Joint Generation of Captions and Subtitles with Dual Decoding

Jitao Xu, François Buet, Josep Crego, Elise Bertin-Lemée, François Yvon

Joint Generation of Captions and Subtitles with Dual Decoding

Jitao Xu[†] **François Buet**[†] **Josep Crego**[‡] **Elise Bertin-Lemée**[‡] **François Yvon**[†]

[†]Université Paris-Saclay, CNRS, LISN, 91400, Orsay, France
[‡]SYSTRAN, 5 rue Feydeau, 75002 Paris, France
{firstname.lastname}@{[†]limsi.fr,[‡]systrangroup.com}

Abstract

As the amount of audio-visual content increases, the need to develop automatic captioning and subtitling solutions to match the expectations of a growing international audience appears as the only viable way to boost throughput and lower the related post-production costs. Automatic captioning and subtitling often need to be tightly intertwined to achieve an appropriate level of consistency and synchronization with each other and with the video signal. In this work, we assess a dual decoding scheme to achieve a strong coupling between these two tasks and show how adequacy and consistency are increased, with virtually no additional cost in terms of model size and training complexity.

1 Introduction

As the amount of online audio-visual content continues to grow, the need for captions and subtitles[1] in multiple languages also steadily increases, as it widens the potential audience of these contents.

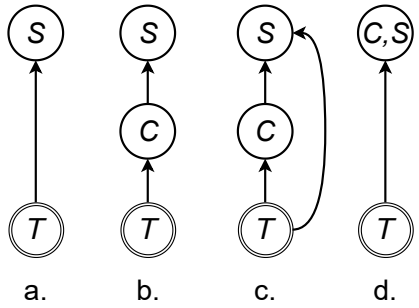

Figure 1: A graphical view of various captioning and subtitling strategies. T refers to transcripts. C and S respectively denote captions and subtitles.

[1]We use 'caption' to refer to a text written in the same language as the audio and 'subtitle' when translated into another language. Captions, which are often meant for viewers with hearing difficulties, and subtitles, which are produced for viewers with an imperfect command of the source language, may have slightly different traits, that we ignore here.

Both activities are closely related: human subtitle translators often generate subtitles directly based on the original captions without viewing or listening to the original audio/video file. This strategy however runs the risk of amplifying, in the subtitle approximations, simplifications or errors present in the captioning. It may even happen that both texts need to be simultaneously displayed on screen: for instance, in countries with several official languages, or to help foreign language learners. This means that captions and subtitles need to be consistent not only with the video content, but also with each other. It also implies that they should be synchronized (Karakanta et al., 2021). Finally, even in scenarios where only subtitles would be needed, generating captions at the same time may still help to better check the correctness of subtitles.

Early approaches to automatic subtitling (e.g. Piperidis et al., 2004) also assumed a pipeline architecture (Figure 1 (b)), where subtitles are translated from captions derived from automatic speech transcripts. A recent alternative (Figure 1 (a)), which mitigates cascading errors, is to independently perform captioning and subtitling in an end-to-end manner (Liu et al., 2020; Karakanta et al., 2020a); the risk however is to generate inconsistencies (both in alignment and content) between the two textual streams. This approach might also be limited by the lack of appropriate training resources (Sperber and Paulik, 2020). Various ways to further strengthen the interactions between these tasks by sharing parameters or loss terms are evaluated by Sperber et al. (2020). Figure 1 (c) illustrates these approaches.

In this work, we explore an even tighter integration consisting of *simultaneously generating both captions and subtitles* from automatic speech recognition (ASR) transcripts *using one single dual decoding process* (Zhou et al., 2019; Wang et al., 2019; Le et al., 2020; He et al., 2021; Xu and Yvon, 2021), illustrated in Figure 1 (d). Generally speak-

Transcript	**i 'm** combining specific types of signals **the** mimic how our body **response** to **in an injury** to help us regenerate
Caption	**I'm** combining specific types of signals **[eob] that** mimic how our body **responds** to **injury [eol]** to help us regenerate**. [eob]**
Subtitle	Je combine différents types de signaux [eob] qui imitent la réponse du corps [eol] aux blessures pour nous aider à guérir. [eob]

Table 1: Example of a triplet (transcript, caption, subtitle) from our tri-parallel data. Differences between transcript and caption are in bold.

ing, automatically turning ASR transcripts into full-fledged captions involves multiple changes, depending on the specification of the captioning task. In our case, this transformation comprises four main aspects: segmentation for display (via tag insertion), removal of certain features from spoken language (eg. fillers, repetitions or hesitations), ASR errors correction, and punctuation prediction. The transcript-to-subtitle task involves the same transformations, with an additional translation step to produce text in another language. Table 1 illustrates the various transformations that occur between input transcripts and the corresponding output segments.

As our experiments suggest, a tighter integration not only improves the quality and the consistency of captions and subtitles, but it also enables a better use of all available data, *with hardly any impact on model size or training complexity*. Our main contributions are the following: (i) we show that simultaneously generating captions and subtitles can improve performance in both languages, reporting significant improvements in BLEU score with respect to several baselines; (ii) we initialize dual decoder from a standard encoder-decoder model trained with large scale data, thereby mitigating the data scarcity problem; (iii) we explore a new parameter sharing scheme, where the two decoders share all their parameters, and achieve comparable performance at a much reduced model size in our experimental conditions; (iv) using 2-round decoding, we show how to alleviate the exposure bias problem observed in dual decoding, leading to a clear boost in performance.

2 Dual Decoding

2.1 Model

In a nutshell, dual decoding aims to generate two output sentences $\mathbf{e}^1$ and $\mathbf{e}^2$ for each input sentence $\mathbf{f}$. This means that instead of having two independent models (Eq. (1)), the generation of each target is influenced by the other output (Eq. (2)):

$$P(\mathbf{e}^1, \mathbf{e}^2|\mathbf{f}) = \prod_{t=1}^{T} P(\mathbf{e}_t^1|\mathbf{f}, \mathbf{e}_{<t}^1) P(\mathbf{e}_t^2|\mathbf{f}, \mathbf{e}_{<t}^2) \quad (1)$$

$$P(\mathbf{e}^1, \mathbf{e}^2|\mathbf{f}) = \prod_{t=1}^{T} P(\mathbf{e}_t^1|\mathbf{f}, \mathbf{e}_{<t}^1, \mathbf{e}_{<t}^2) \times P(\mathbf{e}_t^2|\mathbf{f}, \mathbf{e}_{<t}^1, \mathbf{e}_{<t}^2), \quad (2)$$

where $T = max(|\mathbf{e}^1|, |\mathbf{e}^2|)$.

In our experiments, ASR transcripts are considered as the source language while captions and subtitles are the two target languages (Wang et al., 2019; He et al., 2021; Xu and Yvon, 2021). The dual decoder model has also been proposed in several application scenarios other than multi-target translation such as bi-directional translation (Zhou et al., 2019; Zhang et al., 2020a; He et al., 2021), and also to simultaneously generate transcripts and translations from the audio source (Le et al., 2020).

To implement the interaction between the two decoders, we mostly follow Le et al. (2020) and Xu and Yvon (2021) who add a decoder cross-attention layer in each decoder block, so that the hidden states of previous layers of each decoder H_l^1 and H_l^2 can attend to each other. The decoder cross-attention layers take the form:[2]

$$H_{l+1}^1 = \text{Attention}(H_l^1, H_l^2, H_l^2)$$
$$H_{l+1}^2 = \text{Attention}(H_l^2, H_l^1, H_l^1)$$

Both decoders are thus fully synchronous since each requires the hidden states of the other to compute its own hidden states.

2.2 Sharing Decoders

One weakness of the dual decoder model is that it contains two separate decoders, yielding an increased number of parameters ($\times 1.6$ in our models w.r.t. standard translation models). Inspired by

[2]We define the $\text{Attention}(Q, K, V)$ function as in (Vaswani et al., 2017) as a function of three arguments standing respectively for Query, Key and Value.

the idea of tying parameters in embedding matrices (Inan et al., 2017; Press and Wolf, 2017), we extend the dual decoder model by *sharing all the parameters matrices in the two decoders*: in this way, the total number of parameters remains close to that of a standard translation model ($\times 1.1$), since the only increase comes from the additional decoder cross-attention layer. When implementing inference with this multilingual shared decoder, we prefix each target sentence with a tag indicating the intended output (captioning or subtitling).

2.3 Training and Fine-tuning

The dual decoder model is trained using a joint loss combining the log-likelihood of the two targets:

$$L(\theta) = \sum_{D} \Big(\sum_{t=1}^{|\mathbf{e}^1|} \log P(\mathbf{e}^1_t | \mathbf{e}^1_{<t}, \mathbf{e}^2_{<t}, \mathbf{f}; \theta) + \sum_{t=1}^{|\mathbf{e}^2|} \log P(\mathbf{e}^2_t | \mathbf{e}^2_{<t}, \mathbf{e}^1_{<t}, \mathbf{f}; \theta) \Big) ,$$

where θ represents the set of parameters. Training this model requires triplets of instances associating one source with two targets. Such resources are difficult to find and the largest tri-parallel open source corpus we know of is the MuST-Cinema dataset (Karakanta et al., 2020b), which is clearly smaller than what exists to separately train automatic transcription or translation systems.

In order to leverage large scale parallel translation data for English-French, we adopt a fine-tuning strategy where we initially pre-train a standard (encoder-decoder) translation model using all available resources, which serves to initialize the parameters of our dual decoder model. As the dual decoder network employs two decoders with shared parameters, we use also the decoder of the pre-trained model to initialize this subnetwork. Fine-tuning is performed on a tri-parallel corpus. We discuss the effect of decoder initialization in Section 3.4.1. Finally, for all fine-tuned models, the decoder cross-attention layer which binds the two decoders together is always randomly initialized.

3 Experiments

3.1 Datasets and Resources

For our experiments, we use MuST-Cinema[3] (Karakanta et al., 2020b), a multilingual Speech-to-Subtitles corpus compiled from TED talks, in which subtitles contain additional segmentation tags indicating changes of screen ([eob]) or line ([eol]). Our experiments consider the translation from English (EN) into French (FR). Our tri-parallel data also includes a pre-existing unpunctuated ASR output generated by Karakanta et al. (2020a), which achieves a WER score of 39.2% on the MuST-Cinema test set speech transcripts (details in Appendix A). For pre-training, we use all available WMT14 EN-FR data. During fine-tuning, we follow the recommendations and procedures of Zhou et al. (2019); Wang et al. (2019); He et al. (2021); Xu and Yvon (2021), and use synthetic tri-parallel data, in which we alternatively replace one of the two target side references by hypotheses generated from the baseline system for the corresponding direction via forward-translation. For more details about synthetic tri-parallel data generation, we refer to (Zhou et al., 2019; Xu and Yvon, 2021). We tokenize all data with Moses scripts and use a shared source-target vocabulary of 32K Byte Pair Encoding units (Sennrich et al., 2016) learned with `subword-nmt`.[4]

3.2 Experimental Settings

We implement the dual decoder model based on the Transformer (Vaswani et al., 2017) model using `fairseq`[5] (Ott et al., 2019).[6] All models are trained until no improvement is found for 4 consecutive checkpoints on the development set, except for the EN→FR pre-trained translation model which is trained during 300k iterations (further details in Appendix B). We mainly measure performance with SacreBLEU (Post, 2018);[7] TER and BERTScores (Zhang et al., 2020b) are also reported in Appendix D. Segmentation tags in subtitles are taken into account and BLEU scores are computed over full sentences. In addition to BLEU score, measuring the consistency between captions and subtitles is also an important aspect. We reuse the structural and lexical consistency score proposed by Karakanta et al. (2021). *Structural consistency* measures the percentage of utterances having the same number of blocks in both languages, while *lexical scores* count the proportion of words in the two languages that are aligned in the same block

[3] https://ict.fbk.eu/must-cinema/

[4] https://github.com/rsennrich/subword-nmt

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

[6] Our implementation is open-sourced at https://github.com/jitao-xu/dual-decoding

[7] BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a+version.1.5.1

(refer to Appendix C for additional details).

We call the dual decoder model `dual`. Baseline translation models trained separately on each direction ($T_{en} \rightarrow C_{en}$, $T_{en} \rightarrow S_{fr}$) are denoted by `base`. To study the effectiveness of dual decoding, we mainly compare `dual` with a `pipeline` system. The latter uses the `base` model to produce captions which are then translated into subtitles using an independent system trained to translate from caption to subtitle ($T_{en} \rightarrow C_{en} \rightarrow S_{fr}$).

Like the `dual` model, `base` and `pipeline` systems also benefit from pre-training. For the former, we pre-train the direct transcript-to-subtitle translation model ($T_{en} \rightarrow S_{fr}$); for `pipeline`, the caption-to-subtitle model ($C_{en} \rightarrow S_{fr}$) is pre-trained, while the first step ($T_{en} \rightarrow C_{en}$) remains as in the `base` system. Note that all fine-tuned systems start with the same model pre-trained using WMT EN-FR data.

3.3 Main Results

Model	BLEU			Consistency	
	EN	FR	Avg	Struct.	Lex.
`base`	55.7	23.9	39.8	55.3	70.7
`base` +FT	55.7	24.9	40.3	54.5	71.4
`pipeline`	55.7	23.6	39.7	95.7	96.0
`pipeline` +FT	55.7	24.2	40.0	98.4	98.3
`dual` +FT	**56.9**	25.6	**41.3**	65.1	79.1
`share` +FT	56.5	**25.8**	41.2	**66.7**	**80.0**

Table 2: BLEU scores for captions (EN) and subtitles (FR), with measures of structural and lexical consistency between the two hypotheses. These scores are in percentage (higher is better). The `base` and `pipeline` settings are trained from scratch with original data. `share` refers to tying all decoder parameters.

We only report in Table 2 the performance of the two baselines and fine-tuned (+FT) models, as our preliminary experiments showed that training the dual decoder model with only tri-parallel data was not optimal. The BLEU score of the *do nothing* baseline, which copies the source ASR transcripts to the output, is 28.0, which suggests that the captioning task actually involves much more transformations than simply inserting segmentation tags. We see that fine-tuning improves subtitles generated by `base` and `pipeline` systems by $\sim$1 BLEU. Our `dual` decoder model, after fine-tuned using synthetic tri-parallel data, respectively outperforms `base`+FT by 0.7 BLEU, and `pipeline`+FT by 1.4 BLEU. Sharing all parameters of both decoders yields further increase of 0.2 BLEU, with about one third less parameters.

We also measure the structural and lexical consistency between captions and subtitles generated by our systems (see Table 2). As expected, `pipeline` settings always generate very consistent pairs of captions and subtitles, as subtitles are direct translations of the captions; all other methods generate both outputs from the ASR transcripts. `dual` models do not perform as well, but are still able to generate captions and subtitles with a much higher structural and lexical consistency between the two outputs than in the `base` systems. Xu and Yvon (2021) show that dual decoder models generate translations that are more consistent in content. We further show here that our `dual` models generates hypotheses which are also more consistent in structure. Examples output captions and subtitles are in Appendix E.

3.4 Analyses and Discussions

3.4.1 The Effect of Fine-tuning

As the pre-trained uni-directional translation model has never seen sentences in the source language on the target side, we first only use it to initialize the subtitling decoder, and use a random initialization for the captioning decoder. To study the effect of initialization, we conduct an ablation study by comparing three settings: initializing only the subtitling decoder, both decoders or the shared decoder (see Table 3). Initializing both decoders brings improvements in both directions, with a gain of 1.6 BLEU for captioning and 0.3 BLEU for subtitling. Moreover, sharing parameters between decoders further boost the subtitling performance by 0.2 BLEU. As it seems, the captioning decoder also benefits from a decoder pre-trained in another language.

Model	EN	FR	Avg
`dual` 1-decoder +FT	55.3	25.3	40.3
`dual` +FT	56.9	25.6	41.3
`share` +FT	56.5	25.8	41.2

Table 3: BLEU scores for multiple initializations.

3.4.2 Exposure Bias

Due to error accumulations in both decoders, the exposure bias problem seems more severe for dual decoder model than for regular translation models (Zhou et al., 2019; Zhang et al., 2020a; Xu and Yvon, 2021). These authors propose to use *pseudo tri-parallel data with synthetic references* to alleviate this problem. We analyze the influence of this

exposure bias issue in our application scenario.

To this end, we compare fine-tuning the `dual` model with original vs artificial tri-parallel data. For simplicity, we only report in Table 4 the average BLEU scores of captioning and subtitling. Results show that fine-tuning with the original data (w.real) strongly degrades the automatic metrics for the generated text , resulting in performance that are worse than the baseline.

Model	Normal	2-round	Ref
`dual` +FT w.real	39.2	40.9	45.0
`share` +FT w.real	38.6	40.1	43.9
`dual` +FT	41.3	41.2	41.0
`share` +FT	41.2	40.9	40.5

Table 4: Performance of various decoding methods. All BLEU scores are averaged over the two outputs. *2-round* (resp. *Ref*) refers to decoding with model predictions (resp. references) as forced prefix in one direction.

In another set of experiments, we follow Xu and Yvon (2021) and perform asynchronous 2-round decoding. We first decode the `dual` models to obtain hypotheses in both languages $\mathbf{e}'_1$ and $\mathbf{e}'_2$. During the second decoding round, we use the output English caption $\mathbf{e}'_1$ as a forced prefix when generating the French subtitles $\mathbf{e}''_2$. The final English caption $\mathbf{e}''_1$ is obtained similarly. Note that when generating the t-th token in $\mathbf{e}''_2$, the decoder cross-attention module only attends to the t first tokens of $\mathbf{e}'_1$, even though the full of $\mathbf{e}'_1$ is actually known. The 2-round scores for $\mathbf{e}''_1$ and $\mathbf{e}''_2$ are in Table 4, and compared with the optimal situation where we use references instead of model predictions as forced prefix in the second round (in col. 'Ref').

Results in Table 4 suggest that dual decoder models fine-tuned with original data (w.real) are quite sensible to exposure bias, which can be mitigated with artificial tri-parallel data. Their performance can however be improved by $\sim$1.5 BLEU when using 2-round decoding, thereby almost closing the initial gap with models using synthetic data. The latter approach is overall slightly better and also more stable across decoding configurations.

4 Conclusion

In this paper, we have explored dual decoding to jointly generate captions and subtitles from ASR transcripts. Experimentally, we found that dual decoding improves translation quality for both captioning and subtitling, while delivering more consistent output pairs. Additionally, we showed that (a) model sharing on the decoder side is viable and effective, at least for related languages; (b) initializing with pre-trained models vastly improves performance; (c) 2-round decoding allowed us to mitigate the exposure bias problem in our model. In the future, we would like to experiment on more distant language pairs to validate our approach in a more general scenario.

5 Acknowledgement

The authors wish to thank Alina Karakanta for providing the ASR transcripts and the evaluation script for the consistency measures. We would also like to thank the anonymous reviewers for their valuable suggestions. This work was granted access to the HPC resources of IDRIS under the allocation 2021-[AD011011580R1] made by GENCI. The first author is partly funded by SYSTRAN and by a grant Transwrite from Région Ile-de-France. This work has also been funded by the BPI-France investment programme "Grands défis du numérique", as part of the ROSETTA-2 project (Subtitling RObot and Adapted Translation).

References

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Eunah Cho, Jan Niehues, and Alex Waibel. 2012. Segmentation and punctuation prediction in speech language translation using a monolingual translation system. In *Proceedings of the 9th International Workshop on Spoken Language Translation: Papers*, pages 252–259, Hong Kong, Table of contents.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Kyle Gorman. 2016. Pynini: A Python library for weighted finite-state grammar compilation. In *Proceedings of the SIGFSM Workshop on Statistical NLP and Weighted Automata*, pages 75–80, Berlin, Germany. Association for Computational Linguistics.

Hao He, Qian Wang, Zhipeng Yu, Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2021. Synchronous interactive decoding for multilingual neural machine

translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12981–12988.

Hakan Inan, Khashayar Khosravi, and Richard Socher. 2017. Tying word vectors and word classifiers: A loss framework for language modeling. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Alina Karakanta, Marco Gaido, Matteo Negri, and Marco Turchi. 2021. Between flexibility and consistency: Joint generation of captions and subtitles. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 215–225, Bangkok, Thailand (online). Association for Computational Linguistics.

Alina Karakanta, Matteo Negri, and Marco Turchi. 2020a. Is 42 the answer to everything in subtitling-oriented speech translation? In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 209–219, Online. Association for Computational Linguistics.

Alina Karakanta, Matteo Negri, and Marco Turchi. 2020b. MuST-cinema: a speech-to-subtitles corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3727–3734, Marseille, France. European Language Resources Association.

Hang Le, Juan Pino, Changhan Wang, Jiatao Gu, Didier Schwab, and Laurent Besacier. 2020. Dual-decoder transformer for joint automatic speech recognition and multilingual speech translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3520–3533, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Danni Liu, Jan Niehues, and Gerasimos Spanakis. 2020. Adapting end-to-end speech recognition for readable subtitles. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 247–256, Online.

Mehryar Mohri. 2002. Semiring frameworks and algorithms for shortest-distance problems. *J. Autom. Lang. Comb.*, 7(3):321–350.

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.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*, pages 5206–5210. IEEE.

Stelios Piperidis, Iason Demiros, Prokopis Prokopidis, Peter Vanroose, Anja Hoethker, Walter Daelemans, Elsa Sklavounou, Manos Konstantinou, and Yannis Karavidas. 2004. Multimodal, multilingual resources in the subtitling process. In *Proceedings of LREC*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al. 2011. The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*, CONF. IEEE Signal Processing Society.

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163, Valencia, Spain. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Matthias Sperber and Matthias Paulik. 2020. Speech translation and the end-to-end promise: Taking stock of where we are. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7409–7421, Online. Association for Computational Linguistics.

Matthias Sperber, Hendra Setiawan, Christian Gollan, Udhyakumar Nallasamy, and Matthias Paulik. 2020. Consistent transcription and translation of speech. *Transactions of the Association for Computational Linguistics*, 8:695–709.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Yining Wang, Jiajun Zhang, Long Zhou, Yuchen Liu, and Chengqing Zong. 2019. Synchronously generating two languages with interactive decoding. In

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 3350–3355, Hong Kong, China. Association for Computational Linguistics.

Jitao Xu and François Yvon. 2021. One source, two targets: Challenges and rewards of dual decoding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8533–8546, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiajun Zhang, Long Zhou, Yang Zhao, and Chengqing Zong. 2020a. Synchronous bidirectional inference for neural sequence generation. *Artificial Intelligence*, 281:103234.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations*.

Long Zhou, Jiajun Zhang, and Chengqing Zong. 2019. Synchronous bidirectional neural machine translation. *Transactions of the Association for Computational Linguistics*, 7:91–105.

A Data Processing Details

For the English to French language pair, MuST-Cinema[8] (Karakanta et al., 2020b) contains 275k sentences for training and 1079 and 544 lines for development and testing, respectively. The ASR system used by Karakanta et al. (2020a) to produce transcripts was based on the KALDI toolkit (Povey et al., 2011), and had been trained on the clean portion of LibriSpeech (Panayotov et al., 2015) (∼460h) and a subset of MuST-Cinema (∼450h). In order to emulate a real production scenario, we segment these transcripts as if they were from an ASR system performing segmentation based on prosody. As this kind of system tends to produce longer sequences compared to typical written text (Cho et al., 2012), we randomly concatenate the English captions into longer sequences, to which we align the ASR transcripts using the conventional edit distance, thus adding a subsegmentation aspect to the translation task. Edit distance computations are based on a Weighted Finite-State Transducer (WSFT), implemented with Pynini (Gorman, 2016), which represents editing operations (match, insertion, deletion, replacement) at the character level, with weights depending on the characters and the previous operation context. After composing the edit WFST with the transcript string and the caption string, the optimal operation sequence is computed using a shortest-distance algorithm (Mohri, 2002). The number of sentences to be concatenated is sampled normally, with an average around of 2. This process results in 133k, 499 and 255 lines for training, development and testing, respectively.

For pre-training, we use all available WMT14 EN-FR data,[9] in which we discard sentence pairs with invalid language label as computed by `fasttext` language identification model[10] (Bojanowski et al., 2017). This pre-training data contains 33.9M sentence pairs.

B Experimental Details

We build our dual decoder model with a hidden size of 512 and a feedforward size of 2048. We optimize with Adam, set up with a maximum learning rate of 0.0007 and an inverse square root decay schedule, as well as 4000 warmup steps. For fine-tuning, we use Adam with a fixed learning rate of 8e−5. For all models, we share lexical embeddings between the encoder and the input and output decoder matrices. All models are trained with mixed precision and a batch size of 8192 tokens on 4 V100 GPUs.

The two models in the `base` setting are trained separately using transcript→caption and transcript→subtitle data. The second model of the `pipeline` setting is trained using caption→subtitle data. When performing fine-tuning, we first pre-train an EN→FR translation model `pre-train` using WMT EN-FR data. For `base`+FT setting, the transcript→subtitle model is fine-tuned from `pre-train`, while the transcript→caption is the same as `base` since languages on both source and target sides are English. For `pipeline`+FT, the caption→subtitle model is fine-tuned from `pre-train`. For `dual`+FT, the encoder and the two decoders are fine-tuned from the same `pre-train` model. The decoder cross-attention layers cannot be fine-tuned and are randomly initialized. Due to computation limits, we are not able to conduct multiple runs for our models. However, all results are obtained by using the parameters averaged over the last 5 checkpoints.

[8] License: CC BY-NC-ND 4.0

[9] https://statmt.org/wmt14

[10] https://dl.fbaipublicfiles.com/fasttext/supervised-models/lid.176.bin

C Consistency Score

Consider the following example from (Karakanta et al., 2021):

0:00:50,820, 00:00:53,820
To put the assumptions very clearly:
Enonçons clairement nos hypothèses : le capitalisme,

00:00:53,820, 00:00:57,820
capitalism, after 150 years, has become acceptable,
après 150 ans, est devenu acceptable, au même titre

00:00:58,820, 00:01:00,820
and so has democracy.
que la democratie.

As defined by Karakanta et al. (2021), for the stuctural consistency, both captions (EN) and subtitles (FR) have the same number of 3 blocks. For lexical consistency, there are 6 tokens of the subtitles which are not aligned to captions in the same block: *"le capitalisme ," , "au même titre"*. The $Lex_{C \to S}$ is calculated as the percentage of aligned words normalized by number of words in the caption. Therefore, $Lex_{C \to S} = \frac{20}{22} = 90.9\%$; the computation is identical in the other direction, yielding $Lex_{S \to C} = \frac{17}{23} = 73.9\%$, the average lexical consistency of this segment is thus $Lex_{pair} = \frac{Lex_{C \to S} + Lex_{S \to C}}{2} = 82.4\%$.

When computing the *lexical consistency* between captions and subtitles, we use the WMT14 EN-FR data to train an alignment model using `fast_align`[11] (Dyer et al., 2013) in both directions and use it to predict word alignments for model outputs.

D Additional Metric

Table 5 reports TER and BERTScores[12] (Zhang et al., 2020b). Note that for BERTScores, we remove segmentation tokens ([eob] and [eol]) from hypotheses and references, as special tokens are out-of-vocabulary for pre-trained BERT models.

E Examples

Some examples of dual decoding improving the quality of both captioning and subtitling compared to the pipeline system are in Table 6.

[11] https://github.com/clab/fast_align
[12] https://github.com/Tiiiger/bert_score

	TER ↓			BERTScore-F1 ↑			BLEU ↑			Consistency ↑	
Model	EN	FR	Avg	EN	FR	Avg	EN	FR	Avg	Struct.	Lex.
`base`	0.264	0.662	0.463	0.7346	0.3961	0.5654	55.7	23.9	39.8	55.3	70.7
`base` +FT	0.264	0.654	0.459	0.7346	0.4026	0.5686	55.7	24.9	40.3	54.5	71.4
`pipeline`	0.264	0.650	0.457	0.7346	0.3912	0.5629	55.7	23.6	39.7	95.7	96.0
`pipeline` +FT	0.264	0.652	0.458	0.7346	0.3924	0.5635	55.7	24.2	40.0	98.4	98.3
`dual` +FT	**0.256**	**0.640**	**0.448**	0.7378	**0.4074**	0.5726	**56.9**	25.6	**41.3**	65.1	79.1
`share` +FT	0.259	**0.640**	0.450	**0.7396**	0.4066	**0.5731**	56.5	**25.8**	41.2	**66.7**	**80.0**

Table 5: TER, BERTScore and BLEU scores for captions (EN) and subtitles (FR), with measures of structural and lexical consistency between the two hypotheses. The `base` and `pipeline` settings are trained from scratch with original data. `share` refers to tying all decoder parameters. Signature of BERTScore (EN): microsoft/deberta-xlarge-mnli_L40_no-idf_version=0.3.11(hug_trans=4.10.3)-rescaled_fast-tokenizer. Signature of BERTScore (FR): bert-base-multilingual-cased_L9_no-idf_version=0.3.11(hug_trans=4.10.3)-rescaled_fast-tokenizer.

Source	take time to write down your values your objectives and your key results do it today
EN `pipeline` +FT	Take time to write down [eol] your values, your objectives, [eob] and your key results do it today. [eob]
EN `share` +FT	Take time to write down your values, **[eol]** your objectives, [eob] and your key results do it today. [eob]
EN ref	Take time to write down your values, [eob] your objectives and your key results. [eob] Do it today. [eob]
FR `pipeline` +FT	Prenez le temps d'écrire vos valeurs, [eol] vos objectifs, [eob] et vos principaux résultats [eol] le font aujourd'hui. [eob]
FR `share` +FT	Prenez le temps d'écrire vos valeurs, **[eob]** vos objectifs et **vos résultats clés. [eob] Faites-le** aujourd'hui. [eob]
FR ref	Prenez le temps d'écrire vos valeurs, [eob] vos objectifs et vos résultats clés. [eob] Faites-le aujourd'hui. [eob]
Source	and as it turns out what are you willing to give up is exactly the right question to ask
EN `pipeline` +FT	And as it turns out, what are you willing [eol] to give up is exactly [eob] the right question to ask? [eob]
EN `share` +FT	And as it turns out, what are you willing [eol] to give up **[eob]** is exactly the right question to ask? [eob]
EN ref	And as it turns out, [eob] "What are you willing to give up?" [eob] is exactly the right question to ask. [eob]
FR `pipeline` +FT	Et il s'avère que ce que vous voulez abandonner [eol] est exactement [eob] la bonne question à poser ? [eob]
FR `share` +FT	Et il s'avère que ce que vous voulez abandonner **[eob]** est exactement la bonne question à poser**.** [eob]
FR ref	Et il s'avère que [eob] « Qu'êtes-vous prêts à abandonner ? » [eob] est exactement la question à poser. [eob]

Table 6: Examples of dual decoding improving both captioning and subtitling. Major improvements are marked in bold.