Better Document-Level Machine Translation with Bayes' Rule

Lei Yu[1], Laurent Sartran[1], Wojciech Stokowiec[1], Wang Ling[1], Lingpeng Kong[1], Phil Blunsom[1,2], Chris Dyer[1]

[1]DeepMind, [2]University of Oxford
{leiyu, lsartran, wstokowiec, lingwang, lingpenk, pblunsom, cdyer}@google.com

Abstract

We show that Bayes' rule provides an effective mechanism for creating document translation models that can be learned from only parallel sentences and monolingual documents—a compelling benefit because parallel documents are not always available. In our formulation, the posterior probability of a candidate translation is the product of the unconditional (prior) probability of the candidate output document and the ''reverse translation probability'' of translating the candidate output back into the source language. Our proposed model uses a powerful autoregressive language model as the prior on target language documents, but it assumes that each sentence is translated independently from the target to the source language. Crucially, at test time, when a source document is observed, the document language model prior induces dependencies between the translations of the source sentences in the posterior. The model's independence assumption not only enables efficient use of available data, but it additionally admits a practical left-to-right beam-search algorithm for carrying out inference. Experiments show that our model benefits from using cross-sentence context in the language model, and it outperforms existing document translation approaches.

1 Introduction

There have been many recent demonstrations that neural language models based on transformers (Vaswani et al., 2017; Dai et al., 2019) are capable of learning to generate remarkably coherent documents with few (Zellers et al., 2019) or no (Radford et al., 2019) conditioning variables. Despite this apparent generation ability, in practical applications, unconditional language models are most often used to provide representations for natural language understanding applications (Devlin et al., 2019; Yang et al., 2019; Peters et al., 2018), and how to use them for conditional generation applications remains an open question.

Our hypothesis in this work is that Bayes' rule provides an effective way to leverage powerful unconditional document language models to improve a conditional task: machine translation. The application of Bayes' rule to transform the translation modeling problem $p(\boldsymbol{y} \mid \boldsymbol{x})$, where $\boldsymbol{y}$ is the target language, and $\boldsymbol{x}$ is the source language, has a long tradition and was the dominant paradigm in speech and language processing for many years (Brown et al., 1993), where it is often called a ''noisy channel'' decomposition, by analogy to an information theoretic conception of Bayes' rule.

Whereas several recent papers have demonstrated that the noisy channel decomposition has benefits when translating sentences one-by-one (Yu et al., 2017; Yee et al., 2019; Ng et al., 2019), in this paper we show that this decomposition is particularly suited to tackling the problem of translating complete documents. Although using cross-sentence context and maintaining cross-document consistency has long been recognized as essential to the translation problem (Tiedemann and Scherrer, 2017; Bawden et al., 2018, inter alia), operationalizing this in models has been challenging for several reasons. Most prosaically, parallel documents are not generally available (whereas parallel sentences are much more numerous), making direct estimation of document translation probabilities challenging. More subtly, documents are considerably more diverse than sentences, and models must be carefully biased so as not to pick up spurious correlations.

Our Bayes' rule decomposition (§2) permits several innovations that enable us to solve these problems. Rather than directly modeling the conditional distribution, we rewrite it as $p(\boldsymbol{y} \mid \boldsymbol{x}) \propto p(\boldsymbol{y}) \times p(\boldsymbol{x} \mid \boldsymbol{y})$. This changes the learning problem from estimating a single complex

Transactions of the Association for Computational Linguistics, vol. 8, pp. 346–360, 2020. https://doi.org/10.1162/tacl_a_00319
Action Editor: David Chiang. Submission batch: 12/2019; Revision batch: 2/2020; Published 6/2020.

conditional distribution to learning two different distributions: a language model $p(\boldsymbol{y})$, which provides unconditional estimates of the output (in this paper, documents); and $p(\boldsymbol{x} \mid \boldsymbol{y})$, which provides the probability of translating a candidate output $\boldsymbol{y}$ into the (observed) source document $\boldsymbol{x}$.

As we will discuss subsequently, although the problems of estimating $p(\boldsymbol{y} \mid \boldsymbol{x})$ and $p(\boldsymbol{x} \mid \boldsymbol{y})$ are formally similar, independence assumptions made in $p(\boldsymbol{x} \mid \boldsymbol{y})$ are less statistically costly than they might otherwise be since, at test time, we will be conditioning on $\boldsymbol{x}$ and reasoning about a posterior distribution over $\boldsymbol{y}$, which will be jointly dependent on all (conditionally independent) parts of $\boldsymbol{x}$. This statistical fact—which is the same trick that gives naïve Bayes classifiers their expressiveness and ease of estimation—permits us to assume independence between sentence translations in the reverse translation model, and therefore to use parallel sentences (rather than parallel documents) to train it. In the posterior, we thus have an implicit estimate of a document-level translation system, even though we made no use of parallel documents when estimating the prior or likelihood models. This is particularly useful because parallel sentences are much more readily available than parallel documents. A second benefit of our approach is that the unconditional language model can be estimated from nonparallel data, which exists in vast quantities.

Although the noisy channel model is ideal for exploiting the data resources that naturally exist in the world (large corpora of parallel but independent sentences, and large corpora of monolingual documents), we are faced with a much harder decoding problem (§3). To address this problem, we propose a new beam-search algorithm, exploiting the fact that our document language model operates left-to-right, and our reverse translation model treats sentences independently. The search is guided by a proposal distribution that provides candidate continuations of a document prefix, and these are reranked according to the posterior distribution. In particular, we compare two proposal models: one based on estimates of independent sentence translations (Vaswani et al., 2017) and one that conditions on the source document context (Zhang et al., 2018). Although closely related, our algorithm is much simpler and faster than that proposed in Yu et al. (2017). Rather than using a specially designed channel model (Yu et al., 2016) which is limited in processing long sequences like documents, our conditional sentence independence assumptions allow us to use any sequence-to-sequence model as the channel model, making it a better option for document-level translation.

To explore the performance of our proposed model, we focus on Chinese–English translation, following a series of papers on document translation (Zhang et al., 2018; Werlen et al., 2018; Tu et al., 2018; Xiong et al., 2019). Although in general it is unreasonable to expect that independent translations of sentences would lead to coherent translations of documents, the task of translating Chinese into English poses some particularly acute challenges. As Chinese makes fewer inflectional distinctions than English does, and the relevant clues for predicting, for example, what tense an English verb should be in, or whether an English noun should have singular or plural morphology, may be spread throughout a document, it is crucial that extra-sentential context is used.

Our experiments (§4) explore: (1) different approaches to reranking, (2) different independence assumptions when modeling documents (i.e., whether sentences are generated independently or not), (3) different amounts of language modeling data, and (4) different proposal models. Briefly summarized, we find that document-context language models significantly improve the translation quality obtained with our system, both in terms of BLEU scores, and in terms of a human evaluation. Targeted error analysis demonstrates the document prior is capable of enforcing consistency of tense and number and lexical choice across documents.

2 Model Description

We define $\underline{\boldsymbol{x}} = (\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^I)$ as the source document with I sentences, and similarly, $\underline{\boldsymbol{y}} = (\boldsymbol{y}^1, \boldsymbol{y}^2, \ldots, \boldsymbol{y}^J)$ as the target document with J sentences. During the (human) translation process, translators may split or recombine sentences, but we will assume that $I = J$.[1] Let $\boldsymbol{x}^i = (x_1^i, x_2^i, \ldots, x_M^i)$ represent the ith sentence in the document, consisting of M words; likewise $\boldsymbol{y}^i = (y_1^i, y_2^i, \ldots, y_N^i)$ denote the ith sentence in the target document, containing N words.

[1] Size mismatches are addressed by merging sentences using sentence alignment algorithms (Gale and Church, 1993).

The translation of a document $\underline{\boldsymbol{x}}$ is determined by finding the document $\underline{\hat{\boldsymbol{y}}}$, where $p(\underline{\hat{\boldsymbol{y}}} \mid \underline{\boldsymbol{x}})$ is optimal.

$$\underline{\hat{\boldsymbol{y}}} = \arg\max_{\underline{\boldsymbol{y}}} p(\underline{\boldsymbol{y}} \mid \underline{\boldsymbol{x}}). \tag{1}$$

Instead of modeling the probability $p(\underline{\boldsymbol{y}} \mid \underline{\boldsymbol{x}})$ directly, we factorize it using Bayes' rule:

$$\begin{aligned}\underline{\hat{\boldsymbol{y}}} &= \arg\max_{\underline{\boldsymbol{y}}} \frac{p(\underline{\boldsymbol{x}} \mid \underline{\boldsymbol{y}}) \times p(\underline{\boldsymbol{y}})}{p(\underline{\boldsymbol{x}})} \\ &= \arg\max_{\underline{\boldsymbol{y}}} \underbrace{p(\underline{\boldsymbol{x}} \mid \underline{\boldsymbol{y}})}_{\text{channel model}} \times \underbrace{p(\underline{\boldsymbol{y}})}_{\text{language model}}.\end{aligned} \tag{2}$$

We further assume that sentences are independently translated, and that the sentences are generated by a left-to-right factorization according to the chain rule. Therefore, we have

$$\underline{\hat{\boldsymbol{y}}} \approx \arg\max_{\underline{\boldsymbol{y}}} \prod_{i=1}^{|\underline{\boldsymbol{x}}|} p(\boldsymbol{x}^i \mid \boldsymbol{y}^i) \times p(\boldsymbol{y}^i \mid \underline{\boldsymbol{y}}^{<i}), \tag{3}$$

where $\underline{\boldsymbol{y}}^{<i} = (\boldsymbol{y}^1, \ldots, \boldsymbol{y}^{i-1})$ denotes a document prefix consisting of the first $i-1$ target sentences. Thus conceived, this is a generative model of parallel documents that makes a particular independence assumption; we illustrate the corresponding graphical model on the top of Figure 1.

2.1 Impact of the Conditional Independence Assumption

At first glance, the conditional independence assumption we have made might seem to be the very independence assumption that bedevils conventional sentence-based approaches to document translation—translations of sentence i appear to be uninfluenced by the translation of any sentence $j \neq i$. However, although this is the case during training, this is *not* the case at test time. Then, we will be conditioning on the $\boldsymbol{x}_i$'s (the source language sentences), and reasoning about the posterior distribution over the ''underlying'' $\boldsymbol{y}_i$'s. By conditioning on the child variables, conditional dependencies between all $\boldsymbol{y}_i$'s and between each $\boldsymbol{y}_i$ and all $\boldsymbol{x}_i$'s are created (Shachter, 1998). The (in)dependencies that are present in the posterior distribution are shown in the bottom of Figure 1.

Thus, although modeling $p(\underline{\boldsymbol{y}} \mid \underline{\boldsymbol{x}})$ or $p(\underline{\boldsymbol{x}} \mid \underline{\boldsymbol{y}})$ would appear to be superficially similar, the statistical impact of making a conditional independence assumption is quite different. This is fortunate, as it makes it straightforward to use parallel sentences, rather than assuming we have parallel documents, which are less often available (Voita et al., 2019b; Zhang et al., 2018; Maruf et al., 2019, inter alia). Finally, because we only need to learn to model the likelihood of sentence translations (rather than document translations), the challenges of guiding the learners to make robust generalizations in direct document translation models (Voita et al., 2019b; Zhang et al., 2018; Maruf et al., 2019, inter alia) are neatly avoided.

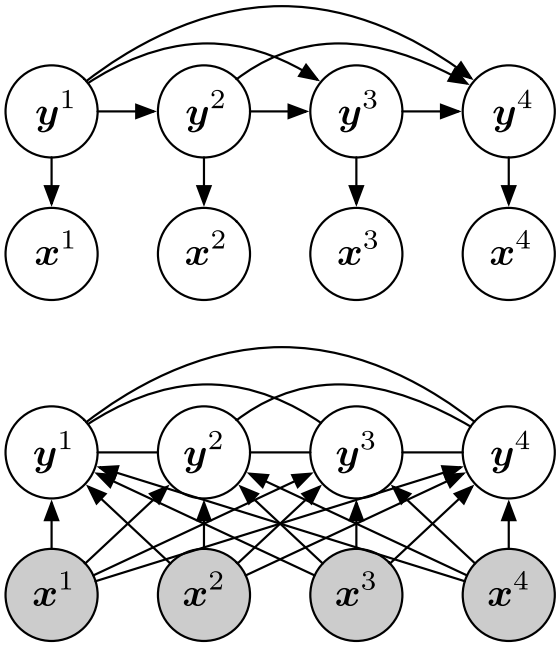

Figure 1: Graphical model showing the factorization of our noisy channel model where $\boldsymbol{y}^i$ indicates the ith target language sentence and $\boldsymbol{x}^i$ indicates the ith source language sentence. In the prior (top) the target sentences (the $\boldsymbol{y}^i$'s) only influence the corresponding source sentence and therefore can be learned and modeled independently, but at test time (bottom), when the target is not observed, each $\boldsymbol{y}^i$ depends on every $\boldsymbol{x}^i$.

2.2 Learning

We can parameterize the channel probability $p(\boldsymbol{x}^i \mid \boldsymbol{y}^i)$ using any sequence-to-sequence model and parameterize the language model $p(\boldsymbol{y}^i \mid \underline{\boldsymbol{y}}^{<i})$ using any language model. It is straightforward to learn our model: We simply optimize the channel model and the language model separately on parallel data and monolingual data, respectively. We remark that it is a significant practical

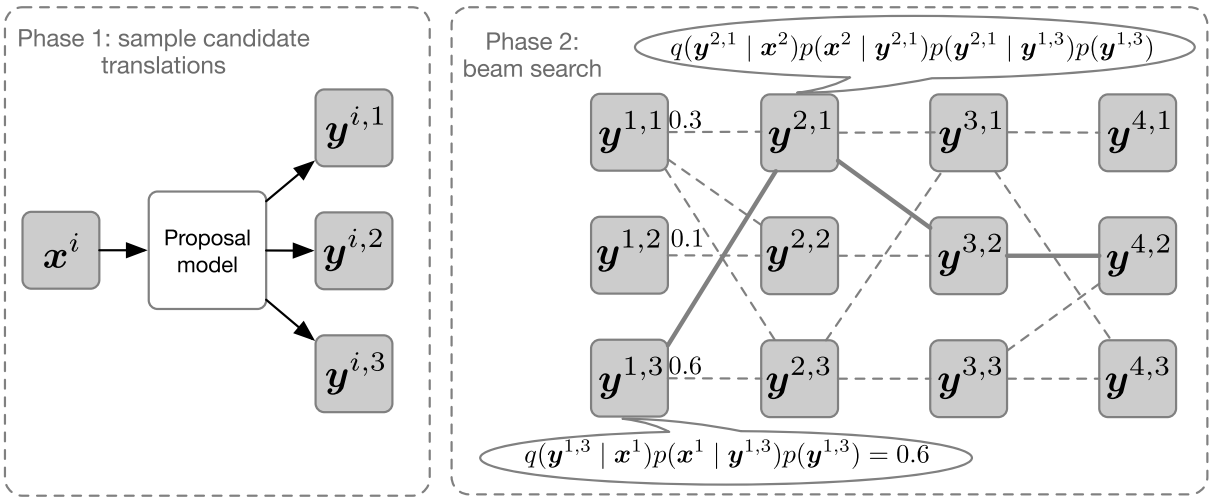

Figure 2: The decoding process. In Phase 1, the auxiliary proposal model generates candidate translations (3 candidates in the diagram) for each sentence in the document (containing 4 sentences). In Phase 2, beam search is employed to search for the best path from the candidate translations.

advantage of this parameterization that we can retrain the channel and language models independently—for example, if we acquire more monolingual data, or use different language models with the same channel model conditioned on the domain of the source text.

3 Decoding

Because of the global dependencies in the posterior distribution, decoding in our model is computationally complex. On one hand, similar to the decoding problem faced in standard sequence-to-sequence models, we must search over the space of all possible outputs with a model that makes no Markov assumptions. On the other hand, unlike traditional models, we have to have a complete $\boldsymbol{y}_i$ before we can compute $p(\boldsymbol{x}_i \mid \boldsymbol{y}_i)$, making greedy and near-greedy algorithms ineffective. To address this issue, we use an auxiliary proposal model $q(\underline{\boldsymbol{y}} \mid \underline{\boldsymbol{x}})$, that approximates the posterior distribution using a direct model, to focus our search on promising parts of the output space.

Because of the autoregressive factorization of the language model (p_{LM}), and the independent sentence translation assumption in the channel model (p_{TM}), we can carry out the reranking process using a left-to-right beam search strategy with the aid of our proposal model (q). Figure 2 illustrates the decoding process. For an input document of ℓ sentences, we let the proposal model propose K candidate translations for each sentence in the document.[2] We then search for the best document path through this lattice—or confusion network (Mangu et al., 2000)—of candidate sentence translations. To do so, we maintain a beam of the B active hypotheses (i.e., when considering the ith sentence, the prefix consists of $i-1$ sentences), and we consider the proposal's K one-sentence extensions (which we write $\boldsymbol{y}^i$). We retain B partial translations from the $K \times B$ candidates according to the following linear objective,

$$\begin{aligned}\mathcal{O}(\underline{\boldsymbol{x}}, \underline{\boldsymbol{y}}^{<i}, \boldsymbol{y}^i) = &\lambda_1 \log q(\boldsymbol{y}^i \mid \underline{\boldsymbol{x}}) + \\ &\log p_{\text{LM}}(\boldsymbol{y}^i \mid \underline{\boldsymbol{y}}^{<i}) + \\ &\lambda_2 \log p_{\text{TM}}(\boldsymbol{x}^i \mid \boldsymbol{y}^i) + \lambda_3 |\boldsymbol{y}^i| + \\ &\mathcal{O}(\underline{\boldsymbol{x}}, \underline{\boldsymbol{y}}^{<i-1}, \boldsymbol{y}^{i-1}),\end{aligned} \tag{4}$$

where $|\boldsymbol{y}|$ denotes the number of tokens in the sentence $\boldsymbol{y}$, and where the base case $\mathcal{O}(\underline{\boldsymbol{x}}, \underline{\boldsymbol{y}}^{<0}, \boldsymbol{y}^0) = 0$. Note that Eq. 4 is a generalization of Eq. 3 in log space—if we set $\lambda_1 = \lambda_3 = 0$ and $\lambda_2 = 1$ and take the log of Equation 3 the two objectives are equivalent. The extra factors—the proposal probability and the length of the output—provide improvements (e.g., by calibrating the expected length of the output), and can be incorporated at no cost in the model; they are widely used in prior work (Koehn et al., 2007; Yu et al., 2017; Yee

[2] Our proposal model can optionally use document context on the source (conditioning) side, but sentences are generated independently.

et al., 2019; Ng et al., 2019). The elements on the beam after considering the ℓth sentence are reranked one final time by adding $\log p_{\text{LM}}(\langle \text{STOP} \rangle \mid \underline{\boldsymbol{y}}^{\leq \ell})$ to the final score; this accounts for the language model's assessment that the candidate document has been appropriately ended.[3]

4 Experiments

We evaluate our model on two translation tasks, the NIST Open MT Chinese–English task[4] and the WMT19 Chinese–English news translation task.[5] On both tasks, we use the standard parallel training data, and compare our model with a strong transformer baseline, as well as related models from prior work.

4.1 Dataset Description

The NIST training data is composed from LDC-distributed news articles and broadcast transcripts and consists of 1.5M sentence pairs. The document-level parallel corpus is a subset of the full training set, including 55K documents with 1.2M sentences. Following prior work, we use the MT06 dataset as validation set and MT03, MT04, MT05, and MT08 as test sets. There are 79 documents and 1,649 sentences in the validation set and in total 509 documents and 5,146 sentences in the test set. On average, documents in the test set has 10 sentences, and 250 words and 330 words on the Chinese and English sides, respectively. We preprocess the dataset by doing punctuation normalization, tokenization, and lower-casing. We use byte pair encoding (Sennrich et al., 2016b) with 32K merges to segment words into sub-word units for both Chinese and English. The evaluation metric is case-insensitive BLEU calculated using `multi-bleu.perl`, which is consistent with prior work on this task.

The training data for the WMT19 Chinese–English task includes the UN corpus, CWMT, and news commentary. The total number of sentence pairs is 18M after filtering the data by removing duplicate sentences and sentences longer than 250 words. The validation sets that we use in the experiment are newstest2017 and newstest2018, which contains 169 documents, 2,001 sentences and 275 documents, 3,981 sentences, respectively. The test set is newstest2019, containing 163 documents and 2,000 sentences. On average, documents in the test set have 12 sentences, and 360 words and 500 words on the Chinese and English sides, respectively. The dataset is preprocessed by segmenting Chinese sentences and normalizing punctuation, tokenizing, and true-casing English sentences. As for NIST, we learn a byte pair encoding (Sennrich et al., 2016b) with 32K merges to segment words into sub-word units for both Chinese and English. The evaluation metric is *sacreBLEU* (Post, 2018).

4.2 Model Configuration

For NIST, we use the transformer (Vaswani et al., 2017) as the channel model and the document transformer (Zhang et al., 2018) as the proposal model. The hyperparameters for training the transformer are the same as *transformer base* (Vaswani et al., 2017), that is, 512 hidden size, 2,048 filter size, 8 attention heads, and 6 layers for both the encoder and decoder. We follow Zhang et al. (2018)'s configuration to train the *document transformer*: Context length is set to 2 and all other hyperparameters are the same as *transformer base*. Both models are optimized using Adam (Kingma and Ba, 2015) with approximately 24K BPE tokens per mini-batch. For the language model, we train the transformer-XL (Dai et al., 2019) on a combination of the English side of NIST training data as well as three sections of Gigaword: XIN, AFP, APW, resulting in a total of 7.3M documents and 115M sentences. We use an architecture with 24 layers, 16 attention heads, and embeddings of dimension 1024. The input sequences to the language model are encoded into bytes using the byte-level encoder provided by GPT2 (Radford et al., 2019).

For WMT19, we use the transformer as both the channel and proposal model. The hyperparameters for training the transformer is the same as *transformer big* (Vaswani et al., 2017), namely, 1,024 hidden size, 4,096 filter size, 16 attention heads, and 6 layers. The model is trained on 8 GPUs with batch size of 4,096. The setup for the language model is the same as that of NIST except that the training data is the English side of the parallel training data and Gigaword.

[3] When sentences are modeled independently, this quantity is constant and can be ignored.

[4] `https://www.nist.gov/itl/iad/mig/open-machine-translation-evaluation`.

[5] `http://www.statmt.org/wmt19/translation-task.html`.

Method	Model	Proposal	MT06	MT03	MT04	MT05	MT08
(Wang et al., 2017)	RNNsearch	–	37.76	–	–	36.89	27.57
(Kuang et al., 2017)	Transformer + cache	–	48.14	48.05	47.91	48.53	38.38
(Zhang et al., 2018)	Doc-transformer	–	49.69	50.21	49.73	49.46	39.69
Baseline	Sent-transformer	–	47.72	47.21	49.08	46.86	40.18
	Doc-transformer (q)	–	49.79	49.29	50.17	48.99	41.70
	Backtranslation (q')	–	50.77	51.80	51.61	51.81	42.47
	Sent-reranker	q	51.33	52.23	52.36	51.63	43.63
This work	Doc-reranker	q	51.99	52.77	52.84	51.84	44.17
	Doc-reranker	q'	**53.63**	**54.51**	**54.23**	**54.86**	**45.17**

Table 1: Comparison with prior work on NIST Chinese–English translation task. The evaluation metric is tokenized case-insensitive BLEU. The first three rows are numbers reported in the papers of prior work. The first two baselines are the results that we obtained by running the transformer (Vaswani et al., 2017) and the document transformer (Zhang et al., 2018) on the NIST dataset. The sent-reranker is a variation of our model in which sentences in documents are assumed to be independent. The backtranslation baseline is obtained by training the document transformer using additional synthetic parallel documents generated by backtranslation.

For both tasks, the weights $\boldsymbol{\lambda}$ are selected using grid search, from $[0.8, 1., 1.5, 2., 2.2, 2.5, 3.]$ for the weights of channel model λ_2 and proposal model λ_1, and from $[0.2, 0.5, 0.8, 1.]$ for the length penalty λ_3. The size of the n-best list used in the reranker is set to $K = 50$.[6] The beam size in the document decoding algorithm is $B = 5$.

The running time for our decoding algorithm (Section 3) highly depends on the language model's speed of calculating probabilities of partial documents. Using the transformer-XL language model with the aforementioned configuration, it takes approximately 90 seconds to decode a document on a Google Cloud TPU v3. We leave systematic exploration of inference algorithms for better solving the decoding problem to future work.

4.3 Experimental Results

Table 1 presents the best result from our model (doc-reranker) in comparison with prior work on the NIST Chinese–English translation task. The first three rows are numbers reported in prior work. Wang et al. (2017) incorporate document context by introducing a hierarchical RNN to an LSTM sequence-to-sequence model. Kuang et al. (2017) use a cache to store previously translated words across sentences, which they then use in sequence-to-sequence models. Zhang et al. (2018) extend the transformer model with an extra context encoder to capture information from previous source sentences. Apart from prior work, we also compare our doc-reranker with four baselines: the transformer (Vaswani et al., 2017), document transformer (Zhang et al., 2018), the sentence-level reranker (sent-reranker), and the document transformer with backtranslation.

In the sent-reranker, we assume sentences in the document are independent (formulation $\hat{\underline{\boldsymbol{y}}} = \arg\max_{\underline{\boldsymbol{y}}} \prod_{i=1}^{|\boldsymbol{x}|} p(\boldsymbol{x}^i \mid \boldsymbol{y}^i) \times p(\boldsymbol{y}^i)$), and therefore we train a sentence-level language model and rerank each sentence independently. This sent-reranker setup is close to the work from Yee et al. (2019) and Ng et al. (2019) with the difference that rather than using a language model trained on documents we use a language model trained on sentences, which is more statistically consistent.

Table 1 shows that our reranker outperforms previous models as well as strong transformer baselines by a significant margin—approximately 2.5 BLEU on all test sets—achieving new state of the art. Although the gap between the doc-reranker and sent-reranker is smaller, as we will show in §A.1 and §5.2 that translations generated by doc-reranker are preferred by humans and are more consistent across documents, in line with concerns

[6] $K = 50$ gives the best compromise between performance and inference time.

Proposal model	Language model	Sent-reranker	Doc-reranker
Sent-transformer	LSTM: NIST	49.92	50.24
	transformer-XL: NIST	50.29	50.56
	transformer-XL: NIST + Gigaword	50.19	50.93
Doc-transformer	LSTM: NIST	50.75	51.20
	transformer-XL: NIST	51.27	51.68
	transformer-XL: NIST + Gigaword	51.33	51.99

Table 2: BLEU scores on NIST dev set MT06 from rerankers which are incorporated with various language models. In the language model column X: Y means the language model X is trained on dataset Y. A bigger language model improves the doc-reranker but does not help the sent-reranker.

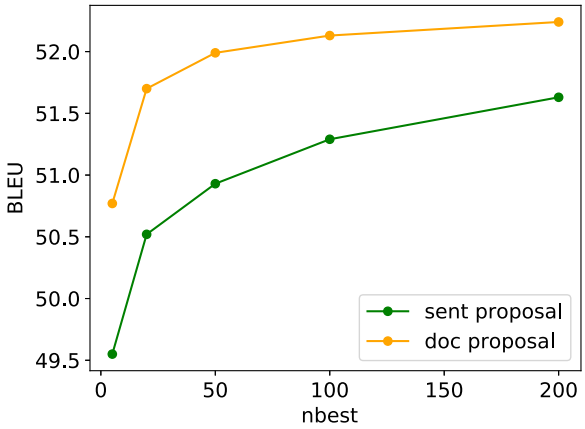

Figure 3: Effect of n-best list.

Architecture	Data	PPL
transformer-XL	NIST sent	83.3
transformer-XL	NIST + GW sent	96.5
LSTM	NIST doc	71.6
transformer-XL	NIST doc	43.8
transformer-XL	NIST + GW doc	43.4

Table 3: Perplexity per word of language models on NIST dev set. GW refers to Gigaword.

about the reliability of using BLEU at assessing cross-sentential consistency (Voita et al., 2019b).

To compare the effectiveness of leveraging monolingual data between backtranslation (Edunov et al., 2018; Sennrich et al., 2016a) and our model, we train the document transformer (Zhang et al., 2018) using additional synthetic parallel documents generated by backtranslation (q'). For fair comparison we use the same monolingual data for both models. As shown in Table 1, although both techniques improve translation, backtranslation is less effective than our model. Because we have a new model q', we can use it as a proposal model for our doc-reranker—effectively using the monolingual data twice. We find that this improves results even further, indicating that the effect of both approaches is additive.

To understand the rerankers better, we investigate the effect of different proposal models, different language models, and various numbers of candidates in the n-best list. Table 2 and Figure 3 show that better proposal models and bigger n-best lists lead to consistently better reranking results. This is an appealing behavior showing that our reranker is able to pick better translations from higher quality and more diverse candidate pools generated by better proposal models and bigger n-best lists. To compare the effect of language models, we train an LSTM language model (Merity et al., 2018b,a) and a transformer-XL language model on the English side of NIST parallel training data in addition to the transformer-XL trained on NIST and Gigaword. Table 3 lists the perplexity per word on the NIST validation set for different language models. Given the same training data, the transformer-XL performs significantly better than the LSTM-based language model, which in turn results in a higher BLEU score from the doc-reranker. By adding more training data, the transformer-XL language model achieves even lower perplexity and that gives a further boost to the performance of the doc-reranker. Notably, when the strong transformer-XL language model is incorporated

Reranker	Models	MT06
–	Doc-transformer	49.79
Doc-reranker	Proposal + LM	49.79
	Channel + LM	51.93
	Proposal + Channel	50.40
	Proposal + Channel + LM	**51.99**

Table 4: Effect of different components.

into the doc-reranker, the best weight ratio of the channel and language model is 1:1, indicating that the doc-reranker depends heavily on the language model. By contrast, if a weak language model is incorporated, the best ratio is approximately $2:1$. A further observation is that although a larger-scale language model improves the doc-reranker, it does not help the sent-reranker.

We perform an ablation study to explore what each component of the doc-reranker contributes to the overall performance. Table 4 shows BLEU scores on the NIST validation set for the optimal interpolation of various component models. No gains are observed if the language model is combined with the proposal model (a probabilistically unsound combination, although one that often worked in pre-neural approaches to statistical translation). We find that as we increase the weight of the language model, the results become worse. The interpolation of the proposal model and channel model slightly outperforms the proposal model baseline but considerably underperforms the interpolation of the proposal model, channel model, and the language model. This difference indicates the key roles that the language model plays in the doc-reranker. When the channel model is combined with the language model the performance of the doc-reranker is comparable to that with all three components included. We conclude from the ablation study that both the channel and language models are indispensable for the doc-reranker, indicating that Bayes' rule provides reliable estimates of translation probabilities.

Table 5 presents the results of our model together with baselines on the WMT19 Chinese–English translation task. We find that the results follow the same pattern as those on NIST: A better language model leads to better translation results and overall the reranker outperforms the transformer-big by approximately 2.5 BLEU. The two best systems submitted to the WMT19 Chinese–English translation task are Microsoft Research Asia's system (Xia et al., 2019) and Baidu's system (Sun et al., 2019), both of which use multiple techniques to improve upon the transformer big model. Here, we mainly compare our results with those from Xia et al. (2019) because we use the same evaluation metric *SacreBLEU* (Post, 2018) and the same validation and test sets. Using extra parallel training data and the techniques of masked sequence-to-sequence pretraining (Song et al., 2019), sequence-level knowledge distillation (Kim and Rush, 2016), and backtranslation (Edunov et al., 2018), the best model from Xia et al. (2019) achieves 30.8, 30.9, and 39.3 on newstest2017, newstest2018, and newstest2019, respectively. Although our best results are lower than this, it is notable that our model achieves comparable results to their model, which was trained on 56M sentences of parallel data—over two times more training data than we use. However, our method is orthogonal to these works and can be combined with other techniques to make further improvement.

5 Analysis

In this section, we present the quantitative and qualitative analysis of our models. The analysis is performed on the experiments of the NIST dataset.

5.1 Quantitative Analysis

We do oracle experiments in order to assess our models' ability to select good translation candidates. We create our candidate pool by mixing the proposals generated from the transformer model (Vaswani et al., 2017) and the four references. We subsequently calculate how many cases over the entire validation dataset in which different models (the proposal model, sent-reranker, and doc-reranker) assign the highest model scores to the reference translations. As shown in Figure 4, while the proposal model selects one of the references as the best candidate for 22% of the sentences in the validation dataset, both rerankers double the ratio and the doc-reranker achieves 2% higher accuracy than the sent-reranker. This observation provides further evidence that if we improve the quality of the candidate pool our model will generate better translations.

Method	Model	Unpaired Data	LM PPL	Test17	Test18	Test19
Baseline	transformer big	–	–	23.9	23.9	24.5
This work	Doc-reranker	WMT	106.3	24.9	26.0	27.1
		Gigaword + WMT	63.8	**25.5**	**26.3**	**27.1**

Table 5: SacreBLEU of different models on WMT19 validation and test sets and perplexity per word of the language models on the English side of WMT19 validation set.

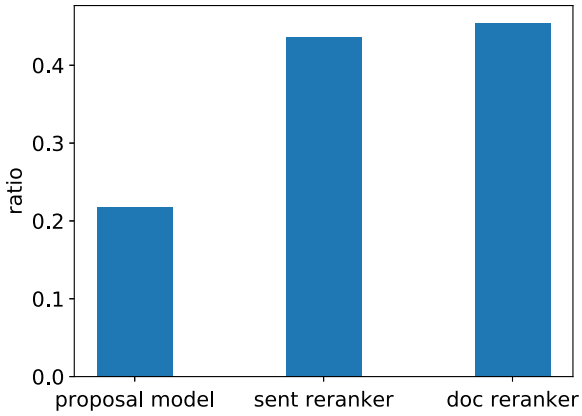

Figure 4: Ratio of different models picking true targets.

We also assess the diversity of the candidate pool and investigate the effect of their diversity on our model's performance. Table 6 lists pairwise-BLEU[7] scores (Shen et al., 2019) of different candidate pools (of size 50) and their corres ponding BLEU scores from the doc-reranker. We use the document transformer (Zhang et al., 2018) trained with additional backtranslated synthetic documents as the proposal models (q' in Table 1) in the doc-reranker. Table 6 shows that the candidates generated from our proposal model (by taking 50 best sentences from the beam search) are much less diverse than human translations. We conjecture that the lack of diversity in the candidate pool may harm the performance of our model.

To increase the diversity of candidate translations, we create candidate pools by composing translations generated from different ''experts'', which are simply document transformer models trained from different random initializations. As illustrated in Table 6, we find that a candidate pool from more experts results in more diverse translations (quantified by pairwise BLEU) and better reranking results (quantified by BLEU).

[7]Pairwise-BLEU (Shen et al., 2019) is a metric of measuring the similarity of candidate translations. The lower the pairwise-BLEU is, the more diverse the candidate translations are. We refer the readers to Shen et al. (2019) for the definition of the metric.

Proposal	#Experts	pBLEU	BLEU
human	4	21.40	–
Doc-transformer	1	70.41	53.63
	2	59.09	54.70
	4	**53.54**	**55.21**

Table 6: Pairwise-BLEU (pBLEU) (Shen et al., 2019) for candidate translations generated from different number of experts. BLEU from the doc-reranker taking different sets of candidate translations. We obtain different experts by training the document transformer (Zhang et al., 2018) with backtranslation with different random initialization. The size of the candidate pool is 50. The experts for the human proposal baseline are the reference translations.

5.2 Qualitative Analysis

To investigate how the rerankers improve translation quality, we analyze the output from different models: The document transformer (Zhang et al., 2018) (our proposal model), the sent-reranker, and the doc-reranker. We observe that in general the doc-reranker improves adequacy of translations and can generate more fluent and natural sentences compared with the document transformer. More importantly, our doc-reranker shows its superiority over the others in terms of exploiting context, improving consistency of tense, number, and lexical choice across entire articles. Tables 7 and 8 in Appendix A present example output from the aforementioned systems.

In Example 1, the pronoun *he* is omitted in the Chinese sentence. While the document transformer misses this pronoun resulting in a translation of completely different meaning, the doc-reranker is able to recover it. Likewise, in Example 6 *them* is dropped in the source sentence and this pronoun can only be inferred from previous context. Although both rerankers recover some pronoun, only the doc-reranker gets it right, by relying on cross-sentential context. Example 2 is a good example showing that the doc-reranker is better at generating adequate translations than the proposal model: the document transformer ignores the phrase *with these people*, but the doc-reranker covers it.

Chinese does not mark nouns for number, and it therefore has to be inferred from context to generate accurate English translations. It is not possible for a sentence-level MT system to capture this information if the relevant context is not from the current sentence. In Example 3 and 5 the plural *problems* and *identities* can only be inferred from previous sentences (the immediate previous sentence in Example 3 and the sentence 4-5 sentences away from the current one in Example 5). While neither the document transformer nor the sent-reranker makes the right predictions in both examples, the doc-reranker translates correctly, indicating its strength in capturing extra-sentential information. In addition to making inference across sentences, the doc-reranker is also capable of maintaining consistency of tense and lexical choice, as demonstrated in Examples 4, 7, and 9. Furthermore, it improves the consistency of writing style. To illustrate, in Example 8, the context is that of a list of bullet points starting with *continue*. The doc-reranker follows in this style by starting the translation with the verb *continue*. However, the sent-reranker starts the sentence with *we should continue*. Although both translations are reasonable, the former one is more natural within the document since it preserves stylistic consistency.

6 Related Work

Our work is closely related to three lines of research: context-aware neural machine translation, large-scale language models for language understanding, and semi-supervised machine translation. Recent studies (Tiedemann and Scherrer, 2017; Bawden et al., 2018, inter alia) have shown that exploiting document-level context improves translation performance, and in particular improves lexical consistency and coherence of the translated text. Existing work in the area of context-aware NMT typically adapts the MT system to take additional context as input, either a few previous sentences (Jean et al., 2017; Wang et al., 2017; Tu et al., 2018; Voita et al., 2018; Zhang et al., 2018; Werlen et al., 2018) or the full document (Haffari and Maruf, 2018; Maruf et al., 2019). These methods vary in the method of encoding the additional context and the way of integrating the context with the existing sequence-to-sequence models. For example, Werlen et al. (2018) encode the context with a separate transformer encoder (Vaswani et al., 2017) and use a hierarchical attention model to integrate the context into the rest of transformer model. Zhang et al. (2018) introduce an extra self-attention layer in the encoder to attend over the the context.

Strategies for exploiting monolingual document-level data have been explored in two recent studies (Voita et al., 2019a; Junczys-Dowmunt, 2019). Both use backtranslation (Edunov et al., 2018; Sennrich et al., 2016a) to create synthetic parallel documents as additional training data. In contrast, we train a large-scale language model and use it to refine the consistency between sentences under a noisy channel framework. Advantages of our model over back-translation are that 1) the language model is portable across domain and language pairs; 2) our model involves straightforward training procedures. Specifically, for backtranslation to succeed, monolingual data that will be back-translated must be carefully selected; the ratio of backtranslated data and original data must be balanced carefully. While techniques for doing this are fairly well established for single sentence models, no such established techniques exist for documents.

More generally, strategies for using monolingual data in nueral MT systems is an active research area (Gülçehre et al., 2015; Cheng et al., 2016, inter alia). Backtranslation (Edunov et al., 2018; Sennrich et al., 2016a), originally invented for semi-supervised MT, has been used as a standard approach for unsupervised MT (Lample et al., 2018a,b; Artetxe et al., 2019, 2018). Noisy channel decompositions have been a standard approach in statistical machine translation (Brown et al., 1993; Koehn et al., 2007) and recently have been applied to neural models (Yu et al., 2017; Yee et al., 2019;

Ng et al., 2019). Unlike prior work, we adopt noisy channel models for document-level MT. While the model from Yu et al. (2017) could be used on documents by concatenating their sentences to form a single long sequence, this would not let us use the conditional sentence independence assumptions that gives our model the flexibility to use just parallel sentences. Secondarily, their inference algorithm is specialized to their channel model, and it has a quadratic complexity, which would be prohibitive for sequence longer than a single sentence; in practice our inference technique is much faster.

Large-scale pretrained language models have achieved success in improving systems in language understanding, leading to state-of-the-art results on a wide range of tasks (Peters et al., 2018; Devlin et al., 2019; Radford et al., 2018; McCann et al., 2017; Yang et al., 2019; Chronopoulou et al., 2019; Lample and Conneau, 2019). Language generation is another area where pretrained language models have been applied, with existing work focusing on fine-tuning for repurposing an unconditional language model (Zhang et al., 2019; Edunov et al., 2019; Song et al., 2019; Dong et al., 2019; Ziegler et al., 2019; de Oliveira and Rodrigo, 2019). In contrast to our work, which uses probabilities from languagemodels, that work uses model internal representations.

7 Conclusion

We have presented a noisy channel reranker and empirically validated it on Chinese–English document-level translation tasks. The noisy channel formulation requires only parallel sentences (rather than documents) but we can use abundant monolingual documents to train the language model component. Experiments show that our proposed model considerably improves translation quality—it achieves approximately 2.5 BLEU higher than transformer baselines. Subjective evaluation further confirms that the language model helps enforce consistency of tense, number, and lexical choice across documents.

A Appendix

A.1 Human Evaluation

We selected 50 translation triplets (reference translation, translation from the doc-reranker, translation from the sent-reranker) sampled from the validation and test sets of NIST for evaluation by four native English speakers. The samples are selected by taking the triplets where the output from the sent-reranker and the doc-reranker have a translation edit rate (Snover et al., 2006) above 17.5%.

Each of these documents was presented with a reference translation, and with two translations, labeled A and B, one generated by the doc-reranker and one generated by the sent-reranker. They were tasked with indicating which of these two they found better overall, considering fluency, idiomaticness and correctness (relatively to the reference).

Each of the human evaluators preferred a majority of doc-reranker translations. When aggregated for each document by majority vote, the doc-reranker translations were considered better in 25 documents, worse for 13, and tied for 12. A statistically significant preference at $p < 0.05$ according to an exact one-tailed Binomial test ($n = 38$).

A.2 Comparison of Output from Different Systems

To investigate how the rerankers improve translation quality, we manually inspect the output from three different systems: the document transformer (Zhang et al., 2018), the sent-reranker, and the doc-reranker. Tables 7 and 8 present the comparison between the output from the document transformer (Zhang et al., 2018) and sent-reranker and between the output from sent-reranker and doc-reranker, respectively. In general, we find that the doc-reranker outperforms other systems in terms of maintaining consistency of tense, number, and lexical choices across documents. For detailed analysis, we refer readers to §5.2.

1	**src:**	霍夫曼在接受美国哥伦比亚广播公司新闻杂志「六十分钟」访问时轻叹,那段时期为了得到毒品和酒,真是不择手段。
	ref:	in an interview on us cbs news magazine 60 minutes, hoffman softly sighed that in such period he would truly do anything to get drugs and alcohol.
	out1:	in an interview with the cbs news magazine ‘‘60 minutes", hoffmann sighed that those days were really unscrupulous in getting drugs and alcohol.
	out2:	in an interview with the cbs news magazine ‘‘60 minutes", hoffmann sighed that at that time in order to obtain drugs and alcohol, he was really unscrupulous.
2	**ref:**	in the meantime, more than 10 chinese personnel working in the same place with these people have been called back to karachi. at present they are emotionally stabilized.
	out1:	at the same time, more than ten chinese personnel working at the same site have also withdrawn to karachi. their sentiments are now stable.
	out2:	at the same time, more than ten chinese personnel working with these people on the same site have also withdrawn to karachi. at present, their sentiments are stable.
3	**src:**	基本的问题是什么呢?
	cxt:	. . . however, legislator yeung, i wish to tell you what i am doing today is to ensure every matter can proceed smoothly after the political review starts. therefore, we have to solve some basic problems first and this is a different thing all together.
	ref:	what are the basic problems?
	out1:	what is the basic problem?
	out2:	what are the basic questions?
4	**cxt:**	sword of justice: prospects for 2006
	ref:	author: sword of justice
	out1:	author: the sword of righteousness
	out2:	author: the sword of justice

Table 7: Example outputs from the document transformer (out1) and our doc-reranker (out2).

5	**src:**	同时我们在国内用最短的时间，核实清楚了死亡人员的身份。
	cxt:	. . . the criminal used a submachine gun to fire a barrage of shots, and three engineers died unfortunately. . . .
	ref:	at the same time, we in china verified the identities of the dead within the shortest possible time.
	out1:	at the same time, we spent the shortest time in china to verify the identity of the deceased.
	out2:	at the same time, we spent the shortest time in china to verify the identities of the deceased.
6	**src:**	现在又要平安的送到家里。
	cxt:	. . . when the plane carrying the three survivors and 11 other personnel arrived in Hefei, people waiting at the airport heaved a long sigh of relief. . . . after the incident occurred, it made proper arrangements for them.
	ref:	now they will also be escorted home safely.
	out1:	now they have to send it home safely.
	out2:	now they want to send them safely to their homes.
7	**cxt:**	. . . a traffic accident occurred at the 58 kilometer point of the beijing-harbin highway, with a spill from an oil tanker leading to the closure of a section of the highway. . . . it was learned that the oil tanker contained waste oil from charcoal production. . . .
	ref:	the section of the highway from harbin to shuangcheng was closed, with many vehicles detoured.
	out1:	part of the roads heading towards shuangcheng in harbin are closed, and many vehicles are bypassing.
	out2:	part of the road from harbin to shuangcheng was closed , and many vehicles were bypassing.
8	**cxt:**	. . . with regard to coalmine safety this year, saws will effectively carry out the following three tasks: –continue to effectively tackle the tough issue of controlling methane. . . .
	ref:	– continue to effectively tackle the tough issue of restructuring and shutting down.
	out1:	– we should continue to make a success of the rectification and closure battle.
	out2:	– continue to fight the battle of rectification and closure.
9	**cxt:**	. . . first, such abuse of ‘‘quota" restricts the thorough implementation of world trade organization’s free trade principle. on one hand, u.s. is talking in high-sounding tone about ‘‘free trade". on the other hand, it re-establishes trade barriers and stabs your back at will with ‘‘quotas". does it appear too arbitrary and unfair?
	ref:	second, ‘‘quota" limits the nice growth trend in sino-america trade relation.
	out1:	second, the ‘‘restriction" restricts the good development momentum of sino-us economic and trade relations.
	out2:	second, the ‘‘quota" restricts the good development momentum of sino-us economic and trade relations.

Table 8: Example outputs from the sent-reranker (out1) and the doc-reranker (out2). cxt refers to context.

Acknowledgment

We would like to thank Gábor Melis for helpful comments on an earlier draft of this paper and the language team at DeepMind for valuable discussions.

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2019. An effective approach to unsupervised machine translation. In *Proceedings of ACL*.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *Proceedings of ICLR*.

Rachel Bawden, Rico Sennrich, Alexandra Birch, and Barry Haddow. 2018. Evaluating discourse phenomena in neural machine translation. In *Proceedings of NAACL-HLT*.

Peter F. Brown, Stephen 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.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. In *Proceedings of ACL*.

Alexandra Chronopoulou, Christos Baziotis, and Alexandros Potamianos. 2019. An embarrassingly simple approach for transfer learning from pretrained language models. In *Proceedings of NAACL-HLT*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc Viet Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *CoRR*, abs/1905.03197.

Sergey Edunov, Alexei Baevski, and Michael Auli. 2019. Pre-trained language model representations for language generation. In *Proceedings of NAACL-HLT*.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of EMNLP*.

William A. Gale and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102.

Çaglar Gülçehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *CoRR*, abs/1503.03535.

Gholamreza Haffari and Sameen Maruf. 2018. Document context neural machine translation with memory networks. In *Proceedings of ACL*.

Sébastien Jean, Stanislas Lauly, Orhan Firat, and Kyunghyun Cho. 2017. Does neural machine translation benefit from larger context? *CoRR*, abs/1704.05135.

Marcin Junczys-Dowmunt. 2019. Microsoft translator at WMT 2019: Towards large-scale document-level neural machine translation. In *Proceedings of WMT*.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of EMNLP*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL*.

Shaohui Kuang, Deyi Xiong, Weihua Luo, and Guodong Zhou. 2017. Cache-based document-level neural machine translation. *CoRR*, abs/1711.11221.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018a. Unsupervised machine translation using monolingual corpora only. In *Proceedings of ICLR*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018b. Phrase-based & neural unsupervised machine translation. In *Proceedings of EMNLP*.

Lidia Mangu, Eric Brill, and Andreas Stolcke. 2000. Finding consensus in speech recognition: word error minimization and other applications of confusion networks. *Computer Speech & Language*, 14(4).

Sameen Maruf, André F. T. Martins, and Gholamreza Haffari. 2019. Selective attention for context-aware neural machine translation. In *Proceedings of NAACL-HLT*.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Proceedings of NeurIPS*.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018a. An analysis of neural language modeling at multiple scales. *CoRR*, abs/1803.08240.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018b. Regularizing and optimizing LSTM language models. In *Proceedings of ICLR*.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook fair's WMT19 news translation task submission. In *Proceedings of WMT*.

Luke de Oliveira and Alfredo Láinez Rodrigo. 2019. Repurposing decoder-transformer language models for abstractive summarization. *ArXiv*, abs/1909.00325.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of WMT*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of ACL*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of ACL*.

Ross D. Shachter. 1998. Bayes-ball: The rational pastime (for determining irrelevance and requisite information in belief networks and influence diagrams). In *Proceedings of UAI*.

Tianxiao Shen, Myle Ott, Michael Auli, and Marc'Aurelio Ranzato. 2019. Mixture models for diverse machine translation: Tricks of the trade. In *Proceedings of ICML*.

Matthew Snover, Bonnie Dorr, Richard Schwarz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of AMTA*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked sequence to sequence pre-training for language generation. In *Proceedings of ICML*.

Meng Sun, Bojian Jiang, Hao Xiong, Zhongjun He, Hua Wu, and Haifeng Wang. 2019. Baidu neural machine translation systems for WMT19. In *Proceedings of WMT*.

Jörg Tiedemann and Yves Scherrer. 2017. Neural machine translation with extended context. In *Proceedings of DiscoMT@EMNLP*.

Zhaopeng Tu, Yang Liu, Shuming Shi, and Tong Zhang. 2018. Learning to remember translation history with a continuous cache. *Transactions of the Association for Computational Linguistics*, 6:407–420.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NeurIPS*.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019a. Context-aware monolingual repair for neural machine translation. In *Proceedings of EMNLP-IJCNLP*.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019b. When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion. In *Proceedings of ACL*.

Elena Voita, Pavel Serdyukov, Rico Sennrich, and Ivan Titov. 2018. Context-aware neural machine translation learns anaphora resolution. In *Proceedings of ACL*.

Longyue Wang, Zhaopeng Tu, Andy Way, and Qun Liu. 2017. Exploiting cross-sentence context for neural machine translation. In *Proceedings of EMNLP*.

Lesly Miculicich Werlen, Dhananjay Ram, Nikolaos Pappas, and James Henderson. 2018. Document-level neural machine translation with hierarchical attention networks. In *Proceedings of EMNLP*.

Yingce Xia, Xu Tan, Fei Tian, Fei Gao, Di He, Weicong Chen, Yang Fan, Linyuan Gong, Yichong Leng, Renqian Luo, et al. 2019. Microsoft Research Asia's systems for WMT19. In *Proceedings of WMT*.

Hao Xiong, Zhongjun He, Hua Wu, and Haifeng Wang. 2019. Modeling coherence for discourse neural machine translation. In *Proceedings of AAAI*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.

Kyra Yee, Yann Dauphin, and Michael Auli. 2019. Simple and effective noisy channel modeling for neural machine translation. In *Proceedings of EMNLP*.

Lei Yu, Phil Blunsom, Chris Dyer, Edward Grefenstette, and Tomás Kociský. 2017. The neural noisy channel. In *Proceedings of ICLR*.

Lei Yu, Jan Buys, and Phil Blunsom. 2016. Online segment to segment neural transduction. In *Proceedings of EMNLP*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *arXiv preprint arXiv:1905.12616*.

Haoyu Zhang, Yeyun Gong, Yu Yan, Nan Duan, Jianjun Xu, Ji Wang, Ming Gong, and Ming Zhou. 2019. Pretraining-based natural language generation for text summarization. *CoRR*, abs/1902.09243.

Jiacheng Zhang, Huanbo Luan, Maosong Sun, Feifei Zhai, Jingfang Xu, Min Zhang, and Yang Liu. 2018. Improving the transformer translation model with document-level context. In *Proceedings of EMNLP*.

Zachary M. Ziegler, Luke Melas-Kyriazi, Sebastian Gehrmann, and Alexander M. Rush. 2019. Encoder-agnostic adaptation for conditional language generation. *CoRR*, abs/1908.06938.