Journalistic Guidelines Aware News Image Captioning

Xuewen Yang[1], Svebor Karaman[2], Joel Tetreault[2], and Alex Jaimes[2]

[1]Stony Brook University
[2]Dataminr Inc.
[1]xuewen.yang@stonybrook.edu
[2]{skaraman, jtetreault, ajaimes}@dataminr.com

Abstract

The task of news article image captioning aims to generate descriptive and informative captions for news article *images*. Unlike conventional image captions that simply describe the content of the image in general terms, news image captions follow journalistic guidelines and rely heavily on named entities to describe the image content, often drawing context from the whole article they are associated with. In this work, we propose a new approach to this task, motivated by caption guidelines that journalists follow. Our approach, Journalistic Guidelines Aware News Image Captioning (JoGANIC), leverages the structure of captions to improve the generation quality and guide our representation design. Experimental results, including detailed ablation studies, on two large-scale publicly available datasets show that JoGANIC substantially outperforms state-of-the-art methods both on caption generation and named entity related metrics.

1 Introduction

Research on generating textual descriptions of images has made great progress in recent years with the introduction of encoder-decoder architectures (Xu et al., 2015; Johnson et al., 2016; Venugopalan et al., 2017; Karpathy and Fei-Fei, 2017; Anderson et al., 2018; Lu et al., 2018b; Aneja et al., 2018). Those models are generally trained and evaluated on image captioning datasets like COCO (Lin et al., 2014; Chen et al., 2015) and Flickr (Hodosh et al., 2013) that only contain generic object categories but no details such as names, locations, or dates. The captions generated by these methods are thus generic descriptions of the images.

The news image captioning problem (Feng and Lapata, 2013; Ramisa et al., 2018; Biten et al., 2019; Tran et al., 2020) can be seen as a multimodal extension of the image captioning task with additional context provided in the form of a news article. Specifically, given image-article pairs as input, the news captioning task aims to generate an informative caption that describes the image with proper named entities and context extracted from the article. The development of automatic news image caption generation methods can ease the process of adding images to articles and produce more engaging content. According to *The News Manual*[1] and *International Journalists' Network*[2], a caption should help news readers understand six main components (*who*, *when*, *where*, *what*, *why*, *how*) related to the image and article. As shown in Fig. 1, different journalists can write captions to cover different components for the same image and article pair. Previous news image captioning work (Biten et al., 2019; Tran et al., 2020) has not directly addressed the challenge of generating a caption that follows those journalistic principles.

Figure 1: Three possible captions (bottom) for one image-article pair input (top). These three captions follow different 'templates' composed of *who* (in green), *when* (in red), *where* (in blue), *context* (in purple) and *misc* (in orange) components.

In this work, we tackle the news image captioning problem by introducing these guidelines in our modeling through a new concept called a 'caption template', which is composed of 5 key components, detailed in Section 3. We propose a Journalistic Guidelines Aware News Image Captioning (JoGANIC) model that, given an image-article pair,

[1]https://www.thenewsmanual.net/Manuals%20Volume%202/volume2_47.htm
[2]https://ijnet.org/en/resource/writing-photo-captions

aims to predict the most likely active template components and, using component-specific decoding block, produces a caption following the provided template guidance. JoGANIC thus models the underlying structure of the captions, which helps to improve the generation quality.

Captions for images that accompany news articles often include named entities *and* rely heavily on context found throughout the article (making the text encoding process especially challenging). We propose two techniques to address these issues: (i) integration of features specifically to extract relevant named entities, and (ii) a multi-span text reading (MSTR) method, which first splits long articles into multiple text spans and then merges the extracted features of all spans together.

Our work has two main contributions: (i) the definition of the template components of a news caption based on journalistic guidelines, and their explicit integration in the caption generation process of our JoGANIC model; (ii) the design of encoding mechanisms to extract relevant information for the news image captioning task throughout the article, specifically a dedicated named entity representation and the ability to process longer article. Experimental results show better performance than state of the art on news image caption generation. We release the source code of our method at https://github.com/dataminr-ai/JoGANIC.

2 Related Work

2.1 Generic Image Captioning

State-of-the-art approaches (Johnson et al., 2016; Wang et al., 2020; He et al., 2020; Sammani and Melas-Kyriazi, 2020) mainly use encoder-decoder frameworks with attention to generate captions for images. Xu et al. (2015) developed soft and hard attention mechanisms to focus on different regions in the image when generating different words. Similarly, Anderson et al. (2018) used a Faster R-CNN (Ren et al., 2015) to extract regions of interest that can be attended to. Yang et al. (2020) used self-critical sequence training for image captioning. Lu et al. (2018a) and Whitehead et al. (2018) introduced a knowledge aware captioning method where the knowledge comes from metadata associated with the datasets.

Our work differs from generic image captioning in three aspects: (i) our model's input consists of image-article pairs; (ii) our caption generation is a guided process following news image captioning journalistic guidelines; (iii) news captions contain named entities and additional context extracted from the article, making them more complex.

Type	Description	Component
PERSON	People, including fictional	who
NORP	Political groups	who
ORG	Companies, agencies, etc	who
DATE	Dates or periods	when
TIME	Times smaller than a day	when
FAC	Buildings, airports, highways	where
GPE	Countries, cities, states	where
LOC	Locations, mountains, waters	where
PRODUCT	Objects, vehicles, foods	misc
EVENT	Named wars, sports events	misc
ART	Titles of books, songs	misc
LAW	Laws	misc
LAN	Any named language	misc
PERCENT	Percentage, including "%"	misc
MONEY	Monetary values	misc
QUANTITY	Measurements	misc
ORDINAL	"first", "second", etc	misc
CARDINAL	Numerals	misc

Table 1: Named Entities type, description and assigned component category.

2.2 News Article Image Captioning

One of the earliest works in news article image captioning, Ramisa et al. (2018), proposed an encoder-decoder architecture with a deep convolutional model VGG (Simonyan and Zisserman, 2015) and Word2Vec (Mikolov et al., 2013) as the image and text feature encoder, and an LSTM as the decoder.

Biten et al. (2019) introduced the GoodNews dataset, and proposed a two-step caption generation process using ResNet-152 (He et al., 2016) as the image representation and a sentence-level aggregated representation using GloVe embeddings (Pennington et al., 2014). First, a caption is generated with placeholders for the different types of named entities: *PERSON*, *ORGANIZATION*, *etc.* shown in the left column of Table 1. Then, the placeholders are filled in by matching entities from the best ranked sentences of the article. This two-step process aims to deal with rare named entities but prevents the captions from being linguistically rich and is can induce error propagation between steps.

More recently, Liu et al. (2020); Hu et al. (2020); Tran et al. (2020) proposed one step, end-to-end methods. They all used ResNet-152 as image encoder, while for the text encoder: Hu et al. (2020) applied BiLSTM, Liu et al. (2020) used BERT and Tran et al. (2020) used RoBERTa. Hu et al. (2020); Liu et al. (2020) used LSTM as the decoder. Tran et al. (2020) introduced the NYTimes800k dataset, and a model named Transform and Tell, which we

	who	when	where	misc	context
GoodNews	93.02	44.06	58.59	31.69	78.44
NYTimes800k	93.77	41.54	51.08	30.92	77.07

Table 2: Template components percentage in the GoodNews and NYTimes800k datasets.

refer to as Tell. This model exploits a Transformer decoder and byte-pair-encoding (BPE) (Sennrich et al., 2016) allowing to generate captions with unseen or rare named entities from common tokens.

As in other multimodal tasks, where studies (Shekhar et al., 2019; Caglayan et al., 2019; Li et al., 2020) have shown that the exploitation of both modalities is essential for achieving a good performance, Tran et al. (2020) evaluated a text only model showing that it performs worse than the multimodal model. We will also evaluate single visual and text modality models in our experiments.

Our work differs from previous work in news image captioning in that JoGANIC is an end-to-end framework that (i) integrates journalistic guidelines through a template guided caption generation process; and (ii) exploits a dedicated named entity representation and a long text encoding mechanism. Our experiments show that our framework significantly outpeforms the state of the art.

3 Defining Caption Templates

The objective of news image captioning is to give the reader a clear understanding of the main components *who*, *when*, *where*, *what*, *why* and *how* depicted in the image given the context of the article. We propose to exploit the idea of components in the caption generation process, but we first need to define components that can be automatically detected in the ground truth caption for training.

The *who*, *when* and *where* components can be retrieved via Named Entity Recognition (NER). As shown in the right column of Tab. 1, we define named entities with type 'PERSON', 'NORP' and 'ORG' as *who*, those with type 'DATE' and 'TIME' as *when*, and ones with type 'FAC', 'GPE' and 'LOC' as *where*. We define the component *misc* as the rest of the named entities. The *what*, *why* and *how* components are hard to define and can correspond to a wide range of elements, we propose to merge them into a *context* component, which we assume is present if a verb is detected by a part-of-speech (POS) tagger[3]. In Fig. 1, captions 1 and 2 have an *context* component, but caption 3 does not contain a verb and thus has no *context*.

In summary, our proposed news caption template consists of *at most* five components: *who*, *when*, *where*, *context* and *misc*. We report the percentage of each component in the captions of the GoodNews and NYTimes800k datasets in Tab. 2. The *who* is present in almost all the captions, and all components appear commonly in both datasets.

4 Template-Guided News Image Captioning

In this section, we formally define the news captioning task and introduce the idea of template guidance and our Journalistic Guidelines Aware News Image Captioning (JoGANIC) approach. We then propose two strategies to address the specific challenges of named entities and long articles.

4.1 News Captioning Problem Formulation

Given an image and article pair (X^I, X^A), the objective of news captioning is to generate a sentence $\mathbf{y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ with a sequence of N tokens, $\mathbf{y}_i \in V^K$ being the i-th token, V^K being the vocabulary of K tokens. The problem can be solved by an encoder-decoder model. The decoder predicts the target sequence $\mathbf{y}$ conditioned on the source inputs X^I and X^A. The decoding probability $P(\mathbf{y}|X^I, X^A)$ is modeled using the probability of each target token $\mathbf{y}_n$ at time step n conditioned on the source input X^I and X^A and the current partial target sequence $\mathbf{y}_{<n}$:

$$P(\mathbf{y}|X^I, X^A; \boldsymbol{\theta}) = \prod_{n=1}^{N} P(\mathbf{y}_n|X^I, X^A, \mathbf{y}_{<n}; \boldsymbol{\theta}) \quad (1)$$

where, $\boldsymbol{\theta}$ denotes the parameters of the model.

4.2 Template Guidance

To make our model capable of generating captions following different templates, we introduce a new variable $\boldsymbol{\alpha}$ for template guidance. The new decoding probability can be defined as:

$$P(\mathbf{y}|X^I, X^A) = \prod_{n=1}^{N} P(\mathbf{y}_n|X^I, X^A, \boldsymbol{\alpha}, \mathbf{y}_{<n}) \quad (2)$$

where we ignore $\boldsymbol{\theta}$ for simplicity.

Based on our definition of templates, we could see $\boldsymbol{\alpha}$ as the high-level template class defined by the combination of the active components. As there are 5 template components, the total number of possible template classes is 2^5. However, this poses two challenges to train our model: (i) data imbalance, as the most frequent template corresponds to

[3]We use spaCy which has almost SOTA pos tagging accuracy of 97.8% and NER accuracy of 89.8% on the OntoNotes corpus.

15.2% of captions, while the least common ones appear less than 2% of the time (more details in Tab. 3 of the supplementary material), and (ii) different high-level templates may be similar (i.e. having a single component difference) but would be considered totally different classes.

In order to address these issues we define $\boldsymbol{\alpha}$ as the set of active components of the template $\boldsymbol{\alpha}_{i=1}^{5}$, with $\boldsymbol{\alpha}_i$ being the probability of a template having component i. This formulation enables us to exploit the partial overlap in terms of components between the different templates. Note that the percentage of each component, in Tab. 2, is not as imbalanced as the full template classes. The template guidance $\boldsymbol{\alpha}$ can be provided by the news writer ('oracle' setting in the experiments) or can be estimated ('auto' setting) through a multi-label classification task as detailed in the next section and illustrated in the top-left of Fig. 2(a).

The template guidance variable α is static during the decoding process but that does not prevent our method from generating fluent captions covering the whole set of components. Exploring ways of exploiting dynamically the component specific representations during the caption generation process could be an interesting future work direction.

4.3 Our Model Description

We propose a news image captioning model that generates captions through template guidance and can also generate accurate named entities and cover a larger extent of the article. Our JoGANIC model, illustrated in Fig. 2, is a transformer-based encoder-decoder, with an encoder extracting features from the image X^I and the article X^A, a prediction head estimating the probability of each component and a hybrid decoder to produce the caption.

The encoder consists of three parts: (i) a ResNet-152 pretrained on ImageNet extracting the image feature $\mathbf{X}^I \in \mathbb{R}^{d_I}$; (ii) RoBERTa producing the text features $\mathbf{X}^T \in \mathbb{R}^{d_T}$ from the article; and (iii) a Named Entity Embedder (NEE), detailed in Section 4.3.1, applied to obtain the features $\mathbf{X}^E \in \mathbb{R}^{d_E}$ of the named entities in the article. The components prediction head, taking as input the concatenation of the image, article and named entities features, is a multi-layer perceptron with a sigmoid layer trained (using the components detected in the ground truth caption as target) to output the probability of each component $P(\boldsymbol{\alpha}|X^I, X^A)$.

The hybrid decoder consists of an embedding layer to get the embeddings of the output generated thus far (i.e., the partial generation), followed by 4 blocks of 3 Multi-Head Attention (MHA) modules, denoted as MHA (image/text/NE), to compute the attention across the partial generation and the input image, text and named entities. The final representation $\mathbf{u}_i$ for each block is the concatenation of the 3 modules' output, Fig. 2(b). The first 3 blocks are shared for all components, while the 4-th block consists of 5 parallel component-specific blocks $4_1 - 4_5$ where block $4i$ outputs the representation $\mathbf{u}_4^i$ for the component i. The final representation of the decoder is the average of the weighted sum of all components $\overline{\mathbf{u}} = \frac{1}{5}\sum_{i=1}^{5} \boldsymbol{\alpha}_i \mathbf{u}_4^i$. Then the output probability $P(\mathbf{y}_n|X^I, X^A, \boldsymbol{\alpha}, \mathbf{y}_{<n})$ is obtained by applying a feed-forward (FF) layer, and softmax over the target vocabulary. Note that our "template guided" generation does not limit the number of occurrences of one component in the output caption and does not explicitly constrain the generation of specific components but rather the final representation $\overline{\mathbf{u}}$ will rely more on the component-specific representations corresponding to higher $\boldsymbol{\alpha}_i$ values.

4.3.1 Named Entity Embedding

With over 96% (see Tab. 1 in the supplementary material) of the news captions containing named entities, producing accurate named entities is essential to generating good news captions. However, text encoders like RoBERTa cannot properly represent named entities, and only handle them implicitly through BPE (Byte-Pair Encoding) subwords.

To deal explicitly with named entities, we learn entity embeddings from the Wikipedia knowledge base (KB), following Wikipedia2vec (Yamada et al., 2018) which embeds words and entities into a common space[4]. Given a vocabulary of words V_W and a set of entities V_E, it learns a lookup embedding function $\mathcal{E}_{Wiki} : V_W \cup V_E \rightarrow \mathbb{R}^{d_{Wiki}}$. There are three components in Wikipedia2Vec: (i) a skip-gram model for learning the word similarity in V_W, (ii) a KB graph model to learn the relatedness between pairs of entities (vertices V_E of the Wikipedia entity graph) and (iii) a version of Word2Vec where words are predicted from entities.

Since predicting the correct named entities from context is very important for news captioning, we introduce a fourth component: (iv) a neural entity predictor (NEP). Given a text (sequence of words) $t = \{w_1, \ldots, w_N\}$, we train Wikipedia2vec to pre-

[4] https://wikipedia2vec.github.io/wikipedia2vec/

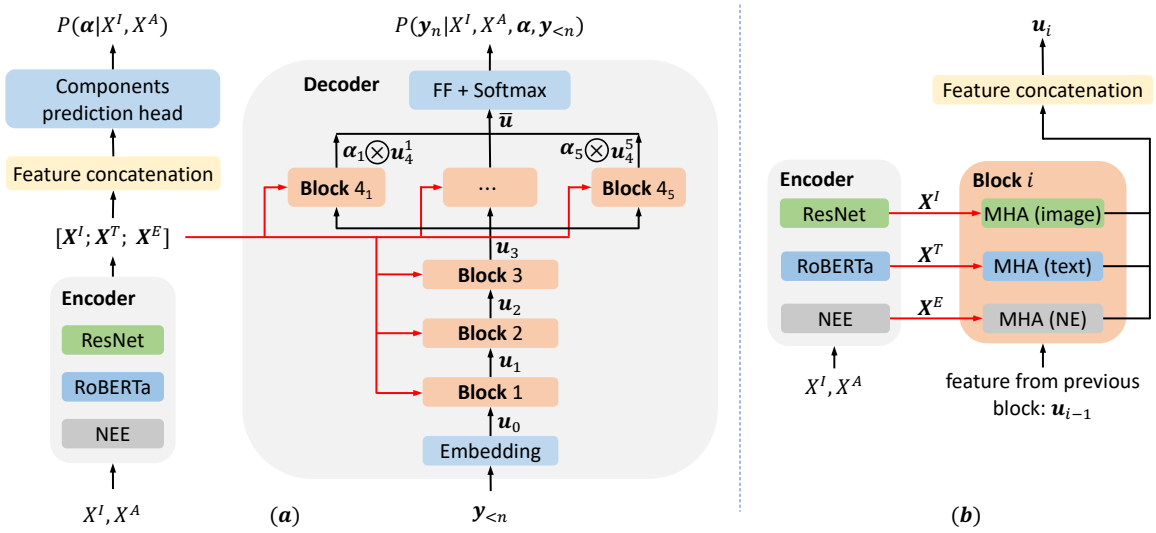

Figure 2: The architecture of our model. (*a*) The Encoder takes image+text+named entities as input and generates features. The Decoder consists blocks 1-4, with blocks 1-3 shared for all template components *who*, *when*, *where*, *context* and *misc*. Block 4 consists of 5 component-specific subblocks (4_1-4_5). A prediction head on top of the encoder predicts the probabilities of the 5 components $\boldsymbol{\alpha}_{1:5}$, which then multiply the representations of the 5 subblocks $\mathbf{u}_4^{1:5}$. The final representation $\bar{\mathbf{u}}$ is obtained by averaging and used to predict the output token probabilities. (*b*) Every block takes as input the representations from previous blocks as well as those from the Encoder via three Multi-Head Attention (MHA) modules designed for image, text and named entities separately.

dict the entities $e_1, \ldots, e_m$ that appear in the sequence. With E_{KB} being the set of all entities in KB, and v_e and v_t (computed as the element-wise mean of all the word vectors in t followed by a fully connected layer) the vector representations of the entity e and the text t, respectively, the probability of an entity e appearing in text t is defined as

$$P(e|t) = \frac{\exp(v_e^T v_t)}{\sum_{e' \in E_{KB}} \exp(v_{e'}^T v_t)}. \quad (3)$$

We optimize the NEP model with a cross-entropy loss, but using Eq. 3 as is would be computationally expensive as it involves a summation over all entities in the KB. We address this by replacing E_{KB} in Eq. 3 with E^*, the union of the positive entity e and 50 randomly chosen negative entities not in t. Through exploiting the Named Entity Embedding (NEE), our model can represent and thus generate more accurate entities. The NEE model is not jointly trained with the template components prediction and caption generation heads of JoGANIC, but pre-trained offline on Wikipedia KB.

The Wikipedia KB contains a large set of NEs but cannot cover all NEs that could appear in a news article (about 40% are not covered in our datasets). The embedding of a new NE cannot be obtained directly by lookup. To alleviate this problem, we set the embedding of any missing NE with v_t which is reasonable as we trained the NEP to maximize the correlation between v_e and v_t in Eq. 3.

4.3.2 Reading Longer Articles

Biten et al. (2019) use sentence-level features obtained by averaging the word features, of a pre-trained GloVe (Pennington et al., 2014) model, in the sentence. While this method can embed the whole article, the averaging makes the feature less informative. Tran et al. (2020) instead use RoBERTa as the text feature extractor, though this has the limitation of exploiting only 512 tokens.

However, processing only the first 512 tokens may ignore important contextual information appearing later in the news article. To alleviate this problem, we propose a *Multi-Span Text Reading* (MSTR) method to read more than 512 tokens from the article. MSTR splits the text into overlapping segments of 512 tokens and pass them to the RoBERTa encoder independently. The representation of any overlapping token in 2 segments is the element-wise interpolation of their representations.

5 Experiments

We evaluate JoGANIC on two large-scale publicly available news captioning datasets: GoodNews (Biten et al., 2019) and NYTimes800k (Tran et al., 2020) both collected using The New York Times public API[5], with the latter being larger and con-

[5]https://developer.nytimes.com/apis

		General Caption Ceneration				Named Entities		Components	
		BLEU-4	ROUGE	METEOR	CIDEr	P	R	$\overline{P}$	$\overline{R}$
GoodNews	SAT (Xu et al., 2015)	0.73	11.88	4.14	12.15	8.19	7.10	–	–
	Att2in2 (Rennie et al., 2017)	0.76	11.58	3.90	11.58	–	–	–	–
	BUTD (Anderson et al., 2018)	0.71	11.06	3.74	11.02	–	–	–	–
	Adaptive Att (Lu et al., 2017)	0.51	10.94	3.59	10.55	–	–	–	–
	Avg+CtxIns (Biten et al., 2019)	0.89	12.20	4.37	13.10	8.23	6.06	20.51	18.72
	TBB+AttIns (Biten et al., 2019)	0.76	12.20	4.17	12.70	8.87	5.64	20.23	18.45
	VGG+LSTM (Ramisa et al., 2018)	0.31	6.38	1.66	1.28	–	–	–	–
	VisualNews (Liu et al., 2020)	5.1	19.3	8.8	43.7	19.6	17.9	–	–
	Tell (Tran et al., 2020)	5.45	20.70	9.74	48.50	21.10	17.40	69.52	63.31
	Tell (full) (Tran et al., 2020)	6.05	21.40	10.30	53.80	22.20	18.70	71.55	64.93
	JoGANIC (zero-out text)	1.71	13.04	5.23	9.61	4.42	3.01	18.92	16.77
	JoGANIC (zero-out image)	4.10	17.33	8.41	38.49	18.03	15.12	48.74	46.29
	JoGANIC (image only)	1.86	13.28	5.97	10.20	4.46	3.31	19.07	17.13
	JoGANIC (text only)	5.28	19.07	9.17	50.04	20.43	18.13	49.56	46.98
	JoGANIC (auto)	6.34	21.65	10.78	59.19	24.60	20.90	75.51	66.27
	JoGANIC+NEE (auto)	6.73	22.68	11.18	59.50	25.87	21.63	74.42	68.53
	JoGANIC+MSTR (auto)	6.45	21.99	10.83	59.65	24.75	21.61	75.57	**70.04**
	JoGANIC+MSTR+NEE (auto)	**6.83**	**23.05**	**11.25**	**61.22**	**26.87**	**22.05**	**75.83**	68.85
	JoGANIC (oracle)	*7.06*	*24.13*	*11.72*	*69.23*	*28.40*	*23.48*	*92.96*	*87.86*
	JoGANIC+MSTR+NEE (oracle)	*7.36*	*24.25*	*11.98*	*69.76*	*28.59*	*23.68*	*92.46*	*87.55*
NYTimes800k	Tell (Tran et al., 2020)	5.01	19.40	9.05	40.30	20.0	18.10	67.13	62.24
	Tell (full) (Tran et al., 2020)	6.30	21.70	10.30	54.40	24.60	22.20	69.72	63.52
	JoGANIC (zero-out text)	1.42	12.66	5.08	9.33	4.23	2.89	18.87	16.53
	JoGANIC (zero-out image)	3.88	15.64	7.76	32.01	21.15	14.84	53.71	51.29
	JoGANIC (image only)	1.50	12.58	5.68	9.93	4.49	2.88	19.40	17.12
	JoGANIC (text only)	4.95	18.47	8.54	41.27	20.52	18.48	54.89	52.31
	JoGANIC (auto)	6.39	22.38	10.75	56.54	27.35	23.73	73.37	65.79
	JoGANIC+NEE (auto)	6.66	22.72	10.85	59.02	26.81	23.20	73.02	**66.54**
	JoGANIC+MSTR (auto)	6.44	22.63	10.88	57.61	26.41	23.67	73.36	66.30
	JoGANIC+MSTR+NEE (auto)	**6.79**	**22.80**	**10.93**	**59.42**	**28.63**	**24.49**	**73.51**	65.49
	JoGANIC (oracle)	*7.44*	*24.09*	*11.93*	*65.53*	*28.53*	*26.09*	*90.76*	*87.99*
	JoGANIC+MSTR+NEE (oracle)	*7.68*	*24.09*	*12.09*	*66.15*	*28.79*	*26.35*	*90.07*	*87.92*

Table 3: Results on GoodNews and NYTimes800k. We highlight the **best** model in bold. Note that we directly use the results reported in (Tran et al., 2020) for the baseline models.

taining longer articles. We follow the evaluation protocols defined by the authors of each dataset and used by previous works with 421K training, 18K validation, and 23K test captions for GoodNews and 763K training, 8K validation and 22K test captions for NYTimes800k. We provide further details about the datasets in the supplementary material.

5.1 Methods & Metrics

We implement JoGANIC as a Transfomer-based encoder-decoder architecture similar to Tell but with our proposed template guidance. We introduce JoGANIC+NEE as JoGANIC with enriched named entity embeddings (Section 4.3.1), and JoGANIC+MSTR as JoGANIC with multi-span text reading technique (Section 4.3.2). To evaluate how JoGANIC exploits template guidance, we introduce the JoGANIC (oracle) and JoGANIC+MSTR+NEE (oracle) variants, where ground truth template components are provided through $\boldsymbol{\alpha}$. We evaluate if our model exploits both the text and image input in two ways. We first report results of our multimodal model where at test time we zero-out text features (i.e. $\mathbf{X}^T$ and $\mathbf{X}^E$ are set to all zero vectors) JoGANIC (zero-out text) or image features JoGANIC (zero-out image). We also train single-modality models with only an image encoder (JoGANIC image only) or a text encoder (JoGANIC text only).

We compare against two types of baselines. (i) Two-step generation methods: that are based on conventional image captioning models (Xu et al., 2015; Rennie et al., 2017; Anderson et al., 2018; Lu et al., 2017; Biten et al., 2019) to first generate captions with placeholders and then insert named entities into these placeholders. (ii) End-to-end models: VGG+LSTM (Ramisa et al., 2018), VisualNews (Liu et al., 2020) that uses ResNet as image encoder, BERT article encoder and bi-LSTM as decoder, and Tell, with two variants: (a) Tell, which

uses RoBERTa and ResNet-152 as the encoders and Transformer as the decoder, it is equivalent to JoGANIC without template guidance as they use the same encoders and training settings. (b) Tell (full), which includes two additional visual encoders: YOLOv3 and MTCNN, and Location-Aware and Weighted RoBERTa for text encoding.

For the general caption generation quality evaluation, we use the BLEU-4 (Papineni et al., 2002), ROUGE (Lin, 2004), METEOR (Denkowski and Lavie, 2014) and CIDEr (Vedantam et al., 2015) metrics. We also use named entity precision/recall to evaluate the named entity generation quality. To better understand how well the generated captions follow the ground truth templates, we calculate precision and recall for the five components *who*, *when*, *where*, *context* and *misc* and use the averaged precision and recall[6] as the final metric.

5.2 Implementation and Training details

Following Tran et al. (2020), we set the hidden size of the input features $d_I = 2048$, $d_T = 1024$ and $d_E = 300$ and the number of heads $H = 16$. We use the Adam optimizer (Kingma and Ba, 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-6}$. The number of tokens in the vocabulary $K = 50264$ and $d^{Wiki} = 300$. We limit the text length in MSTR to 1,000 tokens as preliminary studies have shown similar performance with longer text input but at the expense of significant increased training time (Tab. 6 in supplementary). In practice, for an article longer than 512 tokens, we read two overlapping text segments of 512 tokens, one starting from the beginning and another from the end and thus can have $[24 - 511]$ overlapping tokens. The components prediction head in Fig. 2 is a linear layer followed by an output layer of 1024 dimensions.

The training pipeline uses PyTorch (Paszke et al., 2017) and the AllenNLP framework (Gardner et al., 2018). The RoBERTa model and dynamic convolution code are adapted from fairseq (Ott et al., 2019). We use a maximum batch size of 16 and training is stopped after the model has seen 6.6 million examples, corresponding to 16 epochs on GoodNews and 9 epochs on NYTimes800k. Training is done with mixed precision to reduce the memory footprint and allow our full model to be trained on a single V-100 GPU for 4 to 6 days on both datasets.

5.3 Evaluation

5.3.1 General Caption Generation

We first discuss the results with the general caption generation metrics BLEU-4, ROUGE, METEOR and CIDEr reported in Table 8. We report the mean values of three runs, and the maximum standard deviations of our variants on BLEU, ROUGE, METEOR, CIDEr are 0.013, 0.019, 0.016 and 0.069, which shows the stability of our results and that our method improvements are notable. For the GoodNews dataset, JoGANIC (auto) provides an improvement of 0.89, 0.95, 1.04, 10.69 points over Tell on the four metrics respectively, while the full model JoGANIC+MSTR+NEE (auto) has an even bigger improvement of 1.38, 2.35, 1.51, 12.72. The improvement is especially impressive for the CIDEr score. JoGANIC performs much better than all the two-step captioning methods (first group of results) and VGG+LSTM. For the NYTimes800k dataset, we compare our models only to Tell since other models perform much worse. Here, our full model achieves 6.79, 22.80, 10.93 and 59.42 with 1.78, 3.40, 1.88, 19.12 points improvement over Tell. Our JoGANIC+MSTR+NEE (auto) outperforms Tell (full) which exploits additional visual features on both datasets . This demonstrates the effectiveness of our model in generating good captions. By providing the oracle $\boldsymbol{\alpha}$, the JoGANIC+MSTR+NEE (oracle) can achieve even higher performance on almost all metrics, showing the value of our template guidance process.

From the single modality evaluation, we observe that models that exploit the text only (JoGANIC (zero-out image) and JoGANIC (text only)) perform better than those relying on the image only (JoGANIC (zero-out text) and JoGANIC (image only)) but all have lower performance than multimodal models, confirming that both modalities are important for news image captioning.

5.3.2 Named Entity Generation

One of the main objectives of news captioning is to generate captions with accurate named entities. As shown in Tab. 8, compared to Tell, JoGANIC+MSTR+NEE (auto) increases the named entity precision and recall scores by 5.77% and 4.65% on GoodNews, and 8.63% and 6.39% on NYTimes800k. The oracle versions of our models attain even higher performances.

[6]Per-component results are provided in Table 4 of the supplementary material.

In Mr. Pedersen's more than 60 years of architectural practice, the project that means the most is the modest one he just completed here, where he and his wife, Elizabeth Pedersen, have summered since 1975. "I probably had the most pleasurable professional experience of my life," Mr. Pedersen, said during a recent walk through the Shelter Island History Center. The center is the new name for the reconfigured complex run by the Shelter Island Historical Society. Until recently, those treasures were housed in the attic of the historical society's decaying Havens House Museum, built in 1743. Over the last three years, Havens House has been renovated and expanded with a two-level addition designed by Mr. Pedersen to create more storage and a proper art gallery. The effort was initiated by Ms. Pedersen. Four years ago, she received a diagnosis of pancreatic cancer, just as the project was gaining steam. Her husband has finished it for her.

(1) The exterior of the Shelter Island History Center. The building was the former Havens House Museum, which has been renovated and expanded with a two-level addition.	(2) A building with a clock on the side of it.	(3) The Havens House Museum, built in 1743, has been renovated and expanded to the site of the Havens House Museum.	**(4)** The Havens House Museum, built in 1743, has been renovated and expanded with a two-level addition designed by Mr. Pedersen.
(5) The Shelter Island History Center, was the Havens House Museum, designed by Mr. Pedersen and his wife, Ms. Pedersen.	**(6)** The Havens House Museum, built in 1743, has been renovated and expanded, effort initiated by Ms. Pedersen.	**(7)** The Havens House Museum, built in 1743, has a new name the Shelter Island History Center, has been renovated and expanded by Mr. Pedersen.	(8) The Havens House Museum, has a new name the Shelter Island History Center, has been renovated and expanded with a two-level addition designed by Mr. Pedersen.
(9) The Havens House Museum, has been renovated and expanded with a two-level addition designed by William Pedersen to create more storage.	(10) The Havens House Museum, built in 1743.	(11) The Havens House Museum, built in 1743, was renovated and expanded to create more storage and a proper art gallery.	(12) The Havens House Museum.

Figure 3: An example of news caption generation. The captions are generated by: (1) human (ground truth caption). (2) conventional image captioning model SAT. (3) Tell. (4) JoGANIC. (5) JoGANIC+NEE. (6) JoGANIC+MSTR. JoGANIC+MSTR+NEE (7) auto, (8) oracle, with template (9) *who* + *context*, (10) *who* + *when*, (11) *who* + *when* + *context*, and (12) *who*. For the generated captions, we highlight wrong statements in red.

5.3.3 Template Components Evaluation

The average precision and recall of the template components, reported in the two rightmost columns of Tab. 8, of JoGANIC+MSTR+NEE (auto) increases by 6.3% and 5.5% on GoodNews dataset and 6.4% and 3.3% on NYTimes800k dataset compared to Tell. By providing the oracle $\boldsymbol{\alpha}$, even better results are obtained, demonstrating that our model can exploit template guidance.

5.3.4 Qualitative & Human Evaluation

In Figure 3 we show the image, article (shortened for visualization) and the captions generated by a conventional image captioning model SAT (Xu et al., 2015), Tell (Tran et al., 2020) and different JoGANIC variants. The captions generated by all JoGANIC variants are meaningful and closer to the ground truth than the baselines. Interestingly, most captions generated by JoGANIC variants include people's names, e.g. Mr. or Ms. *Pedersen* in addition to the building names probably because people's names are the most common type for the component *who* in the datasets (see Tab. 1 of the supplementary material). As MSTR can read longer text than Tell, JoGANIC+MSTR can exploit the end of the article and generates the text span *effort initiated by Ms. Pedersen*. The caption generated by JoGANIC+MSTR+NEE has all the key factors in the ground truth caption (*the Havens House Museum*, *the Shelter Island History Center*, *been renovated and expanded*) demonstrating the strengths of our model. The captions generated using the oracle $\boldsymbol{\alpha}$ (8) as well as some other manually defined $\boldsymbol{\alpha}$ (9-12) illustrate the benefits and flexibility of our "template components" modeling, showing how the caption generation process can be controlled by the template guidance in JoGANIC.

Finally, we conducted a human evaluation through crowd-sourcing on Amazon Mechanical Turk on 200 random image-article pairs sampled from the test set of the NYT800K dataset. For each image-article pair, three different raters were requested to rate the ground truth caption, the caption generated by Tell, and captions generated by 4 variants of our model, on a 4 point scale. Raters were asked to evaluate separately how well the caption was describing the *image*, how relevant it was to the *article*, and how easy to understand the *sentence* was. We report the average of the three ratings in Tab. 4, showing that all variants of our model produce captions that are rated better than Tell and closer to the ground truth captions ratings on the three aspects. The groundtruth captions have the highest sentence quality score but can have lower score for image and article relatedness as journalists sometimes do not follow guidelines and can write a caption describing the image independently of the article context or on the contrary being more related to the article than the image content. Details on the annotation instructions and results are given in the supplementary material.

6 Conclusion

News image captioning is a challenging task as it requires exploiting both image and text content to produce rich and well structured captions including

Model	image	article	sentence
Ground Truth	2.96	2.86	3.08
Tell	2.80	2.80	2.92
JoGANIC	2.87	2.86	2.97
JoGANIC+NEE	2.88	**2.92**	**2.99**
JoGANIC+MSTR	**2.89**	2.86	2.98
JoGANIC+MSTR+NEE	2.86	2.88	**2.99**

Table 4: **Human evaluation on the generated captions.** We highlight the **best** model in bold.

relevant named entities and information gathered from the whole article. In this work, we presented Journalistic Guidelines Aware News Image Captioning, aiming to solve the news image captioning task by integrating domain specific knowledge in both the representation and caption generation process. On the representation side, we introduced two techniques: named entity embedding (NEE) and multi-span text reading (MSTR). Our decoding process explicitly integrates the key components a journalist would seek to describe to improve the caption generation quality. Our method obtains remarkable gains on both GoodNews and NYTimes800k datasets relative to the state-of-the-art.

7 Ethical Considerations and Broader Impact

Our model is a multi-modality extension of the general image captioning methods. It can further be applied to other applications, including but not limited to, multi-modality machine translation, summarization, etc. By modeling the template components of the captions, our research could be used to explore the underlying structure of each task, improving understanding of the generation decisions or providing explanations. The potential risks of news article image captioning is the generation bias, i.e., the model might tend to use the named entities that have high frequencies. We thus suggest that people use our model as a recommendation for generating captions, people could thus modify the generated captions and control for potential bias. We would also encourage further work to understand the biases and limitations of the datasets used in this paper, including tools to analyze gender bias and other limitations.

Acknowledgments

We thank Mahdi Abavisani, Shengli Hu, and Di Lu for the fruitful discussions during the development of the method, and all the reviewers for their detailed questions, clarification requests, and suggestions on the paper.

References

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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

J. Aneja, A. Deshpande, and A. G. Schwing. 2018. Convolutional image captioning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition*.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.

Ali Furkan Biten, Lluis Gomez, Marcal Rusinol, and Dimosthenis Karatzas. 2019. Good news, everyone! context driven entity-aware captioning for news images. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ozan Caglayan, Pranava Madhyastha, Lucia Specia, and Loïc Barrault. 2019. Probing the need for visual context in multimodal machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *ArXiv*, abs/1504.00325.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*.

Y. Feng and M. Lapata. 2013. Automatic caption generation for news images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778.

Sen He, Wentong Liao, Hamed Rezazadegan Tavakoli, Michael Ying Yang, Bodo Rosenhahn, and Nicolas Pugeault. 2020. Image captioning through image transformer. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Micah Hodosh, Peter Young, and Julia Hockenmaier. 2013. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*.

Anwen Hu, Shizhe Chen, and Qin Jin. 2020. Icecap: Information concentrated entity-aware image captioning. In *Proceedings of the 28th ACM International Conference on Multimedia*.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Andrej Karpathy and Li Fei-Fei. 2017. Deep visual-semantic alignments for generating image descriptions. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 664–676.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization.

Haoran Li, Junnan Zhu, Jiajun Zhang, Xiaodong He, and Chengqing Zong. 2020. Multimodal sentence summarization via multimodal selective encoding. In *Proceedings of the 28th International Conference on Computational Linguistics*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV 2014*.

Fuxiao Liu, Yinghan Wang, Tianlu Wang, and Vicente Ordonez. 2020. Visualnews : Benchmark and challenges in entity-aware image captioning. *ArXiv*.

Di Lu, Spencer Whitehead, Lifu Huang, Heng Ji, and Shih Fu Chang. 2018a. Entity-aware image caption generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2017. Knowing when to look: Adaptive attention via a visual sentinel for image captioning.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2018b. Neural baby talk. *2018 IEEE Conference on Computer Vision and Pattern Recognition*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: 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*.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

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)*.

A. Ramisa, F. Yan, F. Moreno-Noguer, and K. Mikolajczyk. 2018. Breakingnews: Article annotation by image and text processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1072–1085.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28*.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Fawaz Sammani and Luke Melas-Kyriazi. 2020. Show, edit and tell: A framework for editing image captions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725.

Ravi Shekhar, Ece Takmaz, Raquel Fernández, and Raffaella Bernardi. 2019. Evaluating the representational hub of language and vision models. In *Proceedings of the 13th International Conference on Computational Semantics*.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*.

Alasdair Tran, Alexander Mathews, and Lexing Xie. 2020. Transform and Tell: Entity-Aware News Image Captioning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *CVPR*.

Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, Raymond J. Mooney, Trevor Darrell, and Kate Saenko. 2017. Captioning images with diverse objects. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1170–1178.

Zeyu Wang, Berthy Feng, Karthik Narasimhan, and Olga Russakovsky. 2020. Towards unique and informative captioning of images. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Spencer Whitehead, Heng Ji, Mohit Bansal, Shih-Fu Chang, and Clare Voss. 2018. Incorporating background knowledge into video description generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

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 *Proceedings of the 32nd International Conference on Machine Learning*.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. 2018. Wikipedia2vec: An optimized tool for learning embeddings of words and entities from wikipedia. *CoRR*.

X. Yang, H. Zhang, D. Jin, Yingru Liu, Chi-Hao Wu, Jianchao Tan, Dongliang Xie, Jue Wang, and Xin Wang. 2020. Fashion captioning: Towards generating accurate descriptions with semantic rewards. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

A Appendices

The appendices provide more information about the two datasets GoodNews and NYTimes800k, template statistics and prediction results, implementation and training details, model difference between Tell and JoGANIC, details on the human evaluation and ablation evaluations of another sequence length efficient Transfomer - Longformer as well as different sequence length for MSTR.

A.1 Datasets

We use two datasets GoodNews (Biten et al., 2019) and NYTimes800k (Tran et al., 2020). Both datasets are collected by using The New York Times public API[7] For GoodNews dataset, since only the articles, captions, and image URLs are publicly released, the images need to be downloaded from the original source. Out of the 466K image URLs provided by Biten et al. (2019), we were able to download 463K images, the remaining

[7]https://developer.nytimes.com/apis

	GoodNews	NYTimes800k
# of articles	257033	444914
# of images	462642	792971
Average article length	653	892
Average caption length	18	18
% of caption words that are		
nouns	16%	16%
pronouns	1%	1%
proper nouns	23%	22%
verbs	9%	9%
adjectives	4%	4%
named entities	27%	26%
% of captions with		
named entities	97%	96%
people's names	68%	68%

Table 5: Summary of news captioning datasets.

Dataset	average len	% > 512	% > 1000
GoodNews	653	49.7%	18.2%
NYTimes800k	892	54.85%	21.92%

Table 6: Article length statistics for the GoodNews and NYTimes800k dataset.

are broken links. We use the same train, validation and test splits provided as Biten et al. (2019). There are 421K training, 18K validation, and 23K test captions.

NYTimes800k dataset is 70% larger and more complete dataset of New York Times articles, images, and captions. The number of train, validation and test sets are 763K, 8K and 22K respectively. Tab. 5 presents a detailed comparison between GoodNews and NYTimes800k in terms of articles and captions length, and captions composition.

We also show the article length statistics in Tab. 6. With approximate 50% of the training articles having more than 512 tokens, MSTR technique is necessary to deal with this problem.

A.2 Template statistics and prediction results

We show the composition in terms of components and the percentage of the template classes of the whole GoodNews dataset in Tab. 7.

We also report in Tab. 8 detailed template components precision and recall scores for different variants of our model and the Tell baseline on the two datasets.

A.3 Implementation and Training details

Following Tran et al. (2020), we set the hidden size of the input features $d_I = 2048$, $d_T = 1024$ and $d_E = 300$ and the number of heads $H = 16$. We use the Adam optimizer (Kingma and Ba, 2015)

template	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	avg
%	15.2	4.4	4.2	3.5	2.8	2.6	13.1	12.7	7.7	7.3	6.8	5.3	5.1	2.4	2.2	–
who	×	×	×	×	×	×	×	×	×	×	×	×	×	–	–	–
when	×	–	–	×	–	×	–	–	×	–	×	–	×	–	–	–
where	×	–	×	×	–	–	–	×	–	–	×	×	–	×	–	–
misc	×	–	–	–	×	–	–	–	–	×	×	×	×	–	–	–
context	–	–	–	–	–	–	×	×	×	×	×	×	×	×	×	–

Table 7: Template class definition and its relation with different components. Note there are in total 2^5 template class but we show the ones with over 2% of samples which accounts for 96.2% of the training data.

		Average		**who**		**when**		**where**		**misc**		**context**	
		$\overline{P}$	$\overline{R}$	P	R	P	R	P	R	P	R	P	R
GoodNews	Tell (Tran et al., 2020)	69.52	63.31	90.44	83.48	59.25	58.78	62.90	66.67	51.59	42.50	83.43	65.13
	Tell (full) (Tran et al., 2020)	71.55	64.93	91.49	85.89	63.04	60.69	65.45	66.90	53.27	46.00	84.50	65.19
	JoGANIC (zero-out text)	18.92	16.77	23.15	21.28	16.96	15.12	18.63	17.86	13.14	11.58	22.72	18.01
	JoGANIC (zero-out image)	48.74	46.29	63.09	60.21	39.82	37.40	44.96	42.60	25.50	23.03	70.33	68.21
	JoGANIC (image only)	19.07	17.13	23.38	21.61	17.15	15.52	18.70	17.92	13.31	11.72	22.81	18.88
	JoGANIC (text only)	49.56	46.98	63.89	60.89	40.73	38.11	45.81	43.25	26.20	23.73	71.17	68.92
	JoGANIC (Longformer)	74.07	58.86	94.46	87.45	63.29	26.66	71.45	62.66	54.67	45.30	86.46	72.25
	JoGANIC (auto)	75.51	66.27	94.77	86.59	66.15	64.19	71.21	67.79	**58.92**	46.12	86.52	66.65
	JoGANIC+NEE (auto)	74.42	68.53	94.64	88.65	64.66	64.26	70.54	68.22	55.93	48.65	86.34	72.88
	JoGANIC+MSTR (auto)	75.57	**70.04**	94.32	**91.00**	**68.75**	**66.91**	71.50	**70.16**	56.81	**49.07**	86.49	**73.07**
	JoGANIC+MSTR+NEE (auto)	**75.83**	68.85	**95.75**	90.01	66.72	64.45	**72.19**	69.38	57.33	48.48	**87.19**	71.96
	JoGANIC (oracle)	*92.69*	*87.86*	*95.07*	*88.21*	*97.09*	*95.10*	*88.50*	*84.02*	*84.07*	*78.01*	*98.75*	*93.97*
	JoGANIC+MSTR+NEE (oracle)	*92.46*	*87.55*	*95.07*	*88.79*	*97.00*	*93.86*	*88.09*	*83.79*	*83.43*	*75.81*	*98.70*	*95.50*
NYTimes800k	Tell (Tran et al., 2020)	67.13	62.24	86.44	79.65	57.45	63.93	61.08	72.19	46.30	36.99	84.39	58.44
	Tell (full) (Tran et al., 2020)	69.72	63.52	88.91	82.92	61.30	65.83	63.97	73.52	49.40	39.34	85.07	56.01
	JoGANIC (zero-out text)	18.87	16.53	22.96	20.96	16.52	14.88	18.01	17.21	13.02	11.34	23.84	18.26
	JoGANIC (zero-out image)	53.71	51.29	79.74	68.10	41.83	45.91	51.07	48.82	26.89	30.19	69.02	63.43
	JoGANIC (image only)	19.40	17.12	23.51	21.55	16.98	15.51	18.54	17.69	13.61	11.82	24.36	19.03
	JoGANIC (text only)	54.89	52.31	80.96	69.08	44.02	47.02	52.39	49.75	28.15	31.23	68.93	64.47
	JoGANIC (Longformer)	68.93	56.67	88.24	83.72	60.59	28.63	65.10	65.73	46.64	38.87	84.08	66.40
	JoGANIC (auto)	73.37	65.79	92.89	85.61	**64.62**	**66.90**	69.69	72.22	**53.25**	41.53	86.38	62.69
	JoGANIC+NEE (auto)	73.02	**66.54**	**93.54**	**86.45**	63.54	64.44	68.43	**74.59**	52.81	**44.37**	86.80	62.86
	JoGANIC+MSTR (auto)	73.36	66.30	93.21	85.97	64.59	66.60	70.32	73.25	51.67	42.62	87.02	**63.08**
	JoGANIC+MSTR+NEE (auto)	**73.51**	65.49	93.10	86.44	64.09	65.05	**70.40**	74.26	52.82	41.05	**87.15**	60.66
	JoGANIC (oracle)	*90.76*	*87.99*	*93.35*	*87.63*	*95.42*	*94.04*	*87.59*	*86.57*	*78.58*	*76.08*	*98.86*	*95.65*
	JoGANIC+MSTR+NEE (oracle)	*90.07*	*87.92*	*92.88*	*88.16*	*94.70*	*93.59*	*86.86*	*87.28*	*77.26*	*75.03*	*98.67*	*95.55*

Table 8: Precision and Recall results of each template component prediction on GoodNews and NYTimes800k. We highlight the **best** model in bold.

with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-6}$. The number of tokens in the vocabulary $K = 50264$ and $d^{Wiki} = 300$. We use a maximum batch size of 16 and training is stopped after the model has seen 6.6 million examples, corresponding to 16 epochs on GoodNews and 9 epochs on NYTimes800k. The components prediction head in Fig. 2 of the main paper is a Linear layer followed by an output layer with hidden states dimension equal to 1024. The training pipeline is written in PyTorch (Paszke et al., 2017) using the AllenNLP framework (Gardner et al., 2018). The RoBERTa model and dynamic convolution code are adapted from fairseq (Ott et al., 2019). Training is done with mixed precision to reduce the memory footprint and allow our full model to be trained on a single GPU. The models take 4 to 6 days to train on one V-100 GPU on both datasets.

A.4 Model Difference Between Tell and JoGANIC

As shown in Tab. 9, our model shares some components with the baseline model Tell (Tran et al., 2020). JoGANIC and Tell both use an image and text encoder and a Transformer decoder. However, JoGANIC applies template guidance to model the journalistic guidelines for caption generation.

A.5 Human evaluation

We have conducted a human evaluation of 200 article-image pairs. Below the article and the image, we displayed either the ground truth caption or a caption generated by Tell or one of our model variant. We ask the annotators to rate each caption as follows:

- How well does the caption describe the IMAGE? Regardless of how fluent it is.

	image	text	template guidance	faces	objects	weighted RoBERTa	location aware	decoder	# of parameters
Tell	×	×	–	–	–	–	–	Transformer	125M
Tell (full)	×	×	–	×	×	×	×	Transformer	200M
JoGANIC	×	×	×	–	–	–	–	Transformer	205M

Table 9: The difference between JoGANIC and Tell (Tran et al., 2020). Tell can be regarded as a variant of JoGANIC without template guidance. ×: having this technique. –: not having this one.

  - 1 = Very bad (Does not describe the image)
  - 2 = Somewhat bad (Describes the image, but contradictory to or missing key information from the image)
  - 3 = Somewhat good (Describes the image, no contradictions but missing key information from the image)
  - 4 = Very good (Describes the image, no contradictions and contains the key information from the image)

- How well does the caption summarize the ARTICLE? Regardless of how fluent it is.
  - 1 = Very bad (Not relevant to the topic)
  - 2 = Somewhat bad (Covers the right topic, but contradicting the article or missing key facts)
  - 3 = Somewhat good (Covers the right topic, no contradictions with the article, but missing key facts)
  - 4 = Very good (Covers the right topic, no contradictions with the article, and contains the key facts)

- How easy or hard is it to understand the SENTENCE? Regardless of how well it describes the image or article.
  - 1 = Very hard or doesn't make sense
  - 2 = Somewhat hard
  - 3 = Somewhat easy
  - 4 = Very easy to understand

Each image-article pair is shown to three different annotators, thus each caption is rated three times. We average the rating for each caption, and then plot the image relevance, article relevance and sentence quality ratings statistics as violin plots in Figure 4, Figure 5 and Figure 6, respectively. In each of these plots, the median is reported as a large dashed line, the first and third quartile as thinner dashed lines and the mean score as the black diamond. The varying height of each violin represent the number of samples having the corresponding rating. We can observe that all distributions are somewhat similar, but the Tell model is generally produces the lowest rated captions. The basic JoGANIC is a bit better, while more advanced variations of our model produce captions that are rated higher and really similarly to the ground truth captions.

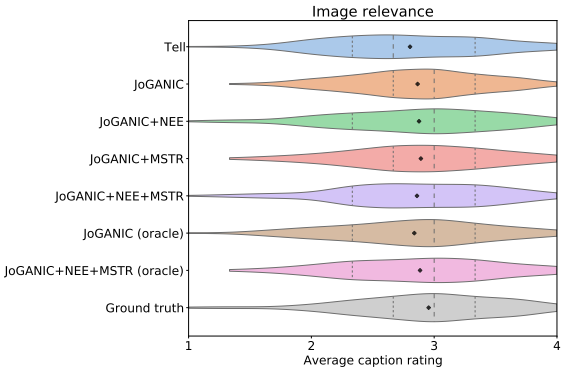

Figure 4: Image relevance ratings distributions.

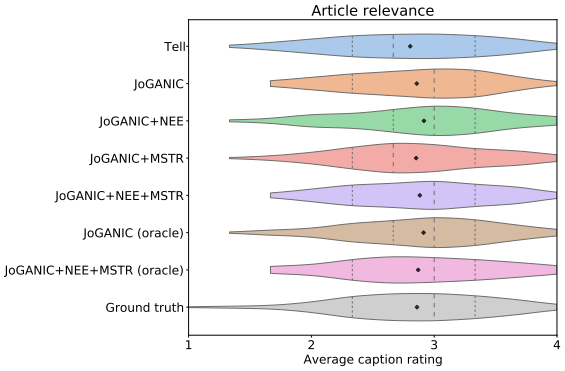

Figure 5: Article relevance ratings distributions.

A.6 Ablation Study

In addition to the *Multi-Span Text Reading* (MSTR) method proposed as an efficient technique to read long articles, we also try out Longformer (Beltagy et al., 2020), which is proposed to read long articles efficiently with an attention mechanism that scales linearly with sequence length. This attention mechanism is a drop-in replacement for the standard self-attention and combines a local windowed

		General Caption Ceneration				Named Entities		Training Time
		BLEU-4	ROUGE	METEOR	CIDEr	P	R	h/epoch
GoodNews	JoGANIC (Longformer)	5.69	21.08	9.97	52.04	24.63	19.63	**0.91**
	JoGANIC (RoBERTa)	6.34	21.65	10.78	59.19	24.60	20.90	1.02
	JoGANIC (RoBERTa+MSTR 800)	6.38	21.72	10.80	59.33	24.63	21.22	1.23
	JoGANIC (RoBERTa+MSTR 1000)	**6.45**	**21.99**	10.83	59.65	**24.75**	21.61	1.41
	JoGANIC (RoBERTa+MSTR 1200)	6.44	21.98	**10.85**	59.66	24.74	**21.63**	1.58
	JoGANIC (RoBERTa+MSTR 1400)	6.45	21.96	10.80	**59.67**	24.74	21.60	1.83
NYTimes800k	JoGANIC (Longformer)	5.72	19.55	9.87	41.66	22.89	18.09	**0.94**
	JoGANIC (RoBERTa)	6.39	22.38	10.75	56.54	**27.35**	**23.73**	1.09
	JoGANIC (RoBERTa+MSTR 800)	6.41	22.40	10.79	56.92	27.01	23.70	1.26
	JoGANIC (RoBERTa+MSTR 1000)	**6.44**	22.63	**10.88**	**57.61**	26.41	23.67	1.47
	JoGANIC (RoBERTa+MSTR 1200)	6.42	**22.64**	10.83	57.59	26.43	23.61	1.69
	JoGANIC (RoBERTa+MSTR 1400)	6.42	22.62	10.81	57.60	26.40	23.58	1.88

Table 10: Results on GoodNews and NYTimes800k. We highlight the **best** model in bold. Note that we report the mean values of three runs, and the maximum standard derivations of our variants on BLEU, ROUGE, METEOR, CIDEr are 0.013, 0.019, 0.016 and 0.069, which shows the stability of our results and that our method improvements are notable.

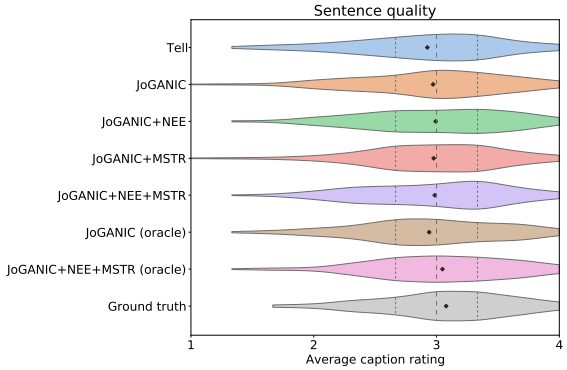

Figure 6: Sentence quality ratings distributions.

attention with a task motivated global attention. In this experiment, we replace RoBERTa with Longformer as the text feature extractor. Results are shown in Tab. 10. Unexpectedly, the Longformer variant of JoGANIC underperforms the RoBERTa variant. The possible reason is that in order to improve training efficiency, Longformer applies local windowed attentions with sparse global attentions. However, in this task, global attention is needed in every token. One possible solution for Longformer is to re-do the pretraining with fully global attention. However, this might be a non-trivial task and we will explore this in the future work.

We also conduct experiments to get the best possible number of tokens for MSTR. We applied number of tokens equal to 512, 800, 1000, 1200 and 1400 respectively. We found that the best choice is 1000 as it provides nearly the best performance while the training time per epoch is still good enough.