Dynamic Context-guided Capsule Network for Multimodal Machine Translation

Huan Lin  
Xiamen University, China

Fandong Meng  
Tencent WeChat AI, China

Jinsong Su  
Xiamen University, China

Yongjing Yin  
Xiamen University, China

Zhengyuan Yang  
University of Rochester, USA

Yubin Ge  
University of Illinois at Urbana-Champaign, USA

Jie Zhou  
Tencent WeChat AI, China

Jiebo Luo  
University of Rochester, USA

ABSTRACT

Multimodal machine translation (MMT), which mainly focuses on enhancing text-only translation with visual features, has attracted considerable attention from both computer vision and natural language processing communities. Most current MMT models resort to attention mechanism, global context modeling or multimodal joint representation learning to utilize visual features. However, the attention mechanism lacks sufficient semantic interactions between modalities while the other two provide fixed visual context, which is unsuitable for modeling the observed variability when generating translation. To address the above issues, in this paper, we propose a novel Dynamic Context-guided Capsule Network (DCCN) for MMT. Specifically, at each timestep of decoding, we first employ the conventional source-target attention to produce a timestep-specific source-side context vector. Next, DCCN takes this vector as input and uses it to guide the iterative extraction of related visual features via a context-guided dynamic routing mechanism. Particularly, we represent the input image with global and regional visual features, we introduce two parallel DCCNs to model multimodal context vectors with visual features at different granularities. Finally, we obtain two multimodal context vectors, which are fused and incorporated into the decoder for the prediction of the target word. Experimental results on the Multi30K dataset of English-to-German and English-to-French translation demonstrate the superiority of DCCN. Our code is available on https://github.com/DeepLearnXMU/MM-DCCN.

CCS CONCEPTS

• Computing methodologies → Natural language generation; Computer vision; Machine translation.

KEYWORDS

Multimodal Machine Translation, Capsule Network, Transformer

ACM Reference Format:
Huan Lin, Fandong Meng, Jinsong Su, Yongjing Yin, Zhengyuan Yang, Yubin Ge, Jie Zhou, and Jiebo Luo. 2020. Dynamic Context-guided Capsule Network for Multimodal Machine Translation. In Proceedings of the 28th ACM International Conference on Multimedia (MM ’20), October 12–16, 2020, Seattle, WA, USA. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3394171.3413715

1 INTRODUCTION

MMT significantly extends the conventional text-based machine translation by taking corresponding images as additional inputs [5, 19, 42]. The assumption behind this is that the translation is expected to be more accurate compared to purely text-based translation, since the visual context helps to solve data sparsity and ambiguity problems [28]. As shown in Figure 1, with the help of the image, MMT model is able to correctly translate “bank” to “erdwall”. Overall, the research on MMT is of great significance. On the one hand, similar to other multimodal tasks such as image captioning [11, 12, 24, 41] and visual question answering [22, 33, 37, 64], MMT involves computer vision and natural language processing (NLP) and proves the effectiveness of visual features in translation tasks. In other words, it not only requires an algorithm with in-depth understanding of visual contexts, but also connects its interpretation with a language model to create a natural sentence. On the other hand, MMT has wide applications, such as translating multimedia news, product information and movie subtitles [63]. Therefore, MMT has become an attractive but challenging multimodal task.

Very importantly, one of the key issues in MMT is how to effectively utilize visual features during the process of translation. To achieve this goal, three categories of methods have been investigated: (1) exploiting visual features as global visual context [8, 23, 27]; (2) applying attention mechanism to extract visual context, where one common approach is to employ a timestep-specific attention mechanism to extract visual context [9, 17, 26] and another way is to use source hidden states to consider visual features and then use the obtained invariant visual context as a complement to source hidden states [3, 16]; (3) learning multimodal joint representations [10, 21, 63]. Despite their successes, these approaches still have various shortcomings. First, global visual context and
learning multimodal joint representations cannot encode the observed variability when generating translation. Second, according to previous studies [34, 52], extracting visual context is beyond the capacity of a single-step attention due to the complexity of multimodal tasks. Although multiple attention layers can refine the extraction of visual context, the improvement is still limited. One possible reason is that too many parameters of multiple attention layers make the model vulnerable to over-fitting, especially when limited training examples are given in MMT. Moreover, the above methods only use visual features at global or regional level, which is unable to offer sufficient visual guidance. As a result, visual features are not fully utilized, limiting the potential of MMT models.

To overcome these issues, in this paper, we propose a novel Dynamic Context-guided Capsule Network (DCCN) for MMT. At each timestep of decoding, we first employ the standard source-target attention to produce a timestep-specific source-side context vector. Next, DCCN takes this context vector as input and uses it to guide the iterative extraction of related visual context during the dynamic routing process, where a multimodal context vector is updated simultaneously. In particular, to fully exploit image information, we employ DCCN to extract visual features at two complementary granularities: global visual features and regional visual features, respectively. In this way, we can obtain two multimodal context vectors that are then fused for the prediction of the current target word. Compared with previous studies, DCCN is able to dynamically extract visual context without introducing a large number of parameters, which is suitable to model such kind of variability observed in machine translation. Potentially, DCCN learns a better multimodal joint representation for MMT. Therefore, it is also applicable to other related tasks that require a joint representation of different modalities, such as visual question answering. In summary, the major contributions of our work are three-fold:

- We introduce a capsule network to effectively capture visual features at different granularities for MMT, which has advantage of effectively capturing visual features without explosive parameters. To the best of our knowledge, our work is the first attempt to explore a capsule network to extract visual features for MMT.
- We propose a novel context-guided dynamic routing for the capsule network, which uses the timestep-specific source-side context vector as the guiding signal to dynamically produce a multimodal context vector for MMT.

- We conduct experiments on the Multi30K English-German and English-French datasets. Experimental results show that our model significantly outperforms several competitive MMT models.

2 RELATED WORK

The related work mainly includes the studies of multimodal context modeling in MMT and capsule networks.

**Multimodal context modeling in MMT.** How to fully exploit context for neural machine translation has always been a hot research topic [44–47, 49, 58–61], which is the same for MMT. The commonly-used approaches to extract multimodal context for MMT can be classified into three categories: (1) Learning global visual features for MMT. For instance, Huang et al. [27] concatenated global and regional visual features with source sequences. Calixto and Liu [8] utilized global visual features as additional tokens in the source sequence, to initialize the encoder hidden states or initialize the first decoder hidden state. (2) Leveraging attention mechanism to exploit visual features. In this aspect, Caglayan et al. [7] and Calixto et al. [9] incorporated spatial visual features into the MMT model via an independent attention mechanism. Furthermore, Delbruck and Dupont [17] employed Compact Bilinear Pooling to fuse the attention-based context vectors of two modalities. Meanwhile, Libovický and Helcl [32] explored flat and hierarchical combinations to fuse the attention-based context vectors of two modalities. Instead of using attention mechanism, Grönroos et al. [23] introduced a gating layer to modify the prediction distribution on both visual features and decoder states. Unlike previous studies, Delbruck and Dupont [16] utilized the attention mechanism on visual inputs for the source hidden states. Along this line, Arslan et al. [3] extended this approach into Transformer, and Helcl et al. [26] used timestep-specific source-side context vector as attention query to dynamically produce the visual context vectors. (3) Applying multi-task learning to jointly model translation task with other visual related tasks. For example, Elliott and Kádár [21] decomposed multimodal translation into two sub-tasks: learning to translate and learning visually grounded representations. Zhou et al. [63] optimized the learning of a shared visual language embedding and a multimodal attention-based translator. Calixto et al. [10] introduced a continuous latent variable for MMT, which contains the underlying semantic information extracted from texts and images. Recently, Yin et al. [57] uses a unified multi-modal graph to capture various semantic relationships between multi-modal semantic units.

**Capsule Network.** Recently, capsule network has been widely used in computer vision [29, 31, 35, 40, 53, 54] and NLP [1, 13, 54, 55] tasks. Specific to machine translation, Wang et al. [50] employed dynamic routing algorithm to model child-parent relationships between lower and higher encoder layers. Yang et al. [56] proposed a query-guided capsule networks to cluster context information into different perspectives from which the target translation may concern. Zheng et al. [62] separated translated and untranslated source words into different groups of capsules.

To the best of our knowledge, our work is the first attempt to introduce the capsule network into MMT. Furthermore, we use the timestep-specific source-side context vector rather than static
source hidden states to guide the routing procedure. Therefore, we can dynamically produce visual context for MMT.

## 3 OUR MODEL

As shown in Figure 2, our model is based on Transformer [48]. The most important feature of our model is that two DCCNs are introduced to dynamically learn multimodal context vectors for generating translations.

### 3.1 Encoder

Given the source sentence $X$, we represent each source word as the sum of its word embedding and positional encoding. Next, we follow Vaswani et al. [48] to use a stack of $L_e$ identical layers to encode $X$, where each layer consists of two sub-layers. Note that we also introduce residual connection and layer normalization to each sub-layer, of which the descriptions are omitted. Specifically, at the $l$-th layer ($1 \leq l \leq L_e$), the first sub-layer is a multi-head self-attention:

$$ H_e^{(l)} = \text{MultiHead}(S^{(l-1)}, S^{(l-1)}, S^{(l-1)}), $$

where $H_e^{(l)}$ is the temporary encoder hidden states. MultiHead($\ast$) is a multi-head self-attention function, and $S^{(l-1)} \in \mathbb{R}^{d_w \times |X|}$ is the representation of the source sentence at the $(l-1)$-th layer, $d_w$ is the model dimension. Particularly, $S^{(0)}$ is the concatenation of all source word embeddings. Taking a query matrix $Q$, a key matrix $K$ and a value matrix $V$ as inputs, MultiHead($\ast$) is defined as follows:

$$ \text{MultiHead}(Q, K, V) = \text{Concat}(head_1, \ldots, head_N)W^C $$

where $head_k = \text{Attention}(QW^Q_k, KW^k_k, VW^V_k)$

$$ = \text{Softmax}(\frac{QW^Q_kKW^k_k^T}{\sqrt{d_w}})VW^V_k, $$

where $W^C$ and $W^*_k$ are learnable parameter matrices.

The second sub-layer is a position-wise fully connected feed-forward network. It is applied to each position separately and identically, forming the representation $S^{(l)}$ of the source sentence as

$$ S^{(l)} = \text{FFN}(H_e^{(l)}), $$

where FFN($\ast$) is a position-wise feed-forward function.

### 3.2 Decoder

As shown in Figure 2, our decoder is an extension of the Transformer decoder [48]. It takes the already generated sequence as inputs and uses a stack of $L_d$ identical layers to produce target-side hidden states. Similar to the standard Transformer decoder, each layer of our decoder contains three sub-layers. The only difference is that at the last decoder layer, two DCCNs are equipped to produce timestep-specific multimodal context vectors for MMT.

Specifically, the first sub-layer is also a multi-head self-attention:

$$ H_d^{(l)} = \text{MultiHead}(T^{(l-1)}, T^{(l-1)}, T^{(l-1)}), 1 \leq l \leq L_d, $$

where $T^{(l-1)}$ is the temporary decoder hidden states.
where $\mathbf{H}_1^{(l)}$ denotes the temporary decoder hidden states, produced by a multi-head self-attention mechanism fed with the target-side hidden states $\mathbf{T}^{(l-1)}$ at the previous layer. Typically, at the second sub-layer, a multi-head source-target attention mechanism is used to dynamically produce the timestep-specific source-side context vectors $\mathbf{C}_1^{(l)}$:  

$$\mathbf{C}_1^{(l)} = \text{MultiHead}(\mathbf{H}_1^{(l)}, \mathbf{S}^{(l_1)}, \mathbf{S}^{(l_2)}), 1 \leq l \leq L_d.$$  

The third sub-layer is a position-wise fully connected feed-forward neural network, of which the definition depends on the decoder layer. At the first $(L_d - 1)$ layers, this sub-layer produces the target-side hidden states $\mathbf{T}^{(l)}$ as follows:  

$$\mathbf{T}^{(l)} = \text{FFN}(\mathbf{C}_1^{(l)}), 1 \leq l \leq L_d - 1.$$  

Very importantly, at the last $(L_d)$th decoder layer, we introduce two DCCNs between the second and third sub-layers to learn multimodal context representations at the $t$-th timestep as follows:

$$\bar{\mathbf{m}}_g = \text{CapsuleNet}(\mathbf{C}_1^{(L_d)}, \mathbf{I}_g),$$  

$$\bar{\mathbf{m}}_r = \text{CapsuleNet}(\mathbf{C}_1^{(L_d)}, \mathbf{I}_r),$$

where CapsuleNet(·) is a context-guided dynamic routing function, $\mathbf{C}_1^{(L_d)}$ is the $t$-th column vector of $\mathbf{C}_1^{(L_d)}$, representing the source-side context vectors at the $t$-th timestep, $\mathbf{I}_g$ and $\mathbf{I}_r$ represent global visual features and local visual features respectively, as depicted in subsection 3.2.1. By using DCCN, we can iteratively extract the related visual features to dynamically produce better multimodal context representations for MMT. We will describe this procedure in subsection 3.2.2. Next, we fuse $\bar{\mathbf{m}}_g$ and $\bar{\mathbf{m}}_r$ via the following gating mechanism:

$$\mathbf{M}_{1-t}^{(L_d)} = \alpha \bar{\mathbf{m}}_g + (1 - \alpha) \bar{\mathbf{m}}_r,$$  

$$\alpha = \text{Sigmoid}(\mathbf{W}_g \bar{\mathbf{m}}_g + \mathbf{W}_r \bar{\mathbf{m}}_r),$$

where $\mathbf{W}_g$ and $\mathbf{W}_r$ are learnable parameters. Correspondingly, the Eq. 6 at the last decoder layer becomes

$$\mathbf{T}^{(L_d)} = \text{FFN}(\mathbf{M}_{1-t}^{(L_d)}).$$  

Finally, with the target-side hidden states generated by Eq. 11, our decoder adopts a Softmax layer to generate the probability distribution of the current target word $y_t$:

$$P(y_t | X, Y_{<t}; \theta) \propto \exp(\mathbf{W}_g \mathbf{T}^{(L_d)}),$$

where $Y_{<t}$ is the previously generated target words $y_1 y_2 \ldots y_{t-1}$, $\mathbf{W} \in \mathbb{R}^{V_g \times d_v}$ is a model parameter, $V_g$ is the target vocabulary, and $\mathbf{T}^{(L_d)}$ is the $t$-th column vector of $\mathbf{T}^{(l)}$ for predicting $y_t$.

### 3.2.2 Dynamic Context-guided Capsule Network

Our DCCN is a significant extension of the conventional capsule network. Thus, it retains the advantages on iterative feature extraction of capsule network, which has shown effective in many computer vision [29, 31, 35, 40, 53, 54] and NLP [1, 13, 50, 54–56, 62] tasks. More importantly, unlike the conventional capsule network that only captures static visual context, DCCN introduces the timestep-specific source-side context vector to guide the extraction of multimodal context, which can model the observed variability during translation. Figure 3 shows the architecture of DCCN. Similar to the conventional capsule network, DCCN consists of (1) low-level capsules $\{u_j\}_{j=1}^{N_u}$ (the first column from the right) encoding the visual features of the input image and (2) high-level capsules $\{v_j\}_{j=1}^{N_v}$ (the second column from the left) encoding the related visual features. Besides, it includes multimodal context capsules $\{m_j\}_{j=1}^{N_m}$ (the first column from the left), where $m_j$ indicates a temporary multimodal context vector iteratively updated with $v_j$ and is used as query to guide the extraction of related visual features.

We summarize the training procedure in Algorithm 1. Given a visual features matrix $X$, we use low-level capsules $\{u_j\}_{j=1}^{N_u}(u_j \in \mathbb{R}^{256})$ to individually encode the semantic representation of each row of $X$ (Line 4), and initialize each multimodal context capsule $m_j$, each of the 196 rows consists of a 256D feature vector; and (2) **regional visual features** that illustrate class annotations of each region (e.g. cat, arms, peak). Following Anderson et al. [2], we employ the R-CNN based bottom-up attention to identify the regions with class annotations. For each region, we generate the corresponding prediction probability distribution over 1,600 classes from Visual Genome. To represent each region as a vector, we project its class annotations into word embeddings and define the region vector as the weighted sum of its class annotation embeddings. Finally, all region vectors are concatenated to represent the semantics of input image. In practice, we keep the number of predicted regions up to 10 so as to reduce negative effects of abundant regions, therefore the regional visual features can be represented as a 10A⇒256 matrix $I_r$, where each of the 10 rows consists of a 256D feature vector.
Algorithm 1 Context-guided Dynamic Routing Mechanism.

1: **Input**: the timestep-specific source-side context vector $C_{t \rightarrow t}^{(L_d)}$, the input image $I$, the low-level capsule number $N_v$, the high-level capsule number $N_m$, the iteration number $N_{itr}$
2: **Output**: the multimodal context vector $\overrightarrow{m}$
3: for $i = 1 \ldots N_v$ do
4: Initialize the low-level capsules $u_i$ with $I$
5: end for
6: for $j = 1 \ldots N_m$ do
7: Initialize the multimodal context capsules $m_j$ with $C_{t \rightarrow t}^{(L_d)}$
8: end for
9: for each low-level capsule $u_i$ do
10: for each high-level capsule $v_j$ do
11: $b_{ij} \leftarrow 0$
12: $\hat{u}_{ij} = W_j u_i$
13: $\rho_{ij} \leftarrow \tanh(\text{PCC}(u_i, W_m m_j))$
14: end for
15: end for
16: for $itr = 1 \ldots N_{itr}$ do
17: for each low-level capsule $u_i$ do
18: for each high-level capsule $v_j$ do
19: $(\epsilon_{ij})_{j=1}^{N_m} \leftarrow \text{Softmax}(b_{ij})_{j=1}^{N_m}$
20: end for
21: end for
22: for each high-level capsule $v_j$ do
23: $v_j = \sum_i (\epsilon_{ij} + \rho_{ij} \hat{u}_{ij})$
24: $m_j = m_j \odot W_v v_j$
25: end for
26: for each low-level capsule $u_i$ do
27: for each high-level capsule $v_j$ do
28: $\rho_{ij} \leftarrow \tanh(\text{PCC}(u_i, W_m m_j))$
29: $b_{ij} \leftarrow b_{ij} + \rho_{ij}(\hat{u}_{ij} \cdot v_j)$
30: end for
31: end for
32: end for
33: $\overrightarrow{m} = \text{FuseMultimodalContext}(m_1, \ldots, m_{N_m})$

with the timestep-specific source-side context vector $C_{t \rightarrow t}^{(L_d)}$ (Line 7). Following the conventional capsule network, we introduce the matrix $W_{ij}$ to transform the $i$-th low-level capsule $u_i$ into $\hat{u}_{ij}$ (Line 12), which will be used to generate the $j$-th high-level capsule $v_j$.

Next, we introduce the coefficient $\rho_{ij}$ to measure the cross-modal correlation between $u_i$ and the multimodal context vector $m_j$ (Line 13), which can be subsequently used to generate high-level capsules $v_j$ and update $b_{ij}$, benefiting the extraction of related visual features. Formally, the correlation function is defined as

$$\rho_{ij} = \tanh(\text{PCC}(u_i, W_m m_j)) = \tanh\left(\frac{\text{Cov}(u_i, W_m m_j)}{\sigma(u_i) \sigma(W_m m_j)}\right), \quad (13)$$

where PCC(*) indicates the Pearson Correlation Coefficients, $W_m$ is a parameter matrix that maps $m_j$ to the same semantic space of $u_i$, Cov(*) is the covariance and $\sigma(*)$ is the standard deviation. When $\rho_{ij}$ is close to +1, the visual features encoded by $u_i$ is closely related to $m_j$, otherwise it indicates negative correlation.

Then, we conduct $N_{itr}$ iterations of routing to capture related visual features at current timestep (Line 16 to Line 32). At each iteration, we generate high-level capsules from low-level capsules. To do this, we employ a Softmax function along columns with the logits $b_{ij}$ initialized as 0 to calculate the coupling coefficient $\epsilon_{ij}$ (Line 19). Afterwards, we generate the high-level capsule $v_j$ to represent visual context as the weighted sum of $\hat{u}_{ij}$ according to their corresponding $\epsilon_{ij}$ and the cross-modal correlation coefficient $\rho_{ij}$ (Line 23). Note that unlike the conventional dynamic routing algorithm [38] where $v_j$ only depends on $\epsilon_{ij}$ and $\hat{u}_{ij}$, we further introduce $\rho_{ij}$ that enables the most relevant visual features to be iteratively clustered into high-level capsules. Note that the conventional capsule network [38] uses the norm of high-level capsule to represent prediction probability, thus the norm of $v_j$ is adjusted to $[0, 1]$ using the “squashing” function. Different from that, $v_j$ represents visual context in DCCN. Therefore we do not apply “squashing” function in our algorithm. Further, we introduce a transformation matrix $W_v$ to map visual context $v_j$ into the semantic space of multimodal context and follow Wu and Mooney [51] to update $m_j$ with the captured visual context $v_j$ (Line 24). By doing so, we expect the updated multimodal context capsule can be better exploited to guide the routing procedure at the next iteration.

Finally, we update $\rho_{ij}$ (Line 28), and then use it to guide the updating of $b_{ij}$ (Line 29). Different from conventional dynamic routing algorithm, where $b_{ij}$ only depends on the cumulative “agreement” between $u_i$ and $v_j$, we control the updating range of $b_{ij}$ according to the cross-modal correlation $\rho_{ij}$.

Through $N_{itr}$ iterations of routing, we obtain $N_m$ multimodal context capsules $\{m_j\}_{j=1}^{N_m}$ fused by a linear transformation to produce the final multimodal context vector $\overrightarrow{m}$ (Line 33).

4 EXPERIMENTS

4.1 Dataset

To investigate the effectiveness of our proposed model, we conduct experiments on the Multi30K dataset [20], which is an extended version of the Flickr30K Entities and has been widely used in MMT [5, 19, 42]. For each image, one of the English (EN) descriptions was selected and manually translated into German (DE) and French (FR) by professional translators [42]. The dataset contains 29,000 instances for training, 1,024 for validation and 1,000 for testing. We also report results on the WMT2017 test set with 1,000 instances and the MSCOCO test set containing 461 out-of-domain instances with ambiguous verbs. Besides, as mentioned in subsection 3.2.1, we represent the input image with visual features in two granularities.

We apply the MOSES scripts1 to preprocess datasets. We then employ the Byte Pair Encoding [39] with 10,000 merging operations to convert tokens into subwords.

4.2 Setup

We develop our proposed model based on OpenNMT Transformer [30]. Since the size of training corpus is small and the trained model tends to be over-fitting, we first perform a small grid search to obtain a set of hyper-parameters on the EN⇒DE validation set, specifically, the layer numbers of both encoder and decoder are set

1http://www.statmt.org/moses/
to 4 and the number of attention heads is set to 8. Both hidden size and embedding size are set to 256.

As implemented in [48], we use the Adam optimizer with $\beta_2 = 0.998$ and scheduled learning rate to optimize various models. The learning rate is initialized as 1. During training, each batch consists of approximately 3,700 source and target tokens. Besides, we employ the dropout strategy [43] with rate 0.5 to enhance the robustness of our model. Finally, we adopt the MultiEval scripts [14] to evaluate the translation quality in terms of BLEU [36] and METEOR [18]. Particularly, we run all models three times for each experiment and report the average results.

The context-guided dynamic routing is important for the generation of multimodal context. Therefore, we investigate the impacts of its hyper parameters on the routing mechanism: high-level capsule number $N_v$ and routing iteration number $N_{itr}$. To this end, we try different numbers of high-level capsules and routing iteration numbers to train our model: $N_v$ from 1 to 3, $N_{itr}$ from 1 to 4 on the validation set. We observe that $N_v$ larger than 1 and $N_{itr}$ larger than 3 do not lead to significant improvements and increase the GPU memory requirement. Hence, we use $N_v=1$ and $N_{itr}=3$ in all subsequent experiments.

### 4.3 Baselines

We directly refer to our MMT model as DCCN and compare it with the following commonly-used MMT baselines:

- **Transformer** [48] A text-only machine translation model.
- **Encoder-attention** [16]. It incorporates an encoder-based visual attention mechanism into Transformer, which uses source hidden states to consider visual features and then augment each source hidden state with its corresponding visual context. Please note that [16] is based on RNN and we implement it on Transformer for comparability.
- **Doubly-attention** [26]. A doubly attentive Transformer that introduces an additional visual attention sub-layer to exploit visual features. Specifically, this sub-layer is inserted between the source-target attention and feed-forward sub-layer. For visual attention, the context vectors from the source-target attention are used as queries, and the context vectors of visual attention are fed into the feed-forward sub-layer.

We also display the performance of several dominant MMT models on the same datasets. **Stochastic attention** [15] is a stochastic and sampling-based attention mechanism, which focuses on only one spatial location of the image at every timestep. It is also the model of best performance in [15]. **Imagination** [21] employs multi-task learning to jointly two sub-tasks: translating and visually grounded representation prediction. **Fusion-conv** [6] employs a single feed-forward network to establish the attention alignment between visual features and target-side hidden states at each timestep, where all spatial locations of image are considered to derive the context vector. **Trg-mul** [6] modulates each target word embedding with visual features using element-wise multiplication. **Latent Variable MMT** [10] exploits the interactions between visual and textual features for MMT through a latent variable, which can be seen as a multimodal stochastic embedding of an image and its target language description. **Deliberation Network** [28] is based on a translate-and-refine strategy, where visual features are only used by the decoder at the second stage.

### 4.4 Results on the EN$\Rightarrow$DE Translation Task.

**Parameters** Introducing visual feature features into Transformer model brings more parameters. As shown in Table 1, Transformer (Row 7) has 16.1M parameters. Encoder-attention (Row 8) adds a layer normalization, a fully connected layer and two multi-head attention layers, introducing 1.1M parameters, and Doubly attention (Row 9) increases 4.0M parameters by adding two layer normalization and two multi-head attention layers. By contrast, DCCN (Row 10) only introduces 1.0M extra parameters. Thus, DCCN introduces a small number of extra parameters compared to Transformer, and requires smaller parameters than two multimodal baselines.

**Model Performance** Table 1 shows the translation quality on EN$\Rightarrow$DE translation task. It is obvious that DCCN outperforms most of the existing models and all baselines except for lower METEOR scores than Fusion-conv and Trg-mul, of which model parameters are selected based on METEOR.
Figure 4: BLEU scores on different translation groups divided according to source sentence lengths. Since MSCOCO only contains two sentences longer than 20, we divide all sentences longer than 20 into one group for testing.

Table 2: Ablation study of our model on the EN⇒DE translation task.

| # | Model                                   | Test2016 |           | Test2017 |           | MSCOCO |           |
|---|-----------------------------------------|----------|-----------|----------|-----------|--------|-----------|
|   |                                        | BLEU     | METEOR    | BLEU     | METEOR    | BLEU   | METEOR    |
| 1 | DCCN                                    | 39.7     | 56.8      | 31.0     | 49.9      | 26.7   | 45.7      |
| 2 | dynamic routing (global)                | 39.2     | 56.5      | 30.1     | 49.1      | 25.6   | 44.3      |
| 3 | dynamic routing (regional)              | 39.1     | 56.6      | 29.9     | 48.8      | 26.1   | 44.8      |
| 4 | dynamic routing (global) + attention (regional) | 39.4 | 56.7      | 30.2     | 49.4      | 26.1   | 45.0      |
| 5 | dynamic routing (regional) + attention (global) | 39.3 | 56.8      | 30.0     | 49.3      | 26.3   | 45.1      |
| 6 | attention (global) + attention (regional) | 38.9 | 56.3      | 29.8     | 48.8      | 25.9   | 44.7      |
| 7 | w/o context guidance in dynamic routing | 39.5     | 56.5      | 30.3     | 49.4      | 26.1   | 45.0      |

4.5 Ablation Study

To explore the effectiveness of different components in DCCN, we further compare our model with the following variants in Table 2:

(1) **dynamic routing (global)**. To build this variant, we only use global visual features in our model. The result in Row 2 indicates that removing the regional visual features lead to performance drop. This result suggests that regional visual features are indeed useful for multimodal representation learning.

(2) **dynamic routing (regional)**. Unlike the above variant, we only use regional visual features to represent the input image in this variant. According to the result shown in Row 3, we observe this change results in a significant performance decline, demonstrating that global visual features also bring useful visual information to our model.

(3) **dynamic routing ⇒ attention**. Apparently, one advantage of our model lies in leveraging context-guided dynamic routing to exploit the semantic interactions between different modalities for learning multimodal representation. Here we separately replace the context-guided dynamic routing with the conventional attention mechanism to exploit global visual features, regional visual features, and both of them. Then we investigate the change of model performance. To facilitate the following descriptions, we refer to these three variants as dynamic routing (global) + attention (regional), dynamic routing (regional) + attention (global), and attention (global) + attention (regional), respectively. From Row 4 to Row 6 of Table 2, we observe that dynamic routing (global) + attention (regional) slightly outperforms dynamic routing (global) while is inferior to DCCN. Similarly, the performance of dynamic routing (regional) + attention (global) is between dynamic routing (regional) and DCCN. Moreover, when we use attention mechanism rather than dynamic routing to extract two kinds of visual features, the performance of our model degrades most. Based on these experimental results, we can draw the conclusion that context-guided dynamic routing is able to better extract two types of visual features than conventional attention mechanism.

(4) **w/o context guidance in dynamic routing**. By removing the context guidance from DCCN, we adopt the standard capsule network to extract visual features. As shown in Row 7, the model performance drops drastically in this case. This result is consistent with our intuition that the ideal visual features required for translation should be dynamically captured at different timesteps.
4.6 Case Study

When encountering ambiguous source words or complicated sentences, it is difficult for MMT models to translate correctly without corresponding visual features. To further demonstrate the effectiveness of DCCN, we display the 1-best translations of the four cases generated by different models, as shown in Figure 5.

When encountering ambiguous nouns, the regional visual features are more helpful. For example, in case (a), both Transformer and Encoder-attention miss the translation of source word "cliff", while Doubly-attention and DCCN translate it correctly according to the detected object "rock". In case (b), we can find that the ambiguous source word "student" is translated to "étudiants (college students)" by all baselines, while only DCCN correctly translates it with "étudiants (generally refers to all students)" with the help of the detected object "kid".

When the model has difficulty in translating words out of the predicted objects such as verbs and adjectives, the global visual features are more helpful. In case (c), the source word "rides" is not associated with any object and thus all baselines choose "fuhrt (drive)". While only DCCN translates it correctly. In case (d), three baselines translate "harvest rice" to "circulant (flow)". "travaillent (work)" and "pagaient (paddle)", respectively. By contrast, only DCCN can produce the correct translation with the help of image information.

These cases reveal that DCCN can fully utilize complementary visual information to learn more accurate representations and disambiguate during translation in different cases.

4.7 Results on the EN⇒FR Translation Task

To investigate the generality of our proposed model, we also conduct experiments on the EN⇒FR translation task. Table 3 reports the final experimental results. Likewise, no matter which evaluation metric is used, our model still achieves better performance than all baselines. This result strongly demonstrates again that DCCN is effective and general to different language pairs in MMT.

5 CONCLUSION

In this paper, we have proposed a novel context-guided capsule network (DCCN) for MMT. As a significant extension of the conventional capsule network, DCCN utilizes the timestep-specific source-side context vector to dynamically guide the extraction of visual features at different timesteps, where the semantic interactions between modalities can be fully exploited for MMT via context-guided dynamic routing mechanism. Moreover, we employ DCCN to extract visual features in two complementary granularities: global visual features and regional visual features, respectively. Experimental results on English-to-German and English-to-French MMT tasks strongly demonstrate the effectiveness of our model. In the future, we plan to apply DCCN to other multimodal tasks such as visual question answering and multimodal text summarization.

6 ACKNOWLEDGMENTS

This work was supported by the Beijing Advanced Innovation Center for Language Resources (No. TYR17002), the National Natural Science Foundation of China (No. 61672440), and the Scientific
