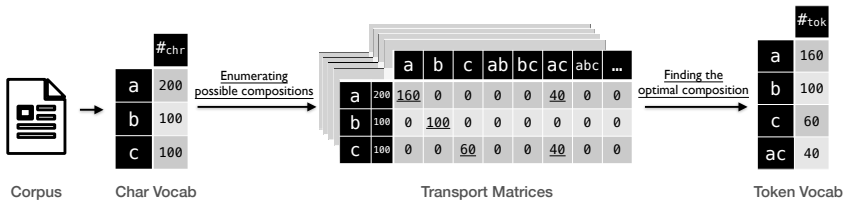

Figure 2: An illustration of vocabulary construction from a transport view. Given a corpus, we can calculate a char distribution and a token candidate distribution. A vocabulary can be built upon a transport matrix deciding how much chars are transported to different tokens.

training data. By contrast, the learning direction has a higher performance ceiling. In this work, we take a first step in the learning direction to explore "how far can an information-theoretic learning approach can reach".

4 Our Proposed Approach: *Info*-VOT

4.1 Objective Formulation

We formulate vocabulary construction as an optimization problem whose target is to find a vocabulary with the highest AMD based on Eq. 2. Since AMD is a dynamic feature depending on a margin difference, we also formulate a dynamic process here. Given an incremental integer sequence $\boldsymbol{S} = \{1, 1+i, 1+2\cdot i, ..., 1+(t-1)\cdot i, \cdots\}$ where $1+(t-1)\cdot i$ is the upper bound of vocabulary size at t-th timestep. With seqence $\boldsymbol{S}$, Eq. 2 can be rewritten as:

$$\underset{v(t-1)\in\mathbb{V}_{\boldsymbol{S}[t-1]}, v(t)\in\mathbb{V}_{\boldsymbol{S}[t]}}{\arg\max} \boldsymbol{D}_{v(t-1)\to v(t)} = -\frac{1}{i}\big[\boldsymbol{B}(v(t)) - \boldsymbol{B}(v(t-1))\big] \tag{3}$$

where $\mathbb{V}_{\boldsymbol{M}[t-1]}$ and $\mathbb{V}_{\boldsymbol{M}[t]}$ are two sets containing all vocabularies with upper bound of size $\boldsymbol{M}[t-1]$ and $\boldsymbol{M}[t]$. Since it is intractable to directly solve this problem, we propose to optimize the surrogated loss which is the lower bound of Eq. 3:

$$\arg\max_{t} -\frac{1}{i}\big[\max_{v(t)\in\mathbb{V}_{\boldsymbol{S}[t]}} \boldsymbol{B}(v(t)) \; - \max_{v(t-1)\in\mathbb{V}_{\boldsymbol{S}[t-1]}} \boldsymbol{B}(v(t-1))\big] \tag{4}$$

Based on this equation, the whole solution is split into two steps: search for the optimal vocabulary with the highest BPC at each timestep t; 2) enumerate all timesteps and output the vocabulary satisfying Eq. 4.

4.2 Maximization of BPC

The first step of our approach is to search for a vocabulary with the highest BPC given an upper bound of vocabulary size $\boldsymbol{S}[t]$. Formally, the goal is to find a vocabulary $v(t)$ such that BPC is maximized,

$$\underset{v(t)\in\mathbb{V}_{\boldsymbol{S}[t]}}{\arg\max} -\frac{1}{l_{v(t)}} \sum_{x\in v(t)} P(x)\log P(x), \tag{5}$$

where l_v is the average length for tokens in $v(t)$, $P(x)$ is the probability of token x. However, notice that this problem is in general intractable due to the extensive vocabulary size. Therefore, we instead propose a relaxation in the formulation of discrete Optimal Transport, which can then be solved efficiently via the well-known Sinkhorn algorithm (Cuturi, 2013).

To be specific, vocabulary construction can be viewed as a transport process that transfers char distributions into token distributions. Given two sets of chars and tokens, we can define a transport matrix with each item (i, j) deciding how much chars are transported from char i to token j. Since the number of chars is limited, and not all token candidates can get enough chars, different transport metrics result in different vocabularies and different costs. Figure 2 illustrates an example to understand this process. The objective function is to find a transport matrix with the lowest costs. In the

next part, we will show the definition of optimal transport problem and the reformulation of Eq. 5 with an optimal transport objective.

More precisely, given a cost matrix $\boldsymbol{C} \in \mathbb{R}^{|\mathcal{C}|\times|\mathcal{T}|}$ and two discrete distributions: char distribution $\mathcal{C}$ and token distribution $\mathcal{T}$, $\{a_i\}_{i=1}^{|\mathcal{C}|}, \{b_j\}_{j=1}^{|\mathcal{T}|}$ are the corresponding probability mass. The discrete OT considers the following optimization problem

$$\min_{\boldsymbol{A}\in\mathbb{R}^{m\times n}} \langle \boldsymbol{A}, \boldsymbol{C}\rangle, \quad \text{s.t.} \quad \boldsymbol{A}\cdot\mathbf{1}_n = \vec{a}, \quad \boldsymbol{A}^{\mathsf{T}}\cdot\mathbf{1}_m = \vec{b}, \tag{6}$$

where $\boldsymbol{A}$ is the transport matrix. Intuitively, optimal transport is about finding the best plan of transporting mass from the source distribution $\mathcal{C}$ to the target distribution $\mathcal{T}$ with the minimum work defined by $\langle \boldsymbol{A}, \boldsymbol{C}\rangle$. Since the original OT problem is a linear programming which requires $O(N^3 \log N)$ time complexity to solve, Cuturi (2013) proposed to add an entropy regularization term to accelerate the convergence . More precisely, the objective function for entropy regularized OT is

$$\min_{\boldsymbol{A}\in\mathbb{R}^{m\times n}} \langle \boldsymbol{A}, \boldsymbol{C}\rangle - \gamma H(\boldsymbol{A}), \quad \text{s.t.} \quad \boldsymbol{A}\cdot\mathbf{1}_n = \vec{a}, \quad \boldsymbol{A}^{\mathsf{T}}\cdot\mathbf{1}_m = \vec{b}, \tag{7}$$

where $H(\boldsymbol{A}) = -\sum A_{ij}\log A_{ij}$ is the entropy term. The entropy regularization makes the problem convex. Moreover, there is an efficient algorithm, the Sinkhorn algorithm, that allows us to solve the problem in nearly linear time.

4.3 Vocabulary Optimal Transport: Objective Formulation

A tractable lower bound of BPC Given an upper bound of the vocabulary size $\boldsymbol{S}$, we want to find a joint probability distribution of chars and candidate tokens in order to maximize BPC. Let $\mathbb{V}_{\boldsymbol{S}}$ be the set containing all vocabularies with size $\boldsymbol{S}$. Consequently, the objective function in Eq. 5 becomes

$$\min_{v\in\mathbb{V}_{\boldsymbol{S}}} \frac{1}{l_v}\sum_{i\in v} P(i)\log P(i),$$
$$\text{s.t.} \quad P(i) = \frac{\text{Token}(i)}{\sum_{i\in v}\text{Token}(i)}, \; l_v = \frac{\sum_{i\in v} len(i)}{|v|}$$

where $\text{Token}(i)$ is the frequency of token i in the vocabulary v. Notice that both the distribution $P(i)$ and the average length l_v depend on the choice of $v \in \mathbb{V}_{\boldsymbol{S}}$.

To obtain a tractable lower bound of BPC, it suffices to give a tractable upper bound of the above objective function. To this end, let $\mathbb{T} \in \mathbb{V}_{\boldsymbol{S}}$ be the vocabulary containing top S most frequent tokens, $\mathbb{C}$ be the set of chars and $|\mathbb{T}|, |\mathbb{C}|$ be their sizes respectively. Clearly, we have

$$\min_{v\in\mathbb{V}_{\boldsymbol{S}}} \frac{1}{l_v}\sum_{i\in v} P(i)\log P(i) \le \frac{1}{l_{\mathbb{T}}}\sum_{i\in\mathbb{T}} P(i)\log P(i). \tag{8}$$

Let $P(i,j)$ be the joint probability distribution of the tokens and chars that we want to learn. Then we have

$$\begin{aligned}\sum_{i\in\mathbb{T}} P(i)\log P(i) &= \sum_{i\in\mathbb{T}}\sum_{j\in\mathbb{C}} P(i,j)\log P(i) \\ &= \underbrace{\sum_{i\in\mathbb{T}}\sum_{j\in\mathbb{C}} P(i,j)\log P(i,j)}_{\mathcal{L}_1} + \underbrace{\sum_{i\in\mathbb{T}}\sum_{j\in\mathbb{C}} P(i,j)(-\log P(j|i))}_{\mathcal{L}_2}.\end{aligned} \tag{9}$$

The details of proof can be found at Appendix B.

Since $\mathcal{L}_1$ is nothing but the negative entropy of the joint probability distribution $P(i,j)$, we shall denote it as $-H(P)$. Let $\boldsymbol{C}$ be the $|\mathbb{C}| \times |\mathbb{T}|$ matrix whose (i,j)-th entry is given by $-\log P(j|i)$, then we can write

$$\mathcal{L}_2 = \langle \boldsymbol{P}, \boldsymbol{C}\rangle$$

where $C_{ij} = -\log P(j|i) = +\infty$ if $j \notin i$ and $-\log\frac{\#c\in t}{len(t)}$ otherwise.

Note that we have the hard constraints $\sum_j P(i,j) = P(i)$ and $\sum_i P(i,j) = P(j)$ where $P(i), P(j)$ are the char distribution and candidate token distribution of $\mathbb{T}$, respectively. However, in order to obtain a refined token set from $\mathbb{T}$ with larger BPC, we need to relax the hard constraint on the token distribution matching to a soft constraint. This formulation then allows us to drop out tokens with low joint probability distribution. See the discussion at the end of this section for more implementation details. In summary, our final objective function is

$$\begin{aligned} &\underset{\boldsymbol{P}\in\mathbb{R}^{|\mathbb{C}|\times|\mathbb{T}|}}{\arg\min} \; -H(\boldsymbol{P}) + \langle \boldsymbol{P}, \boldsymbol{C} \rangle \,, \\ \text{s.t.} \quad &\sum_i \boldsymbol{P}(i,j) = P(j), \quad |\sum_j \boldsymbol{P}(i,j) - P(i)| \le \epsilon, \quad \forall i, j. \end{aligned}$$

with small $\epsilon > 0$.

Notice that the objective function has the same form as the entropy regularized OT Eq. 7 with $\gamma = 1$ except for the soft constraint on the token distribution matching. Strictly speaking, this is an unbalanced entropy regularized Optimal Transport problem. Nonetheless, we can still use the generalized Sinkhorn algorithm to efficiently find the target vocabulary as detailed in Section 4.6 of Peyré & Cuturi (2020).

Enumerate Elements in S for Optimal Vocabulary Due to the large space of token candidates, here we adopt BPE generated tokens with a large merging size (100K in implementation) as target token candidates. Each merging action generates a new token. We range all tokens based on the generated rank. In this way, we can get a sequence of tokens associated with their probabilities $\boldsymbol{S}_{bpe}$. For $\mathbb{V}_{\boldsymbol{S}[t]}$, we use $\boldsymbol{S}_{bpe}[:\boldsymbol{S}[t]]$ as the target token candidate distribution. For char distribution, we use char probability in the raw text.

At each timestep, by taking char distribution and token candidate distribution $\boldsymbol{S}_{bpe}[:\boldsymbol{S}[t]]$ as input, we can get the maximum BPC value based on our OT objective function. Then, based on BPC values in different timesteps, the optimal AMD score can be obtained via Eq. 4. Finally, according to the transport matrix at the timestep with the highest AMD, we generate the final vocabulary via a post-processing pipeline. It is inevitable to handle illegal transport case in the transport matrix. We remove tokens with distributed chars less than one-tenth token frequencies, and the rest tokens are combined into the final vocabulary.

5 EXPERIMENTS

To evaluate the performance of *Info*-VOT, we conduct experiments on three datasets, including WMT-14 English-German translation, WMT-16 English-Romania translation, and TED dataset.

5.1 SETTINGS

We run experiments on three bilingual machine translation datasets.

1. WMT-14 English-German (En-De) dataset: This dataset has 4.5M sentence pairs. The dataset is processed following Ott et al. (2018). We choose newstest14 as the test set.
2. WMT-16 English-Romanian (En-Ro) dataset: It has 2.2M training sentence pairs. We use the officially released validation and test sets. We adopt the same pre-processing pipeline in the WMT14 En-De dataset.
3. TED dataset: We include two settings: many-to-English multilingual translation and English-to-many multilingual translation. We choose 12 language-pairs with the most training corpus. We list the language code according to ISO-639-1 standard [1] for the languages used in our experiments in Appendix C. We use the officially released validation and test sets. We adopt the same pre-processing pipeline in the WMT-14 En-De dataset.

[1] http://www.lingoes.net/en/translator/langcode.htm

Table 1: Comparison between *Info*-VOT and widely-used BPE Vocabularies. * means WMT translation results and the rest columns are TED results. *Info*-VOT can achieve competitive BLEU scores with a large size reduction, even over 80% on TED.

En-X	**De***	**Ro***	**Es**	**PTbr**	**Fr**	**Ru**	**He**	**Ar**	**Ko**	**It**	**Nl**	**Ro**	**Tr**	**De**
BPE-10K	29.84	36.60	40.78	41.10	44.98	20.90	29.41	18.83	10.94	37.43	33.84	29.65	19.75	30.61
BPE-20K	29.70	36.50	40.27	40.96	44.65	20.57	29.03	18.46	10.50	36.90	33.34	29.06	19.75	30.61
BPE-30K	29.51	36.90	39.97	40.32	43.98	20.11	28.76	18.25	10.40	36.88	33.42	28.50	19.12	30.28
***Info*-VOT**	**30.00**	**36.60**	**40.75**	**41.65**	**45.02**	**20.38**	**29.82**	**18.65**	**10.41**	**37.24**	**33.83**	**29.56**	**20.06**	**31.52**
X-En	**De***	**Ro***	**Es**	**PTbr**	**Fr**	**Ru**	**He**	**Ar**	**Ko**	**It**	**Nl**	**Ro**	**Tr**	**De**
BPE-10K	-	-	45.11	47.21	43.12	29.10	41.47	35.60	22.66	41.96	40.33	38.44	29.97	39.30
BPE-20K	-	-	44.44	47.05	43.05	29.32	40.38	34.12	22.35	41.22	39.57	38.29	29.73	38.52
BPE-30K	-	-	44.37	47.08	42.70	28.21	39.93	33.83	22.20	41.44	39.43	37.65	28.89	38.91
***Info*-VOT**	-	-	**45.47**	**47.72**	**43.49**	**28.78**	**41.31**	**35.01**	**22.78**	**41.67**	**39.80**	**39.15**	**30.20**	**39.95**
Size (K)	**De***	**Ro***	**Es**	**PTbr**	**Fr**	**Ru**	**He**	**Ar**	**Ko**	**It**	**Nl**	**Ro**	**Tr**	**De**
BPE-10K	13.6	10.3	10.3	10.2	10.2	10.3	10.2	10.4	13.7	18.2	10.2	10.2	10.2	10.2
BPE-20K	23.6	20.2	20.1	20.1	20.1	20.2	20.2	20.4	23.6	20.1	20.0	20.1	20.1	20.1
BPE-30K	33.6	30.0	29.9	29.8	29.8	30.1	30.0	30.3	33.5	29.8	29.8	29.9	30.0	29.9
***Info*-VOT**	**8.5**	**1.8**	**1.9**	**1.7**	**1.5**	**1.7**	**1.5**	**1.7**	**5.2**	**1.9**	**2.0**	**1.9**	**1.8**	**1.7**

Models We use Fairseq to train a Transformer-big model with the same setting in the original paper (Ott et al., 2018). The input embedding and output embeddings are shared. We use the Adam optimizer (Kingma & Ba, 2015) with a learning rate 0.0005 and an inverse_sqrt decay schedule. The warm-up step is $4,000$, the dropout rate is 0.3, the update frequency is 8, the number of tokens is $9,600$, or $4,800$ in a single batch.

Training and Evaluation We run WMT-14 En-De experiments with 4 GPUs, WMT-16 En-Ro with 1 GPU, TED bilingual translation with 1 GPU. We set a beamwidth to 4 for En-De and 5 for the other. We average the last five models on all datasets and use the averaged model to generate translation results. We calculate case-sensitive tokenized BLEU for evaluation except for En-Ro which adopts tokenized SacreBLEU.

6 Results and Analysis

Vocabularies Searched by *Info*-VOT are Much Better than Widely-used BPE Vocabularies. We first compare our methods with widely-used BPE variants: BPE-10K, BPE-20K, BPE-30K. 10K, 20K, 30K represent the size of the merging operation. It needs to notice that BPE-30K is the dominant choice. The results are listed in Table 1. As we can see, *Info*-VOT achieves competitive BLEU scores with a much smaller vocabulary than all BPE variants. Compared to dominant BPE-30K, *Info*-VOT brings significant performance gains with over 80% percent vocabulary size reduction on TED. The promising results demonstrate that *Info*-VOT is a practical approach that can find a well-performing vocabulary with better BLEU scores and less vocabulary size.

A Simple Baseline with *Info*-VOT-generated Vocabularies Beats Existing Strong Approaches. We implement *Info*-VOT on a widely-used baseline, Transformer-big. Here we are curious about how much a baseline can reach only with the change of vocabulary. We compare *Info*-VOT and several strong approaches on WMT-14 En-De dataset. Table 2 shows surprisingly good results of our method. Compared to existing approaches in the top block, *Info*-VOT achieves almost the best performance with a much smaller vocabulary. These results demonstrate that a simple baseline can achieve good results with a well-defined vocabulary.

***Info*-VOT Beats SentencePiece and WordPiece.** SentencePiece and WordPiece are two variants of subword vocabularies. We also compare our approach with them on WMT-14 En-De dataset to evaluate the effectiveness of *Info*-VOT. The middle block of Table 2 lists the results of SentenPiece and WordPiece. We implement these two approaches with the default setting, with 32K vocabulary

Table 2: Comparison between *Info*-VOT and strong baselines. *Info*-VOT achieves almost the best performance with a much smaller vocabulary.

WMT-14 En-De	**BLEU**	**Merging Size**	**Vocabulary Size**	**# Parameters**
Vaswani et al. (2017)	28.4	32K	33.6K	210M
Shaw et al. (2018)	29.2	32K	33.6K	213M
Ott et al. (2018)	29.3	32K	33.6K	210M
So et al. (2019)	29.8	32K	33.6K	218M
Liu et al. (2020)	30.1	32K	33.6K	256M
SentencePiece	28.7	32K	33.6K	210M
WordPiece	29.0	32K	33.6K	210M
***Info*-VOT**	**30.0**	**8.5K**	**8.7K**	**188M**

Table 3: Comparison between *Info*-VOT and AMD-Search. AMD-Search randomly samples 10 BPE-variants, size varying from 1K to 50K.

En-X	**Es**	**PTbr**	**Fr**	**Ru**	**He**	**Ar**	**Ko**	**It**	**Nl**	**Ro**	**Tr**	**De**	**Time (min)**
AMD-Search	**40.98**	41.24	44.91	**20.57**	29.41	18.46	**10.94**	**37.50**	**34.05**	29.06	19.75	31.36	340
Info-VOT	40.75	**41.65**	**45.02**	20.38	**29.82**	**18.65**	10.41	37.24	33.83	**29.56**	**20.06**	**31.52**	95

size. We can observe that *Info*-VOT outperforms SentencePiece and WordPiece by a large margin, with over 1 BLEU score improvements.

Comparison between *Info*-VOT and *Info*-VOT-based Rules Our method is based on an exciting finding that AMD, an information-theoretic feature, indicates downstream performance. In addition to guiding vocabulary learning, this finding can also be used in a heuristic manner. We can randomly search several vocabularies as candidates and take the vocabulary with the highest AMD as the final vocabulary. We name this baseline AMD-Search in the subsequent analysis. The vocabularies generated by the two approaches are similar, around 2K in *Info*-VOT, and 5K in AMD-Search. As Table 3 shows, *Info*-VOT and AMD-Search have similar BLEU scores, but *Info*-VOT takes fewer minutes, demonstrating the effectiveness of our solution on optimizing AMD. Despite the better performance of *Info*-VOT, we argue that AMD-Search is also a good information-theoretic solution, which outperforms widely-used vocabularies.

In this work, we start from the connection between AMD and downstream performance and aim to build a unified information-theoretic framework for automatic vocabulary learning. AMD-Search is one of the solutions in this framework. We do not stop at this simple solution and further explore a more challenging and promising direction: learning-based vocabularization. *Info*-VOT is a preliminary exploration approach. We believe that idea of *Info*-VOT can motivate more researchers for further exploration in the future.

7 Conclusion

In this work, we propose a unified information-theoretic vocabulary learning framework. The whole framework starts from an exciting finding that AMD, an information-theoretic feature, is related to model performance. Based on this finding, we design a two-step discrete optimization objective and a principled optimal transport solution: *Info*-VOT. Experiments show that *Info*-VOT is an effective approach that can quickly find a well-performing vocabulary with a much smaller size. The searched vocabulary outperforms current widely-used vocabularies, as well as several strong network variants.

Although *Info*-VOT can find a well-performing vocabulary, it still has several limitations that need to be improved in future work. First, *Info*-VOT relies on the initialized token distribution. For simplification, we directly adopt BPE-100K generated tokens associated with their probabilities as initialization. The sensitivity to initialized token distributions should be considered in the future. Second, the transport matrix in *Info*-VOT still needs an additional post-processing pipeline to get the final vocabulary. Although we give a recommended post-processing setting, we believe that an advanced algorithm in the future can reduce the dependency on post-processing.

REFERENCES

Rami Al-Rfou, Dokook Choe, Noah Constant, Mandy Guo, and Llion Jones. Character-level language modeling with deeper self-attention. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pp. 3159–3166. AAAI Press, 2019.

Wenhu Chen, Yu Su, Yilin Shen, Zhiyu Chen, Xifeng Yan, and William Yang Wang. How large a vocabulary does text classification need? A variational approach to vocabulary selection. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 3487–3497. Association for Computational Linguistics, 2019.

Marta R. Costa-jussà and José A. R. Fonollosa. Character-based neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer Linguistics, 2016.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pp. 2292–2300, 2013.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 4171–4186. Association for Computational Linguistics, 2019.

Shuoyang Ding, Adithya Renduchintala, and Kevin Duh. A call for prudent choice of subword merge operations in neural machine translation. In Mikel L. Forcada, Andy Way, Barry Haddow, and Rico Sennrich (eds.), *Proceedings of Machine Translation Summit XVII Volume 1: Research Track, MTSummit 2019, Dublin, Ireland, August 19-23, 2019*, pp. 204–213. European Association for Machine Translation, 2019.

Marylou Gabrié, Andre Manoel, Clément Luneau, Jean Barbier, Nicolas Macris, Florent Krzakala, and Lenka Zdeborová. Entropy and mutual information in models of deep neural networks. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pp. 1826–1836, 2018.

Sebastian Gehrmann, Yuntian Deng, and Alexander M. Rush. Bottom-up abstractive summarization. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pp. 4098–4109. Association for Computational Linguistics, 2018.

Ziv Goldfeld, Ewout van den Berg, Kristjan H. Greenewald, Igor Melnyk, Nam Nguyen, Brian Kingsbury, and Yury Polyanskiy. Estimating information flow in deep neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2299–2308. PMLR, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Eduardo Blanco and Wei Lu (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pp. 66–71. Association for Computational Linguistics, 2018.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5: 365–378, 2017.

Xiaodong Liu, Kevin Duh, Liyuan Liu, and Jianfeng Gao. Very deep transformers for neural machine translation. *CoRR*, abs/2008.07772, 2020.

David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pp. 3111–3119, 2013.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. Scaling neural machine translation. In Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno-Yepes, Philipp Koehn, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana L. Neves, Matt Post, Lucia Specia, Marco Turchi, and Karin Verspoor (eds.), *Proceedings of the Third Conference on Machine Translation: Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pp. 1–9. Association for Computational Linguistics, 2018.

Gabriel Peyré and Marco Cuturi. Computational optimal transport, 2020.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. Bpe-dropout: Simple and effective subword regularization. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 1882–1892. Association for Computational Linguistics, 2020.

Elizabeth Salesky, Andrew Runge, Alex Coda, Jan Niehues, and Graham Neubig. Optimizing segmentation granularity for neural machine translation. *Machine Translation*, pp. 1–19, 2020.

Andrew M. Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan D. Tracey, and David D. Cox. On the information bottleneck theory of deep learning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In Marilyn A. Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pp. 464–468. Association for Computational Linguistics, 2018.

David R. So, Quoc V. Le, and Chen Liang. The evolved transformer. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5877–5886. PMLR, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pp. 5998–6008, 2017.

Changhan Wang, Kyunghyun Cho, and Jiatao Gu. Neural machine translation with byte-level subwords. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 9154–9160. AAAI Press, 2020.

Lei Zhang, Shuai Wang, and Bing Liu. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253, 2018.

Yi Zhao, Yanyan Shen, and Junjie Yao. Recurrent neural network for text classification with hierarchical multiscale dense connections. In Sarit Kraus (ed.), *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pp. 5450–5456. ijcai.org, 2019.