QIAI at MEDIQA 2021: Multimodal Radiology Report Summarization

Jean-Benoit Delbrouck and **Han Zhang** and **Daniel L. Rubin**
Laboratory of Quantitative Imaging and Artificial Intelligence
Stanford University
{jbdel, hzhang20, dlrubin}@stanford.edu

Abstract

This paper describes the solution of the QIAI lab sent to the Radiology Report Summarization (RRS) challenge at MEDIQA 2021. This paper aims to investigate whether using multimodality during training improves the summarizing performances of the model at test-time. Our preliminary results shows that taking advantage of the visual features from the x-rays associated to the radiology reports leads to higher evaluation metrics compared to a text-only baseline system. These improvements are reported according to the automatic evaluation metrics METEOR, BLEU and ROUGE scores. Our experiments can be fully replicated at the following address : https://github.com/jbdel/vilmedic.

1 Introduction

Radiology report summarization is a growing area of research. Given the Findings and Background sections of a radiology report, the goal is to generate a summary (called an impression section in radiology reports) that highlights the key observations and conclusion of the radiology study. Automating this summarization task is critical because the impression section is the most important part of a radiology report, and manual summarization can be time-consuming and error-prone.

This paper describes the solution of the QIAI lab sent to the Radiology Report Summarization (RRS) challenge at MEDIQA 2021 (Ben Abacha et al., 2021). This challenge aims to promote the development of clinical summarization models that generate radiology impression statements by summarizing textual findings written by radiologists. Since for most reports, the associated x-rays are available, we aim to evaluate if incorporating visual features from x-rays helps our systems for the report summarization task. This task could be defined as Multimodal Radiology Report Summarization (MRRS) as depicted in Figure 1.

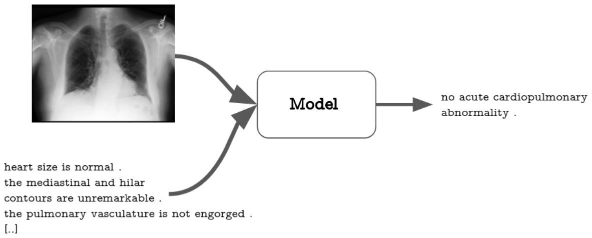
Model
no acute cardiopulmonary
abnormality .
heart size is normal .
the mediastinal and hilar
contours are unremarkable .
the pulmonary vasculature is not engorged .
[..]

Figure 1: An example of Multimodal Radiology Report Summarization.

2 Data Collection

The training set consists of 91,544 examples taken from the MIMIC-CXR v2.0 dataset (Johnson et al., 2019). Each training example is a free-text chest radiology report that contains the Background, Findings and Impression sections. Two validation sets, each with 2,000 reports were used. One validation set was collected from MIMIC, and the other was collected from the Indiana University Chest X-Rays Report dataset (Indiana-University). The test set contains 300 reports from the Indiana University dataset, and 300 reports from Stanford University School of Medicine. All report sections were tokenized using the Stanford CoreNLP tokenizer (Manning et al., 2014).

split	#report	#report w/o image
mimic-train	91,544	0
mimic-dev	2,000	0
indiana-dev	2,000	53
stanford-test	300	300
indiana-test	300	4

Table 1: Splits statistics from the MEDIQA 2021 challenge.

Proceedings of the BioNLP 2021 workshop, pages 285–290
June 11, 2021. ©2021 Association for Computational Linguistics

3 Model

This section describes the two architectures that will be bench-marked in the result section. We start by describing the text-based monomodal architecture at section 3.1. This model only takes as input the findings section and outputs the impression section (the summary). In section 3.2, we incorporate visual information into the monomodal architecture to make it multimodal.

3.1 Monomodal architecture

Given the report's Findings section of M words $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_M)$, an attention-based encoder-decoder model (Bahdanau et al., 2014) outputs its summary $\boldsymbol{Y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N)$. If we denote $\boldsymbol{\theta}$ as the model parameters, then $\boldsymbol{\theta}$ is learned by maximizing the likelihood of the observed sequence $\boldsymbol{Y}$ or in other words by minimizing the cross entropy loss. The objective function is given by:

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{t=1}^{n} \log p_{\boldsymbol{\theta}}(\boldsymbol{y}_t | \boldsymbol{y}_{<t}, X) \tag{1}$$

The encoder-decoder model consists of three components : an encoder, a decoder and an attention mechanism.

Encoder At every time-step t, an encoder creates an annotation $\boldsymbol{h}_t$ according to the current embedded word $\boldsymbol{x}'_t$ and internal state $\boldsymbol{h}_{t-1}$:

$$\boldsymbol{h}_t = f_{\text{enc}}(\boldsymbol{x}'_t, \boldsymbol{h}_{t-1}) \tag{2}$$

Every word $\boldsymbol{x}_t$ of the input sequence $\boldsymbol{X}$ is an index in the embedding matrix $\boldsymbol{E}^x$ so that the following formula maps the word to the f_{enc} size S:

$$\boldsymbol{x}'_t = \boldsymbol{W}^x \boldsymbol{E}^x x_t \tag{3}$$

The total size of the embeddings matrix $\boldsymbol{E}^x$ depends on the source vocabulary size $|\mathcal{Y}_s|$ and the embedding dimension d such that $\boldsymbol{E}^x \in \mathbb{R}^{|\mathcal{Y}_s| \times d}$. The mapping matrix $\boldsymbol{W}^x$ also depends on the embedding dimension because $\boldsymbol{W}^x \in \mathbb{R}^{d \times S}$.

The encoder function f_{enc} is a bi-directional GRU (Cho et al., 2014). The following equations define a single GRU block (called f_{gru} for future references) :

$$\begin{aligned}
\boldsymbol{z}_t &= \sigma\left(\boldsymbol{x}'_t + \boldsymbol{W}^z \boldsymbol{h}_{t-1}\right) \\
\boldsymbol{r}_t &= \sigma\left(\boldsymbol{x}'_t + \boldsymbol{W}^r \boldsymbol{h}_{t-1}\right) \\
\underline{\boldsymbol{h}}_t &= \tanh\left(\boldsymbol{x}'_t + \boldsymbol{r}_t \odot (\boldsymbol{W}^h \boldsymbol{h}_{t-1})\right) \\
\boldsymbol{h}_t &= (1 - \boldsymbol{z}_t) \odot \underline{\boldsymbol{h}}_t + \boldsymbol{z}_t \odot \boldsymbol{h}_{t-1}
\end{aligned} \tag{4}$$

where $\boldsymbol{h}_t \in \mathbb{R}^S$. Our encoder consists of two GRUs, one is reading the input sentence from 1 to M and the second from M to 1. Therefore the encoder annotation $\overline{\boldsymbol{h}_t}$ for timestep t is the concatenation of both GRUs annotations $\boldsymbol{h}_t$. The encoder set of annotations $\boldsymbol{H}$ contains the annotations $\overline{\boldsymbol{h}}$ of each timestep and is of size $M \times 2S$.

Decoder At every time-step t, a decoder outputs probabilities $\boldsymbol{p}_t$ over the target vocabulary $\mathcal{Y}_d$ according to previously generated word $\boldsymbol{y}_{t-1}$, internal state $\boldsymbol{s}_{t-1}$ and encoder annotations $\boldsymbol{H}$:

$$y_t \sim \boldsymbol{p}_t = f_{\text{dec}}(\boldsymbol{y}'_{t-1}, \boldsymbol{s}_{t-1}, \boldsymbol{H}) \tag{5}$$

Every word $\boldsymbol{y}_t$ of the summarized report $\boldsymbol{Y}$ is an index in the embedding matrix $\boldsymbol{E}^y$ so that the following formula maps the word in the f_{dec} size D:

$$\boldsymbol{y}'_t = \boldsymbol{W}^y \boldsymbol{E}^y \boldsymbol{y}_{t-1} \tag{6}$$

The decoder function f_{dec} consists of two parts: a conditional GRU (f_{cgru}) and a bottleneck function (f_{bot}).
The following equations describe the cGRU function f_{cgru}:

$$\begin{aligned}
\boldsymbol{s}'_t &= f_{\text{gru}_1}(\boldsymbol{y}'_t, \boldsymbol{s}_{t-1}) \\
\boldsymbol{c}_t &= f_{\text{att}}(\boldsymbol{s}'_t, \boldsymbol{H}) \\
\boldsymbol{s}_t &= f_{\text{gru}_2}(\boldsymbol{s}'_t, \boldsymbol{c}_t)
\end{aligned} \tag{7}$$

where f_{att} is the soft linguistic attention module over the set of source annotation $\boldsymbol{H}$:

$$\begin{aligned}
\boldsymbol{a}'_t &= \boldsymbol{W}^a \tanh(\boldsymbol{W}^{\mathrm{s}} \boldsymbol{s}'_t + \boldsymbol{W}^{\mathrm{H}} \boldsymbol{H}) \\
\boldsymbol{a}_t &= \text{softmax}(\boldsymbol{a}'_t) \\
\boldsymbol{c}'_t &= \sum_{i=0}^{M-1} \boldsymbol{a}_{t_i} \boldsymbol{h}_i \\
\boldsymbol{c}_t &= \boldsymbol{W}^c \boldsymbol{c}'_t
\end{aligned} \tag{8}$$

The bottleneck function f_{bot} projects the cGRU output $\boldsymbol{s}_t$ into probabilities over the target vocabulary. It is defined as such:

$$\begin{aligned} \boldsymbol{b}_t &= \tanh(\boldsymbol{W}^{\text{bot}}[\boldsymbol{s}_t, \boldsymbol{c}_t] \\ y_t \sim \boldsymbol{p}_t &= \text{softmax}(\boldsymbol{W}^{\text{proj}}\boldsymbol{b}_t) \end{aligned} \tag{9}$$

where $[\cdot, \cdot]$ denotes the concatenation operation.

3.2 Multimodal architecture

In the MIMIC dataset, a report can be associated with multiple x-rays images. We pick only one image according to the following priority: PA, AP, LATERAL, AP AXIAL, LL. Using this setting, we can select one image to each report. The Indiana dataset has at most one image associated with each report. In case no image is provided, we input a representation of "zeros" to the pipeline.

For each image, we extract the "pool0" representation of a DenseNet121 (Huang et al., 2017) architecture pretrained on x-rays images made available by the TorchXRayVision library (Cohen et al., 2020). The representation for each image is a vector of 1024 features that we call $\boldsymbol{v}$ in the following equations.

We consider three approaches to integrate the vector $\boldsymbol{v}$ to the monomodal architecture presented in Section 3.1. First, the **encdecinit** policy that consists of initializing both the encoder and decoder state $\boldsymbol{h}_0$ and $\boldsymbol{s}_0$ with the visual features as such:

$$\begin{aligned} \boldsymbol{h}_0 &= \tanh(\boldsymbol{W}^{vh0}\boldsymbol{v}) \\ \boldsymbol{s}_0 &= \tanh(\boldsymbol{W}^{vs0}\boldsymbol{v}) \end{aligned} \tag{10}$$

The second one is **ctxmul** that performs the element-wise product of each encoder annotations $\overline{\boldsymbol{h}_i}$ with $\boldsymbol{v}$:

$$\overline{\boldsymbol{h}_i} = \overline{\boldsymbol{h}_i} \odot \boldsymbol{W}^{vhi}\boldsymbol{v} \text{ for i} = 1 \text{ to } M \tag{11}$$

Finally, the **trgmul** policy consists of the element-wise product of each target embedding of equation 6 with $\boldsymbol{v}$:

$$\boldsymbol{y}_t' = \boldsymbol{y}_t' \odot \boldsymbol{W}^{vy}\boldsymbol{v} \tag{12}$$

Matrices $\boldsymbol{W}^{vh0}$, $\boldsymbol{W}^{vs0}$, $\boldsymbol{W}^{vhi}$, $\boldsymbol{W}^{vy}$ are trainable weights that transform and map $\boldsymbol{v}$ to right dimension.

Finally, we define a fourth approach, **allv**, using all the aforementioned interactions.

4 Settings

Both monomodal and multimodal architectures use a 2-layered bi-directional GRU for the encoder, and 1-layered GRU for the decoder. Each GRU has a hidden size of 320 units and our embeddings are of size 200. We apply dropout of 0.4 on the source embeddings $\boldsymbol{x}_t'$, 0.5 on the source annotations $\boldsymbol{H}$ and 0.5 on the bottleneck $\boldsymbol{b}_t$.

We chose Adam (Kingma and Ba, 2014) as the optimizer with a learning rate of 0.0004 and batch size 64. Model parameters are initialized using the He initialization method (He et al., 2015). We evaluate the model performance using the ROUGE-2 F1 metrics (Lin, 2004), which is commonly used for evaluating machine summarization task. We stop training when the ROUGE score does not improve for 10 evaluations on the validation set. In the experiment section, we also report the METEOR (Banerjee and Lavie, 2005) and BLEU metrics (Papineni et al., 2002).

In the scope of this paper, we only use the findings section as input to our models and discard the background section.

5 Experiments

The experiments are carried out as follows:

1. We define two settings for the dataset splits. The first one is dictated by the challenge as defined in Table 1. We call it **regular-split**. The second setting consists of injecting 1500 out of the 2000 indiana-dev samples into the training set. We keep the remaining 500 for development. This setting allows more training homogeneity compared to regular-split, we refer to it as the **mix-split**;

2. We use our monomodal architecture to predict summarization for both the stanford and indiana test sets. We use the multimodal architecture to predict summarization only on the indiana test set (the stanford test set having no x-rays available). Note that both architectures are trained with the same number of samples.

Figure 2 and 3 depict the results of the best scoring configurations for the monomodal and multimodal models on the development sets. Each results is obtained by using beam-search with width varying from 8 to 12. Finally, E5 means results are from an ensemble of 6 trained models (i.e. model ensembling).

Model	BLEU	METEOR	R2-F1
		indiana-dev	
Mono	13.94	16.47	31.33
Multi allv	13.27	15.19	26.84
Mono E5	**15.88**	**17.67**	**31.37**
Multi E5 allv	15.27	17.12	30.42
		mimic-dev	
Mono	28.67	25.74	47.96
Multi allv	28.90	26.01	48.19
Mono E5	28.66	25.74	48.41
Multi E5 allv	**29.31**	**26.24**	**48.86**

Table 2: Results of our best multimodal and monomodal architectures on the development sets (regular-split).

Model	BLEU	METEOR	R2-F1
		indiana-dev	
Mono	26.93	24.50	52.18
Multi allv	27.21	24.60	51.79
Mono E5	26.61	24.35	52.02
Multi E5 allv	**28.32**	**25.30**	**54.38**
		mimic-dev	
Mono	29.00	25.90	48.10
Multi allv	28.30	25.48	48.47
Mono E5	**28.97**	25.95	48.38
Multi E5 allv	**28.97**	**26.10**	**48.98**

Table 3: Results of our best multimodal and monomodal architectures on the development sets (mix-split).

A few observations can be made. First, three of four best scoring models (highlighted in bold) is the multimodal variant. Each time, the multimodal model is using the *allv* interaction. It means that injecting the visual features from the x-rays in both the encoder and the decoder improves summarization.

Secondly, the only instance where the monomodal variant is better is on the indiana-dev set using the regular-split. One could hypothesize that the multimodal model is sensitive to distribution shift; indeed no indiana samples (and therefore indiana x-rays) are in the training set for this configuration. Though using model ensembling seems to mitigates the performance drop, it is still lower that the monomodal baseline.

Finally, we underline the ROUGE scores from systems that are significantly different (p-value ≤ 0.05) than the baseline *mono* models using the approximate randomization test of multeval (Clark et al., 2011). The underlined scores are all from multimodal systems.

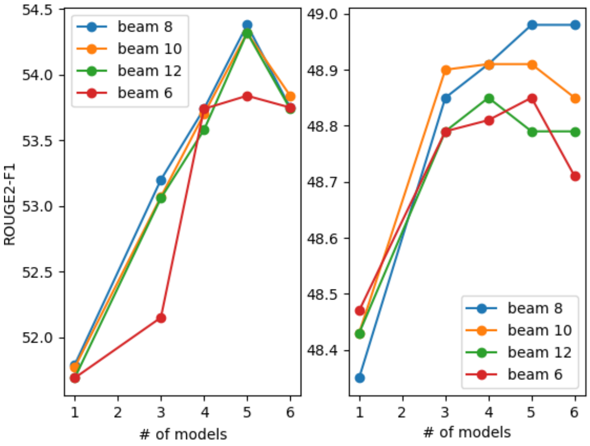

Figure 2: Effect of ensembling and beam-size width for model *Multi E5 allv* (mix-split setting). Left concerns split indiana-dev and right mimic-dev.

6 Related Work

Though relatively new, a few previous work can be denoted in the field of radiology report summarization. Zhang et al. (2018) first studied the problem of automatic generation of radiology impressions by summarizing textual radiology findings, and showed that an augmented pointer-generator model achieves high overlap with human references. This model has been extended with an ontologyaware pointer-generator and showed improved summarization quality (MacAvaney et al., 2019). RL-based approaches have been investigated by Li et al. (2018) and (Liu et al., 2019).

More recently, (Zhang et al., 2020) developed a general framework where the evaluation of the factual correctness of a generated summary is done by factchecking it automatically against its reference using an information extraction module.

To our knowledge, this work is the first attempt to use multimodality for radiology report summarization.

References

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

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Asma Ben Abacha, Yassine Mrabet, Yuhao Zhang, Chaitanya Shivade, Curtis Langlotz, and Dina Demner-Fushman. 2021. Overview of the mediqa 2021 shared task on summarization in the medical domain. In *Proceedings of the 20th SIGBioMed Workshop on Biomedical Language Processing, NAACL-BioNLP 2021*. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merriënboer, Çağlar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 176–181, Portland, Oregon, USA. Association for Computational Linguistics.

Joseph Paul Cohen, Joseph Viviano, Paul Morrison, Rupert Brooks, Mohammad Hashir, and Hadrien Bertrand. 2020. TorchXRayVision: A library of chest X-ray datasets and models. *https://github.com/mlmed/torchxrayvision*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Indiana-University. Indiana university - chest x-rays.

Alistair E. W. Johnson, Tom J. Pollard, Nathaniel R. Greenbaum, Matthew P. Lungren, Chih-ying Deng, Yifan Peng, Zhiyong Lu, Roger G. Mark, Seth J. Berkowitz, and Steven Horng. 2019. MIMIC-CXR-JPG, a large publicly available database of labeled chest radiographs. *arXiv e-prints*, page arXiv:1901.07042.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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*, volume 31. Curran Associates, Inc.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

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 *Proceedings of the 4th Machine Learning for Healthcare Conference*, volume 106 of *Proceedings of Machine Learning Research*, pages 249–269, Ann Arbor, Michigan. PMLR.

Sean MacAvaney, Sajad Sotudeh, Arman Cohan, Nazli Goharian, Ish Talati, and Ross W. Filice. 2019. Ontology-aware clinical abstractive summarization. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 1013–1016, New York, NY, USA. Association for Computing Machinery.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Prismatic Inc, Steven J. Bethard, and David Mcclosky. 2014. The stanford corenlp natural language processing toolkit. In *In ACL, System Demonstrations*.

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 of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Yuhao Zhang, Daisy Yi Ding, Tianpei Qian, Christopher D. Manning, and Curtis P. Langlotz. 2018. Learning to summarize radiology findings. In *EMNLP 2018 Workshop on Health Text Mining and Information Analysis*.

Yuhao Zhang, Derek Merck, Emily Tsai, Christopher D. Manning, and Curtis Langlotz. 2020. Optimizing the factual correctness of a summary: A study of summarizing radiology reports. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5108–5120, Online. Association for Computational Linguistics.

A Multimodal results

Model	BLEU	METEOR	R2-F1
		indiana-dev	
Multi E5 allv	15.27	17.12	**30.42**
Multi E5 encdecinit	**15.37**	**17.32**	30.05
Multi E5 ctxmul	14.57	16.75	29.85
Multi E5 trgmul	15.26	16.75	29.75
		mimic-dev	
Multi E5 allv	**29.31**	**26.24**	**48.86**
Multi E5 trgmul	29.0	26.10	42.56
Multi E5 ctxmul	28.90	26.02	48.47
Multi E5 encdecinit	28.38	25.58	48.42

Table 4: Results of our multimodal architectures on the development sets (regular-split).

Model	BLEU	METEOR	R2-F1
		indiana-dev	
Multi E5 allv	**28.32**	**25.30**	**54.38**
Multi E5 trgmul	27.50	24.72	53.90
Multi E5 ctxmul	26.86	24.53	53.20
Multi E5 encdecinit	26.65	24.52	52.10
		mimic-dev	
Multi E5 allv	**28.97**	**26.10**	**48.98**
Multi E5 trgmul	28.83	25.90	**48.98**
Multi E5 ctxmul	28.83	25.87	48.81
Multi E5 encdecinit	28.31	25.65	48.39

Table 5: Results of our multimodal architectures on the development sets (mix-split).