Attention Strategies for Multi-Source Sequence-to-Sequence Learning

Jindřich Libovický and **Jindřich Helcl**
Charles University, Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Malostranské náměstí 25, 118 00 Prague, Czech Republic
{libovicky, helcl}@ufal.mff.cuni.cz

Abstract

Modeling attention in neural multi-source sequence-to-sequence learning remains a relatively unexplored area, despite its usefulness in tasks that incorporate multiple source languages or modalities. We propose two novel approaches to combine the outputs of attention mechanisms over each source sequence, *flat* and *hierarchical*. We compare the proposed methods with existing techniques and present results of systematic evaluation of those methods on the WMT16 Multimodal Translation and Automatic Post-editing tasks. We show that the proposed methods achieve competitive results on both tasks.

1 Introduction

Sequence-to-sequence (S2S) learning with attention mechanism recently became the most successful paradigm with state-of-the-art results in machine translation (MT) (Bahdanau et al., 2014; Sennrich et al., 2016a), image captioning (Xu et al., 2015; Lu et al., 2016), text summarization (Rush et al., 2015) and other NLP tasks.

All of the above applications of S2S learning make use of a single encoder. Depending on the modality, it can be either a recurrent neural network (RNN) for textual input data, or a convolutional network for images.

In this work, we focus on a special case of S2S learning with multiple input sequences of possibly different modalities and a single output-generating recurrent decoder. We explore various strategies the decoder can employ to attend to the hidden states of the individual encoders.

The existing approaches to this problem do not explicitly model different importance of the inputs to the decoder (Firat et al., 2016; Zoph and Knight, 2016). In multimodal MT (MMT), where an image and its caption are on the input, we might expect the caption to be the primary source of information, whereas the image itself would only play a role in output disambiguation. In automatic post-editing (APE), where a sentence in a source language and its automatically generated translation are on the input, we might want to attend to the source text only in case the model decides that there is an error in the translation.

We propose two interpretable attention strategies that take into account the roles of the individual source sequences explicitly—flat and hierarchical attention combination.

This paper is organized as follows: In Section 2, we review the attention mechanism in single-source S2S learning. Section 3 introduces new attention combination strategies. In Section 4, we evaluate the proposed models on the MMT and APE tasks. We summarize the related work in Section 5, and conclude in Section 6.

2 Attentive S2S Learning

The attention mechanism in S2S learning allows an RNN decoder to directly access information about the input each time before it emits a symbol. Inspired by content-based addressing in Neural Turing Machines (Graves et al., 2014), the attention mechanism estimates a probability distribution over the encoder hidden states in each decoding step. This distribution is used for computing the context vector—the weighted average of the encoder hidden states—as an additional input to the decoder.

The standard attention model as described by Bahdanau et al. (2014) defines the attention energies e_{ij}, attention distribution α_{ij}, and the con-

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 196–202
Vancouver, Canada, July 30 - August 4, 2017.
https://doi.org/10.18653/v1/P17-2031

text vector c_i in i-th decoder step as:

$$e_{ij} = v_a^\top \tanh(W_a s_i + U_a h_j), \quad (1)$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}, \quad (2)$$

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j. \quad (3)$$

The trainable parameters W_a and U_a are projection matrices that transform the decoder and encoder states s_i and h_j into a common vector space and v_a is a weight vector over the dimensions of this space. T_x denotes the length of the input sequence. For the sake of clarity, bias terms (applied every time a vector is linearly projected using a weight matrix) are omitted.

Recently, Lu et al. (2016) introduced *sentinel gate*, an extension of the attentive RNN decoder with LSTM units (Hochreiter and Schmidhuber, 1997). We adapt the extension for gated recurrent units (GRU) (Cho et al., 2014), which we use in our experiments:

$$\psi_i = \sigma(W_y y_i + W_s s_{i-1}) \quad (4)$$

where W_y and W_s are trainable parameters, y_i is the embedded decoder input, and s_{i-1} is the previous decoder state.

Analogically to Equation 1, we compute a scalar energy term for the sentinel:

$$e_{\psi_i} = v_a^\top \tanh\left(W_a s_i + U_a^{(\psi)}(\psi_i \odot s_i)\right) \quad (5)$$

where W_a, $U_a^{(\psi)}$ are the projection matrices, v_a is the weight vector, and $\psi_i \odot s_i$ is the sentinel vector. Note that the sentinel energy term does not depend on any hidden state of any encoder. The sentinel vector is projected to the same vector space as the encoder state h_j in Equation 1. The term e_{ψ_i} is added as an extra attention energy term to Equation 2 and the sentinel vector $\psi_i \odot s_i$ is used as the corresponding vector in the summation in Equation 3.

This technique should allow the decoder to choose whether to attend to the encoder or to focus on its own state and act more like a language model. This can be beneficial if the encoder does not contain much relevant information for the current decoding step.

3 Attention Combination

In S2S models with multiple encoders, the decoder needs to be able to combine the attention information collected from the encoders.

A widely adopted technique for combining multiple attention models in a decoder is concatenation of the context vectors $c_i^{(1)}, \ldots, c_i^{(N)}$ (Zoph and Knight, 2016; Firat et al., 2016). As mentioned in Section 1, this setting forces the model to attend to each encoder independently and lets the attention combination to be resolved implicitly in the subsequent network layers.

In this section, we propose two alternative strategies of combining attentions from multiple encoders. We either let the decoder learn the α_i distribution jointly over all encoder hidden states (*flat* attention combination) or factorize the distribution over individual encoders (*hierarchical* combination).

Both of the alternatives allow us to explicitly compute distribution over the encoders and thus interpret how much attention is paid to each encoder at every decoding step.

3.1 Flat Attention Combination

Flat attention combination projects the hidden states of all encoders into a shared space and then computes an arbitrary distribution over the projections. The difference between the concatenation of the context vectors and the flat attention combination is that the α_i coefficients are computed jointly for all encoders:

$$\alpha_{ij}^{(k)} = \frac{\exp(e_{ij}^{(k)})}{\sum_{n=1}^{N} \sum_{m=1}^{T_x^{(n)}} \exp\left(e_{im}^{(n)}\right)} \quad (6)$$

where $T_x^{(n)}$ is the length of the input sequence of the n-th encoder and $e_{ij}^{(k)}$ is the attention energy of the j-th state of the k-th encoder in the i-th decoding step. These attention energies are computed as in Equation 1. The parameters v_a and W_a are shared among the encoders, and U_a is different for each encoder and serves as an encoder-specific projection of hidden states into a common vector space.

The states of the individual encoders occupy different vector spaces and can have a different dimensionality, therefore the context vector cannot be computed as their weighted sum. We project

them into a single space using linear projections:

$$c_i = \sum_{k=1}^{N} \sum_{j=1}^{T_x^{(k)}} \alpha_{ij}^{(k)} U_c^{(k)} h_j^{(k)} \quad (7)$$

where $U_c^{(k)}$ are additional trainable parameters.

The matrices $U_c^{(k)}$ project the hidden states into a common vector space. This raises a question whether this space can be the same as the one that is projected into in the energy computation using matrices $U_a^{(k)}$ in Equation 1, i.e., whether $U_c^{(k)} = U_a^{(k)}$. In our experiments, we explore both options. We also try both adding and not adding the sentinel $\alpha_i^{(\psi)} U_c^{(\psi)} (\psi_i \odot s_i)$ to the context vector.

3.2 Hierarchical Attention Combination

The hierarchical attention combination model computes every context vector independently, similarly to the concatenation approach. Instead of concatenation, a second attention mechanism is constructed over the context vectors.

We divide the computation of the attention distribution into two steps: First, we compute the context vector for each encoder independently using Equation 3. Second, we project the context vectors (and optionally the sentinel) into a common space (Equation 8), we compute another distribution over the projected context vectors (Equation 9) and their corresponding weighted average (Equation 10):

$$e_i^{(k)} = v_b^\top \tanh(W_b s_i + U_b^{(k)} c_i^{(k)}), \quad (8)$$

$$\beta_i^{(k)} = \frac{\exp(e_i^{(k)})}{\sum_{n=1}^{N} \exp(e_i^{(n)})}, \quad (9)$$

$$c_i = \sum_{k=1}^{N} \beta_i^{(k)} U_c^{(k)} c_i^{(k)} \quad (10)$$

where $c_i^{(k)}$ is the context vector of the k-th encoder, additional trainable parameters v_b and W_b are shared for all encoders, and $U_b^{(k)}$ and $U_c^{(k)}$ are encoder-specific projection matrices, that can be set equal and shared, similarly to the case of flat attention combination.

4 Experiments

We evaluate the attention combination strategies presented in Section 3 on the tasks of multimodal translation (Section 4.1) and automatic post-editing (Section 4.2).

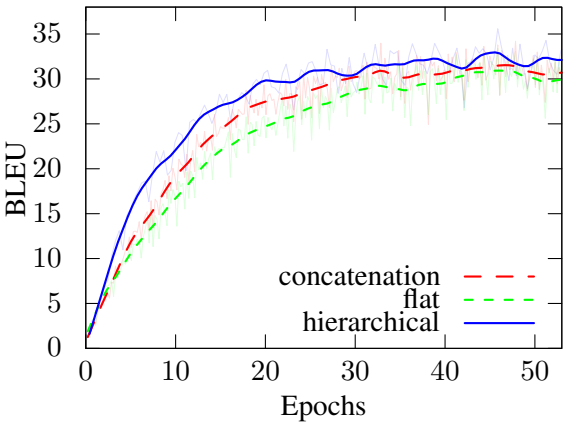


Figure 1: Learning curves on validation data for context vector concatenation (blue), flat (green) and hierarchical (red) attention combination without sentinel and without sharing the projection matrices.

The models were implemented using the Neural Monkey sequence-to-sequence learning toolkit (Helcl and Libovický, 2017).[1][2] In both setups, we process the textual input with bidirectional GRU network (Cho et al., 2014) with 300 units in the hidden state in each direction and 300 units in embeddings. For the attention projection space, we use 500 hidden units. We optimize the network to minimize the output cross-entropy using the Adam algorithm (Kingma and Ba, 2014) with learning rate 10^{-4}.

4.1 Multimodal Translation

The goal of multimodal translation (Specia et al., 2016) is to generate target-language image captions given both the image and its caption in the source language.

We train and evaluate the model on the Multi30k dataset (Elliott et al., 2016). It consists of 29,000 training instances (images together with English captions and their German translations), 1,014 validation instances, and 1,000 test instances. The results are evaluated using the BLEU (Papineni et al., 2002) and METEOR (Denkowski and Lavie, 2011).

In our model, the visual input is processed with a pre-trained VGG 16 network (Simonyan and Zisserman, 2014) without further fine-tuning. Atten-

[1] http://github.com/ufal/neuralmonkey

[2] The trained models can be downloaded from http://ufallab.ms.mff.cuni.cz/~libovicky/acl2017_att_models/

tion distribution over the visual input is computed from the last convolutional layer of the network. The decoder is an RNN with 500 conditional GRU units (Firat and Cho, 2016) in the recurrent layer. We use byte-pair encoding (Sennrich et al., 2016b) with a vocabulary of 20,000 subword units shared between the textual encoder and the decoder.

The results of our experiments in multimodal MT are shown in Table 1. We achieved the best results using the hierarchical attention combination without the sentinel mechanism, which also showed the fastest convergence. The flat combination strategy achieves similar results eventually. Sharing the projections for energy and context vector computation does not improve over the concatenation baseline and slows the training almost prohibitively. Multimodal models were not able to surpass the textual baseline (BLEU 33.0).

Using the conditional GRU units brought an improvement of about 1.5 BLEU points on average, with the exception of the concatenation scenario where the performance dropped by almost 5 BLEU points. We hypothesize this is caused by the fact the model has to learn the implicit attention combination on multiple places – once in the output projection and three times inside the conditional GRU unit (Firat and Cho, 2016, Equations 10-12). We thus report the scores of the introduced attention combination techniques trained with conditional GRU units and compare them with the concatenation baseline trained with plain GRU units.

4.2 Automatic MT Post-editing

Automatic post-editing is a task of improving an automatically generated translation given the source sentence where the translation system is treated as a black box.

We used the data from the WMT16 APE Task (Bojar et al., 2016; Turchi et al., 2016), which consists of 12,000 training, 2,000 validation, and 1,000 test sentence triplets from the IT domain. Each triplet contains an English source sentence, an automatically generated German translation of the source sentence, and a manually post-edited German sentence as a reference. In case of this dataset, the MT outputs are almost perfect in and only little effort was required to post-edit the sentences. The results are evaluated using the human-targeted error rate (HTER) (Snover et al., 2006) and BLEU score (Papineni et al., 2002).

	share	sent.	MMT		APE	
			BLEU	METEOR	BLEU	HTER
concat.			31.4 ±.8	48.0 ±.7	62.3 ±.5	24.4 ±.4
flat	×	×	30.2 ±.8	46.5 ±.7	62.6 ±.5	24.2 ±.4
	×	✓	29.3 ±.8	45.4 ±.7	62.3 ±.5	24.3 ±.4
	✓	×	30.9 ±.8	47.1 ±.7	62.4 ±.6	24.4 ±.4
	✓	✓	29.4 ±.8	46.9 ±.7	62.5 ±.6	24.2 ±.4
hierarchical	×	×	**32.1 ±.8**	**49.1 ±.7**	62.3 ±.5	24.1 ±.4
	×	✓	28.1 ±.8	45.5 ±.7	62.6 ±.6	24.1 ±.4
	✓	×	26.1 ±.7	42.4 ±.7	62.4 ±.5	24.3 ±.4
	✓	✓	22.0 ±.7	38.5 ±.6	62.5 ±.5	24.1 ±.4

Table 1: Results of our experiments on the test sets of Multi30k dataset and the APE dataset. The column 'share' denotes whether the projection matrix is shared for energies and context vector computation, 'sent.' indicates whether the sentinel vector has been used or not.

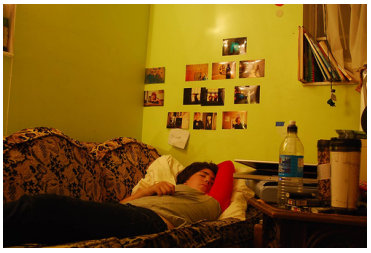
Following Libovický et al. (2016), we encode the target sentence as a sequence of edit operations transforming the MT output into the reference. By this technique, we prevent the model from paraphrasing the input sentences. The decoder is a GRU network with 300 hidden units. Unlike in the MMT setup (Section 4.1), we do not use the conditional GRU because it is prone to overfitting on the small dataset we work with.

The models were able to slightly, but significantly improve over the baseline – leaving the MT output as is (HTER 24.8). The differences between the attention combination strategies are not significant.

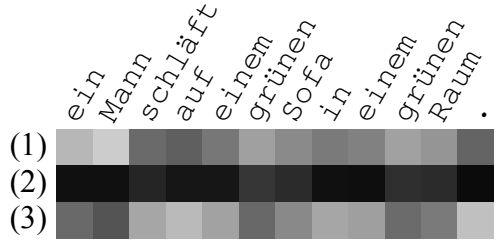
5 Related Work

Attempts to use S2S models for APE are relatively rare (Bojar et al., 2016). Niehues et al. (2016) concatenate both inputs into one long sequence, which forces the encoder to be able to work with both source and target language. Their attention is then similar to our flat combination strategy; however, it can only be used for sequential data.

The best system from the WMT'16 competition (Junczys-Dowmunt and Grundkiewicz, 2016) trains two separate S2S models, one translating from MT output to post-edited targets and the second one from source sentences to post-edited targets. The decoders average their output distributions similarly to decoder ensembling. The biggest source of improvement in this state-of-the-art posteditor came from additional training data generation, rather than from changes in the network architecture.



Source: a man sleeping in a green room on a couch .
Reference: ein Mann schläft in einem grünen Raum auf einem Sofa .
Output with attention:



(1) source, (2) image, (3) sentinel

Figure 2: Visualization of hierarchical attention in MMT. Each column in the diagram corresponds to the weights of the encoders and sentinel. Note that the despite the overall low importance of the image encoder, it gets activated for the content words.

Caglayan et al. (2016) used an architecture very similar to ours for multimodal translation. They made a strong assumption that the network can be trained in such a way that the hidden states of the encoder and the convolutional network occupy the same vector space and thus sum the context vectors from both modalities. In this way, their multimodal MT system (BLEU 27.82) remained far bellow the text-only setup (BLEU 32.50).

New state-of-the-art results on the Multi30k dataset were achieved very recently by Calixto et al. (2017). The best-performing architecture uses the last fully-connected layer of VGG-19 network (Simonyan and Zisserman, 2014) as decoder initialization and only attends to the text encoder hidden states. With a stronger monomodal baseline (BLEU 33.7), their multimodal model achieved a BLEU score of 37.1. Similarly to Niehues et al. (2016) in the APE task, even further improvement was achieved by synthetically extending the dataset.

6 Conclusions

We introduced two new strategies of combining attention in a multi-source sequence-to-sequence setup. Both methods are based on computing a joint distribution over hidden states of all encoders.

We conducted experiments with the proposed strategies on multimodal translation and automatic post-editing tasks, and we showed that the flat and hierarchical attention combination can be applied to these tasks with maintaining competitive score to previously used techniques.

Unlike the simple context vector concatenation, the introduced combination strategies can be used with the conditional GRU units in the decoder. On top of that, the hierarchical combination strategy exhibits faster learning than than the other strategies.

Acknowledgments

We would like to thank Ondřej Dušek, Rudolf Rosa, Pavel Pecina, and Ondřej Bojar for a fruitful discussions and comments on the draft of the paper.

This research has been funded by the Czech Science Foundation grant no. P103/12/G084, the EU grant no. H2020-ICT-2014-1-645452 (QT21), and Charles University grant no. 52315/2014 and SVV project no. 260 453. This work has been using language resources developed and/or stored and/or distributed by the LINDAT-Clarin project of the Ministry of Education of the Czech Republic (project LM2010013).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR* abs/1409.0473. http://arxiv.org/abs/1409.0473.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Névéol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation (WMT16). In *Proceedings of the First Conference on Machine Translation (WMT). Volume 2: Shared Task Papers*. Association for Computational Linguistics, Association for Computational Linguistics, Stroudsburg, PA, USA, volume 2, pages 131–198.

Ozan Caglayan, Walid Aransa, Yaxing Wang, Marc Masana, Mercedes García-Martínez, Fethi Bougares, Loïc Barrault, and Joost van de Weijer. 2016. Does multimodality help human and machine for translation and image captioning? In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, pages 627–633. http://www.aclweb.org/anthology/W16-2358.

Iacer Calixto, Qun Liu, and Nick Campbell. 2017. Incorporating global visual features into attention-based neural machine translation. *CoRR* abs/1701.06521. http://arxiv.org/abs/1701.06521.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Association for Computational Linguistics, Doha, Qatar, pages 103–111. http://www.aclweb.org/anthology/W14-4012.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Edinburgh, United Kingdom, pages 85–91. http://www.aclweb.org/anthology/W11-2107.

Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. Multi30k: Multilingual english-german image descriptions. *CoRR* abs/1605.00459. http://arxiv.org/abs/1605.00459.

Orhan Firat and Kyunghyun Cho. 2016. Conditional gated recurrent unit with attention mechanism. https://github.com/nyu-dl/dl4mt-tutorial/blob/master/docs/cgru.pdf. Published online, version `adbaeea`.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, San Diego, CA, USA, pages 866–875. http://www.aclweb.org/anthology/N16-1101.

Alex Graves, Greg Wayne, and Ivo Danihelka. 2014. Neural turing machines. *CoRR* abs/1410.5401. http://arxiv.org/abs/1410.5401.

Jindřich Helcl and Jindřich Libovický. 2017. Neural monkey: An open-source tool for sequence learning. *The Prague Bulletin of Mathematical Linguistics* (107):5–17. https://doi.org/10.1515/pralin-2017-0001.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.* 9:1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2016. Log-linear combinations of monolingual and bilingual neural machine translation models for automatic post-editing. In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, pages 751–758. http://www.aclweb.org/anthology/W/W16/W16-2378.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR* abs/1412.6980. http://arxiv.org/abs/1412.6980.

Jindřich Libovický, Jindřich Helcl, Marek Tlustý, Ondřej Bojar, and Pavel Pecina. 2016. CUNI system for WMT16 automatic post-editing and multimodal translation tasks. In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, pages 646–654. http://www.aclweb.org/anthology/W/W16/W16-2361.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2016. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. *CoRR* abs/1612.01887. http://arxiv.org/abs/1612.01887.

Jan Niehues, Eunah Cho, Thanh-Le Ha, and Alex Waibel. 2016. Pre-translation for neural machine translation. *CoRR* abs/1610.05243. http://arxiv.org/abs/1610.05243.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, pages 311–318. https://doi.org/10.3115/1073083.1073135.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 379–389. https://aclweb.org/anthology/D/D15/D15-1044.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh neural machine translation systems for WMT 16. In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, pages 371–376. http://www.aclweb.org/anthology/W/W16/W16-2323.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. 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). Association for Computational Linguistics, Berlin, Germany, pages 1715–1725. http://www.aclweb.org/anthology/P16-1162.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *CoRR* abs/1409.1556. http://arxiv.org/abs/1409.1556.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas*. volume 200.

Lucia Specia, Stella Frank, Khalil Sima'an, and Desmond Elliott. 2016. A shared task on multimodal machine translation and crosslingual image description. In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, pages 543–553. http://www.aclweb.org/anthology/W16-2346.

Marco Turchi, Rajen Chatterjee, and Matteo Negri. 2016. WMT16 APE shared task data. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University in Prague. http://hdl.handle.net/11372/LRT-1632.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In David Blei and Francis Bach, editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*. JMLR Workshop and Conference Proceedings, Lille, France, pages 2048–2057. http://jmlr.org/proceedings/papers/v37/xuc15.pdf.

Barret Zoph and Kevin Knight. 2016. Multi-source neural translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, San Diego, CA, USA, pages 30–34. http://www.aclweb.org/anthology/N16-1004.