Cross-modal Memory Networks for Radiology Report Generation

Zhihong Chen[♠♡], **Yaling Shen**[♠], **Yan Song**[♠♡†], **Xiang Wan**[♡]
[♠]The Chinese University of Hong Kong (Shenzhen)
[♡]Shenzhen Research Institute of Big Data
[♠]{zhihongchen,yalingshen}@link.cuhk.edu.cn
[♠]songyan@cuhk.edu.cn [♡]wanxiang@sribd.cn

Abstract

Medical imaging plays a significant role in clinical practice of medical diagnosis, where the text reports of the images are essential in understanding them and facilitating later treatments. By generating the reports automatically, it is beneficial to help lighten the burden of radiologists and significantly promote clinical automation, which already attracts much attention in applying artificial intelligence to medical domain. Previous studies mainly follow the encoder-decoder paradigm and focus on the aspect of text generation, with few studies considering the importance of cross-modal mappings and explicitly exploit such mappings to facilitate radiology report generation. In this paper, we propose a cross-modal memory networks (CMN) to enhance the encoder-decoder framework for radiology report generation, where a shared memory is designed to record the alignment between images and texts so as to facilitate the interaction and generation across modalities. Experimental results illustrate the effectiveness of our proposed model, where state-of-the-art performance is achieved on two widely used benchmark datasets, i.e., IU X-Ray and MIMIC-CXR. Further analyses also prove that our model is able to better align information from radiology images and texts so as to help generating more accurate reports in terms of clinical indicators.[1]

1 Introduction

Interpreting radiology images (e.g., chest X-ray) and writing diagnostic reports are essential operations in clinical practice and normally requires considerable manual workload. Therefore, radiology report generation, which aims to automatically generate a free-text description based on a radiograph, is highly desired to ease the burden of radiologists while maintaining the quality of health care. Recently, substantial progress has been made towards research on automated radiology report generation models (Jing et al., 2018; Li et al., 2018; Johnson et al., 2019; Liu et al., 2019; Jing et al., 2019). Most existing studies adopt a conventional encoder-decoder architecture, with convolutional neural networks (CNNs) as the encoder and recurrent (e.g., LSTM/GRU) or non-recurrent networks (e.g., Transformer) as the decoder following the image captioning paradigm (Vinyals et al., 2015; Anderson et al., 2018). Although these methods have achieved remarkable performance, they are still restrained in fully employing the information across radiology images and reports, such as the mappings demonstrated in Figure 1 that aligned visual and textual features point to the same content. The reason for the restraint comes from both the limitation of annotated correspondences between image and text for supervised learning as well as the lack of good model design to learn the correspondences. Unfortunately, few studies[2] are dedicated to solving the restraint. Therefore, it is expected to have a better solution to model the alignments across modalities and further improve the generation ability, although promising results are continuously acquired by other approaches (Li et al., 2018; Liu et al., 2019; Jing et al., 2019; Chen et al., 2020).

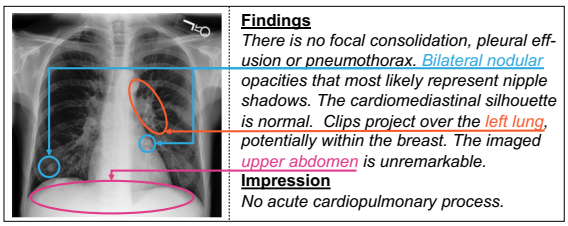

Figure 1: A chest X-ray image and its report including findings and impression, where aligned visual and textual features are marked in different colors.

[†]Corresponding author.

[1]Our code and the best performing models are released at https://github.com/cuhksz-nlp/R2GenCMN.

[2]Along this research track, recently there is only Jing et al. (2018) studying on a multi-task learning framework with a co-attention mechanism to explicitly explore information linking particular parts in a radiograph and its corresponding report.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 5904–5914
August 1–6, 2021.

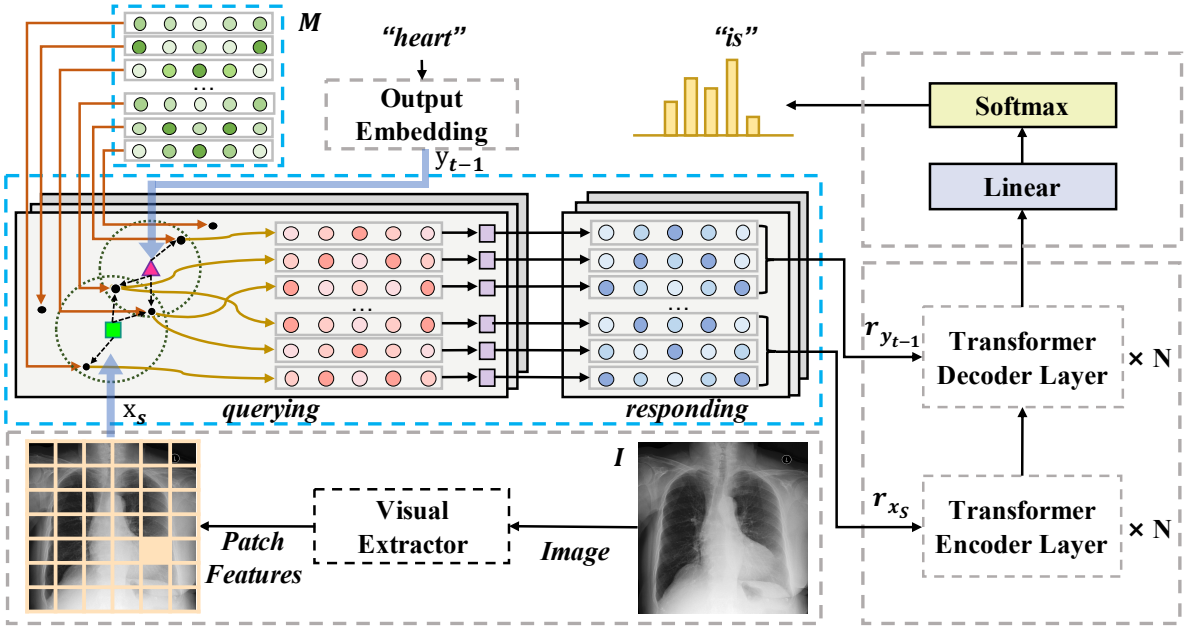

Figure 2: The overall architecture of our proposed approach, where the visual extractor, encoder and decoder are shown in gray dash boxes with the details omitted. The cross-modal memory networks are illustrated in blue dash boxes with presenting the detailed process of memory querying and responding.

In this paper, we propose an effective yet simple approach to radiology report generation enhanced by cross-modal memory networks (CMN), which is designed to facilitate the interactions across modalities (i.e., images and texts). In detail, we use a memory matrix to store the cross-modal information and use it to perform memory querying and memory responding for the visual and textual features, where for memory querying, we extract the most related memory vectors from the matrix and compute their weights according to the input visual and textual features, and then generate responses by weighting the queried memory vectors. Afterwards, the responses corresponding to the input visual and textual features are fed into the encoder and decoder, so as to generate reports enhanced by such explicitly learned cross-modal information. Experimental results on two benchmark datasets, IU X-RAY and MIMIC-CXR, confirm the validity and effectiveness of our proposed approach, where state-of-the-art performance is achieved on both datasets. Several analyses are also performed to analyze the effects of different factors affecting our model, showing that our model is able to generate reports with meaningful image-text mapping while requiring few extra parameters in doing so.

2 The Proposed Approach

We regard radiology report generation as an image-to-text generation task, for which there exist several solutions (Vinyals et al., 2015; Xu et al., 2015; Anderson et al., 2018; Cornia et al., 2019). Although images are organized as 2-D format, we follow the standard sequence-to-sequence paradigm for this task as that performed in Chen et al. (2020). In detail, the source sequence is $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_s, ..., \mathbf{x}_S\}$, where $\mathbf{x}_s \in \mathbb{R}^d$ are extracted by visual extractors from a radiology image $\mathbf{I}$ and the target sequence are the corresponding report $\mathbf{Y} = \{y_1, y_2, ..., y_t, ..., y_T\}$, where $y_t \in \mathbb{V}$ are the generated tokens, T the length of the report and $\mathbb{V}$ the vocabulary of all possible tokens. The entire generation process is thus formalized as a recursive application of the chain rule

$$p(\mathbf{Y}|\mathbf{I}) = \prod_{t=1}^{T} p(y_t|y_1, ..., y_{t-1}, \mathbf{I}) \quad (1)$$

The model is then trained to maximize $p(\mathbf{Y}|\mathbf{I})$ through the negative conditional log-likelihood of $\mathbf{Y}$ given the $\mathbf{I}$:

$$\theta^* = \arg\max_{\theta} \sum_{t=1}^{T} \log p(y_t|y_1, ..., y_{t-1}, \mathbf{I}; \theta) \quad (2)$$

where θ is the parameters of the model. An overview of the proposed model is demonstrated in Figure 2, with cross-modal memories emphasized. The details of our approach are described in following subsections regarding to its three major components, i.e., the visual extractor, the cross-modal memory networks and the encoder-decoder process enhanced by the memory.

2.1 Visual Extractor

To generate radiology reports, the first step is to extract the visual features from radiology images. In our approach, the visual features $\mathbf{X}$ of a radiology image $\mathbf{I}$ are extracted by pre-trained convolutional neural networks (CNN), such as VGG (Simonyan and Zisserman, 2015) or ResNet (He et al., 2016). Normally, an image is decomposed into regions of equal size[3], i.e., patches, and the features (representations) of them are extracted from the last convolutional layer of CNN. Once extracted, the features in our study are expanded into a sequence by concatenating them from each row of the patches on the image. The resulted representation sequence is used as the source input for all subsequent modules and the process is formulated as

$$\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_s, ..., \mathbf{x}_S\} = f_v(\mathbf{I}) \tag{3}$$

where $f_v(\cdot)$ refers to the visual extractor.

2.2 Cross-modal Memory Networks

To model the alignment between image and text, existing studies tend to map between images and texts directly from their encoded representations (e.g., Jing et al. (2018) used a co-attention to do so). However, this process always suffers from the limitation that the representations across modalities are hard to be aligned, so that an intermediate medium is expected to enhance and smooth such mapping. To address the limitation, we propose to use CMN to better model the image-text alignment, so as to facilitate the report generation process.

With using the proposed CMN, the mapping and encoding can be described in the following procedure. Given a source sequence $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_S\}$ (features extracted from the visual extractor) from an image, we feed it to this module to obtain the memory responses of the visual features $\{\mathbf{r}_{\mathbf{x}_1}, \mathbf{r}_{\mathbf{x}_2}, ..., \mathbf{r}_{\mathbf{x}_S}\}$. Similarly, given a generated sequence $\{y_1, y_2, ..., y_{t-1}\}$ with its embedding $\{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{t-1}\}$, it is also fed to the cross-modal memory networks to output the memory responses of the textual features $\{\mathbf{r}_{\mathbf{y}_1}, \mathbf{r}_{\mathbf{y}_2}, ..., \mathbf{r}_{\mathbf{y}_{t-1}}\}$. In doing so, the shared information of visual and textual features can be recorded in the memory so that the entire learning process is able to explicitly map between the images and texts. Specifically, the cross-modal memory networks employs a matrix to preserve information for encoding and decoding process, where each row of the matrix (i.e., a memory vector) records particular cross-modal information connecting images and texts. We denote the matrix as $\mathbf{M} = \{\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_i, ..., \mathbf{m}_\mathcal{N}\}$, where $\mathcal{N}$ represents the number of memory vectors and $\mathbf{m}_i \in \mathbb{R}^d$ the memory vector at row i with d referring to its dimension. During the process of report generation, CMN is operated with two main steps, namely, querying and responding, whose details are described as follows.[4]

Memory Querying We apply multi-thread[5] querying to perform this operation, where in each thread the querying process follows the same procedure described as follows.

In querying memory vectors, the first step is to ensure the input visual and textual features are in the same representation space. Therefore, we convert each memory vector in $\mathbf{M}$ as well as input features through linear transformation by

$$\mathbf{k}_i = \mathbf{m}_i \cdot \mathbf{W}_k \tag{4}$$

$$\mathbf{q}_s = \mathbf{x}_s \cdot \mathbf{W}_q \tag{5}$$

$$\mathbf{q}_t = \mathbf{y}_t \cdot \mathbf{W}_q \tag{6}$$

where $\mathbf{W}_k$ and $\mathbf{W}_q$ are trainable weights for the conversion. Then we separately extract the most related memory vector to visual and textual features according to their distances D_{s_i} and D_{t_i} through

$$D_{s_i} = \frac{\mathbf{q}_s \cdot \mathbf{k}_i^\top}{\sqrt{d}} \tag{7}$$

$$D_{t_i} = \frac{\mathbf{q}_t \cdot \mathbf{k}_i^\top}{\sqrt{d}} \tag{8}$$

where the number of extracted memory vectors can be controlled by a hyper-parameter $\mathcal{K}$ to regularize how much memory is used. We denote the queried memory vectors as $\{\mathbf{k}_{s_1}, \mathbf{k}_{s_2}, ..., \mathbf{k}_{s_j}, ..., \mathbf{k}_{s_\mathcal{K}}\}$ and $\{\mathbf{k}_{t_1}, \mathbf{k}_{t_2}, ..., \mathbf{k}_{t_j}, ..., \mathbf{k}_{t_\mathcal{K}}\}$. Afterwards, the importance weight of each memory vector with respect to visual and textual features are obtained by normalization over all distances by

$$w_{s_i} = \frac{\exp(D_{s_i})}{\Sigma_{j=1}^{\mathcal{K}} \exp(D_{s_j})} \tag{9}$$

$$w_{t_i} = \frac{\exp(D_{t_i})}{\Sigma_{j=1}^{\mathcal{K}} \exp(D_{t_j})} \tag{10}$$

Note that the above steps are applied in each thread to allow memory querying from different memory representation subspaces.

[3] E.g., VGG/ResNet uses region size 32×32 (in pixels).

[4] Note that these two steps are performed in both training and inference stages, where in inference, all textual features are obtained along with the generation process.

[5] Thread number can be arbitrarily set in experiments.

Memory Responding The responding process is also conducted in a multi-thread manner corresponding to the query process. For each thread, we firstly perform a linear transformation on the queried memory vector via

$$\mathbf{v}_i = \mathbf{m}_i \cdot \mathbf{W}_v \tag{11}$$

where $\mathbf{W}_v$ is the trainable weight for $\mathbf{m}_i$. So that all memory vectors $\{\mathbf{v}_{s_1}, \mathbf{v}_{s_2}, ..., \mathbf{v}_{s_j}, ..., \mathbf{v}_{s_\mathcal{K}}\}$ are transferred into $\{\mathbf{v}_{t_1}, \mathbf{v}_{t_2}, ..., \mathbf{v}_{t_j}, ..., \mathbf{v}_{t_\mathcal{K}}\}$. Then, we obtain the memory responses for visual and textual features by weighting over the transferred memory vectors by

$$\mathbf{r}_{\mathbf{x}_s} = \Sigma_{i=1}^{\mathcal{K}} w_{s_i} \mathbf{v}_{s_i} \tag{12}$$

$$\mathbf{r}_{\mathbf{y}_t} = \Sigma_{i=1}^{\mathcal{K}} w_{t_i} \mathbf{v}_{t_i} \tag{13}$$

where w_{s_i} and w_{t_i} are the weights obtained from memory querying. Similar to memory querying, we apply memory responding to all the threads so as to obtain responses from different memory representation subspaces.

2.3 Encoder-Decoder

Since the quality of input representation plays an important role in model performance (Pennington et al., 2014; Song et al., 2017, 2018; Peters et al., 2018; Song and Shi, 2018; Devlin et al., 2019; Song et al., 2021), the encoder-decoder in our model is built upon standard Transformer (which is a powerful architecture that achieved state-of-the-art in many tasks), where memory responses of visual and textual features are functionalized as the input of the encoder and decoder so as to enhance the generation process. In detail, as the first step, the memory responses $\{\mathbf{r}_{\mathbf{x}_1}, \mathbf{r}_{\mathbf{x}_2}, ..., \mathbf{r}_{\mathbf{x}_S}\}$ for visual features are fed into the encoder through

$$\{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_S\} = f_e(\mathbf{r}_{\mathbf{x}_1}, \mathbf{r}_{\mathbf{x}_2}, ..., \mathbf{r}_{\mathbf{x}_S}) \tag{14}$$

where $f_e(\cdot)$ represents the encoder. Then the resulted intermediate states $\{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_S\}$ are sent to the decoder at each decoding step, jointly with the memory responses $\{\mathbf{r}_{\mathbf{y}_1}, \mathbf{r}_{\mathbf{y}_2}, ..., \mathbf{r}_{\mathbf{y}_{t-1}}\}$ for the textual features of generated tokens from previous steps, so as to generate the current output y_t by

$$y_t = f_d(\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_S, \mathbf{r}_{\mathbf{y}_1}, \mathbf{r}_{\mathbf{y}_2}, ..., \mathbf{r}_{\mathbf{y}_{t-1}}) \tag{15}$$

where $f_d(\cdot)$ refers to the decoder. As a result, to generate a complete report, the above process is repeated until the generation is finished.

Dataset	IU X-Ray			MIMIC-CXR		
	Train	Val	Test	Train	Val	Test
Image #	5.2K	0.7K	1.5K	369.0K	3.0K	5.2K
Report #	2.8K	0.4K	0.8K	222.8K	1.8K	3.3K
Patient #	2.8K	0.4K	0.8K	64.6K	0.5K	0.3K
Avg. Len.	37.6	36.8	33.6	53.0	53.1	66.4

Table 1: The statistics of the two benchmark datasets w.r.t. their training, validation and test sets, including the numbers of images, reports and patients, and the averaged word-based length (Avg. Len.) of reports.

3 Experiment Settings

3.1 Datasets

We employ two conventional benchmark datasets in our experiments, i.e., IU X-Ray (Demner-Fushman et al., 2016)[6] from Indiana University and MIMIC-CXR (Johnson et al., 2019)[7] from the Beth Israel Deaconess Medical Center. The former is a relatively small dataset with 7,470 chest X-ray images and 3,955 corresponding reports; the latter is the largest public radiography dataset with 473,057 chest X-ray images and 206,563 reports.

Following the experiment settings from previous studies (Li et al., 2018; Jing et al., 2019; Chen et al., 2020), we only generate the findings section and exclude the samples without the findings section for both datasets. For IU X-Ray, we use the same split (i.e., 70%/10%/20% for train/validation/test set) as that stated in Li et al. (2018) and for MIMIC-CXR we adopt its official split. Table 1 show the statistics of all datasets in terms of the numbers of images, reports, patients and the average length of reports with respect to train/validation/test set.

3.2 Baseline and Evaluation Metrics

To examine our proposed model, we use the following ones as the main baselines in our experiments:

- **Base**: this is the backbone encoder-decoder used in our full model, i.e., a three-layer Transformer model with 8 heads and 512 hidden units without other extensions.
- **Base+Mem**: this is the Transformer model with the same architecture of **Base** where two memory networks are separately applied to image and text, respectively. This baseline aims to provide a reference to the cross-modal memory.

To further demonstrate the effectiveness of our model, we compare it with previous studies, includ-

[6] https://openi.nlm.nih.gov/

[7] https://physionet.org/content/mimic-cxr/2.0.0/

Data	Model	NLG Metrics							CE Metrics		
		BL-1	BL-2	BL-3	BL-4	MTR	RG-L	Avg. Δ	P	R	F1
IU X-Ray	Base	0.396	0.254	0.179	0.135	0.164	0.342	-	-	-	-
	+mem	0.443	0.270	0.191	0.144	0.172	0.351	6.6%	-	-	-
	+cmn	**0.475**	**0.309**	**0.222**	**0.170**	**0.191**	**0.375**	**19.6%**	-	-	-
MIMIC -CXR	Base	0.314	0.192	0.127	0.090	0.125	0.265	-	0.331	0.224	0.228
	+mem	0.340	0.209	0.140	0.100	0.135	0.273	8.2%	0.322	0.255	0.261
	+cmn	**0.353**	**0.218**	**0.148**	**0.106**	**0.142**	**0.278**	**13.1%**	**0.334**	**0.275**	**0.278**

Table 2: NLG and CE evaluations of different models on the test sets of IU X-Ray and MIMIC-CXR datasets. BL-n denotes BLEU score using up to 4-grams; MTR and RG-L denote METEOR and ROUGE-L, respectively. The average improvement over all NLG metrics compared to Base is also presented in the "Avg. Δ" column.

ing conventional image captioning models, e.g., **ST** (Vinyals et al., 2015), **Att2in** (Rennie et al., 2017), **AdaAtt** (Lu et al., 2017), **Topdown** (Anderson et al., 2018), and the ones proposed for the medical domain, e.g., **CoAtt** (Jing et al., 2018), **Hrgr** (Li et al., 2018), **Cmas-Rl** (Jing et al., 2019) and **R2Gen** (Chen et al., 2020).

Following Chen et al. (2020), we evaluate the above models by two types of metrics, conventional natural language generation (NLG) metrics and clinical efficacy (CE) metrics[8]. The NLG metrics[9] include BLEU (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2011) and ROUGE-L (Lin, 2004). For CE metrics, the CheXpert (Irvin et al., 2019)[10] is applied to label the generated reports and compare the results with ground truths in 14 different categories related to thoracic diseases and support devices. We use precision, recall and F1 to evaluate model performance for CE metrics.

3.3 Implementation Details

To ensure consistency with the experiment settings of previous work (Li et al., 2018; Chen et al., 2020), we use two images of a patient as input for report generation on IU X-Ray and one image for MIMIC-CXR. For visual extractor, we adopt the ResNet101 (He et al., 2016) pretrained on ImageNet (Deng et al., 2009) to extract patch features with 512 dimensions for each feature. For the encoder-decoder backbone, we use a Transformer structure with 3 layers and 8 attention heads, 512 dimensions for hidden states and initialize it randomly. For the memory matrix in CMN, its dimension and the number of memory vectors $\mathcal{N}$ are set to 512 and 2048, respectively, and also randomly initialized. For memory querying and responding, thread number and the $\mathcal{K}$ are set to 8 and 32, respectively. We train our model under cross entropy loss with Adam optimizer (Kingma and Ba, 2015). The learning rates of the visual extractor and other parameters are set to 5×10^{-5} and 1×10^{-4}, respectively, and we decay them by a 0.8 rate per epoch for all datasets. For the report generation process, we set the beam size to 3 to balance the effectiveness and efficiency of all models. Note that the optimal hyper-parameters mentioned above are obtained by evaluating the models on the validation sets from the two datasets.

[8]Note that CE metrics only apply to MIMIC-CXR because the labeling schema of CheXpert is designed for MIMIC-CXR, which is different from that of IU X-Ray.

[9]https://github.com/tylin/coco-caption

[10]https://github.com/MIT-LCP/mimic-cxr/tree/master/txt/chexpert

4 Results and Analyses

4.1 Effect of Cross-Modal Memory

The main experimental results on the two aforementioned datasets are shown in Table 2, where Base+cmn represents our model (same below). There are several observations drawn from different aspects. First, both Base+mem and Base+cmn outperform the vanilla Transformer (Base) on both datasets with respect to NLG metrics, which confirms the validity of incorporating memory to introduce more knowledge into the Transformer backbone. Such knowledge may come from the hidden structures and regularity patterns shared among radiology images and their reports, so that the memory modules are able to explicitly and reasonably model them to promote the recognition of diseases (symptoms) and the generation of reports. Second, the comparison between Base+cmn and two baselines on different metrics confirms the effectiveness of our proposed model with significant improvement. Particularly, Base+cmn outperforms Base+mem by a large margin, which indicates the

Data	Model	NLG Metrics						CE Metrics		
		BL-1	BL-2	BL-3	BL-4	MTR	RG-L	P	R	F1
IU X-Ray	ST‡	0.216	0.124	0.087	0.066	-	0.306	-	-	-
	Att2in‡	0.224	0.129	0.089	0.068	-	0.308	-	-	-
	AdaAtt‡	0.220	0.127	0.089	0.068	-	0.308	-	-	-
	CoAtt‡	0.455	0.288	0.205	0.154	-	0.369	-	-	-
	Hrgr‡	0.438	0.298	0.208	0.151	-	0.322	-	-	-
	Cmas-RL‡	0.464	0.301	0.210	0.154	-	0.362	-	-	-
	R2Gen‡	0.470	0.304	0.219	0.165	0.187	0.371	-	-	-
	Ours (cmn)	**0.475**	**0.309**	**0.222**	**0.170**	**0.191**	**0.375**	-	-	-
MIMIC -CXR	ST◇	0.299	0.184	0.121	0.084	0.124	0.263	0.249	0.203	0.204
	Att2in◇	0.325	0.203	0.136	0.096	0.134	0.276	0.322	0.239	0.249
	AdaAtt◇	0.299	0.185	0.124	0.088	0.118	0.266	0.268	0.186	0.181
	Topdown◇	0.317	0.195	0.130	0.092	0.128	0.267	0.320	0.231	0.238
	R2Gen‡	0.353	0.218	0.145	0.103	0.142	0.277	0.333	0.273	0.276
	Ours (cmn)	**0.353**	**0.218**	**0.148**	**0.106**	**0.142**	**0.278**	**0.334**	**0.275**	**0.278**

Table 3: Comparisons of our proposed model with previous studies on the test sets of IU X-Ray and MIMIC-CXR with respect to NLG and CE metrics. ‡ refers to that the result is directed cited from the original paper and ◇ represents our replicated results by their released codes.

usefulness of CMN in learning cross-modal features with a shared structure rather than separate ones. Third, when comparing between datasets, the performance gains from Base+cmn over two baselines (i.e., Base and Base+mem) on MIMIC-CXR are larger than that of IU X-Ray. This observation owes to the fact that MIMIC-CXR is relatively larger, which helps the learning of the alignment between images and texts so that CMN helps more on report generation on MIMIC-CXR. Third, when compared between datasets, the performace gain from Base+cmn over two baselines (i.e., Base and Base+mem) on IU X-Ray are larger than that of MIMIC-CXR. This observation owes to the fact that IU X-Ray is relatively small and has less complicated visual-textual mappings, thus easier for generation by CMN. Moreover, this size effect also helps that our model shows the same trend on the CE metrics on MIMIC-CXR as that for NLG metrics, where it outperforms all its baselines in terms of precision, recall and F1.

4.2 Comparison with Previous Studies

To further demonstrate the effectiveness, we further compare our model with existing models on the same datasets, with their results reported in Table 3 on both NLG and CE metrics. We have following observations. First, cross-modal memory shows its effectiveness in this task, where our model outperforms CoAtt, although both of them improve the report generation by the alignment of visual and textual features. The reason behind might be that our model is able to use a shared memory matrix as the medium to softly align the visual and textual features instead of direct alignment using the co-attention mechanism, thus unifies cross-modal features within same representation space and facilitate the alignment process. Second, our model confirms its superiority of simplicity when comparing with those complicated models. For example, Hrgr uses manually extracted templates and Cmas-RL utilizes reinforcement learning with a careful design of adaptive rewards and our model achieves better results with a rather simpler method. Third, applying memory to both the encoding and decoding can further improve the generation ability of Transformer when compared with R2Gen which only uses memory in decoding. This observation complies with our intuition that the cross-modal operation tightens the encoding and decoding so that our model generates higher quality reports. Fourth, note that although there are other models (i.e., CoAtt and Hrgr) with exploiting extra information (such as private datasets for visual extractor pre-training), our model still achieves the state-of-the-art performance without requiring such information. It reveals that in this task, the hidden structures among the images and texts and a

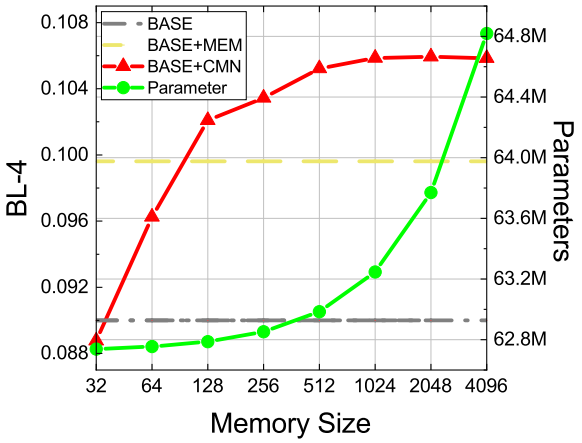

Figure 3: The BLEU-4 score and the number of parameters from BASE+CMN against the memory size (i.e., number of memory vectors) when the model is trained and tested on MIMIC-CXR dataset.

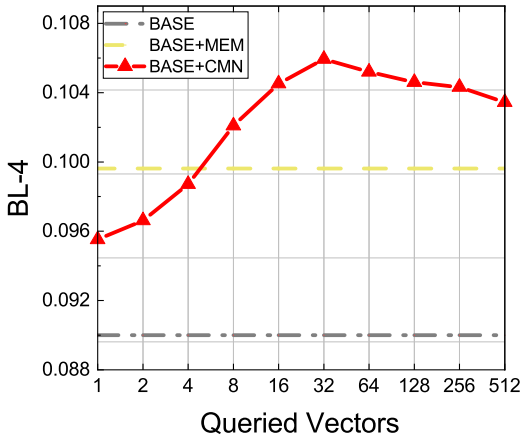

Figure 4: The BLEU-4 score from BASE+CMN when tested on the MIMIC-CXR test set against different numbers of queried memory vectors.

good solution of exploiting them are more essential in promoting the report generation performance.

4.3 Analysis

Memory Size To analyze the impacts of memory size, we train our model with different numbers of memory vectors, i.e., $\mathcal{N}$ ranges from 32 to 4096, with the results on MIMIC-CXR shown in Figure 3. It is observed that, first, enlarging memory by the number of vectors results in better overall performance when the entire memory matrix is relatively small ($\mathcal{N} \leq 1024$), which can be explained by that, within a certain memory capacity, larger memory size helps store more cross-modal information; second, when the memory matrix is larger than a threshold, increasing memory vectors is not able to continue promising a better outcome. An explanation to this observation may be that, when the matrix is getting to large, the memory vectors can not be fully updated so they do not help the generation process other than being played as noise. More interestingly, it is noted that even if we use a rather large memory size (i.e., $\mathcal{N} = 4096$), only 3.34% extra parameters are added to the model compared to BASE, which justifies that introducing memory to report generation process through our model can be done with small price.

Number of Queried Memory Vectors To analyze how querying impacts report generation, we try CMN with different numbers of queried vectors, i.e., $\mathcal{K}$ ranges from 1 to 512, and show the results in Figure 4. It is found that the number of queried vectors should be neither too small nor too big, where enlarging $\mathcal{K}$ leads to better results when $\mathcal{K} \leq 32$ and after this threshold the performance starts to drop. The reason behind might be the overfitting of memory updating since the memory matrix is sparsely updated in each iteration when $\mathcal{K}$ is small, i.e., it is hard to be overfit under this scenario, while more queried vectors should cause intensive updating on the matrix and some of the essential vectors are over-updated accordingly. As a result, it is interesting to find the optimal number (i.e., 32) of queried vectors and this is a useful guidance to further improve report generation with controlling the querying process.

Case Study To further qualitatively investigate how our model learns from the alignments between the visual and textual information, we perform a case study on the generated reports from different models regarding to an input chest X-ray image chosen from MIMIC-CXR. Figure 5 shows the image with ground-truth report, and different reports with selected mappings from visual (some part of the image) and textual features (some words and phrases),[11] where the mapped areas on the image are highlighted with different colors. In general, BASE+CMN is able to generate more accurate descriptions (in terms of better visual-textual mapping) in the report while other baselines are inferior in doing so. For instance, normal medical conditions and abnormalities presented in the chest X-ray image are covered by the generated report from BASE+CMN (e.g., "*severe cardiomegaly*", "*pulmonary edema*" and "*pulmonary arteries*") and the related regions on the image are precisely located regarding to the texts, while the areas highlighted on the image from other models are inaccurate.

[11] The representations of the textual features are extracted from the first layer of the decoder.

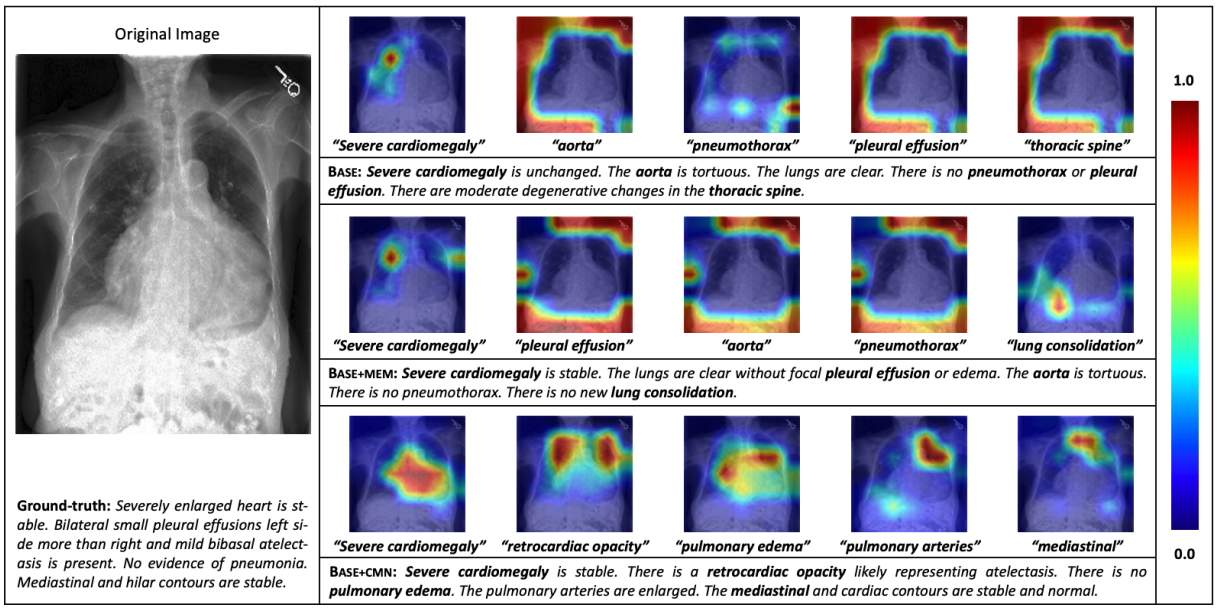

Figure 5: Visualizations of image-text mappings between particular regions (indicated by colored weights) of a chest X-ray image and words/phrases from its reports generated by BASE, BASE+MEM and BASE+CMN, respectively. The color spectrum indicates the value of weight from low to high in the range of [0, 1].

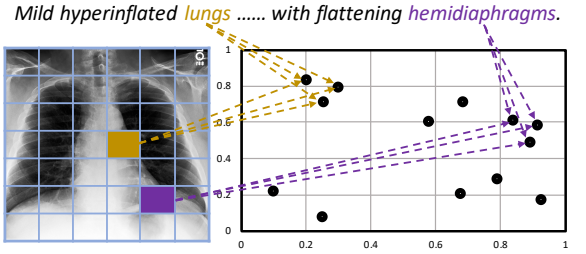

Figure 6: T-SNE visualization of memory vectors with an example input image and its partial generated report from MIMIC-CXR test set. The queried vectors for visual and textual features are indicated by arrows.

To further illustrate how the alignment works between visual and textual features, we perform a t-SNE visualization on the memory vectors linking to an image and its generated report from the MIMIC-CXR test set. It is observed that the word "*lung*" in the report and the visual feature for the region of lung on the image query similar memory vectors from CMN, where similar observation is also drawn for "hemidiaphragms" and its corresponding regions on the image. This case confirms that memory vector is effective intermediate medium to interact between image and text features.

5 Related Work

In general, the most popular related task to ours is image captioning, a cross-modal task involving natural language processing and computer vision, which aims to describe images in sentences (Vinyals et al., 2015; Xu et al., 2015; Anderson et al., 2018; Wang et al., 2019; Cornia et al., 2019). Among these studies, the most related study from Cornia et al. (2019) also proposed to leverage memory matrices to learn a priori knowledge for visual features using memory networks (Weston et al., 2015; Sukhbaatar et al., 2015; Zeng et al., 2018; Santoro et al., 2018; Nie et al., 2020; Diao et al., 2020; Tian et al., 2020b, 2021; Chen et al., 2021), but such operation is only performed during the encoding process. Different from this work, the memory in our model is designed to align the visual and textual features, and the memory operations (i.e., querying and responding) are performed in both the encoding and decoding process.

Recently, many advanced NLP techniques (e.g., pre-trained language models) have been applied to tasks in the medical domain (Pampari et al., 2018; Zhang et al., 2018; Wang et al., 2018; Alsentzer et al., 2019; Tian et al., 2019, 2020a; Wang et al., 2020; Lee et al., 2020; Song et al., 2020). Being one of the applications and extensions of image captioning to the medical domain, radiology report generation aims to depicting radiology images with professional reports. Existing methods were designed and proposed to better align images and texts or to exploit highly-patternized features of texts. For the former studies, Jing et al. (2018) proposed a co-attention mechanism to simultaneously explore visual and semantic information with a multi-task learning framework. For the latter studies, Li et al. (2018) introduced a template database to incorporate patternized information and Chen et al. (2020) improved the performance of radi-

ology report generation by applying a memory-driven Transformer to model patternized information. Compared to these studies, our model offers an effective yet simple alternative to generating radiology reports, where a soft intermediate layer is provided to facilitate the mappings between visual and textual features, so that more accurate descriptions are produced for generation.

6 Conclusion

In this paper, we propose to generate radiology reports with cross-modal memory networks, where a memory matrix is employed to record the alignment and interaction between images and texts, with memory querying and responding performed to obtain the shared information across modalities. Experimental results on two benchmark datasets demonstrate the effectiveness of our model, which achieves the state-of-the-art performance. Further analyses investigate the effects of hyper-parameters in our model and show that our model is able to better align information from images and texts, so as to generate more accurate reports, especially with the fact that enlarging the memory matrix does not significantly affect the entire model size.

Acknowledgments

This work is supported by Chinese Key-Area Research and Development Program of Guangdong Province (2020B0101350001) and NSFC under the project "The Essential Algorithms and Technologies for Standardized Analytics of Clinical Texts" (12026610).

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly Available Clinical BERT Embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086.

Guimin Chen, Yuanhe Tian, Yan Song, and Xiang Wan. 2021. Relation Extraction with Type-aware Map Memories of Word Dependencies. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*.

Zhihong Chen, Yan Song, Tsung-Hui Chang, and Xiang Wan. 2020. Generating Radiology Reports via Memory-driven Transformer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1439–1449.

Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. 2019. M^2: Meshed-Memory Transformer for Image Captioning. *arXiv preprint arXiv:1912.08226*.

Dina Demner-Fushman, Marc D Kohli, Marc B Rosenman, Sonya E Shooshan, Laritza Rodriguez, Sameer Antani, George R Thoma, and Clement J McDonald. 2016. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255.

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*, pages 85–91.

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

Shizhe Diao, Yan Song, and Tong Zhang. 2020. Keyphrase Generation with Cross-Document Attention. *arXiv preprint arXiv:2004.09800*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. 2019. CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels and Expert Comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 590–597.

Baoyu Jing, Zeya Wang, and Eric Xing. 2019. Show, Describe and Conclude: On Exploiting the Structure Information of Chest X-ray Reports. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6570–6580.

Baoyu Jing, Pengtao Xie, and Eric Xing. 2018. On the Automatic Generation of Medical Imaging Reports. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2577–2586.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. 2019. MIMIC-CXR: A large publicly available database of labeled chest radiographs. *arXiv preprint arXiv:1901.07042*.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Yuan Li, Xiaodan Liang, Zhiting Hu, and Eric P Xing. 2018. Hybrid Retrieval-Generation Reinforced Agent for Medical Image Report Generation. In *Advances in neural information processing systems*, pages 1530–1540.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81.

Guanxiong Liu, Tzu-Ming Harry Hsu, Matthew McDermott, Willie Boag, Wei-Hung Weng, Peter Szolovits, and Marzyeh Ghassemi. 2019. Clinically Accurate Chest X-Ray Report Generation. In *Machine Learning for Healthcare Conference*, pages 249–269.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2017. Knowing When to Look: Adaptive Attention via A Visual Sentinel for Image Captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 375–383.

Yuyang Nie, Yuanhe Tian, Yan Song, Xiang Ao, and Xiang Wan. 2020. Improving Named Entity Recognition with Attentive Ensemble of Syntactic Information. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 4231–4245.

Anusri Pampari, Preethi Raghavan, Jennifer Liang, and Jian Peng. 2018. emrQA: A Large Corpus for Question Answering on Electronic Medical Records. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2357–2368.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical Sequence Training for Image Captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024.

Adam Santoro, Ryan Faulkner, David Raposo, Jack Rae, Mike Chrzanowski, Theophane Weber, Daan Wierstra, Oriol Vinyals, Razvan Pascanu, and Timothy Lillicrap. 2018. Relational recurrent neural networks. In *Advances in neural information processing systems*, pages 7299–7310.

Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*, abs/1409.1556.

Yan Song, Chia-Jung Lee, and Fei Xia. 2017. Learning Word Representations with Regularization from Prior Knowledge. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 143–152.

Yan Song and Shuming Shi. 2018. Complementary Learning of Word Embeddings. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4368–4374.

Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018. Directional Skip-Gram: Explicitly Distinguishing Left and Right Context for Word Embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 175–180.

Yan Song, Yuanhe Tian, Nan Wang, and Fei Xia. 2020. Summarizing Medical Conversations via Identifying Important Utterances. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 717–729.

Yan Song, Tong Zhang, Yonggang Wang, and Kai-Fu Lee. 2021. ZEN 2.0: Continue Training and Adaption for N-gram Enhanced Text Encoders. *arXiv preprint arXiv:2105.01279*.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-To-End Memory Networks. In *Advances in neural information processing systems*, pages 2440–2448.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021. Enhancing Aspect-level Sentiment Analysis with Word Dependencies. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3726–3739, Online.

Yuanhe Tian, Weicheng Ma, Fei Xia, and Yan Song. 2019. ChiMed: A Chinese Medical Corpus for Question Answering. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 250–260.

Yuanhe Tian, Wang Shen, Yan Song, Fei Xia, Min He, and Kenli Li. 2020a. Improving Biomedical Named Entity Recognition with Syntactic Information. *BMC Bioinformatics*, 21:1471–2105.

Yuanhe Tian, Yan Song, Fei Xia, Tong Zhang, and Yonggang Wang. 2020b. Improving Chinese Word Segmentation with Wordhood Memory Networks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8274–8285.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and Tell: A Neural Image Caption Generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Nan Wang, Yan Song, and Fei Xia. 2018. Coding Structures and Actions with the COSTA Scheme in Medical Conversations. In *Proceedings of the BioNLP 2018 workshop*, pages 76–86.

Nan Wang, Yan Song, and Fei Xia. 2020. Studying Challenges in Medical Conversation with Structured Annotation. In *Proceedings of the First Workshop on Natural Language Processing for Medical Conversations*, pages 12–21, Online.

Weixuan Wang, Zhihong Chen, and Haifeng Hu. 2019. Hierarchical Attention Network for Image Captioning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8957–8964.

Jason Weston, Sumit Chopra, and Antoine Bordes. 2015. Memory Networks. *CoRR*, abs/1410.3916.

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 *International conference on machine learning*, pages 2048–2057.

Jichuan Zeng, Jing Li, Yan Song, Cuiyun Gao, Michael R Lyu, and Irwin King. 2018. Topic Memory Networks for Short Text Classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3120–3131.

Yuhao Zhang, Daisy Yi Ding, Tianpei Qian, Christopher D Manning, and Curtis P Langlotz. 2018. Learning to Summarize Radiology Findings. In *Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis*, pages 204–213.