

# The Devil is in the Details: On the Pitfalls of Vocabulary Selection in Neural Machine Translation

Tobias Domhan\* Eva Hasler\* Ke Tran Sony Trenous Bill Byrne Felix Hieber

Amazon AI Translate

Berlin, Germany

{domhant, ehasler, trnke, trenous, willbyrn, fhieber}@amazon.com

## Abstract

Vocabulary selection, or lexical shortlisting, is a well-known technique to improve latency of Neural Machine Translation models by constraining the set of allowed output words during inference. The chosen set is typically determined by separately trained alignment model parameters, independent of the source-sentence context at inference time. While vocabulary selection appears competitive with respect to automatic quality metrics in prior work, we show that it can fail to select the right set of output words, particularly for semantically non-compositional linguistic phenomena such as idiomatic expressions, leading to reduced translation quality as perceived by humans. Trading off latency for quality by increasing the size of the allowed set is often not an option in real-world scenarios. We propose a model of vocabulary selection, integrated into the neural translation model, that predicts the set of allowed output words from contextualized encoder representations. This restores translation quality of an unconstrained system, as measured by human evaluations on WMT newstest2020 and idiomatic expressions, at an inference latency competitive with alignment-based selection using aggressive thresholds, thereby removing the dependency on separately trained alignment models.

## 1 Introduction

Neural Machine Translation (NMT) has achieved great improvements in translation quality, largely thanks to the introduction of Transformer models (Vaswani et al., 2017). However, increasingly larger models (Aharoni et al., 2019; Arivazhagan et al., 2019) lead to prohibitively slow inference when deployed in industrial settings. Especially for real-time applications, low latency is key. A number of inference optimization speed-ups have been proposed and are used in practice:

\*Equal contributions.

EN: to swal\_\_ low the bitter pill  
DE: in den sau\_\_ ren **ap\_\_fel** bei\_\_ ßen  
GL: to **bite** into the sour **ap ple**

EN: by ho\_\_ ok or cro\_\_ ok  
DE: auf **biegen** und *brechen*  
GL: by **bending** and *breaking*

EN: to buy a p\_\_ ig in a po\_\_ ke  
DE: die **kat\_\_ze** im *sack* kaufen  
GL: to buy the **cat** in the *bag*

EN: to swe\_\_ at blood  
DE: blu\_\_ t und **wasser** sch\_\_ wit\_\_ zen  
GL: to sweat blood and **water**

EN: make yourself at home !  
DE: machen sie es sich **bequem** !  
GL: make yourself **comfortable** !

Figure 1: Examples of subword-segmented idiomatic expressions (EN) and their German correspondences (DE) as well as an English gloss (GL) of the German expression. Alignment-based vocabulary selection: output tokens missing from the allowed set of top-k output tokens are marked in orange/bold (red/italic) for k=200 (k=1000).

reduced precision (Aji and Heafield, 2020), replacing self-attention with Average Attention Networks (AANs) (Zhang et al., 2018), Simpler Simple Recurrent Units (SSRUs) (Kim et al., 2019), or model pruning (Behnke and Heafield, 2020; Behnke et al., 2021).

Another technique that is very common in practice is vocabulary selection (Jean et al., 2015) which usually provides a good tradeoff between latency and automatic metric scores (BLEU) and reduced inference cost is often preferred over the loss of  $\sim 0.1$  BLEU. Vocabulary selection is effective because latency is dominated by expensive, repeated decoder steps, where the final projection to the output vocabulary size contributes to a large portion of time spent (Bérard et al., 2021). Despite

high parallelization in GPUs, vocabulary selection is still relevant for GPU inference for state-of-the-art models.

However, we show that standard methods of vocabulary selection based on alignment model dictionaries lead to quality degradations not sufficiently captured by automatic metrics such as BLEU. We demonstrate that this is particularly true for semantically non-compositional linguistic phenomena such as idiomatic expressions, and aggressive thresholds for vocabulary selection. For example, see Figure 1 for alignment-model based vocabulary selection failing to include tokens crucial for translating idiomatic expressions in the set of allowed output words. While less aggressive thresholds can reduce the observed quality issues, it also reduces the desired latency benefit. In this paper we propose a neural vocabulary selection model that is jointly trained with the translation model and achieves translation quality at the level of an unconstrained baseline with latency at the level of an aggressively thresholded alignment-based vocabulary selection model.

Our contributions are as follows:

- We demonstrate that alignment-based vocabulary selection is not limited by alignment model quality, but rather inherently by making target word predictions out of context (§2).
- We propose a Neural Vocabulary Selection (NVS) model based on the contextualized deep encoder representation (§3).
- We show that alignment-based vocabulary selection leads to human-perceived translation quality drops not sufficiently captured by automatic metrics and that our proposed model can match an unconstrained model’s quality while keeping the latency benefits of vocabulary selection (§4).

## 2 Pitfalls of vocabulary selection

We first describe vocabulary selection and then analyze its shortcomings. Throughout the paper, we use the recall of unique target sentence tokens as a proxy for measuring vocabulary selection quality, i.e. the reachability of the optimal translation. We use the average vocabulary size in inference decoder steps across sentences as a proxy for translation latency since it directly impacts decoding speed (Kasai et al., 2020).

Scope		$k = 200$		$k = 1000$	
		model	ref.	model	ref.
EN-DE	fast_align	99.6	97.5	99.9	99.7
	GIZA++	99.6	97.8	100.0	99.7
	MaskAlign	96.9	93.1	99.6	98.6
EN-RU	fast_align	98.7	93.8	99.9	98.9
	GIZA++	98.6	94.2	99.9	99.2
	MaskAlign	94.2	87.2	99.1	96.7

Table 1: Recall of unique reference and model tokens on newstest2020 with word probability tables extracted from different alignment models.

### 2.1 Vocabulary selection

Vocabulary selection (Jean et al., 2015), also known as lexical shortlisting or candidate selection, is a common technique for speeding up inference in sequence-to-sequence models, where the repeated computation of the softmax over the output vocabulary  $\mathcal{V}$  of size  $V$  incurs high computational cost in the next word prediction at inference time:

$$p(y_t|y_{1:t-1}, x; \theta) = \text{softmax}(\mathbf{W}\mathbf{h} + \mathbf{b}), \quad (1)$$

where  $\mathbf{W} \in \mathbb{R}^{V \times d}$ ,  $\mathbf{b} \in \mathbb{R}^V$  and  $\mathbf{h} \in \mathbb{R}^d$ ,  $d$  being the hidden size of the network. Vocabulary selection chooses a subset  $\bar{\mathcal{V}} \subset \mathcal{V}$ , with  $\bar{V} \ll V$ , to reduce the size of matrix multiplication in Equation (1) such that

$$\bar{p}(y_t|y_{1:t-1}, x; \theta) = \text{softmax}(\bar{\mathbf{W}}\mathbf{h} + \bar{\mathbf{b}}), \quad (2)$$

where  $\bar{\mathbf{W}} \in \mathbb{R}^{\bar{V} \times d}$  and  $\bar{\mathbf{b}} \in \mathbb{R}^{\bar{V}}$ . The subset  $\bar{\mathcal{V}}$  is typically chosen to be the union of the top- $k$  target word translations for each source token, according to the word translation probabilities of a separately trained word alignment model (Jean et al., 2015; Shi and Knight, 2017). Decoding with vocabulary selection usually yields similar scores according to automatic metrics, such as BLEU (Papineni et al., 2002), compared to unrestricted decoding but at reduced latency (L’Hostis et al., 2016; Mi et al., 2016; Sankaran et al., 2017; Junczys-Dowmunt et al., 2018). In the following, we show that despite its generally solid performance, vocabulary selection based on word alignment models negatively affects translation quality, not captured by standard automatic metrics. We use models trained on WMT20 (Barrault et al., 2020a) data for all evaluations in this section, see Section 4.1 for details.

### 2.2 Alignment model quality

In practice, the chosen subset of allowed output words is often determined by an alignment

model, such as `fast_align` (Dyer et al., 2013), which provides a trade-off between the speed of alignment model training and the quality of alignments (Jean et al., 2015; Junczys-Dowmunt et al., 2018). `fast_align`’s reparametrization of IBM model 2 (Brown et al., 1993) places a strong prior for alignments along the diagonal. We investigate whether more sophisticated alignment models can lead to better vocabulary selection, especially for language pairs with high amount of reordering. To evaluate this we compute the recall of translation model and reference tokens using GIZA++ (Och and Ney, 2003) and MaskAlign<sup>1</sup> (Chen et al., 2021) as seen in Table 1. We extract top-k word translation tables (from `fast_align`, GIZA++, and MaskAlign) by force-aligning the training data. Overall, GIZA++ achieves the best recall, and it is just slightly better than `fast_align`. MaskAlign, a state-of-the-art neural alignment model, underperforms `fast_align` with respect to recall. While performance of MaskAlign may be improved with careful tuning of its hyperparameters via gold alignments (Chen et al., 2021), we choose `fast_align` as a strong, simple baseline for vocabulary selection in the following.

### 2.3 Out-of-context word selection

Alignment-based vocabulary selection does not take source sentence context into account. A top-k list of translation candidates for a source word will likely cover multiple senses for common words, but may be too limited when a translation is highly dependent on the source context. Here we consider *idiomatic expressions* as a linguistic phenomenon that is highly context-dependent due to its semantically non-compositional nature.

Table 2 compares the recall of tokens in the reference translation when querying the translation lexicon of the alignment model for two different top-k settings. Recall is computed as the percentage of unique tokens in the reference translation that appear in the top-k lexicon, or more generally, in the set of predicted tokens according to a vocabulary selection model. We evaluate two scopes for test sets of idiomatic expressions: the full source and target sentence vs. the source and target idiomatic multiword expressions according to metadata. The Idioms test set is an internal set of 100 English idioms in context and their human translations. ITDS is

<sup>1</sup>With default hyper-parameters from <https://github.com/THUNLP-MT/Mask-Align>

	Scope	$k = 200$		$k = 1000$	
		Idioms	ITDS	Idioms	ITDS
EN-DE	sentence	96.0	91.5	99.1	97.3
	idiom	80.0	58.2	92.8	88.0
DE-EN	sentence	-	92.2	-	98.0
	idiom	-	75.5	-	90.1
EN-RU	sentence	93.0	-	98.7	-
	idiom	65.7	-	85.6	-

Table 2: Recall on internal test set of idioms in context and IdiomTranslationDS test set.

	Scope	Wikiquote	
		$k = 200$	$k = 1000$
EN-DE	literal	96.5	99.2
	idiomatic	74.6	91.0
EN-RU	literal	91.2	99.2
	idiomatic	67.6	88.9

Table 3: Recall on English proverbs and their literal and idiomatic translations sourced from Wikiquote.

the IdiomTranslationDS<sup>2</sup> data released by Fadaee et al. (2018) with 1500 test sentences containing English and German idiomatic expressions for evaluation into and out of German, respectively. The results show that recall increases when increasing  $k$  but is consistently lower for the idiomatic expressions than for full sentences. Clearly, the idiom translations contain tokens that are on average less common than the translations of “regular” inputs. As a consequence, increasing the output vocabulary is less effective for idiom translations, with recall lagging behind by up to 9.3%. This can directly affect translation quality because the NMT model will not be able to produce idiomatic translations given an overly restrictive output vocabulary.

Table 3 shows a similar comparison but here we evaluate full literal translations vs. full idiomatic translations on a data set of English proverbs from Wikiquote<sup>3</sup>. For EN-DE, we extracted 94 triples of English sentence and two references, for EN-RU we extracted 262 triples. Although in both cases recall can be improved by increasing  $k$ , it helps considerably less for idiomatic than for literal translations.

Figure 1 shows examples of idiomatic expressions from the ITDS set and the output tokens belonging to an idiomatic translation that are missing from the respective lexicon used for vocabulary

<sup>2</sup><https://github.com/marziehf/IdiomTranslationDS>

<sup>3</sup><https://github.com/awslabs/socketeye/tree/naacl2022/naacl2022/wikiquote>

	Scope	idioms in context	
		w/o adapt	w/ adapt
EN-DE	sentence	96.0	98.0
	idiom	80.0	97.2
EN-RU	sentence	93.0	96.6
	idiom	65.7	91.7

Table 4: Recall on internal test set of idioms in context, with and without adapting the `fast_align` translation lexicons ( $k = 200$ ).

selection. While for some of the examples, increasing the lexicon size solves the problem, for others the idiomatic translation can still not be generated because of missing output tokens.

These results demonstrate that there is room for improvement in vocabulary selection approaches when it comes to non-literal translations.

## 2.4 Domain mismatch in adaptation settings

Using a word alignment model to constrain the NMT output vocabulary means that this model should ideally also be adapted when adapting the NMT model to a new domain. Table 4 shows that adapting the word alignment model with relevant in-domain data (in this case, idiomatic expressions in context) yields strong recall improvements for vocabulary selection. Compared to increasing the per-source-word vocabulary as shown in Table 2, the improvement in recall for idiom tokens is larger which highlights the importance of having a vocabulary selection model which matches the domain of the NMT model. This also corroborates the finding of [Bogoychev and Chen \(2021\)](#) that vocabulary selection can be harmful in domain-mismatched scenarios.

We argue that integrating vocabulary prediction into the NMT model avoids the need for mitigating domain mismatch because domain adaptation will update both parts of the model. This simplifies domain adaptation since it only needs to be done once for a single model and does not require adaptation or re-training of a separate alignment model.

## 2.5 Summary

We use target recall as a measure for selection model quality. We see that alignment model quality only has a limited impact on target token recall with more recent models actually having lower recall overall. In domain adaptation scenarios vocabulary selection limits translation quality if the selection model is not adapted. The main challenge

for alignment-based vocabulary selection comes from its out-of-context selection of target tokens on a token-by-token basis, shown to reduce recall for translation of idiomatic, non-literal expressions. Increasing the size of the allowed set can compensate for this shortcoming at the cost of latency. However, this begs the question of whether context-sensitive selection of target tokens can achieve higher recall without increasing vocabulary size.

## 3 Neural Vocabulary Selection (NVS)

We incorporate vocabulary selection directly into the neural translation model, instead of relying on a separate statistical model based on token translation probabilities. This enables predictions based on contextualized representations of the full source sentence. It further simplifies the training procedure and domain adaptation, as we do not require a separate training procedure for an alignment model.

The goal of our approach is three-fold. We aim to (1) keep the general Transformer ([Vaswani et al., 2017](#)) translation model architecture, (2) incur only a minimal latency overhead that amortizes by cheaper decoder steps due to smaller output vocabularies, and (3) scale well to sentences of different lengths.

Figure 2 shows the Neural Vocabulary Selection (NVS) model. We base the prediction of output tokens on the contextualized hidden representation produced by the Transformer encoder  $\mathbf{H} \in \mathbb{R}^{d \times t}$  for  $t$  source tokens and a hidden size of  $d$ . The  $t$  source tokens are comprised of  $t - 1$  input tokens and a special `<EOS>` token. To obtain the set of target tokens, we first project each source position to the target vocabulary size  $V$ , apply max-pooling across tokens ([Shen et al., 2018](#)), and finally use the sigmoid function,  $\sigma(\cdot)$ , to obtain

$$\mathbf{z} = \sigma(\text{maxpool}(\mathbf{W}\mathbf{H} + \mathbf{b})), \quad (3)$$

where  $\mathbf{W} \in \mathbb{R}^{V \times d}$ ,  $\mathbf{b} \in \mathbb{R}^V$  and  $\mathbf{z} \in \mathbb{R}^V$ . The max-pooling operation takes the per-dimension maximum across the source tokens, going from  $\mathbb{R}^{V \times t}$  to  $\mathbb{R}^V$ . Each dimension of  $\mathbf{z}$  indicates the probability of a given target token being present in the output given the source. To obtain the target Bag-of-words (BOW), we select all tokens where  $z_i > \lambda$  as indicated in the right-hand side of Figure 2, where  $\lambda$  is a free parameter that controls the size  $\bar{V}$  of the reduced vocabulary  $\bar{\mathcal{V}}$ . At inference time, the output projection and softmax at every

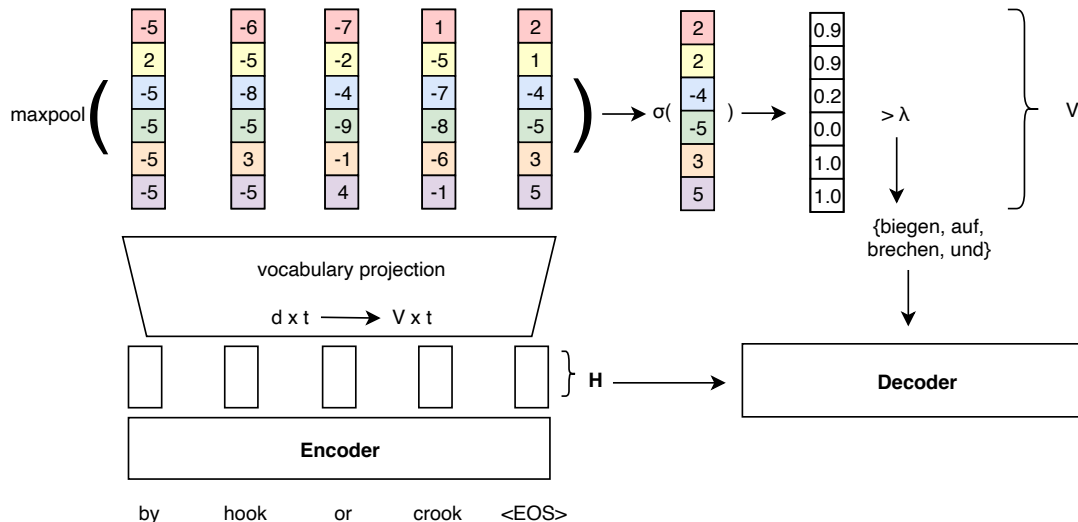


Figure 2: The proposed Neural Vocabulary Selection (NVS) model. A subset of the full vocabulary is selected based on encoder representation  $H$  and passed to the decoder for a reduced output projection at every decoder step.

decoder step are computed over the predicted BOW of size  $\bar{V}$  only.

We achieve goal (1) by basing predictions on the encoder representation already used by the decoder. Goal (2) is accomplished by restricting NVS to a single layer and basing the prediction on the encoder output, where we can parallelize computation across source tokens. Inference latency is dominated by non-parallelizable decoder steps (Kasai et al., 2020). By projecting to the target vocabulary per source token, each source token can “vote” on a set of target tokens. The model automatically scales to longer sentences via the max-pooling operation, acting as a union of per-token choices, fulfilling goal (3). Max-pooling does not tie the predictions across timesteps as they would be with mean-pooling which would also depend on sentence length. Additionally, we factor in a sentence-level target token prediction based on the  $\langle \text{EOS} \rangle$  token. The probability of a target word being present is represented by the source position with the highest evidence, backing off to a base probability of a given word via the bias vector  $\mathbf{b}$ .

To learn the  $V \times d + V$  parameters for NVS, we use a binary cross-entropy loss with the binary ground truth vector  $\mathbf{y} \in \mathbb{R}^V$ , where each entry indicates the presence or absence of target token  $\mathbf{y}_i$ . We define the loss as

$$L_{\text{NVS}} = \frac{1}{Z} \sum_{i=0}^V \mathbf{y}_i \log(z_i) \lambda_p + (1 - \mathbf{y}_i) \log(1 - z_i),$$

where  $\lambda_p$  is a weight for the positive class and

$Z = V + (\lambda_p - 1) * n_p$  is the normalizing factor, with  $n_p = \sum_i \mathbf{y}_i$  being the number of unique target words. Most dimensions of the  $V$ -dimensional vector  $\mathbf{y}$  will be zero as only a small number of target words are present for a given sentence. To counter the class imbalance of the negative class over the positive class the  $\lambda_p$  weight allows overweighting the positive class. This has the same effect as if each target word had occurred  $\lambda_p$  times. The NVS objective ( $L_{\text{NVS}}$ ) is optimized jointly with the standard negative log-likelihood translation model loss ( $L_{\text{MT}}$ ):  $L = L_{\text{NVS}} + L_{\text{MT}}$ .

## 4 Experiments

### 4.1 Setup

Our training setup is guided by best practices for efficient NMT to provide a strong low latency baseline: deep Transformer as encoder with a lightweight recurrent unit in the shallow decoder (Bérard et al., 2021; Kim et al., 2019), int8 quantization for CPU and half-precision GPU inference. We use the constrained data setting from WMT20 (Barrault et al., 2020b) with four language pairs English-German, German-English, English-Russian, Russian-English and apply corpus cleaning heuristics based on sentence length and language identification. We tokenize with *sacremoses*<sup>4</sup> and byte-pair encode (Sennrich et al., 2016) the data with 32k merge operations.

<sup>4</sup><https://github.com/alvations/sacremoses>

	vocabulary selection	newstest2020					ITDS idiom test			
		BLEU $\uparrow$	COMET $\uparrow$	human eval (%) $\uparrow$		CPU $\downarrow$ (ms / %)	GPU $\downarrow$ (ms / %)	BLEU $\uparrow$	human eval (%) $\uparrow$	
EN-DE	-	34.4 $\pm$ 0.2	0.461 $\pm$ 0.003	100	100	999 $\pm$ 12	208 $\pm$ 1	25.5 $\pm$ 0.2	100	100
	align k=200	34.2 $\pm$ 0.1	0.453 $\pm$ 0.004	98.0 $\pm$ 1.9	96.7 $\pm$ 1.9	51.0 $\pm$ 0.6	78.2 $\pm$ 0.9	25.4 $\pm$ 0.2	96.6 $\pm$ 1.9	97.2 $\pm$ 1.8
	align k=1000	34.3 $\pm$ 0.2	0.462 $\pm$ 0.005	100.1 $\pm$ 2.0	100.0 $\pm$ 1.9	66.6 $\pm$ 0.5	82.9 $\pm$ 0.7	25.5 $\pm$ 0.1	98.5 $\pm$ 1.7	100.4 $\pm$ 1.9
	NVS $\lambda=0.99$	34.4 $\pm$ 0.1	0.460 $\pm$ 0.004	-	99.5 $\pm$ 1.7	46.6 $\pm$ 0.5	79.8 $\pm$ 0.9	25.5 $\pm$ 0.2	-	99.3 $\pm$ 1.9
	NVS $\lambda=0.9$	34.4 $\pm$ 0.2	0.461 $\pm$ 0.004	100.0 $\pm$ 1.8	-	54.1 $\pm$ 0.7	82.2 $\pm$ 0.9	25.5 $\pm$ 0.2	99.1 $\pm$ 1.9	-
	NVS $\lambda=0.5$	34.4 $\pm$ 0.2	0.461 $\pm$ 0.004	-	-	65.6 $\pm$ 0.8	86.1 $\pm$ 0.7	25.5 $\pm$ 0.2	-	-
DE-EN	-	40.7 $\pm$ 0.2	0.645 $\pm$ 0.001	100	100	1128 $\pm$ 17	229 $\pm$ 1	29.8 $\pm$ 0.1	100	100
	align k=200	40.7 $\pm$ 0.2	0.643 $\pm$ 0.002	98.1 $\pm$ 2.6	98.8 $\pm$ 2.3	53.6 $\pm$ 0.5	76.9 $\pm$ 0.7	29.8 $\pm$ 0.1	96.9 $\pm$ 2.1	97.9 $\pm$ 2.2
	align k=1000	40.7 $\pm$ 0.2	0.645 $\pm$ 0.001	99.3 $\pm$ 2.5	100.3 $\pm$ 2.2	67.9 $\pm$ 0.5	84.4 $\pm$ 0.6	29.8 $\pm$ 0.1	101.2 $\pm$ 2.1	101.8 $\pm$ 2.2
	NVS $\lambda=0.99$	40.7 $\pm$ 0.1	0.644 $\pm$ 0.003	-	99.8 $\pm$ 2.4	47.6 $\pm$ 0.6	76.4 $\pm$ 0.7	29.7 $\pm$ 0.1	-	99.9 $\pm$ 2.1
	NVS $\lambda=0.9$	40.7 $\pm$ 0.2	0.645 $\pm$ 0.002	101.6 $\pm$ 2.6	-	54.4 $\pm$ 0.6	79.8 $\pm$ 1.3	29.7 $\pm$ 0.1	101.0 $\pm$ 2.2	-
	NVS $\lambda=0.5$	40.7 $\pm$ 0.1	0.645 $\pm$ 0.002	-	-	63.9 $\pm$ 0.5	83.2 $\pm$ 0.7	29.7 $\pm$ 0.1	-	-
EN-RU	-	23.6 $\pm$ 0.1	0.528 $\pm$ 0.002	100	100	673 $\pm$ 7	145 $\pm$ 1			
	align k=200	23.3 $\pm$ 0.1	0.507 $\pm$ 0.003	95.9 $\pm$ 1.8	94.1 $\pm$ 1.8	49.9 $\pm$ 0.8	78.2 $\pm$ 0.9			
	align k=1000	23.5 $\pm$ 0.1	0.527 $\pm$ 0.003	100.4 $\pm$ 1.8	99.9 $\pm$ 1.7	64.8 $\pm$ 0.8	82.9 $\pm$ 0.7			
	NVS $\lambda=0.99$	23.6 $\pm$ 0.1	0.525 $\pm$ 0.002	-	99.2 $\pm$ 1.7	48.1 $\pm$ 0.6	79.8 $\pm$ 0.9			
	NVS $\lambda=0.9$	23.6 $\pm$ 0.1	0.528 $\pm$ 0.003	99.3 $\pm$ 1.7	-	58.2 $\pm$ 0.8	82.2 $\pm$ 0.9			
	NVS $\lambda=0.5$	23.6 $\pm$ 0.0	0.529 $\pm$ 0.002	-	-	67.5 $\pm$ 0.8	86.1 $\pm$ 0.7			
RU-EN	-	35.5 $\pm$ 0.1	0.559 $\pm$ 0.005	100	100	568 $\pm$ 8	123 $\pm$ 1			
	align k=200	35.2 $\pm$ 0.1	0.552 $\pm$ 0.004	96.4 $\pm$ 2.5	97.7 $\pm$ 2.5	49.3 $\pm$ 0.9	79.4 $\pm$ 1.0			
	align k=1000	35.4 $\pm$ 0.1	0.561 $\pm$ 0.004	99.9 $\pm$ 2.5	98.7 $\pm$ 2.3	62.7 $\pm$ 0.5	81.8 $\pm$ 0.8			
	NVS $\lambda=0.99$	35.5 $\pm$ 0.1	0.558 $\pm$ 0.004	-	100.2 $\pm$ 2.5	47.2 $\pm$ 0.5	79.9 $\pm$ 1.0			
	NVS $\lambda=0.9$	35.5 $\pm$ 0.1	0.559 $\pm$ 0.005	101.1 $\pm$ 2.6	-	52.2 $\pm$ 0.6	80.9 $\pm$ 1.3			
	NVS $\lambda=0.5$	35.5 $\pm$ 0.1	0.559 $\pm$ 0.005	-	-	60.2 $\pm$ 0.6	83.6 $\pm$ 0.9			

Table 5: Experimental results for unconstrained decoding (baseline), alignment-based VS with different  $k$ , and Neural Vocabulary Selection with varying  $\lambda$ . BLEU and COMET: mean and std of three runs with different random seeds. Human evaluation: source-based direct assessment renormalized so that the unconstrained baseline is at 100%, with 95% CI of a paired t-test. We ran two sets of human evaluations comparing 4 systems each. CPU/GPU: p90 latency in ms with 95% CI based on 30 runs with batch size 1, shown as a percentage of the baseline.

All models are Transformers (Vaswani et al., 2017) trained with the Sockeye 2 toolkit (Domhan et al., 2020). We release the NVS code as part of the Sockeye toolkit<sup>5</sup>. We use a 20-layer encoder and a 2-layer decoder with self-attention replaced by SSRUs (Kim et al., 2019).

NVS and NMT objectives are optimized jointly, but gradients of the NVS objective are blocked before the encoder. This allows us to compare the different vocabulary selection techniques on the same translation model that is unaffected by the choice of vocabulary selection. All vocabulary selection methods operate at the BPE level. We use the translation dictionaries from `fast_align` for alignment-based vocabulary selection. We use a minimum of  $k = 200$  for alignment-based vocabulary selection which is at the upper end of what is found in previous work. Junczys-Dowmunt et al. (2018) set  $k = 100$ , Kim et al. (2019) set  $k = 75$ , and Shi and Knight (2017) set  $k = 50$ .

Smaller  $k$  would lead to stronger quality degradations at lower latency. GPU and CPU latency is evaluated at single-sentence translation level to match real-time translation use cases where latency is critical. We evaluate translation quality using SacreBLEU (Post, 2018)<sup>6</sup> and COMET (Rei et al., 2020)<sup>7</sup>. Furthermore, we conduct human evaluations with two annotators on the subsets of newstest2020 and IDTS test sentences where outputs differ between NVS  $\lambda = 0.9$  (0.99) and align  $k = 200$ . Professional bilingual annotators rate outputs of four systems concurrently in absolute numbers with increments of 0.2 from 1 (worst) to 6 (best). Ratings are normalized so that the (unconstrained) baseline is at 100%. Complementary details on the training setup, vocabulary selection model size, human and latency evaluation setup can be found in Appendix A.

<sup>5</sup><https://github.com/aws-labs/sockeye/tree/naacl2022>

<sup>6</sup>BLEU+case.mixed+lang.en-de+numrefs.1+smooth.exp+tok.13a+version.1.4.14.

<sup>7</sup>wmt-large-da-estimator-1719

## 4.2 Results

Table 5 shows results of different vocabulary selection models on newstest2020 and the ITDS idiom set, compared to an unconstrained baseline without vocabulary selection. Automatic evaluation metrics show only very small differences between models. For three out of four language pairs, the alignment model with  $k = 200$  performs slightly worse than the unconstrained baseline (0.2-0.3 BLEU). This corroborates existing work that quality measured by automatic metrics is not significantly affected by alignment-based vocabulary selection (Jean et al., 2015; Shi and Knight, 2017; Kim et al., 2019).

However, human-perceived quality of alignment-based vocabulary selection with  $k = 200$  is consistently lower than the baseline. COMET, found to correlate better with human judgements than BLEU (Kocmi et al., 2021), only reflects this drop in two out of the four language pairs, considering confidence intervals across random seeds. Increasing  $k$  to 1000 closes the quality gap with respect to human ratings taking the confidence intervals into account. The same is true for vocabulary selection using NVS at both  $\lambda = 0.9$  and  $\lambda = 0.99$ , where quality is also within the confidence intervals of the unconstrained baseline. However, NVS is consistently faster than the alignment-based model. For  $\lambda = 0.9$  we see CPU latency improvements of 95 ms on average across language arcs. Increasing the threshold to  $\lambda = 0.99$  latency compared to  $k = 1000$  is reduced by 157 ms on average. The same trend holds for GPU latency but with smaller differences. Figure 3 compares the NVS model against the alignment model according to the speed/quality tradeoff reflected by average vocabulary size vs. reference token recall on newstest2020. NVS consistently outperforms the alignment model, especially for small average vocabulary sizes where NVS achieves substantially higher recall. This demonstrates that the reduced vocabulary size and therefore faster decoder steps can amortize the cost of running the lightweight NVS model, which is fully parallelized across source tokens as part of the encoder.

To evaluate a domain adaptation setting, we fine-tune the NVS models on a set of 300 held-out sentences of idioms in sentence context for 10 epochs. For a fair comparison, we also include the same data for the alignment-based vocabulary selection. Figure 4 shows that NVS yields pareto optimality over the alignment model with and without domain

adaptation to a small internal training set of idiomatic expressions in context. This highlights the advantage of NVS which is automatically updated during domain fine-tuning as it is part of a single model. See Appendix C for additional figures on the proverbs and ITDS test sets, where the same trend holds.

## 4.3 Analysis

Our proposed neural vocabulary selection model benefits from contextual target word prediction. We demonstrate this by comparing the predicted BOW when using the source sentence context versus predicting BOWs individually for each input word (which may consist of multiple subwords) and taking the union of individual bags. We use the NVS models that are adapted to a set of idiomatic expressions for this analysis to ensure that the unconstrained baseline models produce reasonable translations for the Idiom test set.

	Ref tokens	$\lambda$	Context	No Context
EN-DE	All	0.9	0.93	0.62
		0.99	0.73	0.31
	All excl	0.9	0.32	0.01
		0.99	0.43	0.01
EN-RU	All	0.9	0.75	0.43
		0.99	0.57	0.25
	All excl	0.9	0.37	0.05
		0.99	0.38	0.06

Table 6: Percentage of segments with all idiomatic reference tokens included in the BOW (All), or exclusively included in the contextual or non-contextual BOW (All excl) for NVS threshold  $\lambda$ .

Table 6 shows the percentage of segments for which all reference tokens are included in the contextual vs. the non-contextual BOW for an acceptance threshold of 0.9 and 0.99. Independent of the threshold, predicting the BOW using source context yields significantly larger overlap with idiomatic reference tokens. We also measure the extent to which idiomatic reference tokens are included exclusively in the contextual or non-contextual BOW. For 32% of EN-DE segments, only the contextual BOW contains all idiomatic reference tokens. For non-contextual BOWs, this happens in only 1% of the segments (with  $\lambda=0.9$ ). For EN-RU, the values are 38% versus 6%, respectively. This shows that the model makes extensive use of contextualized source representations in predicting the relevant output tokens for idiomatic expressions.

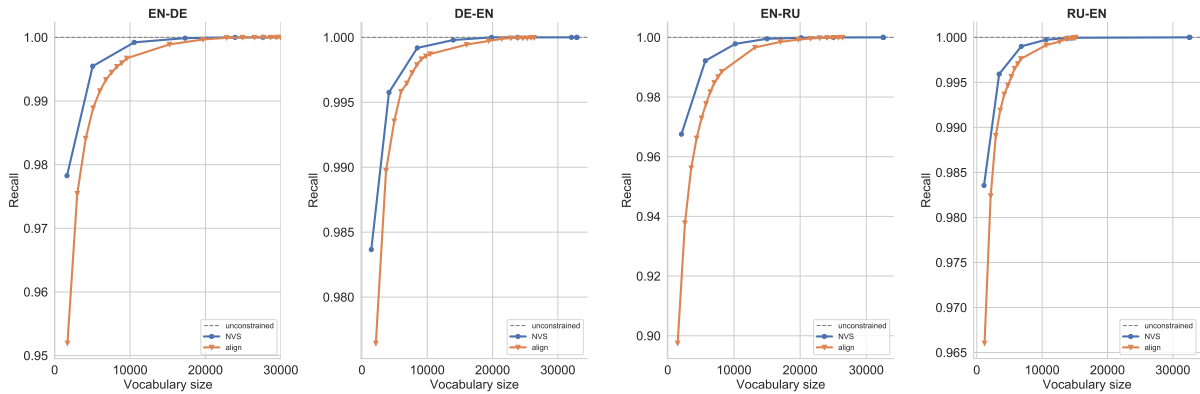


Figure 3: Vocabulary size (speed) vs. recall of reference tokens (quality) for newstest2020. For NVS, values correspond to  $\lambda \in [0.99, 0.9, 0.5, 0.1, 0.01, \dots, 0.000001]$ . For align, values correspond to  $k \in [100, 200, \dots, 1000, 2000, \dots, 10000]$ .

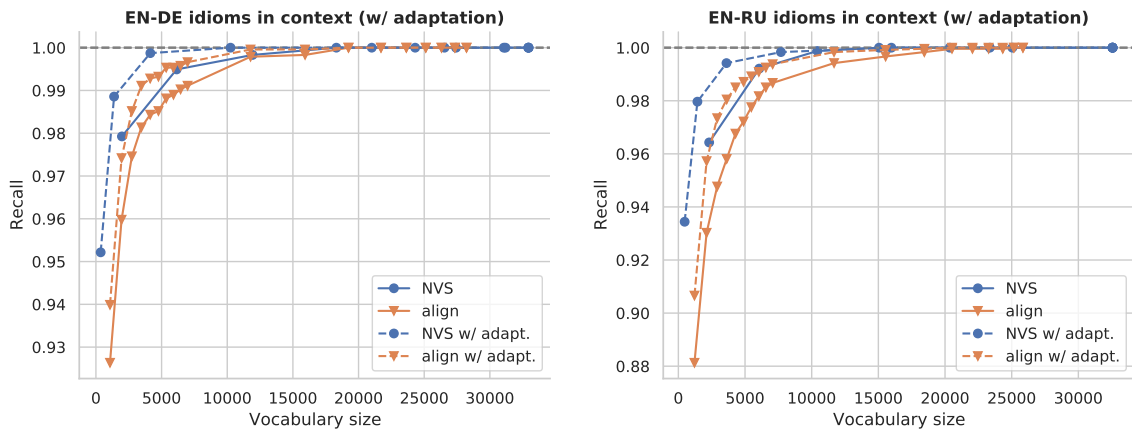


Figure 4: Vocabulary size (speed) vs recall of reference tokens (quality) for the internal Idioms test set *with* and *without* adapting on idioms for NVS and align models. For NVS, the values correspond to setting  $\lambda \in [0.99, 0.9, 0.5, 0.1, 0.01, \dots, 0.000001]$ . For align, the values correspond to setting  $k \in [100, 200, \dots, 1000, 2000, \dots, 10000]$ .

Figure 5 shows a few illustrative examples where the idiomatic reference is only reachable with the contextual BOW prediction. Consider the last example containing the English idiom “to wrap one’s head around it”. Even though the phrase is rather common in English, the German translation “verstehen” (*to understand*) would not be expected to rank high for any of the idiom source tokens. Evaluating the tokens in context however yields the correct prediction.

## 5 Related work

There are two dominant approaches to generate a restricted set of target word candidates (i) using an external model and (ii) using the NMT system itself.

In the first approach, a short-list of translation candidates is generated from word-alignments (Jean et al., 2015; Kim et al., 2019), phrase table, and the most common target words (Mi et al., 2016). L’Hostis et al. (2016) propose an additional method using support vector machines to predict target candidates from a sparse representation of the source sentence.

In the second approach, Sankaran et al. (2017) build alignment probability table from the soft-attention layer from decoder to encoder. However, applying their method to multi-head attention in Transformer is non-trivial as attention may not capture word-alignments in multiple attention layers (Li et al., 2019). Shi and Knight (2017) use local sensitive hashing to shrink the target vocabu-



Source	Idiomatic target
I thought I would be nervous , but I was <b>cool as a cucumber</b> .	die <b>Ruhe</b> selbst
He decides that it is better <b>to face the music</b> , op__ting to stay and conf__ess .	sich den <b>Dingen</b> stellen
The Classic Car Show is held in conjunction with Old Sett__ler 's Day , <b>rain or shine</b> .	bei <b>jedem</b> Wetter
Tools , discipline , formal methods , process , and profession__alism were tou__ted as <b>silver bullets</b> :	<b>Wunder</b> __wa__ffe
They said he was ' a little bit <b>under the weather</b> ' .	sich nicht <b>wohl</b> __ <b>föhlen</b>
I still can 't <b>wrap my head around it</b> .	<b>verstehen</b>

Figure 5: Test inputs from the internal Idioms test set for which the highlighted tokens in the idiomatic reference are *exclusively* included in the contextual BOW (computed for idiom-adapted NVS model with  $\lambda = 0.9$ ).

lary during decoding, though their approach only reduces latency on CPUs instead of GPUs.

Chen et al. (2019) reduce the softmax computation by first predicting a cluster of target words and then perform exact search (i.e., softmax) on that cluster. The clustering process is trained jointly with the translation process in their approach.

Closely related to our work is Weng et al. (2017), who predict all words in a target sentence from the initial hidden state of the decoder. Our NVS model differs from theirs in that we make a prediction for each source token and aggregate the results via max-pooling to scale with sentence length. Recent work of Bogoychev and Chen (2021) illustrates the risk associated with reducing latency via vocabulary selection in domain-mismatched settings. Our work takes this a step further by providing a detailed analysis on the shortcomings of vocabulary selection and proposing a model to mitigate them.

Related to our findings on non-compositional expressions, Renduchintala et al. (2021) evaluate the effect of methods used to speed up decoding in Transformer models on gender bias and find minimal BLEU degradations but reduced gendered noun translation performance on a targeted test set.

## 6 Conclusions

Alignment-based vocabulary selection is a common method to heavily constrain the set of allowed output words in decoding for reduced latency with only minor BLEU degradations. We showed with human evaluations and a targeted qualitative analysis that such translations are perceivably worse. Even recent automatic metrics based on pre-trained neural networks, such as COMET, are only able to capture the observed quality degradations in two out of four language pairs. Human-perceived quality is negatively affected both for generic translations, represented by newstest2020, as well as for idiomatic translations. Increasing the vocabulary

selection threshold can alleviate the quality issues at an increased single sentence translation latency. To preserve both translation latency and quality we proposed a neural vocabulary selection model that is directly integrated into the translation model. Such a joint model further simplifies the training pipeline, removing the dependency on a separate alignment model. Our model has higher reference token recall at similar vocabulary sizes, translating into higher quality at similar latency.

## References

- Roe Aharoni, Melvin Johnson, and Orhan Firat. 2019. [Massively multilingual neural machine translation](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.
- Alham Fikri Aji and Kenneth Heafield. 2020. [Compressing neural machine translation models with 4-bit precision](#). In *Proceedings of the Fourth Workshop on Neural Generation and Translation*, pages 35–42, Online. Association for Computational Linguistics.
- N. Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George F. Foster, Colin Cherry, Wolfgang Macherey, Z. Chen, and Yonghui Wu. 2019. [Massively multilingual neural machine translation in the wild: Findings and challenges](#). *ArXiv*, abs/1907.05019.
- Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri. 2020a. [Findings of the 2020 conference on machine translation \(WMT20\)](#). In *Proceedings of the Fifth Conference on Machine Translation*,

- pages 1–55, Online. Association for Computational Linguistics.
- Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri. 2020b. [Findings of the 2020 conference on machine translation \(WMT20\)](#). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1–55, Online. Association for Computational Linguistics.
- Maximiliana Behnke, Nikolay Bogoychev, Alham Fikri Aji, Kenneth Heafield, Graeme Nail, Qianqian Zhu, Svetlana Tchistiakova, Jelmer van der Linde, Pinzhen Chen, Sidharth Kashyap, et al. 2021. Efficient machine translation with model pruning and quantization. In *Proceedings of the Six Conference on Machine Translation, Online. Association for Computational Linguistics*.
- Maximiliana Behnke and Kenneth Heafield. 2020. [Losing heads in the lottery: Pruning transformer attention in neural machine translation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2664–2674, Online. Association for Computational Linguistics.
- Alexandre Bérard, Dain Lee, Stéphane Clinchant, Kweonwoo Jung, and Vassilina Nikoulina. 2021. [Efficient inference for multilingual neural machine translation](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Alexandre Bérard, Dain Lee, Stéphane Clinchant, Kweonwoo Jung, and Vassilina Nikoulina. 2021. Efficient inference for multilingual neural machine translation. *arXiv preprint arXiv:2109.06679*.
- Nikolay Bogoychev and Pinzhen Chen. 2021. [The highs and lows of simple lexical domain adaptation approaches for neural machine translation](#). In *Proceedings of the Second Workshop on Insights from Negative Results in NLP*, pages 74–80, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Peter F Brown, Stephen A Della Pietra, Vincent J Della Pietra, and Robert L Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311.
- Chi Chen, Maosong Sun, and Yang Liu. 2021. [Mask-align: Self-supervised neural word alignment](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4781–4791, Online. Association for Computational Linguistics.
- Patrick H Chen, Si Si, Sanjiv Kumar, Yang Li, and Choji Hsieh. 2019. Learning to screen for fast softmax inference on large vocabulary neural networks. In *ICLR (Poster)*.
- Tobias Domhan, Michael Denkowski, David Vilar, Xing Niu, Felix Hieber, and Kenneth Heafield. 2020. [The sockeye 2 neural machine translation toolkit at AMTA 2020](#). In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 110–115, Virtual. Association for Machine Translation in the Americas.
- Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. [A simple, fast, and effective reparameterization of IBM model 2](#). In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.
- Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2018. [Examining the tip of the iceberg: A data set for idiom translation](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. [On using very large target vocabulary for neural machine 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 1–10, Beijing, China. Association for Computational Linguistics.
- Marcin Junczys-Dowmunt, Kenneth Heafield, Hieu Hoang, Roman Grundkiewicz, and Anthony Aue. 2018. [Marian: Cost-effective high-quality neural machine translation in C++](#). In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 129–135, Melbourne, Australia. Association for Computational Linguistics.
- Jungo Kasai, Nikolaos Pappas, Hao Peng, James Cross, and Noah A Smith. 2020. Deep encoder, shallow decoder: Reevaluating non-autoregressive machine translation. *arXiv preprint arXiv:2006.10369*.
- Young Jin Kim, Marcin Junczys-Dowmunt, Hany Hassan, Alham Fikri Aji, Kenneth Heafield, Roman Grundkiewicz, and Nikolay Bogoychev. 2019. [From research to production and back: Ludicrously fast neural machine translation](#). In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 280–288, Hong Kong. Association for Computational Linguistics.

- Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Third International Conference on Learning Representations*.
- Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494.
- Gurvan L’Hostis, David Grangier, and Michael Auli. 2016. Vocabulary selection strategies for neural machine translation. *arXiv preprint arXiv:1610.00072*.
- 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.
- Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, pages 25–30, Jeju Island, Korea. Association for Computational Linguistics.
- Haitao Mi, Zhiguo Wang, and Abe Ittycheriah. 2016. Vocabulary manipulation for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 124–129, Berlin, Germany. Association for Computational Linguistics.
- Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.
- 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.
- Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.
- Adithya Renduchintala, Denise Diaz, Kenneth Heafield, Xian Li, and Mona Diab. 2021. Gender bias amplification during speed-quality optimization in neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 99–109, Online. Association for Computational Linguistics.
- Baskaran Sankaran, Markus Freitag, and Yaser Al-Onaizan. 2017. Attention-based vocabulary selection for nmt decoding. *arXiv preprint arXiv:1706.03824*.
- 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 (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Dinghan Shen, Guoyin Wang, Wenlin Wang, Martin Renqiang Min, Qinliang Su, Yizhe Zhang, Chunyuan Li, Ricardo Henao, and Lawrence Carin. 2018. Baseline needs more love: On simple word-embedding-based models and associated pooling mechanisms. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 440–450.
- Xing Shi and Kevin Knight. 2017. Speeding up neural machine translation decoding by shrinking runtime vocabulary. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 574–579, Vancouver, Canada. Association for Computational Linguistics.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Rongxiang Weng, Shujian Huang, Zaixiang Zheng, Xinyu Dai, and Jiajun Chen. 2017. Neural machine translation with word predictions. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 136–145, Copenhagen, Denmark. Association for Computational Linguistics.
- Biao Zhang, Deyi Xiong, and Jinsong Su. 2018. Accelerating neural transformer via an average attention network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1789–1798, Melbourne, Australia. Association for Computational Linguistics.

## A Reproducibility Details

**Data** We use the constrained data setting from WMT20 (Barrault et al., 2020b) with four language pairs English-German, German-English, English-Russian, Russian-English. Noisy sentence pairs are removed based on heuristics, namely sentences with a length ratio  $> 1.5$ ,  $> 70\%$  token overlap,  $> 100$  BPE tokens and those where source or target language does not match according to LangID (Lui and Baldwin, 2012) are filtered.

**Model** We train pre-norm Transformer (Vaswani et al., 2017) models with an embedding dimension of 1024 and a hidden dimension of 4096.

Model	Size
align k=200	6,590,600
align k=1000	32,953,000
NVS	33,776,825

Table 7: Model size in terms of in-memory float numbers for EN-DE model with a target vocabulary size of 32953. This does not reflect actual memory consumption, as the computation of the NVS layer may require more intermediate memory.

Table 7 compares the memory consumption of the different vocabulary selection models in terms of float numbers. We see that the NVS model requires a similar number of floating point numbers as the alignment-based model at  $k = 1000$ . Note, that this only represent the disk space requirements as other intermediate outputs would be required at runtime for either vocabulary selection model.

**Training** The NMT objective uses label smoothing with constant 0.1, the NVS objective sets the positive class weight  $\lambda_p$  to 100,000. Models train on 8 Nvidia Tesla V100 GPUs on AWS p3.16xlarge instances with an effective batch size of 50,000 target tokens accumulated over 40 batches. We train for 70k updates with the Adam (Kingma and Ba, 2015) optimizer, using an initial learning rate of 0.06325 and linear warmup over 4000 steps. Checkpoints are saved every 500 updates and we average the weights of the 8 best checkpoints according to validation perplexity.

**Inference** For GPU latency, we run in half-precision mode (FP16) on AWS g4dn.xlarge instances. CPU benchmarks are run with INT8 quantized models run on AWS c5.2xlarge instances. We decode using beam search with a beam of size

5. Each test set is decoded 30 times on different hosts, and we report the mean p90 latency with its 95% confidence interval. Alignment-based vocabulary selection includes the top  $k$  most frequently aligned BPE tokens for each source token based on a `fast_align` model trained on the same data as the translation model. NVS includes all tokens that are scored above the threshold  $\lambda$ . All vocabulary selection methods operate at the BPE level.

**Evaluation** Human Evaluations and COMET / BLEU use full precision (FP32) inference outputs. We decided to use FP32 for human evaluation as we wanted to evaluate the quality of the underlying model independent of whether it gets used on CPU or GPU and the output differences between FP16/FP32/INT8 being small. We report mean and standard deviation of SacreBLEU (Post, 2018)<sup>8</sup> and COMET (Rei et al., 2020) scores on detokenized outputs for three runs with different random seeds. For human evaluations, bilingual annotators see a source segment and the output of a set of 4 systems at once when assigning an absolute score to each output. The size of the evaluation set was 350 for EN-DE and EN-RU and 200 for DE-EN and RU-EN for newstest2020. We used the full sets of sentences differing between NVS  $\lambda = 0.9$ , align  $k = 200$  for the ITDS test set (309 for EN-DE and 273 for DE-EN).

**Adaptation** For domain adaptation, we fine-tune the NVS model for 10 epochs using a learning rate of 0.0001 and a batch size of 2048 target tokens. To adapt the alignment-based vocabulary selection model, we include the adaptation data as part of the training data for the alignment model. We up-sample the adaptation data by a factor of 10 for a comparable setting with NVS fine-tuning.

## B Positive class weight ablation

Based on preliminary experiments we had used a weight for the positive class ( $\lambda_p$ ) of 100k in the experiments in §4. Here the positive class refers to tokens being present on the target side and the negative class to tokens being absent from the target side. For a Machine Translation setting there are many more words that are not present than are present on the target side. The negative class therefore dominates the positive class. This can be

<sup>8</sup>BLEU+case.mixed+lang.en-  
de+numrefs.1+smooth.exp+tok.13a+version.1.4.14.

	pos. weight	BLEU	COMET
EN-DE	auto $x = 10$	34.4	0.459
	auto $x = 1$	34.1	0.458
	100k	34.4	0.461
	10k	34.2	0.460
	1k	34.4	0.463
	100	34.2	0.456
	10	32.5	0.295
	1	15.9	-0.498
DE-EN	auto $x = 10$	40.9	0.644
	auto $x = 1$	40.8	0.640
	100k	40.8	0.642
	10k	41.0	0.645
	1k	40.8	0.643
	100	40.8	0.638
	10	40.2	0.558
	1	25.3	-0.608
EN-RU	auto $x = 10$	23.6	0.524
	auto $x = 1$	23.5	0.524
	100k	23.6	0.524
	10k	23.7	0.528
	1k	23.6	0.526
	100	23.3	0.497
	10	20.6	0.128
	1	5.6	-1.509
RU-EN	auto $x = 10$	35.6	0.564
	auto $x = 1$	35.6	0.563
	100k	35.6	0.557
	10k	35.8	0.565
	1k	35.4	0.556
	100	35.5	0.551
	10	34.2	0.452
	1	20.1	-0.622

Table 8: Translation quality in terms of BLEU and COMET on newstest2020 with different weights for the positive class. *auto x* refers to setting the weight according to the ratio of the negative class to the positive class with a factor  $x$ .

counteracted by using a large value for the positive weight  $\lambda_p$ .

Instead of setting  $\lambda_p$  to a fixed weight one can also define it as

$$\lambda_p = x \frac{n_n}{n_p}$$

with  $n_p$  as the number of unique target words,  $n_n = V - n_p$  as the number of remaining words and  $x$  being a factor to increase the bias towards recall. This way the positive class and negative class

are weighted equally. Table 8 shows the result of different positive weights, including the automatic setting according to the ratio (*auto*). We see that not increasing the weight of the positive class results in large quality drops. For positive weights  $> 1000$  the quality differences are small. The *auto* setting provides an alternative that is easier to set than finding a fixed positive weight.

### C Additional vocabulary size vs. recall plots

Figures 6 and 7 provide results for the proverbs and ITDS test sets, respectively. We see the same trend across all test sets of NVS offering higher recall at the same vocabulary size compared to alignment-based vocabulary selection. For the proverbs test set this is true both for the literal and the idiomatic translations.

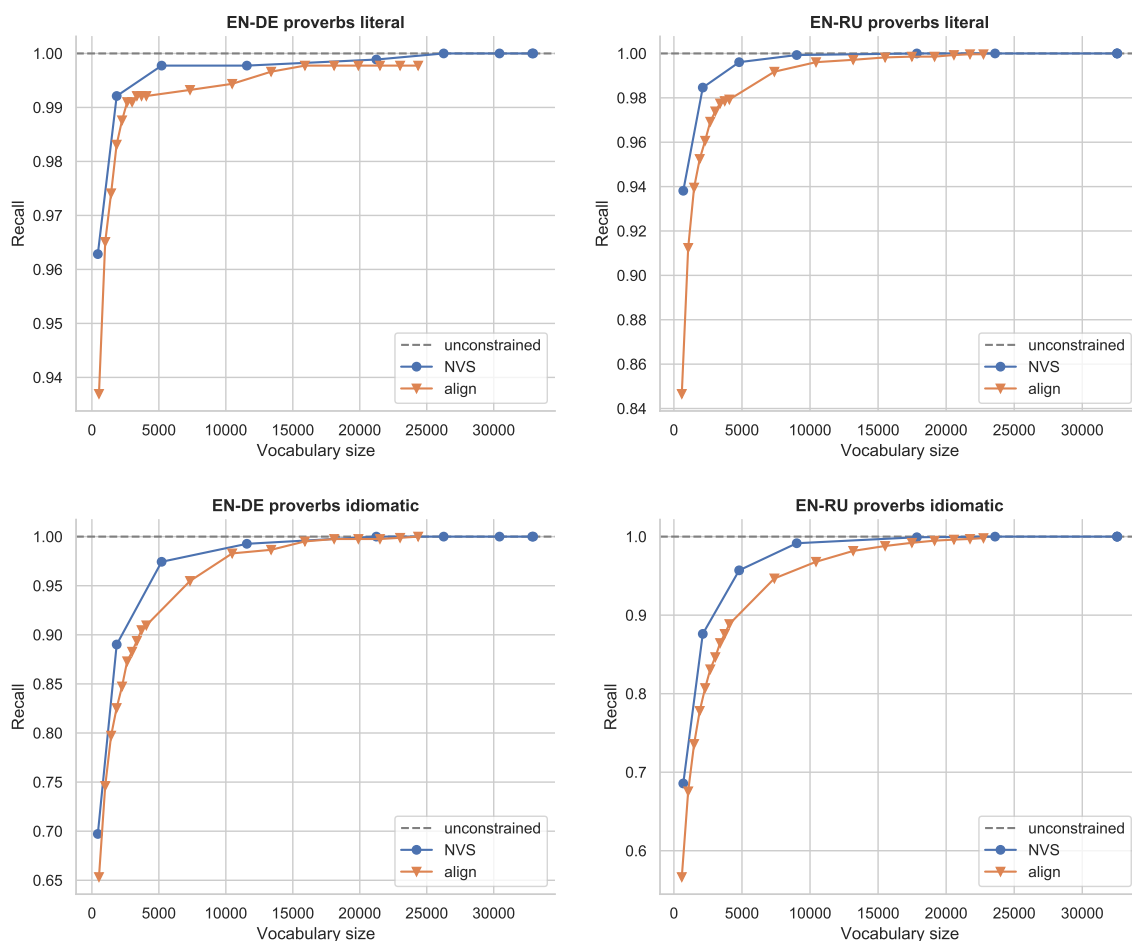


Figure 6: Vocabulary size (speed) vs. recall of reference tokens (quality) for proverbs test set. For NVS, the values correspond to setting  $\lambda \in [0.99, 0.9, 0.5, 0.1, 0.01, \dots, 0.000001]$ . For align, the values correspond to setting  $k \in [100, 200, \dots, 1000, 2000, \dots, 10000]$ .

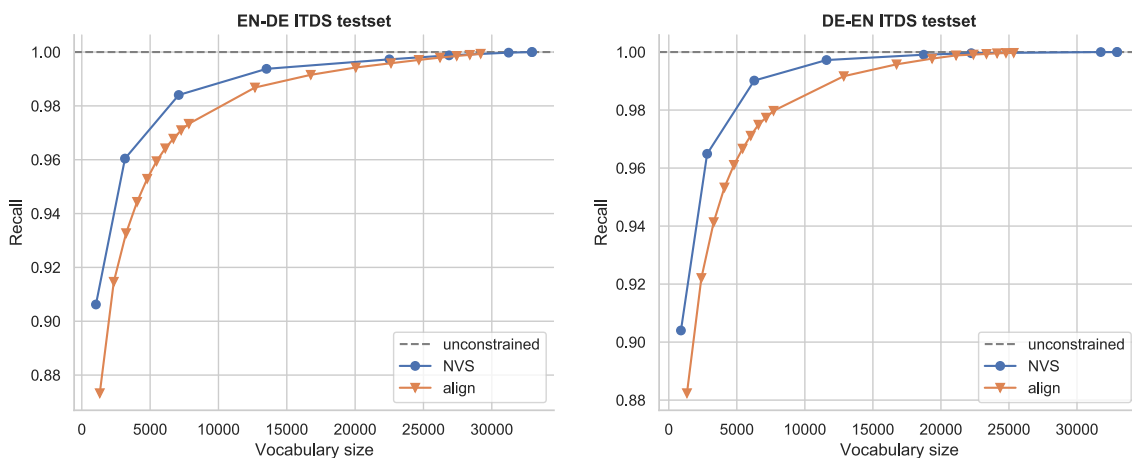


Figure 7: Vocabulary size (speed) vs. recall of reference tokens (quality) for ITDS test set. For NVS, the values correspond to setting  $\lambda \in [0.99, 0.9, 0.5, 0.1, 0.01, \dots, 0.000001]$ . For align, the values correspond to setting  $k \in [100, 200, \dots, 1000, 2000, \dots, 10000]$ .