Encoding Sentence Position in Context-Aware Neural Machine Translation with Concatenation

Lorenzo Lupo[1] **Marco Dinarelli**[1] **Laurent Besacier**[2]
[1]Université Grenoble Alpes, France
[2]Naver Labs Europe, France
lorenzo.lupo@univ-grenoble-alpes.fr
marco.dinarelli@univ-grenoble-alpes.fr
laurent.besacier@naverlabs.com

Abstract

Context-aware translation can be achieved by processing a concatenation of consecutive sentences with the standard Transformer architecture. This paper investigates the intuitive idea of providing the model with explicit information about the position of the sentences contained in the concatenation window. We compare various methods to encode sentence positions into token representations, including novel methods. Our results show that the Transformer benefits from certain sentence position encodings methods on En→Ru, if trained with a context-discounted loss (Lupo et al., 2022b). However, the same benefits are not observed on En→De. Further empirical efforts are necessary to define the conditions under which the proposed approach is beneficial.

1 Introduction

Current neural machine translation (NMT) systems have reached human-like quality in translating standalone sentences, but there is still room for improvement when it comes to translating entire documents (Läubli et al., 2018; Castilho et al., 2020). Researchers have attempted to close this gap by developing various context-aware NMT (CANMT) approaches, where *context* refers to the sentences preceding or following the *current* sentence to be translated. A common approach to CANMT is sentence concatenation (Tiedemann and Scherrer, 2017; Agrawal et al., 2018; Junczys-Dowmunt, 2019). The current sentence and its context are concatenated into a unique sequence that is fed to the standard Transformer architecture (Vaswani et al., 2017). Despite its simplicity, the concatenation approach has been shown to achieve competitive or superior performance to more sophisticated, multi-encoding systems (Lopes et al., 2020; Lupo et al., 2022a). However, learning with long concatenation sequences has been proven challenging for the Transformer architecture, because the self-attention can be "distracted" by long context (Zhang et al., 2020; Bao et al., 2021).

Recently, Lupo et al. (2022b) introduced the *segment-shifted position embeddings* as a way to help concatenation approaches discerning the sentences concatenated in the processed sequence and improve attention's local focus. Explicitly telling the model which tokens belong to each sentence is not a new idea, but an intuitive one that was already tested successfully in other tasks and approaches (Devlin et al., 2019; Voita et al., 2018; Zheng et al., 2020). We believe that encoding into token representations explicit information about the position of the sentences in the concatenation sequence can improve translation quality. The temporal structure of the document constitutes essential information for its understanding and for the correct disambiguation of inter-sentential discourse phenomena. This work investigates this intuitive idea by comparing various approaches to encoding sentence position in concatenation approaches.

Our contributions are the following: (i) we compare segment-shifted position embeddings with three kinds of segment embeddings, evaluating their impact on the performance of the concatenation approach; (ii) we propose and evaluate making sentence position encodings persistent over layers, adding them to the input of every layer in addition to the first; (iii) we propose and evaluate fusing position embeddings and segment embeddings into a single vector where token and sentence positions are encoded in two orthogonal sets of dimensions, allowing a clearer distinction between them, along with memory savings.

To the best of our knowledge, this is the first comparative study on the employment of sentence position encodings for CANMT. The sentence position encoding variants proposed are not found to improve the performance of the concatenation approach except for one specific setting where a context-discounted training loss is employed (Lupo

The Fourth Workshop on Insights from Negative Results in NLP, pages 33–44
May 5, 2023

et al., 2022b). More empirical studies are needed to clearly define the conditions under which the proposed approaches are beneficial to CANMT with concatenation. Nonetheless, we find it useful to share these preliminary results with the scientific community. In fact, the proposed approaches are intuitive and easy to implement, hence something that many practitioners would presumably try. We hope that our findings can guide future research on sentence position encodings, by avoiding redundant experiments on failing settings.

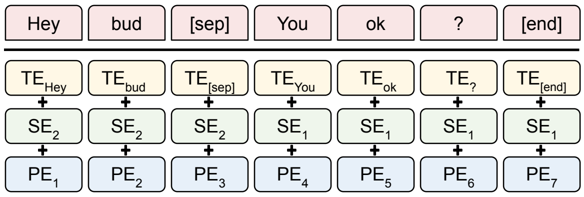
2 Proposed approach

A common method for training a concatenation model and translating is by sliding windows (Tiedemann and Scherrer, 2017). The sliding concatenation approach sKtoK translates a window $\boldsymbol{x}_K^j = \boldsymbol{x}^{j-K+1}\boldsymbol{x}^{j-K+2}\cdots\boldsymbol{x}^{j-1}\boldsymbol{x}^j$, of K consecutive sentences belonging to the source document, including the current (jth) sentence and $K-1$ context sentences, into $\boldsymbol{y}_K^j$. In this work we only consider past context, although future context can also be present in the concatenation window. At training time, the standard NMT loss is calculated over the whole output $\boldsymbol{y}_K^j$. At inference time, only the translation $\boldsymbol{y}^j$ of the current sentence is kept, while the context translation is discarded. Then, the window is slid by one sentence forward to repeat the process for the $(j+1)$th sentence and its context.

2.1 Sentence position encodings

To improve the discernability of the sentences concatenated in the window, we propose to equip the sKtoK approach with sentence position encodings. In particular, we experiment with segment-shifted position embeddings and three segment embedding methods. **Segment-shifted position embeddings** (Lupo et al., 2022b) consist in a slight modification of the Transformer's token position scheme, where the original token positions are shifted by a constant factor every time a new sentence is encountered in the concatenation window. The resulting positions are encoded with sinusoidal embeddings as for Vaswani et al. (2017).

We also experiment with **one-hot**, **sinusoidal**, and **learned segment embeddings**, like BERT's segment embeddings (Devlin et al., 2019). Segment embeddings encode the position k of each sentence within the window of K concatenated sentences into a vector of size d. We attribute sentence positions $k = 1, 2, ..., K$ starting from right to left. The underlying rationale is always to attribute the position $k = 1$ to the current sentence, no matter how many sentences are concatenated as context. The simplest strategy to integrate segment embeddings (SE) with position embeddings (PE) and token embeddings (TE) is by adding them (Devlin et al., 2019). This operation requires that all three embeddings have same dimensionality d_{model}:



2.2 Persistent encodings

We propose to make sentence position encodings persistent across Transformer's blocks, as Liu et al. (2020) did for position embeddings. In other words, we propose adding segment-shifted position embeddings or segment embeddings to each block's input instead of limiting to the first one.

2.3 Position-segment embeddings (PSE)

In the Transformer, position embeddings are sinusoidal. Their sum with the learnable token embeddings is based on the premise that the model can still distinguish both signals after being added up. This distinction is accomplished by learning token embeddings in a way that guarantees them to be distinguishable. Adding non-learnable segment embedding to this sum, however, rises the question whether they can be distinguished from the sinusoidal position embeddings. In some cases, learning to distinguish these two sources of information after their sum might be impossible. For instance, if segment embeddings are sinusoidal too, their sum with sinusoidal position embeddings is not bijective.[1]

Instead, concatenating PE and SE would make them perfectly distinguishable because they would belong to orthogonal spaces. Unfortunately, concatenating two d_{model}-dimensional embeddings would then oblige to project the resulting vector back to a d_{model}-dimensional space. To avoid this expensive operation, we propose to reduce the dimensionality of PE and SE from $d_{PE} = d_{SE} = d_{model}$ to values that sum up to the model dimension, i.e., $d_{PE} + d_{SE} = d_{model}$. Thus, each

[1] Consider, for example, the equivalence between, $PE_t + SE_k$ and $PE_k + SE_t$.

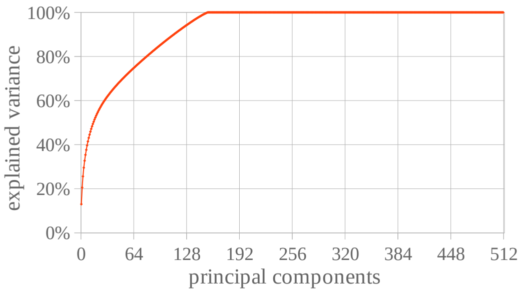


Figure 1: Cumulative ratio of the variance explained by the principal components of the of the sinusoidal position embedding matrix $PE \in \mathcal{R}^{1024\times512}$, representing 1024 positions with 512 dimensions. Less than half of the principal components can explain the entirety of the variance represented in the sinusoidal embeddings. In other words, 1024 positions can be represented with the same resolution using less than half the dimensions.

PE-SE pair can be concatenated into a unique vector named *position-segment embedding* (PSE): $PSE_{t,k} = [PE_t, SE_k]$, of size d_{model}. Reducing the dimensionality of PE and SE can be made without loss of information up to a certain degree, as it can be shown with a Principal Component Analysis (Jolliffe and Cadima, 2016) of the sinusoidal position embedding matrix (Figure 1).

In the experimental section, we will empirically evaluate the impact of representing token and sentence positions with PSE, where the former are encoded with sinusoids and the latter with either one-hot, sinusoidal, or learned representations.

3 Experiments

We experiment with two models: ***base***, a context-agnostic *Transformer-base* (Vaswani et al., 2017), and **s4to4**, a context-sensitive concatenation approach with the same architecture as *base*. s4to4 process sliding windows of 4 concatenated sentences in input and decodes the whole window into the target language. We equip s4to4 with the sentence position encoding options presented in the previous Section, and we evaluate their impact on performance. When experimenting with PSE, we allocate 4 dimensions to segment embeddings ($d_{SE} = 4$), which is enough to encode the position of each of the 4 sentences in the concatenation window, with both one-hot and sinusoidal encodings. Since $d_{model} = 512$, this leaves $d_{PE} = 508$ dimensions available to the sinusoidal representation of token positions.

The models are trained and evaluated on two language pairs covering different domains: En→Ru movie subtitles prepared by Voita et al. (2019), and En→De TED talk subtitles released by IWSLT17 (Cettolo et al. (2012), see Table 6 for statistics). In addition to evaluating the average translation quality with BLEU[2], we employ two contrastive sets to evaluate the translation of context-dependent anaphoric pronouns. For En→Ru, we adopt Voita et al. (2019)'s set for the evaluation of inter-sentential deixis, lexical cohesion, verb-phrase ellipsis, and inflectional ellipsis. For En→De, we evaluate the models on the translation of context-dependent ambiguous pronouns with ContraPro (Müller et al., 2018), a large set of contrastive translations of inter-sentential pronominal anaphora. Appendix B includes more setup details. The implementation of our experiments is open-sourced on GitHub.[3]

3.1 Results

First, we study the impact of sentence position encodings in the En→Ru setting. In Table 1, we compare models equipped with different combinations of encodings (Enc.) and integration methods: persistency (Pers.) and fusion with position encodings (PSE). We primarily focus on the contrastive evaluation of discourse translation since average translation quality metrics like BLEU have been repeatedly shown to be ill-equipped to detect improvements in CANMT (Hardmeier, 2012). Indeed, BLEU displays negligible fluctuations throughout the whole table. However, the performance on the contrastive sets is not encouraging either: most of the encoding variants degrade s4to4'sperformance. The one-hot encoding helps, but only by a thin margin. Making encoding persistent or concatenating them into PSE does not help either. The only exception is s4to4+lrn+pers+PSE (last line), which gains more than two accuracy points over baseline. However, this result is solely driven by the net improvement on deixis disambiguation (almost +5 points, see Table 10), while the performance is degraded on the other three discourse phenomena. In conclusion, sentence position encodings do not seem to benefit the vanilla s4to4 approach.

3.1.1 Training with context-discounted loss

Following Lupo et al. (2022b), we hypothesize that sentence position encodings can be leveraged

[2] Moses' *multi-bleu-detok* (Koehn et al., 2007) for De, *multi-bleu* for lowercased Ru as Voita et al. (2019).

[3] https://github.com/lorelupo/focused-concat

System	Enc.	Pers.	PSE	Voita	BLEU
base				46.64	31.98
s4to4				72.02	32.45
s4to4	shift			71.28	32.27
s4to4	shift	✓		71.80	31.93
s4to4	1hot			**72.52**	32.61
s4to4	1hot	✓		71.44	32.42
s4to4	1hot		✓	71.24	32.33
s4to4	1hot	✓	✓	71.16	32.41
s4to4	sin			71.92	32.39
s4to4	sin	✓		71.20	32.38
s4to4	sin		✓	71.26	32.56
s4to4	sin	✓	✓	71.68	32.38
s4to4	lrn			71.80	32.56
s4to4	lrn	✓		71.40	32.50
s4to4	lrn		✓	70.36	32.37
s4to4	lrn	✓	✓	**73.20**	32.38

Table 1: En→Ru models' accuracy on Voita's contrastive set and BLEU on the test set. s4to4 models are equipped with sentence position encodings (Enc.) of four kinds: segment-shifted position embeddings, one-hot segment embeddings, sinusoidal segment embeddings, or learned segment embeddings. Persistent encodings (Pers.) are added to the input of each Transformer's block. Alternatively to being added, segment embeddings can be concatenated with position embeddings (PSE). Values in bold are the best within their block of rows and outperform the baselines (base, s4to4).

more effectively by training the concatenation approach with a context-discounted objective (see Appendix A for details). Indeed, the context-discounted objective function incentivizes distinguishing among different sentences. Table 2 displays the results of the s4to4+CD model equipped with the various combinations of encodings tested before, except the *non-persistent* PSE.[4] In this case, too, vanilla sentence encoding methods do not significantly help the s4to4+CD model. However, making the encodings persistent boosts performance in the case of segment-shifted positions (+2.52 accuracy points over s4to4+CD) and learned embeddings (+2.14). One-hot segment embeddings benefit only slightly (+0.48) from being persistent, while no improvement is measured in the case of sinusoidal segment embeddings. As discussed in Section 2.3, this was expected since one-hot or sinusoidal segment embeddings might not be distinguishable from sinusoidal position embeddings once they are added together. Instead, when one-hot and sinusoidal segment embeddings are concatenated to position embeddings into a unique PSE and made persistent, they boost s4to4+CD by +1.08 and +1.26 accuracy points, respectively.

[4]Since preliminary experiments where not encouraging, we do not provide results for the non-persistent PSE combination in order to economize experiments.

System	Enc.	Pers.	PSE	Voita	BLEU
base				46.64	31.98
s4to4				72.02	32.45
s4to4+CD				73.42	32.37
s4to4+CD	shift			73.56	32.45
s4to4+CD	shift	✓		**75.94**	31.98
s4to4+CD	1hot			73.06	32.35
s4to4+CD	1hot	✓		73.90	32.56
s4to4+CD	1hot	✓	✓	**74.50**	32.33
s4to4+CD	sin			73.48	32.53
s4to4+CD	sin	✓		73.40	32.52
s4to4+CD	sin	✓	✓	**74.68**	32.27
s4to4+CD	lrn			73.68	32.45
s4to4+CD	lrn	✓		**75.56**	32.43
s4to4+CD	lrn	✓	✓	74.48	32.35

Table 2: En→Ru context-discounted s4to4's accuracy on Voita's contrastive set and BLEU. Values in bold are the best within their block of rows and outperform the baselines (base, s4to4, s4to4+CD).

With the aim of evaluating the generalizability of these results to another language pair and domain, we train the context-discounted approach on the En→De IWSLT17 dataset and evaluate it on ContraPro (Müller et al., 2018).[5] Table 3 summarizes the results. Unfortunately, the improvements achieved on En→Ru do not transfer to this setting. The s4to4+CD slightly benefits from segment-shifted position embeddings, but the other approaches degrade its performance. We hypothesize that the model does not undergo sufficient training in this setting to reap the benefits of sentence position encodings. In En→De IWSLT17, the training data volume is smaller than in the En→Ru setting by an order of magnitude: 0.2 million sentences versus 6 million (see Table 6). Therefore, we extended the experiments on En→De by training models on millions of sentences. The details and results are presented in Appendix C and Table 7. Unfortunately, even in this case, the En→De s4to4+*CD* does not benefit from the proposed sentence position encoding options.

[5]We don't experiment again with one-hot encodings since it was the less promising approach on the En→Ru setting.

System	Enc.	Pers.	PSE	ContraPro	BLEU
base				43.57	29.63
s4to4				72.12	29.48
s4to4+CD				74.78	29.32
s4to4+CD	shift			74.56	29.20
s4to4+CD	shift	✓		71.46	27.50
s4to4+CD	sin			74.46	29.23
s4to4+CD	sin	✓		74.35	29.26
s4to4+CD	sin	✓	✓	74.02	28.73
s4to4+CD	lrn			72.49	28.35
s4to4+CD	lrn	✓		71.07	27.87
s4to4+CD	lrn	✓	✓	71.89	28.63

Table 3: Accuracy on ContraPro of models trained on En→De IWSLT17, and BLEU on the test set.

System[6]	Voita
Chen et al. (2021)	55.61
Sun et al. (2022)	58.13
Zheng et al. (2020)	63.30
Kang et al. (2020)	73.46
Lupo et al. (2022b)	73.56
Zhang et al. (2020)	75.61
s4to4 + shift$_{pers}$ + CD	**75.94**

Table 4: Benchmarking on En→Ru (accuracy).

4 Benchmarking

In Tables 4 and 5, we compare our best performing systems with other CANMT systems from the literature. For En→Ru (Table 4), we compare with works that adopted the same experimental conditions as ours. Our s4to4 concatenation approach trained with context discounting and persistent segment-shifted positions achieves the best accuracy on Voita's contrastive set. For En→De (Table 5), we compare to the works adopting Müller et al. (2018)'s contrastive set for evaluation, even if the training conditions are not comparable. Our s4to4+CD trained on the high resource setting (see Appendix C) is second of the list, by a negligible margin. Notably, Huo et al. (2020)'s system is also a concatenation approach, but trained on x10 parallel sentences with respect to our system. This comparison indicates that context discounting (Lupo et al., 2022b) makes training efficient.

[6]Whenever the cited works present and evaluate multiple systems, we compare to the best performing one. For the majority of these works, BLEU scores are not available for comparison on the same test set.

[7]Reported in Müller et al. (2018).

System[6]	ContraPro
Maruf et al. (2019)	45.04
Voita et al. (2018)[7]	49.04
Stojanovski and Fraser (2019)	57.64
Müller et al. (2018)	59.51
Lupo et al. (2022a)	61.09
Lopes et al. (2020)	70.8
Lupo et al. (2022b)	74.56
Majumder et al. (2022)	78.00
Fernandes et al. (2021)	80.35
Huo et al. (2020)	**82.60**
s4to4 + CD	**82.54**

Table 5: Benchmarking on En→De (accuracy).

5 Conclusions

Intending to improve concatenation approaches to context-aware NMT (CANMT), we investigated an intuitive idea: encoding into token representations the position of their sentence within the processed sequence. Besides adopting existing encoding methods (segment-shifted position embeddings and segment embeddings), we proposed a novel approach to integrate token and sentence position embeddings in a unique vector called position-segment embedding (PSE). We also propose to make sentence position encodings persistent throughout the model's layers.

We compared these encoding approaches on the En→Ru/De language pairs. Consistent improvements were observed on En→Ru when persistent sentence position encoding methods were used in conjunction with the context-discounted training objective proposed by Lupo et al. (2022b). However, results on En→De were negative.

Further research is needed to clearly define the conditions under which the proposed approaches are beneficial to CANMT with concatenation. We encourage practitioners to test the most promising sentence-position encodings - **persistent segment-shifted positions** - should they want to get the most out of their CANMT systems, but only in conjunction with **context discounting**.

Acknowledgements

We thank the anonymous reviewers for their insightful comments. This work has been partially supported by the Multidisciplinary Institute in Artificial Intelligence MIAI@Grenoble Alpes (ANR-19-P3IA-0003).

References

Ruchit Rajeshkumar Agrawal, Marco Turchi, and Matteo Negri. 2018. Contextual Handling in Neural Machine Translation: Look Behind, Ahead and on Both Sides. In *Proceedings of the 21st Annual Conference of the European Association for Machine Translation*, pages 11–20, Alacant, Spain.

Guangsheng Bao, Yue Zhang, Zhiyang Teng, Boxing Chen, and Weihua Luo. 2021. G-transformer for document-level 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 1: Long Papers)*, pages 3442–3455, Online. Association for Computational Linguistics.

Sheila Castilho, Maja Popović, and Andy Way. 2020. On context span needed for machine translation evaluation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3735–3742, Marseille, France. European Language Resources Association.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. WIT3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Annual conference of the European Association for Machine Translation*, pages 261–268, Trento, Italy. European Association for Machine Translation.

Linqing Chen, Junhui Li, Zhengxian Gong, Boxing Chen, Weihua Luo, Min Zhang, and Guodong Zhou. 2021. Breaking the corpus bottleneck for context-aware neural machine translation with cross-task pre-training. 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 2851–2861, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Patrick Fernandes, Kayo Yin, Graham Neubig, and André F. T. Martins. 2021. Measuring and increasing context usage in context-aware 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 1: Long Papers)*, pages 6467–6478, Online. Association for Computational Linguistics.

Christian Hardmeier. 2012. Discourse in Statistical Machine Translation. A Survey and a Case Study. *Discours. Revue de linguistique, psycholinguistique et informatique. A journal of linguistics, psycholinguistics and computational linguistics*, 1(11).

Jingjing Huo, Christian Herold, Yingbo Gao, Leonard Dahlmann, Shahram Khadivi, and Hermann Ney. 2020. Diving deep into context-aware neural machine translation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 604–616, Online. Association for Computational Linguistics.

Ian T. Jolliffe and Jorge Cadima. 2016. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202. Publisher: Royal Society.

Marcin Junczys-Dowmunt. 2019. Microsoft translator at WMT 2019: Towards large-scale document-level neural machine translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 225–233, Florence, Italy. Association for Computational Linguistics.

Xiaomian Kang, Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2020. Dynamic context selection for document-level neural machine translation via reinforcement learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2242–2254, Online. Association for Computational Linguistics.

Yunsu Kim, Duc Thanh Tran, and Hermann Ney. 2019. When and why is document-level context useful in neural machine translation? In *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, pages 24–34, Hong Kong, China. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Samuel Läubli, Rico Sennrich, and Martin Volk. 2018. Has machine translation achieved human parity? a case for document-level evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4791–4796, Brussels, Belgium. Association for Computational Linguistics.

Xuanqing Liu, Hsiang-Fu Yu, Inderjit S. Dhillon, and Cho-Jui Hsieh. 2020. Learning to encode position for transformer with continuous dynamical model. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020,* *Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6327–6335. PMLR.

António Lopes, M. Amin Farajian, Rachel Bawden, Michael Zhang, and André F. T. Martins. 2020. Document-level neural MT: A systematic comparison. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 225–234, Lisboa, Portugal. European Association for Machine Translation.

Lorenzo Lupo, Marco Dinarelli, and Laurent Besacier. 2022a. Divide and rule: Effective pre-training for context-aware multi-encoder translation models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4557–4572, Dublin, Ireland. Association for Computational Linguistics.

Lorenzo Lupo, Marco Dinarelli, and Laurent Besacier. 2022b. Focused Concatenation for Context-Aware Neural Machine Translation. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 830–842, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Zhiyi Ma, Sergey Edunov, and Michael Auli. 2021. A Comparison of Approaches to Document-level Machine Translation. *ArXiv preprint*, abs/2101.11040.

Suvodeep Majumder, Stanislas Lauly, Maria Nadejde, Marcello Federico, and Georgiana Dinu. 2022. A baseline revisited: Pushing the limits of multi-segment models for context-aware translation.

Sameen Maruf, André F. T. Martins, and Gholamreza Haffari. 2019. Selective attention for context-aware 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 3092–3102, Minneapolis, Minnesota. Association for Computational Linguistics.

Mathias Müller, Annette Rios, Elena Voita, and Rico Sennrich. 2018. A large-scale test set for the evaluation of context-aware pronoun translation in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 61–72, Brussels, Belgium. Association for Computational Linguistics.

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

Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing Neural Networks by Penalizing Confident Output Distributions. *ArXiv preprint*, abs/1701.06548.

Martin Popel and Ondřej Bojar. 2018. Training Tips for the Transformer Model. *ArXiv preprint*, abs/1804.00247.

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.

Dario Stojanovski and Alexander Fraser. 2019. Improving anaphora resolution in neural machine translation using curriculum learning. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 140–150, Dublin, Ireland. European Association for Machine Translation.

Zewei Sun, Mingxuan Wang, Hao Zhou, Chengqi Zhao, Shujian Huang, Jiajun Chen, and Lei Li. 2022. Rethinking document-level neural machine translation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3537–3548, Dublin, Ireland. Association for Computational Linguistics.

Jörg Tiedemann and Yves Scherrer. 2017. Neural machine translation with extended context. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 82–92, Copenhagen, Denmark. Association for Computational Linguistics.

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

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1198–1212, Florence, Italy. Association for Computational Linguistics.

Elena Voita, Pavel Serdyukov, Rico Sennrich, and Ivan Titov. 2018. Context-aware neural machine translation learns anaphora resolution. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1264–1274, Melbourne, Australia. Association for Computational Linguistics.

Pei Zhang, Boxing Chen, Niyu Ge, and Kai Fan. 2020. Long-short term masking transformer: A simple but effective baseline for document-level neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1081–1087, Online. Association for Computational Linguistics.

Zaixiang Zheng, Xiang Yue, Shujian Huang, Jiajun Chen, and Alexandra Birch. 2020. Towards making the most of context in neural machine translation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3983–3989. ijcai.org.

A Context-discounted loss

In CANMT with sliding concatenation windows we should prioritize the quality of the translation of the current sentence because the context translation will be discarded during inference. Therefore, the standard NMT objective function is not suitable in this case. Lupo et al. (2022b) propose to encourage the concatenation approach to focus on the translation of the current sentence $\boldsymbol{x}^j$ by applying a discount $0 \leq \text{CD} < 1$ to the loss generated by context tokens:

$$\begin{aligned} \mathcal{L}_{\text{CD}}(\boldsymbol{x}_K^j, \boldsymbol{y}_K^j) &= \text{CD}\cdot\mathcal{L}_{context} + \mathcal{L}_{current} \\ &= \text{CD}\cdot\mathcal{L}(\boldsymbol{x}_K^j, \boldsymbol{y}_{K-1}^{j-1}) + \mathcal{L}(\boldsymbol{x}_K^j, \boldsymbol{y}^j). \end{aligned} \quad (1)$$

with $\mathcal{L}(\boldsymbol{x}, \boldsymbol{y})$ being the standard NMT objective function:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{t=1}^{|\boldsymbol{y}|} \log P(y_t | \boldsymbol{y}_{<t}, \boldsymbol{x}), \quad (2)$$

The authors demonstrate the efficacy of this loss function, that leads to a self-attentive mechanism that is less influenced by noisy contextual information. As a result, they show a marked improvement in the translation of inter-sentential discourse phenomena.

B Details on experimental setup

All experiments are implemented in *fairseq* (Ott et al., 2019). All models follow the *Transformer-base* architecture (Vaswani et al., 2017): hidden size of 512, feed forward size of 2048, 6 layers, 8 attention heads. They are trained on 4 Tesla V100, with a fixed batch size of approximately 32k tokens for En→Ru and 16k for En→De, as it has been shown that Transformers need a large batch size to optimize performance (Popel and Bojar, 2018). We stop training after 12 consecutive non-improving validation steps (in terms of loss on dev), and we average the weights of the best-performing checkpoint and the 4 checkpoints that follow it. We train models with the optimizer configuration and learning rate (LR) schedule described in Vaswani et al. (2017). The maximum LR is optimized for each model over the search space $\{7e-4, 9e-4, 1e-3, 3e-3\}$. The LR achieving the best loss on the validation set after convergence was selected. We use label smoothing with an epsilon value of 0.1 (Pereyra et al., 2017) for all settings. We adopt strong model regularization (dropout=0.3) following Kim et al. (2019) and Ma et al. (2021). At inference time, we use beam search with a beam of 4 for all models. We adopt a length penalty of 0.6 for all models. The other hyperparameters were set according to the relevant literature (Vaswani et al., 2017; Popel and Bojar, 2018; Voita et al., 2019; Ma et al., 2021; Lopes et al., 2020). When experimenting with segment-shifted position embeddings, the shift is equal to the average sentence length calculated over the training data, following (Lupo et al., 2022b). In particular, we set shift$=8$ for En→Ru, shift$=21$ for En→De.

B.1 Data pre-processing

Since Voita's data have already been pre-processed (Voita et al., 2019), we only apply byte pair encoding (Sennrich et al., 2016) with 32k merge operations jointly for English and Russian. For IWSLT17, instead, we tokenize data with the Moses toolkit (Koehn et al., 2007), clean them by removing long sentences, and encode them with byte pair encoding. The byte pair encoding is learned on the En→De training data released by WMT17 for the news translation task using 32k merge operations jointly for source and target languages, to be compatible with the experiments presented in the next section of the Appendix (C).

C Increasing training data for the English to German pair

We hypothesize that the model does not undergo sufficient training in the En→De setting to reap the benefits of segment embeddings. Indeed, the training data volume is smaller than in the En→Ru setting: 0.2 million sentences versus 6 million (see Table 6). Therefore, we choose to experiment with more En→De training data, employing the same high-resource setting of Lupo et al. (2022a). This setting expands the IWSLT17 training data (Cettolo et al., 2012) by adding the News-Commentary-v12 and Europarl-v7 sets released

Corpus	Tgt	Docs	Sents	Doc Length			Sent Length			Sent Length (BPE)		
				mean	std	max	mean	std	max	mean	std	max
Voita	Ru	1.5M	6.0M	4.0	0.0	4	8.3	4.7	64	8.6	4.9	69
IWSLT17	De	1.7k	0.2M	117.0	58.4	386	20.8	14.3	153	23.3	16.3	195
High	De	12.2k	2.3M	188.4	36.2	386	27.3	16.1	249	29.1	17.4	408
Voita	Ru	10k	40k	4.0	0.0	4	8.2	4.8	50	8.5	5.0	58
Both	De	62	5.4k	87.6	53.5	296	19.0	12.5	114	21.1	14.0	132
Voita	Ru	10k	40k	4.0	0.0	4	8.2	4.8	42	8.5	5.0	50
Both	De	12	1.1k	90.0	29.2	151	19.3	12.7	102	21.6	14.3	116

Table 6: Statistics for the training (1st block), validation (2nd block) and test set (3rd block) after pre-processing, and after BPE tokenization. All figures refer to the English text (source side).

System	Enc.	Pers.	PSE	CP	BLEU
s4to4+CD				**82.24**	31.69
s4to4+CD	shift	✓		80.45	30.71
s4to4+CD	sin	✓	✓	80.85	31.40
s4to4+CD	lrn	✓		79.82	31.58

Table 7: Context-discounted s4to4 trained on the En→De high-resource setting, evaluated with the accuracy on ContraPro (CP) and BLEU on the test set.

by WMT17[8]. The resulting training set comprises 2.3M sentences (see statistics in Table 6). Training on this data is more expensive than training on the En→Ru setting, considering that the average sentence length is 27.3 tokens versus 8.3 tokens, respectively. Therefore, we only train the most promising approaches.[9] Their performances are compared in Table 7. As expected, the s4to4+CD model drastically improves its performance compared to training on IWSLT17 alone: +7.93 accuracy points on ContraPro and +2.37 BLEU points on the test set (c.f. Table 3). However, even with larger training volumes, segment position encodings do not seem to help s4to4+CD on the En→De language pair.

D Allocating more space to segments in PSE

For the En→Ru language pair, we have found that one-hot and sinusoidal segment embeddings need to be integrated into PSE for being leveraged by s4to4+CD (Section 3.1.1). Instead, learned embeddings worked best when added to position embeddings.

Here, we evaluate whether PSE with learned segment embeddings would perform better if more dimensions were allocated to segments. In particular, we let the model learn to represent sentence positions in $d_{SE} = 128$ dimensions, which leaves $d_{PE} = d_{model} - d_{SE} = 384$ dimensions to position embeddings, largely enough as shown in Section 2.3.

As shown in Table 8, increasing the number of dimensions allocated to segment embeddings deteriorates the performance on Voita's contrastive set. The reason could simply be that adding more learnable parameters makes the task harder.

E Persistent positions

Making sentence position encodings persistent across the layers have been found beneficial for context-discounted models on the En→Ru setting (Table 2). The best-performing model, s4to4+CD+shift+pers, shifts token positions by a constant factor every time we pass from one sentence to the next and makes the resulting position embeddings persistent throughout Transformer's blocks. In Table 9, we benchmark this model against models employing persistent token position embeddings but without segment-shifting. Both vanilla and context-discounted s4to4 perform better when positions are persistent across Transformer's blocks, as suggested by Liu et al. (2020) and Chen et al. (2021). Segment-shifting further enhances performance, which confirms that the model benefits from a sharper distinction between sentences.

[8]http://www.statmt.org/wmt17/translation-task.html

[9]We set shift= 27 for segment-shifted position embeddings, consistently with the average sentence length of the training data.

System	Enc.	Pers.	PSE	Deixis	Lex co.	Ell. inf	Ell. vp	Voita	BLEU
s4to4+CD	lrn	✓	4	93.20	47.40	72.20	64.40	74.48	32.35
s4to4+CD	lrn		128	83.88	46.33	65.20	50.20	67.38	32.43
s4to4+CD	lrn	✓	128	78.20	46.40	40.60	30.60	60.14	32.35

Table 8: s4to4 trained on En→Ru OpenSubtitles. Accuracy on Voita's En→Ru contrastive set and BLEU on the test set. The accuracy on the contrastive set is detailed on the left, with the accuracy on each subset corresponding to a specific discourse phenomenon. Result: allocating more dimensions to segments in PSE deteriorates performance.

System	Enc.	Pers.	PSE	Voita	BLEU
s4to4				72.02	32.45
s4to4		✓		**72.44**	32.29
s4to4+CD				73.42	32.37
s4to4+CD		✓		74.10	32.12
s4to4+CD	shift	✓		**75.94**	31.98

Table 9: En→Ru: making positions persistent across Transformer's blocks improve discourse disambiguation performance both for vanilla and context-discounted s4to4. Segment-shifting positions further improves performance.

F Details of the evaluation on discourse phenomena

In Tables 10 and 11, we provide more details on the evaluation of the models presented in the tables of the paper, documenting their accuracy on the different subsets of the contrastive sets employed. For Voita's En→Ru contrastive set (Voita et al., 2019), we report the accuracy on each of the 4 discourse phenomena included in it; for the En→De ContraPro (CP, Müller et al. (2018)), the accuracy on anaphoric pronouns with antecedents at different distances $d = 1, 2, ...$ (in number of sentences). We complement Voita/CP with two other metrics, Voita/CP$_{\text{avg}}$ and CP$_{d>0}$. Metrics are calculated as follow:

$$\text{Voita} = \frac{2500{*}\text{Deixis}+1500{*}\text{Lex co.}+500{*}\text{Ell. inf}+500{*}\text{Ell. vp}}{5000} \tag{3}$$

$$\text{CP}_{alld} = \frac{2400{*}(\text{d=0})+7075{*}(\text{d=1})+1510{*}(\text{d=2})+573{*}(\text{d=3})+442{*}(\text{d>3})}{12000} \tag{4}$$

$$\text{CP}_{d>0} = \frac{7075{*}(\text{d=1})+1510{*}(\text{d=2})+573{*}(\text{d=3})+442{*}(\text{d>3})}{9600} \tag{5}$$

$$\text{Voita}_{\text{avg}}/\text{CP}_{\text{avg}} = \frac{(\text{d=1}) + (\text{d=2}) + (\text{d=3}) + (\text{d=4})}{4} \tag{6}$$

System	Enc.	Pers.	PSE	Deixis	Lex co.	Ell. inf	Ell. vp	Voita	Voita$_{avg}$
base				50.00	45.87	51.80	27.00	46.64	43.67
s4to4				85.80	46.13	79.60	73.20	72.02	71.18
s4to4	shift			85.24	46.07	77.20	71.20	71.28	69.93
s4to4	shift	✓		**85.96**	**46.33**	75.20	**74.00**	71.80	70.37
s4to4	sin			**86.36**	45.80	76.40	73.60	71.92	70.54
s4to4	sin	✓		84.96	46.13	74.80	74.00	71.20	69.97
s4to4	sin		✓	84.64	**46.40**	76.60	73.60	71.26	70.31
s4to4	sin	✓	✓	85.24	46.33	76.40	**75.20**	71.68	70.79
s4to4	lrn			85.48	46.27	76.20	**75.60**	71.80	70.89
s4to4	lrn	✓		84.84	45.93	77.60	74.40	71.40	70.69
s4to4	lrn		✓	83.60	**46.67**	74.80	70.80	70.36	68.97
s4to4	lrn	✓	✓	**90.52**	46.00	74.80	66.60	**73.20**	69.48
s4to4	1hot			**86.08**	47.07	78.00	**75.60**	**72.52**	**71.69**
s4to4	1hot	✓		83.76	**47.53**	78.00	75.00	71.44	71.07
s4to4	1hot		✓	84.56	46.13	78.20	73.00	71.24	70.47
s4to4	1hot	✓	✓	84.56	46.47	76.00	73.40	71.16	70.11
s4to4+CD				87.16	46.40	81.00	78.20	73.42	73.19
s4to4+CD	shift			85.76	48.33	81.40	80.40	73.56	73.97
s4to4+CD	shift	✓		**88.76**	**52.13**	**83.00**	76.20	**75.94**	**75.02**
s4to4+CD	sin			87.96	46.80	78.00	76.60	73.48	72.34
s4to4+CD	sin	✓		86.80	**47.00**	80.80	78.20	73.40	73.20
s4to4+CD	sin	✓	✓	**89.28**	46.67	**83.20**	77.20	**74.68**	**74.09**
s4to4+CD	lrn			88.12	46.47	81.20	75.60	73.68	72.85
s4to4+CD	lrn	✓		86.84	**52.27**	**84.60**	**80.00**	**75.56**	**75.93**
s4to4+CD	lrn	✓	✓	**93.20**	47.40	72.20	64.40	74.48	69.30
s4to4+CD	1hot			86.40	46.73	82.00	76.40	73.06	72.88
s4to4+CD	1hot	✓		87.68	46.80	81.60	**78.60**	73.90	73.67
s4to4+CD	1hot	✓	✓	**88.88**	**47.67**	**82.20**	75.40	**74.50**	73.54
Sample size				2500	1500	500	500	5000	5000

Table 10: Accuracy on the En→Ru contrastive set for the evaluation of discourse phenomena (Voita, %), and on its 4 subsets: deixis, lexical cohesion, inflection ellipsis, and verb phrase ellipsis. Voita$_{avg}$ denotes the average on the 4 discourse phenomena, while Voita represents the average weighted by the frequency of each phenomenon in the test set (see row "Sample size").

System	Enc.	Pers.	PSE	d=0	d=1	d=2	d=3	d>3	$CP_{d>0}$	CP_{avg}	CP
base				68.75	32.89	43.97	47.99	70.58	37.27	48.86	43.57
s4to4				75.20	68.89	74.96	79.58	87.78	71.35	77.80	72.12
s4to4+CD				76.66	72.86	75.96	80.10	84.38	74.31	78.33	74.78
s4to4+CD	shift			75.25	72.56	77.15	80.27	86.65	74.39	79.16	74.56
s4to4+CD	shift	✓		72.41	69.15	74.23	77.13	86.42	71.22	76.73	71.46
s4to4+CD	sin			76.75	71.83	76.82	80.97	87.55	73.88	79.29	74.46
s4to4+CD	sin	✓		76.50	72.08	76.35	79.23	85.97	73.82	78.41	74.35
s4to4+CD	sin	✓	✓	77.25	71.22	76.42	78.88	86.87	73.22	78.35	74.02
s4to4+CD	lrn			73.91	70.21	75.29	77.66	85.06	72.14	77.06	72.49
s4to4+CD	lrn	✓		73.66	68.53	72.51	75.74	86.65	70.42	75.86	71.07
s4to4+CD	lrn	✓	✓	73.54	68.40	79.07	80.27	83.48	71.48	77.81	71.89
High Resource Setting											
base				82.83	35.18	44.90	51.13	66.28	39.09	49.37	47.84
s4to4				82.41	80.66	81.72	84.29	88.00	81.38	83.67	81.59
s4to4+CD				83.70	81.79	82.11	82.19	90.04	82.24	84.03	82.54
s4to4+CD	shift	✓		81.70	79.61	81.45	83.42	86.65	80.45	82.78	80.70
s4to4+CD	sin	✓	✓	84.12	79.85	82.38	84.46	86.87	80.85	83.39	81.50
s4to4+CD	lrn	✓		83.12	79.13	79.73	82.19	88.00	79.82	82.26	80.48
Sample size				2400	7075	1510	573	442	9600	9600	12000

Table 11: Accuracy on the En→De contrastive set for the evaluation of anaphoric pronouns (CP = ContraPro, %). The columns titled d=* represent the accuracy for each subset of pronouns with antecedents at a specific distance $d \in [0, 1, 2, 3, > 3]$ (in number of sentences). CP_{avg} denotes the average on the 4 subsets of pronouns with extra-sentential antecedents ($d > 0$) while $CP_{d>0}$ represents the average weighted by the size of each of the 4 subsets (see row "Sample size"). CP is equivalent to $CP_{d>0}$, but it includes the accuracy on $d = 0$.