The MeMAD Submission to the WMT18 Multimodal Translation Task

Stig-Arne Grönroos
Aalto University

Benoit Huet
EURECOM

Mikko Kurimo
Aalto University

Jorma Laaksonen
Aalto University

Bernard Merialdo
EURECOM

Phu Pham
Aalto University

Mats Sjöberg
Aalto University

Umut Sulubacak
University of Helsinki

Jörg Tiedemann
University of Helsinki

Raphael Troncy
EURECOM

Raúl Vázquez
University of Helsinki

Abstract

This paper describes the MeMAD project entry to the WMT Multimodal Machine Translation Shared Task.

We propose adapting the Transformer neural machine translation (NMT) architecture to a multi-modal setting. In this paper, we also describe the preliminary experiments with text-only translation systems leading us up to this choice.

We have the top scoring system for both English-to-German and English-to-French, according to the automatic metrics for *flickr18*.

Our experiments show that the effect of the visual features in our system is small. Our largest gains come from the quality of the underlying text-only NMT system. We find that appropriate use of additional data is effective.

1 Introduction

In multi-modal translation, the task is to translate from a source sentence and the image that it describes, into a target sentence in another language. As both automatic image captioning systems and crowd captioning efforts tend to mainly yield descriptions in English, multi-modal translation can be useful for generating descriptions of images for languages other than English. In the MeMAD project[1], multi-modal translation is of interest for creating textual versions or descriptions of audio-visual content. Conversion to text enables both indexing for multi-lingual image and video search, and increased access to the audio-visual materials for visually impaired users.

Data set	images	en	de	fr	sentences
Multi30k	✓	✓	✓	✓	29k
MS-COCO	✓	✓	+	+	616k
OpenSubtitles		✓	✓	✓	23M/42M
					1M, 3M, and 6M subsets used.

Table 1: Summary of data set sizes. ✓means attribute is present in original data. + means data set augmented in this work.

We adapt[2] the Transformer (Vaswani et al., 2017) architecture to use global image features extracted from Detectron, a pre-trained object detection and localization neural network. We use two additional training corpora: MS-COCO (Lin et al., 2014) and OpenSubtitles2018 (Tiedemann, 2009). MS-COCO is multi-modal, but not multi-lingual. We extended it to a synthetic multi-modal and multilingual training set. OpenSubtitles is multilingual, but does not include associated images, and was used as text-only training data. This places our entry in the unconstrained category of the WMT shared task. Details on the architecture used in this work can be found in Section 4.1. Further details on the synthetic data are presented in Section 2. Data sets are summarized in Table 1.

2 Experiment 1: Optimizing Text-Based Machine Translation

Our first aim was to select the text-based MT system to base our multi-modal extensions on.

[1] https://www.memad.eu/

[2] Our fork available from https://github.com/Waino/OpenNMT-py/tree/develop_mmod

Proceedings of the Third Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 603–611
Belgium, Brussels, October 31 - Novermber 1, 2018. ©2018 Association for Computational Linguistics
https://doi.org/10.18653/v1/W18-64066

EN-FR	flickr16	flickr17	mscoco17
multi30k	61.4	54.0	43.1
+SUBS$_{\text{full}}$	53.7	48.9	47.0
+domain-tuned	66.1	59.7	**51.7**
+ensemble-of-3	**66.5**	**60.2**	51.6

EN-DE	flickr16	flickr17	mscoco17
multi30k	38.9	32.0	27.7
+SUBS$_{\text{full}}$	41.3	34.1	31.3
+domain-tuned	43.3	38.4	35.0
+ensemble-of-3	**43.9**	**39.6**	**37.0**

Table 2: Adding subtitle data and domain tuning for image caption translation (BLEU% scores). All results with Marian Amun.

	EN-FR	flickr16	flickr17	mscoco17
A	SUBS1M$_H$+MS-COCO	66.3	60.5	52.1
A	+domain-tuned	66.8	60.6	52.0
A	+labels	**67.2**	60.4	51.7
T	SUBS1M$_{LM}$+MS-COCO	66.9	60.3	**52.8**
T	+labels	**67.2**	**60.9**	52.7

	EN-DE	flickr16	flickr17	mscoco17
A	SUBS1M$_H$+MS-COCO	43.1	39.0	35.1
A	+domain-tuned	43.9	39.4	35.8
A	+labels	43.2	39.3	34.3
T	SUBS1M$_{LM}$+MS-COCO	**44.4**	39.4	35.0
T	+labels	44.1	**39.8**	**36.5**

Table 3: Using automatically translated image captions and domain labels (BLEU% scores). A is short for Amun, T for Transformer.

We tried a wide range of models, but only include results with the two strongest systems: Marian NMT with the *amun* model (Junczys-Dowmunt et al., 2018), and OpenNMT (Klein et al., 2017) with the *Transformer* model.

We also studied the effect of additional training data. Our initial experiments showed that movie subtitles and their translations work rather well to augment the given training data. Therefore, we included parallel subtitles from the OpenSubtitles2018 corpus to train better text-only MT models. For these experiments, we apply the Marian amun model, an attentional encoder-decoder model with bidirectional LSTM's on the encoder side. In our first series of experiments, we observed that domain-tuning is very important when using Marian. The domain-tuning was accomplished by a second training step on in-domain data after training the model on the entire data set. Table 2 shows the scores on development data. We also tried decoding with an ensemble of three independent runs, which also pushed the performance a bit.

Furthermore, we tried to artificially increase the amount of in-domain data by translating existing English image captions to German and French. For this purpose, we used the large MS-COCO data set with its 100,000 images that have five image captions each. We used our best multidomain model (see Table 2) to translate all of those captions and used them as additional training data. This procedure also transfers the knowledge learned by the multidomain model into the caption translations, which helps us to improve the coverage of the system with less out-of-domain data. Hence, we filtered the large collection of translated movie subtitles to a smaller portion of reliable sentence pairs (one million in the experiment we report) and could train on a smaller data set with better results.

We experimented with two filtering methods. Initially, we implemented a basic heuristic filter (SUBS$_H$), and later we improved on this with a language model filter (SUBS$_{LM}$). Both procedures consider each sentence pair, assign it a quality score, and then select the highest scoring 1, 3, or 6 million pairs, discarding the rest. The SUBS$_H$ method counts terminal punctuation (*'.', '...', '?', '!'*) in the source and target sentences, initializing the score as the negative of the absolute value of the difference between these counts. Afterwards, it further decrements the score by 1 for each occurrence of terminal punctuation beyond the first in each of the sentences. The SUBS$_{LM}$ method first preprocesses the data by filtering samples by length and ratio of lengths, applying a rule-based noise filter, removing all characters not present in the Multi30k set, and deduplicating samples. Afterwards, target sentences in the remaining pairs are scored using a character-based deep LSTM language model trained on the Multi30k data. Both selection procedures are intended for noise filtering, and SUBS$_{LM}$ additionally acts as domain adaptation. Table 3 lists the scores we obtained on development data.

To make a distinction between automatically translated captions, subtitle translations and human-translated image captions, we also introduced domain labels that we added as special tokens to the beginning of the input sequence. In this way, the model can use explicit information about the domain when deciding how to translate given input. However, the effect of such labels is not consistent between systems. For Marian amun, the effect is negligible as we can see in Table 3. For the Transformer, domain labels had little effect on BLEU but were clearly beneficial according to chrF-1.0.

2.1 Preprocessing of textual data

The final preprocessing pipeline for the textual data consisted of lowercasing, tokenizing using Moses, fixing double-encoded entities and other encoding problems, and normalizing punctuation. For the OpenSubtitles data we additionally used the SUBS_{LM} subset selection.

Subword decoding has become popular in NMT. Careful choice of translation units is especially important as one of the target languages of our system is German, a morphologically rich language. We trained a shared 50k subword vocabulary using Byte Pair Encoding (BPE) (Sennrich et al., 2015). To produce a balanced multi-lingual segmentation, the following procedure was used: First, word counts were calculated individually for English and each of the 3 target languages Czech[3], French and German. The counts were normalized to equalize the sum of the counts for each language. This avoided imbalance in the amount of data skewing the segmentation in favor of some language. Segmentation boundaries around hyphens were forced, overriding the BPE.

Multi-lingual translation with target-language tag was done following Johnson et al. (2016). A special token, e.g. <TO_DE> to mark German as the target language, was prefixed to each paired English source sentence.

3 Experiment 2: Adding Automatic Image Captions

Our first attempt to add multi-modal information to the translation model includes the incorporation of automatically created image captions in a purely text-based translation engine. For this, we generated five English captions for each of the images in the provided training and test data. This was done by using our in-house captioning system (Shetty et al., 2018). The image captioning system uses a 2-layer LSTM with residual connections to generate captions based on scene context and object location descriptors, in addition to standard CNN-based features. The model was trained with the MS-COCO training data and used to be state of the art in the COCO leaderboard[4] in Spring 2016. The beam search size was set to five.

EN-FR	flickr16	flickr17	mscoco17
multi30k	61.4	54.0	43.1
+autocap (dual attn.)	60.9	52.9	43.3
+autocap 1 (concat)	61.7	53.7	43.9
+autocap 1-5 (concat)	**62.2**	**54.4**	**44.1**
EN-DE	**flickr16**	**flickr17**	**mscoco17**
multi30k	38.9	32.0	27.7
+autocap (dual attn.)	37.8	30.2	27.0
+autocap 1 (concat)	39.7	**32.2**	**28.8**
+autocap 1-5 (concat)	**39.9**	32.0	28.7

Table 4: Adding automatic image captions (only the best one or all 5). The table shows BLEU scores in %. All results with Marian Amun.

We tried two models for the integration of those captions: (1) a dual attention multi-source model that adds another input sequence with its own decoder attention and (2) a concatenation model that adds auto captions at the end of the original input string separated by a special token. In the second model, attention takes care of learning how to use the additional information and previous work has shown that this, indeed, is possible (Niehues et al., 2016; Östling et al., 2017). For both models, we applied Marian NMT that already includes a working implementation of dual attention translations. Table 4 summarizes the scores on the three development test sets for English-French and English-German.

We can see that the dual attention model does not work at all and the scores slightly drop. The concatenation approach works better probably because the common attention

[3] Czech was later dropped as a target language due to time constraints.

[4] https://competitions.codalab.org/competitions/3221

model learns interactions between the different types of input. However, the improvements are small if any and the model basically learns to ignore the auto captions, which are often very different from the original input. The attention pattern in the example of Figure 1 shows one of the very rare cases where we observe at least some attention to the automatic captions.

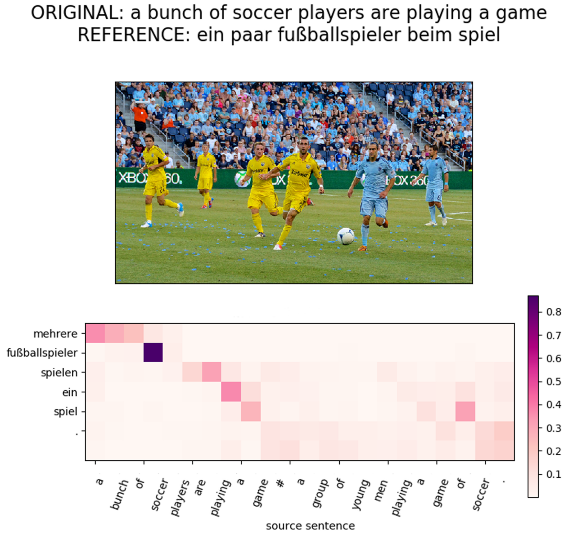

Figure 1: Attention layer visualization for an example where at least one of the attention weights for the last part of the sentence, which corresponds to the automatically generated captions, obtains a value above 0.3

4 Experiment 3: Multi-modal Transformer

One benefit of NMT, in addition to its strong performance, is its flexibility in enabling different information sources to be merged. Different strategies to include image features both on the encoder and decoder side have been explored. We are inspired by the recent success of the Transformer architecture to adapt some of these strategies for use with the Transformer.

Recurrent neural networks start their processing from some **initial hidden state**. Normally, a zero vector or a learned parameter vector is used, but the initial hidden state is also a natural location to introduce additional context e.g. from other modalities. Initializing can be applied in either the encoder (IMG$_E$) or decoder (IMG$_D$) (Calixto et al., 2017). These approaches are not directly applicable to the Transformer, as it is not a recurrent model, and lacks a comparable initial hidden state.

Double attention is another popular choice, used by e.g. Caglayan et al. (2017). In this approach, two attention mechanisms are used, one for each modality. The attentions can be separate or hierarchical. While it would be possible to use double attention with the Transformer, we did not explore it in this work. The multiple multi-head attention mechanisms in the Transformer leave open many challenges in how this integration would be done.

Multi-task learning has also been used, e.g. in the Imagination model (Elliott and Kádár, 2017), where the auxiliary task consists of reconstructing the visual features from the source encoding. Imagination could also have been used with the Transformer, but we did not explore it in this work.

The **source sequence** itself is also a possible location for including the visual information. In the IMG$_W$ approach, the visual features are encoded as a pseudo-word embedding concatenated to the word embeddings of the source sentence. When the encoder is a bidirectional recurrent network, as in Calixto et al. (2017), it is beneficial to add the pseudo-word both at the beginning and the end to make it available for both encoder directions. This is unnecessary in the Transformer, as it has equal access to all parts of the source in the deeper layers of the encoder. Therefore, we add the pseudo-word only to the beginning of the sequence. We use an affine projection of the image features $V \in \mathbb{R}^{80}$ into a pseudo-word embedding $x_I \in \mathbb{R}^{512}$

$$x_I = W_{src} \cdot V + b_I.$$

In the LIUM *trg-mul* (Caglayan et al., 2017), the **target embeddings** and visual features are interacted through elementwise multiplication.

$$y'_j = y_j \odot \tanh(W^{dec}_{mul} \cdot V)$$

Our initial gating approach resembles *trg-mul*.

4.1 Architecture

The baseline NMT for this experiment is the OpenNMT implementation of the Transformer. It is an encoder-decoder NMT system using the Transformer architecture (Vaswani et al., 2017) for both the encoder and decoder side. The Transformer is a deep, non-recurrent network for processing variable-length sequences. A Transformer is a stack of layers, consisting of two types of sub-layer: multi-head (MH) attention (Att) sub-layers and feed-forward (FF) sub-layers:

$$\begin{aligned}\mathrm{Att}(Q,K,V) &= \mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V\\ a_i &= \mathrm{Att}(QW_i^Q, KW_i^K, VW_i^V)\\ \mathrm{MH}(Q,K,V) &= [a_1;\ldots;a_h]W^O\\ \mathrm{FF}(x) &= \max(0, xW_1+b_1)W_2+b_2\end{aligned} \tag{1}$$

where Q is the input query, K is the key, and V the attended values. Each sub-layer is individually wrapped in a residual connection and layer normalization.

When used in translation, Transformer layers are stacked into an encoder-decoder structure. In the encoder, the layer consists of a self-attention sub-layer followed by a FF sub-layer. In self-attention, the output of the previous layer is used as queries, keys and values $Q = K = V$. In the decoder, a third context attention sub-layer is inserted between the self-attention and the FF. In context attention, Q is again the output of the previous layer, but $K = V$ is the output of the encoder stack. The decoder self-attention is also masked to prevent access to future information. Sinusoidal position encoding makes word order information available.

Decoder gate. Our first approach is inspired by *trg-mul*. A gating layer is introduced to modify the pre-softmax prediction distribution. This allows visual features to directly suppress a part of the output vocabulary. The probability of correctly translating a source word with visually resolvable ambiguity can be increased by suppressing the unwanted choices.

At each timestep the decoder output s_j is projected to an unnormalized distribution over the target vocabulary.

$$y_j = W \cdot s_j + b$$

Before normalizing the distribution using a softmax layer, a gating layer can be added.

EN-FR	flickr16	flickr17	mscoco17
IMG$_W$	*68.30*	**62.45**	52.86
enc-gate	68.01	61.38	**53.40**
dec-gate	67.99	61.53	52.38
enc-gate + dec-gate	**68.58**	*62.14*	*52.98*
EN-DE	flickr16	flickr17	mscoco17
IMG$_W$	*45.09*	40.81	36.94
enc-gate	44.75	**41.44**	**37.76**
dec-gate	**45.21**	40.79	36.47
enc-gate + dec-gate	44.91	*41.06*	*37.40*

Table 5: Comparison of strategies for integrating visual information (BLEU% scores). All results using Transformer, Multi30k+MS-COCO+SUBS3M$_{LM}$, Detectron mask surface, and domain labeling.

$$\begin{aligned} g &= \sigma(W^{dec}_{gate} \cdot V + b^{dec}_{gate})\\ y'_j &= y_j \odot g\end{aligned} \tag{2}$$

Preliminary experiments showed that gating based on only the visual features did not work. Suppressing the same subword units during the entire decoding of the sentence was too disruptive. We addressed this by using the decoder hidden state as additional input to control the gate. This causes the vocabulary suppression to be time dependent.

$$g_j = \sigma(U^{dec}_{gate} \cdot s_j + W^{dec}_{gate} \cdot V + b^{dec}_{gate}) \tag{3}$$

Encoder gate. The same gating procedure can also be applied to the output of the encoder. When using the encoder gate, the encoded source sentence is disambiguated, instead of suppressing part of the output vocabulary.

$$\begin{aligned} g_i &= \sigma(U^{enc}_{gate} \cdot h_i + W^{enc}_{gate} \cdot V + b^{enc}_{gate})\\ h'_i &= h_i \odot g_i\end{aligned} \tag{4}$$

The gate biases b^{dec}_{gate} and b^{enc}_{gate} should be initialized to positive values, to start training with the gates opened. We also tried combining both forms of gating.

4.2 Visual feature selection

Image feature selection was performed using the LIUM-CVC translation system (Caglayan et al., 2017) training on the WMT18 training

EN-FR	flickr16	flickr17	mscoco17
SUBS3M$_{LM}$ detectron	68.30	62.45	52.86
+ensemble-of-3	68.72	62.70	53.06
−visual features	**68.74**	**62.71**	53.14
−MS-COCO	67.13	61.17	**53.34**
−multi-lingual	68.21	61.99	52.40
SUBS6M$_{LM}$ detectron	68.29	61.73	53.05
SUBS3M$_{LM}$ gn2048	67.74	61.78	52.76
SUBS3M$_{LM}$ text-only	67.72	61.75	53.02
EN-DE	flickr16	flickr17	mscoco17
SUBS3M$_{LM}$ detectron	45.09	40.81	36.94
+ensemble-of-3	45.52	**41.84**	**37.49**
−visual features	**45.59**	41.75	37.43
−MS-COCO	45.11	40.52	36.47
−multi-lingual	44.95	40.09	35.28
SUBS6M$_{LM}$ detectron	45.50	41.01	36.81
SUBS3M$_{LM}$ gn2048	45.38	40.07	36.82
SUBS3M$_{LM}$ text-only	44.87	41.27	36.59
+multi-modal finetune	44.56	41.61	36.93

Table 6: Ablation experiments (BLEU% scores). The row SUBS3M$_{LM}$ *detectron* shows our best single model. Individual components or data choices are varied one by one. + stands for adding a component, and − for removing a component or data set. Multiple modifications are indicated by increasing the indentation.

data, and evaluating on the *flickr16, flickr17* and *mscoco17* data sets. This setup is different from our final NMT architecture as the visual feature selection stage was performed at an earlier phase of our experiments. However, the LIUM-CVC setup without training set expansion was also faster to train which enabled a more extensive feature selection process.

We experimented with a set of state-of-the-art visual features, described below.

CNN-based features are 2048-dimensional feature vectors produced by applying reverse spatial pyramid pooling on features extracted from the 5th Inception module of the pre-trained GoogLeNet (Szegedy et al., 2015). For a more detailed description, see (Shetty et al., 2018). These features are referred to as gn2048 in Table 6.

Scene-type features are 397-dimensional feature vectors representing the association score of an image to each of the scene types in SUN397 (Xiao et al., 2010). Each association score is determined by a separate Radial Basis Function Support Vector Machine (RBF-SVM) classifier trained from pre-trained GoogLeNet CNN features (Shetty et al., 2018).

Action-type features are 40-dimensional feature vectors created with RBF-SVM classifiers similarly to the scene-type features, but using the Stanford 40 Actions dataset (Yao et al., 2011) for training the classifiers. Pre-trained GoogLeNet CNN features (Szegedy et al., 2015) were again used as the first-stage visual descriptors.

Object-type and location features are generated using the Detectron software[5] which implements Mask R-CNN (He et al., 2017) with ResNeXt-152 (Xie et al., 2017) features. Mask R-CNN is an extension of Faster R-CNN object detection and localization (Ren et al., 2015) that also generates a segmentation mask for each of the detected objects. We generated an 80-dimensional *mask surface* feature vector by expressing the image surface area covered by each of the MS-COCO classes based on the detected masks.

We found that the Detectron mask surface resulted in the best BLEU scores in all evaluation data sets for improving the German translations. Only for *mscoco17* the results could be slightly improved with a fusion of mask surface and the SUN 397 scene-type feature. For French, the results were more varied, but we focused on improving the German translation results as those were poorer overall. We experimented with different ways of introducing the image features into the translation model implemented in LIUM-CVC, and found as in (Caglayan et al., 2017), that *trg-mul* worked best overall.

Later we learned that the *mscoco17* test set has some overlap with the COCO 2017 training set, which was used to train the Detectron models. Thus, the results on that test set may not be entirely reliable. However, we still feel confident in our conclusions as they are also confirmed by the *flickr16* and *flickr17* test sets.

4.3 Training

We use the following parameters for the network:[6] 6 Transformer layers in both encoder and decoder, 512-dimensional word embeddings and hidden states, dropout 0.1, batch

[5] https://github.com/facebookresearch/Detectron

[6] Parameters were chosen following the OpenNMT FAQ http://opennmt.net/OpenNMT-py/FAQ.html#how-do-i-use-the-transformer-model

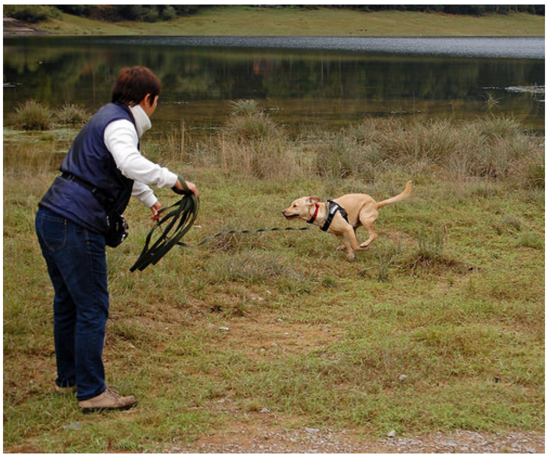

Figure 2: Image 117 was translated correctly as feminine "eine besitzerin steht still und ihr brauner hund rennt auf sie zu ." when not using the image features, but as masculine "ein besitzer …" when using them. The English text contains the word "her". The person in the image has short hair and is wearing pants.

size 4096 tokens, label smoothing 0.1, Adam with initial learning rate 2 and β_2 0.998.

For decoding, we use an ensemble procedure, in which the predictions of 3 independently trained models are combined by averaging after the softmax layer to compute combined prediction.

We evaluate the systems using uncased BLEU using multibleu. During tuning, we also used characterF (Popovic, 2015) with β set to 1.0.

There are no images paired with the sentences in OpenSubtitles. When using OpenSubtitles in training multi-modal models, we feed in the mean vector of all visual features in the training data as a dummy visual feature.

4.4 Results

Based on the previous experiments, we chose the Transformer architecture, Multi30k+MS-COCO+SUBS3M$_{LM}$ data sets, Detectron mask surface visual features, and domain labeling.

Table 5 shows the BLEU scores for this configuration with different ways of integrating the visual features. The results are inconclusive. The ranking according to chrF-1.0 was not any clearer. Considering the results as a whole and the simplicity of the method, we chose IMG$_W$ going forward.

Table 6 shows results of ablation experiments removing or modifying one component or data choice at a time, and results when using ensemble decoding. Using ensemble decoding gave a consistent but small improvement. Multi-lingual models were clearly better than mono-lingual models. For French, 6M sentences of subtitle data gave worse results than 3M.

We experimented with adding multi-modality to a pre-trained text-only system using a fine tuning approach. In the fine tuning phase, a *dec-gate* gating layer was added to the network. The parameters of the main network were frozen, allowing only the added gating layer to be trained. Despite the freezing, the network was still able to unlearn most of the benefits of the additional text-only data. It appears that the output vocabulary was reduced back towards the vocabulary seen in the multi-modal training set. When the experiment was repeated so that the fine-tuning phase included the text-only data, the performance returned to approximately the same level as without tuning (+multi-modal finetune row in Table 6).

To explore the effect of the visual features on the translation of our final model, we performed an experiment where we retranslated using the ensemble while "blinding" the model. Instead of feeding in the actual visual features for the sentence, we used the mean vector of all visual features in the training data. The results are marked *-visual features* in Table 6. The resulting differences in the translated sentences were small, and mostly consisted of minor variations in word order. BLEU scores for French were surprisingly slightly improved by this procedure. We did not find clear examples of successful disambiguation. Figure 2 shows one example of a detrimental use of visual features.

It is possible that adding to the training data forward translations of MS-COCO captions from a text-only translation system introduced a biasing effect. If there is translational ambiguity that should be resolved using the image, the text-only system will not be able to resolve it correctly, instead likely yielding the word that is most frequent in that textual context. Using such data for training a multi-modal system might bias it towards ignoring the image.

On this year's *flickr18* test set, our system scores 38.54 BLEU for English-to-German and 44.11 BLEU for English-to-French.

5 Conclusions

Although we saw an improvement from incorporating multi-modal information, the improvement is modest. The largest differences in quality between the systems we experimented with can be attributed to the quality of the underlying text-only NMT system.

We found the amount of in-domain training data and multi-modal training data to be of great importance. The synthetic MS-COCO data was still beneficial, despite being forward translated, and the visual features being overconfident due to being extracted from a part of the image classifier training data.

Even after expansion with synthetic data, the available multi-modal data is dwarfed by the amount of text-only data. We found that movie subtitles worked well for this purpose. When adding text-only data, domain adaptation was important, and increasing the size of the selection met with diminishing returns.

Current methods do not fully address the problem of how to efficiently learn from both large text-only data and small multi-modal data simultaneously. We experimented with a fine tuning approach to this problem, without success.

Although the effect of the multi-modal information was modest, our system still had the highest performance of the task participants for the English-to-German and English-to-French language pairs, with absolute differences of +6.0 and +3.5 BLEU%, respectively.

Acknowledgments

This work has been supported by the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No 780069, and by the Academy of Finland in the project 313988. In addition the Finnish IT Center for Science (CSC) provided computational resources. We would also like to acknowledge the support by NVIDIA and their GPU grant.

References

Ozan Caglayan, Walid Aransa, Adrien Bardet, Mercedes García-Martínez, Fethi Bougares, Loïc Barrault, Marc Masana, Luis Herranz, and Joost Van de Weijer. 2017. LIUM-CVC submissions for WMT17 multimodal translation task. In *Proceedings of the Second Conference on Machine Translation*.

Iacer Calixto, Koel Dutta Chowdhury, and Qun Liu. 2017. DCU system report on the WMT 2017 multi-modal machine translation task. In *Proceedings of the Second Conference on Machine Translation*. pages 440–444.

Desmond Elliott and Àkos Kádár. 2017. Imagination improves multimodal translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. volume 1, pages 130–141.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask R-CNN. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, pages 2980–2988.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2016. Google's multilingual neural machine translation system: enabling zero-shot translation. *arXiv preprint arXiv:1611.04558* .

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*. Association for Computational Linguistics, Melbourne, Australia, pages 116–121. http://www.aclweb.org/anthology/P18-4020.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proc. ACL*. https://doi.org/10.18653/v1/P17-4012.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: common objects in context. *CoRR* abs/1405.0312. http://arxiv.org/abs/1405.0312.

Jan Niehues, Eunah Cho, Thanh-Le Ha, and Alex Waibel. 2016. Pre-translation for neural machine translation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. pages 1828–1836.

Robert Östling, Yves Scherrer, Jörg Tiedemann, Gongbo Tang, and Tommi Nieminen. 2017. The helsinki neural machine translation system. In *Proceedings of the Second Conference on Machine Translation*. pages 338–347.

Maja Popovic. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *WMT15*. pages 392–395.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems (NIPS)*. pages 91–99.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. In *ACL16*.

Rakshith Shetty, Hamed Rezazadegan Tavakoli, and Jorma Laaksonen. 2018. Image and video captioning with augmented neural architectures. *IEEE MultiMedia* To appear.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. pages 1–9.

Jörg Tiedemann. 2009. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In N. Nicolov, K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing*, John Benjamins, Amsterdam/Philadelphia, Borovets, Bulgaria, volume V, pages 237–248.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. pages 6000–6010.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE conference on Computer vision and pattern recognition (CVPR)*. IEEE, pages 3485–3492.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2017. Aggregated residual transformations for deep neural networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pages 5987–5995.

Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas J. Guibas, and Fei-Fei Li. 2011. Human action recognition by learning bases of action attributes and parts. In *International Conference on Computer Vision (ICCV)*. Barcelona, Spain, pages 1331–1338.