ZmBART: An Unsupervised Cross-lingual Transfer Framework for Language Generation

Kaushal Kumar Maurya
Indian Institute of Technology Hyderabad
Hyderabad, India
cs18resch11003@iith.ac.in

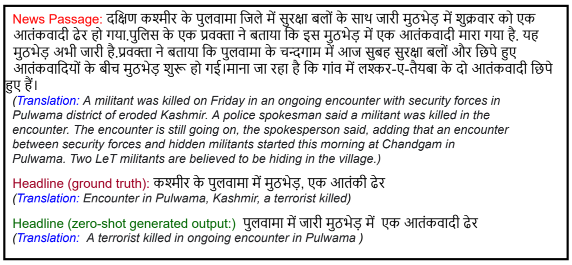
Maunendra Sankar Desarkar
Indian Institute of Technology Hyderabad
Hyderabad, India
maunendra@cse.iith.ac.in

Yoshinobu Kano
Shizuoka University, Japan
kano@inf.shizuoka.ac.jp

Kumari Deepshikha
NVIDIA, India
deepkshikha@gmail.com

Abstract

Despite the recent advancement in NLP research, cross-lingual transfer for natural language generation is relatively understudied. In this work, we transfer supervision from high resource language (HRL) to multiple low-resource languages (LRLs) for natural language generation (NLG). We consider four NLG tasks (text summarization, question generation, news headline generation, and distractor generation) and three syntactically diverse languages, i.e., English, Hindi, and Japanese. We propose an unsupervised cross-lingual language generation framework (called ZmBART) that does not use any parallel or pseudo-parallel/back-translated data. In this framework, we *further* pre-train mBART sequence-to-sequence denoising auto-encoder model with an auxiliary task using monolingual data of three languages. The objective function of the auxiliary task is close to the target tasks which enriches the multi-lingual latent representation of mBART and provides good initialization for target tasks. Then, this model is fine-tuned with task-specific supervised English data and directly evaluated with low-resource languages in the Zero-shot setting. To overcome catastrophic forgetting and spurious correlation issues, we applied freezing model component and data argumentation approaches respectively. This simple modeling approach gave us promising results. We experimented with few-shot training (with 1000 supervised data-points) which boosted the model performance further. We performed several ablations and cross-lingual transferability analysis to demonstrate the robustness of ZmBART.



News Passage: दक्षिण कश्मीर के पुलवामा जिले में सुरक्षा बलों के साथ जारी मुठभेड़ में शुक्रवार को एक आतंकवादी ढेर हो गया.पुलिस के एक प्रवक्ता ने बताया कि इस मुठभेड़ में एक आतंकवादी मारा गया है. यह मुठभेड़ अभी जारी है.प्रवक्ता ने बताया कि पुलवामा के चन्दगाम में आज सुबह सुरक्षा बलों और छिपे हुए आतंकवादियों के बीच मुठभेड़ शुरू हो गई।माना जा रहा है कि गांव में लश्कर-ए-तैयबा के दो आतंकवादी छिपे हुए हैं।

(Translation: A militant was killed on Friday in an ongoing encounter with security forces in Pulwama district of eroded Kashmir. A police spokesman said a militant was killed in the encounter. The encounter is still going on, the spokesperson said, adding that an encounter between security forces and hidden militants started this morning at Chandgam in Pulwama. Two LeT militants are believed to be hiding in the village.)

Headline (ground truth): कश्मीर के पुलवामा में मुठभेड़, एक आतंकी ढेर
(Translation: Encounter in Pulwama, Kashmir, a terrorist killed)

Headline (zero-shot generated output:) पुलवामा में जारी मुठभेड़ में एक आतंकवादी ढेर
(Translation: A terrorist killed in ongoing encounter in Pulwama)

Figure 1: Zero-shot news headline generation from ZmBART in Hindi language

1 Introduction

Recent advancement in natural language generation (NLG) is heavily oriented towards large annotated training data. Such large task-specific annotated data is available for high resource language (HRL) like English. The tasks become challenging when limited training data is available. This is often observed for low-resource languages (LRLs) like Hindi, Japanese, etc. Manually annotating large data is time-consuming, expensive and uninteresting. This limits the model development and product deployment for LRLs. Moreover, despite large active research in cross-lingual representation learning (Hu et al., 2020; Conneau et al., 2020; Lewis et al., 2020b), the area of cross-lingual transfer and generation is relatively under-explored. Motivated by these factors, we propose a novel framework to transfer supervision from HRL to LRLs where model is trained on one language and directly evaluated for unseen languages. This enables cross-lingual transfer and generation for low resource languages in zero and few-shot settings for different tasks. The framework can be easily extended to other tasks and languages.

We carefully selected four challenging NLG tasks i.e., news headline-generation (NHG), question generation (QG), abstractive text summarization (ATS) and distractor generation (DG) to validate the framework's performance. NHG and ATS require understanding of input passage to generate meaningful headline and summary respectively. QG task should accumulate information from a passage and answer to generate high-quality questions.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 2804–2818
August 1–6, 2021. ©2021 Association for Computational Linguistics

Distractor generation is the task of generating incorrect options from reading comprehension MCQ. It is challenging because generated distractors should be in the context with question but should not be semantically equivalent to the answer. We consider two LRLs i.e., Hindi and Japanese from two different language families. English is selected as the HRL from which the learning would be transferred to the LRLs. All three selected languages are different in their syntactic structures and typologically diverse. As there is no established publicly available dataset for DG in Hindi, we also create a new DG dataset for Hindi called as **HiDG**[1].

Our proposed framework to achieve this transfer of supervision from HRL to LRL under multiple languages and multiple tasks is named as ZmBART. ZmBART is based on mBART (Liu et al., 2020), a pre-trained model for cross-lingual natural language generation (NLG). We *further* pretrain mBART with a novel auxiliary task. Then the trained model is fine-tuned on large task-specific supervised data in English and evaluated directly on Hindi and Japanese languages in zero/few-shot setting for the tasks under consideration. We observe that the auxiliary task plays a critical role on the model's performance and needs to be carefully designed. This framework can be directly applied to multiple cross-lingual generation tasks without even the need to modify model hyper-parameters. Figure-1 shows a zero-shot NHG sample output generated by the ZmBART model. Our main contributions in this work can be summarized as:

1. We propose a novel zero-shot cross-lingual generation framework called ZmBART without parallel data and without back-translation. The framework can be directly applied across multiple tasks without even modifications in hyper-parameter values.
2. We demonstrate the effectiveness of ZmBART on four cross-lingual generation tasks across three typologically diverse languages.
3. We have created *HiDG*, a high-quality distractor generation dataset for the Hindi language.

2 Related Work

Early works on cross-lingual generation rely on machine translation (MT). In the very first work, Wan et al. (2010) leveraged the MT pipeline for cross-language document summarization. They first translate the non-English test instances to English. This translated text is fed through the supervised model (trained with document summarization data in English) to generate English summaries. Finally, these summaries are translated back to the target language. Shen et al. (2018) and Duan et al. (2019) used MT systems to generate pseudo training data for cross-lingual summarization and news headline generation respectively. However these MT based models are not suitable for low resource languages as they do not share parameters across-languages and generated translations are error-prone.

Recently there are a few works in the direction of supervision transfer from HRL(s) to LRL(s) for language generation. Kumar et al. (2019) used back-translation (needs MT system) and annotated supervised data for cross-lingual question generation. Chi et al. (2020) used parallel data to train a sequence-to-sequence model for zero-shot cross-lingual abstractive text summarization and question generation. Lewis et al. (2020a) proposed a pre-training based on mono-lingual paragraphs. Then this pre-trained model is used for zero-shot abstractive text summarization (ATS) in multiple languages. They trained a model on the ATS dataset on all the languages except the test language. This approach needs annotated data in multiple languages. Existing supervision transfer methods require parallel data for the cross-lingual tasks. Either they use available parallel corpora directly, or they translate/ back-translate data to generate pseudo-parallel corpora. Both these approaches pose significant challenges, as task-specific parallel data for multiple languages is difficult to obtain, and MT are far from perfect, especially for low resource languages.

Unlike the previous approaches, we did not use any parallel data or back-translation in our proposed framework. We did not pre-train any model from scratch. Instead, we leveraged the existing pre-trained model mBART. We included four challenging generation tasks across three syntactically diverse languages. Even we did not modify any hyper-parameters across the tasks and languages. All these considerations make the framework simple and easy to use. Further, it enables the addition of different other languages and NLG tasks in the proposed framework a simple extension exercise.

[1] HiDG dataset download link: `https://github.com/kaushal0494/ZmBART`

3 Methodology

Figure 2 shows an outline of our proposed ZmBART framework. ZmBART is based on pre-trained mBART (Liu et al., 2020) model. In our framework, we take the mBART model and *further pre-train* it on an auxiliary task. The auxiliary task is designed in such a way that the objective function of auxiliary task is close to fine-tuning tasks and only utilizes the mono-lingual data from the selected languages. Similar to mBART model we use language identifier tag with slight modification. We concatenate $< fxx >< 2xx >$ tags in input data instance where xx indicates the language tag. Given an input sentence and the language tag the model encodes the sentence in multi-lingual space. By conditioning on the encoded representation and language tag the decoder generates output text in target language.

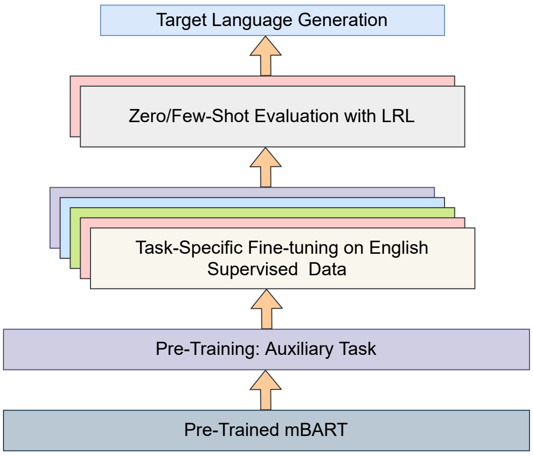

Target Language Generation
Zero/Few-Shot Evaluation with LRL
Task-Specific Fine-tuning on English Supervised Data
Pre-Training: Auxiliary Task
Pre-Trained mBART

Figure 2: Architecture diagram of ZmBART

3.1 Multilingual BART (mBART)

Multilingual BART (Liu et al., 2020) is an extension of BART model (Lewis et al., 2020c) to multiple languages. It is a transformer-based sequence-to-sequence pre-trained model. The model is trained on monolingual data in many languages from Wikipedia Common Crawl corpus with BART language model objective. Particularly, The training data is concatenation of data from K languages i.e., $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2 \dots \mathcal{D}_K\}$ where $\mathcal{D}_i$ is a collection of monolingual documents in language i. They introduced two types of noises to corrupt the text: (1) random token span masking and (2) sentence order permutation. mBART is trained as denoising autoencoder. During training, the model has to predict text X from it's corrupted version $g(X)$, where g is noise function. The aim is to maximize the following objective function

$$\mathcal{L}_\theta = \sum_{\mathcal{D}_i \in \mathcal{D}} \sum_{x \in \mathcal{D}_i} logP(x|g(x); \theta), \qquad (1)$$

where x is a data instance of language i. Probability distribution P is defined by the sequence-to-sequence model. mBART gave state-of-the-art results in sentence and document level machine translations tasks. Details about mBART model can be found in Liu et al. (2020).

3.2 Unsupervised Auxiliary Task

Although the mBART pre-trained model encodes a multi-lingual latent space, it can not be used directly for cross-lingual generation. This is because the model is jointly trained on denoising objectives which do not directly follow auto-regressive decoding, thereby causing mismatch between pre-training and fine-tuning objectives. To overcome this problem, an unsupervised auxiliary task is introduced. We design the auxiliary task with the following desiderata in mind. It (1) should only utilize mono-lingual data from selected languages, (2) should enrich the mBART latent representations for selected languages and (3) train the decoder in pure auto-regressive manner with a training objective which is close to multiple fine-tuning tasks.

The auxiliary task in ZmBART is an additional pre-training step for *better warm-start* to downstream auto-regressive NLG tasks - although the final task (Distractor/Question/Summary generation) can be different from the auxiliary task. Additionally, this step allows the model to have a closer look at the languages under consideration and enrich/adjust the representations and parameters accordingly.

Outputs of the NLG tasks considered in this work are expected to contain words from different parts of the input. Generation of the output tokens are handled by the framework using an encoder-decoder setup. Hence we decide to have an auxiliary task that also encodes the input, and attends to this encoded representation to generate the output words in auto-regressive manner. This way, a single auxiliary task can help to enrich the token representations, warm up the encoder-decoder weights for fine tuning, and also caters to the multiple final output tasks. We define the auxiliary task as: *Given an input passage, generate few random sentences (called rand-summary) from the passage*. After experimentation we found that randomly generating 20% sentences from passage works the best.

Particularly, the input passage has length between 5-25 sentences and output is 1-5 random sentences from the passage. We do not assume any relations among sentences of the passage. We sample equal proportion of monolingual data from three languages. Data preparation steps for the auxiliary task are given below:

1. Generate a random number $k \in \{5. \cdots, 25\}$. k denotes the size of input passage
2. PASSAGE: Append k continuous sentences, starting from a random index of monolingual corpus D_i of the i^{th} language
3. RAND-SUMMARY: Randomly select 20% sentences from the passage
4. Repeat steps 1 to 3 for p languages
5. Repeat steps 1 to 4 for N times, to collect Np <PASSAGE, RAND-SUMMARY> pairs

3.3 Fine-Tuning on Downstream NLG Tasks

The proposed pre-trained model is directly fine-tuned on four downstream tasks: Question Generation (QG), News Headline Generation (NHG), Abstractive Text Summarization (ATS) and Distractor Generation (DG). First, the model is fine-tuned on large task-specific English supervised data and then this trained model is directly evaluated on Hindi and Japanese evaluation datasets in zero-shot setting. To validate the hypothesis that the ZmBART framework is robust across multiple tasks and languages, we did not modify any hyper-parameters during fine-tuning. It is often observed that including a few instances from LRL to supervised data boosts the model performance. To validate this point we further fine-tuned ZmBART with 1000 task-specific supervised data-points in Hindi and Japanese languages in few-shot setting which boosts the model performance.

3.4 Dealing with Catastrophic Forgetting and Spurious Correlation

During experimentation with the zero shot setup, it is observed that the model always generates the output text in English irrespective of input and language tag. We suspect this to be due to catastrophic forgetting problem (Van de Ven and Tolias, 2019). The supervised training completely overrides/erases the pre-trained learning. The generator (decoder) becomes biased towards English due to the explicit supervision learned from large task-specific English data. To overcome this problem, we freeze all word embeddings and all the parameters of decoder layers during fine-tuning with English data. Although this resolves the problem for NHG, QG and DG, the problem did not get completely resolved for the ATS task. We noticed that the zero-shot ATS output now is not completely in English, but it became of code-mix nature. In other words, the number of English words in the output reduced, but still lot many English words remained. The code-mixed outputs were logical and meaningful. We assume this to be due to spurious correlation issue, also reported in (Gu et al., 2019). To resolve this issue, we added a few examples (25 in number) of the auxiliary-task data during the fine-tuning step. This augmentation was helpful to address the spurious correlation issue for ATS. It is to be noted that the non-English data used for this augmentation is still of unsupervised and monolingual nature.

4 Experimental Setup and Results

We conduct experiments over four NLG tasks in three languages. We compare the performance of ZmBART with strong and MT pipeline based baseline models. We use both automated and manual evaluation metrics to evaluate model performances.

4.1 Baselines

Prior results are not available in literature for selected languages and datasets. Hence, for performance comparison, we developed several strong baselines based on recent models and architectures. Details of these baselines are mentioned below:

- **MT Pipeline (mBART):** Here, we fine-tune mBART on task-specific English data. Non-English test data instances are first translated into English and passed to the fine-tuned model. The output is translated back to the input language. Google Translator is used for translations.
- **mBART+MADMO:** This is an **mBART** based baseline where the auxiliary task has **M**asking **A**nd **D**enoising objective with **Mo**no-lingual data in three languages. The aim is to enrich the cross-lingual latent representation space of mBART for English, Hindi and Japanese.
- **mBART+MADPD:** Inspired from (Chi et al., 2020), we took **P**arallel **D**ata (English-Hindi and English-Japanese) and concatenate each parallel instances of two languages. Then we used this data with **M**asking **A**nd **D**oising objective to further train mBART. Including parallel data provides explicit supervision while generating Hindi and Japanese text.

4.2 Evaluation

We use both automated and manual evaluation metrics for performance comparison. Multiple metrics are used in literature for NLG tasks. Since we are considering multiple tasks, for brevity, against each task we only report values of the metrics commonly used by the community for that particular task. For automatic evaluation we used both lexical match (**BLEU** (Papineni et al., 2002) and **ROUGE** (Lin, 2004)) as well as embedding based evaluation metrics (**BERTScore** (Zhang et al., 2020)). To evaluate question generation and distractor generation tasks we use case-mix BLEU-4 (BL) score from sacreBLEU implementation, ROUGE-L (R-L) and BERTScore (BS). For ATS and NHG tasks ROUGE-1, ROUGE-2 and ROUGE-L are used.

We follow a similar approach for manual evaluation as Chi et al. (2020). We sampled 50 generated data points each for QG, ATS and NHG tasks in both Hindi and Japanese languages. We use three metrics: *Fluency* (Flu), *Relatedness* (Rel) and *Correctness* (Corr). **Fluency** measures *how fluent the generated text is*. **Relatedness** indicates *how much the generated outputs are in the context with input(s)*, **Correctness** measures *semantics and meaningfulness*. For DG, we use an additional metric called **Distractibility** that measures *the degree of confusion for generated incorrect options*. For DG task, there can be large number of good distractors for given input, in such situation the manual evaluation is more reliable. We sample 100 generated outputs for DG task. We employed large pool of evaluators from native Hindi and Japanese speakers to evaluate Hindi and Japanese output texts respectively. We asked each annotator to rate the generated texts on a scale of 1-5 (1 is very bad and 5 is very good) for all the metrics. We intentionally selected outputs of ZmBART and two best baselines to reduce the evaluators workload.

4.3 News Headline Generation (NHG)

In this task, *given a news article, we generate grammatically coherent, semantically correct and abstractive headline*. We use 500k/30k/30k (train/validation/test) English NHG data splits from *Gigaword* headline generation corpus[2]. For Hindi and Japanese we use 1k/1k/5k spilt from Kaggle[3] (we manually filtered high-quality news and headlines) and (Iwama and Kano, 2019) respectively.

In a zero-shot setting we fine-tune ZmBART model on supervised data and directly evaluate results on Hindi and Japanese test datasets. Automated evaluation results are included in Tables 1 and 2. We observe that, quality of generated headlines in Hindi is better compared to Japanese. The possible reasoning can be the input size. ZmBART outperforms the baseline with an absolute difference of 5.22 ROUGE-L score. mBART+MADMo is best among others which shows that masking and denoising with monolingual data indeed enrich the multi-lingual latent space for selected three languages. mBART+MADMo generates code mixed (Hindi-English or Hindi-Japanese) output which degrades the model performance. Few-shot training fills the mistakes of zero-shot models and generates better quality output. Manual evaluation scores (Tables 3 and 4) and automated scores correlate well validating ZmBART's performance on NHG task.

4.4 Question Generation (QG)

In the Question Generation (QG) task, *given an input passage and an answer, the aim is to generate semantically and syntactically correct questions that can produce the answer.* We use SQuAD 1.1 (Rajpurkar et al., 2016) English data for supervised training. SQuAD is popular question answering dataset consisting of 100k+ <passage, question, answer> tuples. Following (Zhao et al., 2018), we combine the train and validation sets of SQuAD and then spilt it as 80k/8k/10k training/validation/test tuples. For Hindi we use 1k/5.5k (train/test) from MLQA (Lewis et al., 2020d) and TyDiQA-GoldP (Clark et al., 2020) datasets. We use 1k/1k/5k for Japanese data from (Takahashi et al., 2019). Hindi and Japanese data are available in SQuAD data format which maintains consistency in terms of passage size, question and number of answers. For given passage and question we randomly selected one answer to form the dataset. We combine answer and passage as single input sequence separated by special token <s>.

Even without any parallel data, ZmBART outperformed all the baselines consistently across all automated evaluation metrics for zero-shot setting. Regarding manual evaluations, we see that Hindi questions received good score from the annotators, whereas the questions generated for the Japanese language inputs were considered as poor. Upon closer inspection of the generated text we find that several generated questions start with English wh-

[2] https://github.com/harvardnlp/sent-summary

[3] https://www.kaggle.com/disisbig/hindi-text-short-summarization-corpus

Model	**News Headline Generation**			**Question Generation**			**Abstractive TS**			**Distractor Generation**		
Metrics	**R-1**	**R-2**	**R-L**	**BL**	**R-L**	**BS**	**R-1**	**R-2**	**R-L**	**BL**	**R-L**	**BS**
Cross-lingual zero-shot generation results												
MT Pipeline(mBART)	16.61	4.91	15.83	2.6	21.31	71.53	11.15	3.11	10.93	1.6	9.66	67.35
mBART+MADMo	29.32	16.36	27.52	3.9	23.70	73.76	18.25	4.92	16.10	2.8	15.86	72.26
mBART+MADPd	24.02	13.41	23.29	4.3	25.29	73.74	10.47	2.55	12.30	2.9	15.43	72.89
ZmBART	**34.94**	**19.38**	**32.74**	**4.4**	**26.51**	**74.19**	**21.27**	**5.30**	**17.64**	**4.1**	**21.05**	**73.39**
Cross-lingual few-shot generation results (with 1000 supervised data points)												
ZmBART	52.37	35.52	50.50	7.6	34.11	78.29	36.29	14.21	27.22	6.5	26.58	78.27

Table 1: Zero and few-shot cross-lingual generation results for Hindi Language

Model	**News Headline Generation**			**Question Generation**			**Abstractive TS**		
Metrics	**R-1**	**R-2**	**R-L**	**BL**	**R-L**	**BS**	**R-1**	**R-2**	**R-L**
Cross-lingual zero-shot generation results									
MT Pipeline(mBART)	13.82	0.38	7.92	8.9	26.92	71.93	17.90	3.98	18.46
mBART+MADMo	33.75	8.12	17.78	16.6	34.80	74.01	28.74	9.01	23.63
mBART+MADPd	31.58	6.98	18.95	18.2	36.22	74.99	19.17	4.89	18.22
ZmBART	**35.25**	**9.24**	**19.92**	**18.8**	**38.74**	**75.91**	**36.60**	**15.26**	**29.85**
Cross-lingual few-shot generation results (with 1000 supervised data points)									
ZmBART	47.06	22.36	31.55	30.4	53.98	82.66	41.65	20.33	33.49

Table 2: Zero and few-shot cross-lingual generation results for Japanese Language

Model	**News Headline Generation**			**Question Generation**			**Abstractive TS**			**Distractor Generation**		
Metrics	**Flu**	**Rel**	**Corr**	**Flu**	**Rel**	**Corr**	**Flu**	**Rel**	**Corr**	**Flu**	**Rel**	**Dist**
Annotator set-01												
mBART+MADMo	3.86	4.34	3.94	2.66	3.38	3.52	3.56	3.58	3.22	3.61	4.08	2.89
mBART+MADPd	2.54	2.96	2.28	3.1	3.4	3.78	2.26	2.62	1.92	2.42	3.72	3.08
ZmBART	**4.14**	**4.22**	**4.04**	**3.24**	**3.44**	**3.9**	**4.02**	**4.12**	**3.54**	**4.12**	**4.19**	**3.83**
Annotator set-02												
mBART+MADMo	3.84	4.18	3.8	3.83	4.63	3.96	3.38	3.96	3.4	3.38	3.00	2.24
mBART+MADPd	2.96	3.02	2.7	**3.98**	4.70	3.98	2.96	3.16	2.84	2.97	3.11	**2.46**
ZmBART	**4.12**	**4.38**	**4.16**	3.95	**4.80**	**4.27**	**4.24**	**4.52**	**4.38**	**3.56**	**3.18**	2.36
Annotator set-03												
mBART+MADMo	3.56	3.74	**3.78**	2.68	3.76	3.32	2.9	3.34	2.9	3.96	3.74	3.12
mBART+MADPd	3.1	3.42	2.91	2.80	3.88	3.56	2.64	2.34	2.46	4.13	3.74	2.94
ZmBART	**3.70**	**3.84**	3.76	**2.86**	**4.04**	**3.76**	**4.06**	**3.56**	**3.56**	**4.44**	**4.12**	**3.12**

Table 3: Manual evaluation results of Zero-shot generated outputs for Hindi language

Model	**News Headline Generation**			**Question Generation**			**Abstractive TS**		
Metrics	**Flu**	**Rel**	**Corr**	**Flu**	**Rel**	**Corr**	**Flu**	**Rel**	**Corr**
Annotator set-01									
mBART+MADMo	2.66	2.98	2.50	1.98	**3.70**	**3.18**	3.04	3.55	3.44
mBART+MADPd	2.26	2.70	2.04	2.00	3.38	2.82	1.44	2.22	2.20
ZmBART	**3.60**	**4.02**	**3.50**	**2.12**	3.30	2.94	**4.24**	**3.90**	**3.90**
Annotator set-02									
mBART+MADMo	2.1	2.58	1.98	1.24	1.70	1.33	2.56	3.40	2.62
mBART+MADPd	1.58	1.78	1.46	**1.46**	1.72	1.78	1.00	1.00	1.00
ZmBART	**3.78**	**4.16**	**3.86**	1.26	**1.76**	**1.88**	**4.04**	**4.26**	**3.84**
Annotator set-03									
mBART+MADMo	2.24	2.72	2.24	**2.34**	2.46	2.39	2.82	3.18	3.52
mBART+MADPd	1.9	2.14	1.82	2.10	2.66	2.28	1.16	1.84	1.44
ZmBART	**2.88**	**3.22**	**2.92**	2.10	**2.70**	**2.46**	**3.32**	**3.52**	**3.04**

Table 4: Manual evaluation results of Zero-shot generated outputs for Japanese language

words. This mixing of English 'code' in the output happened somewhat seamlessly for the Hindi data as tokens in both languages are written in left-to-right manner. Moreover, Hindi-English code-mixed data is now getting very common and the annotators mostly accepted the mixing of the wh-words with the Hindi texts. Such mixing is not very common with Japanese text. As a result, the annotators assigned lower scores to such texts.

We then tried to understand the reason for getting the wh-words at the beginning of the output. English interrogative sentences often introduce wh-words at the beginning even though they are not present in the original data. The model gets exposed to such special characteristics of the English interrogative sentences during the fine tuning. The output from other languages get impacted due to this in zero-shot settings. However, the semantics of the text is captured well for the model as demonstrated by the high BERTScore, indicating good cross-lingual transfer of semantic knowledge.

4.5 Abstractive Text Summarization (ATS)

In Abstractive Text Summarization (ATS), we aim to *generate grammatically coherent, semantically correct and abstractive summary given an input document*. We use recently released WikiLingua (Ladhak et al., 2020) cross-lingual abstractive summarization dataset containing data in 18 languages. Prior splits are not available for this dataset. We use 131k/5k/5k (train/validation/test) splits for English, and 1k/1k/5k splits for Hindi and Japanese.

By skimming through data in Hindi we observe that many input documents consist of technical instructions on usage of softwares/tools. Summarizing these instructions are challenging. Zero-shot ZmBART performed better as compared to baselines as shown in human evaluation (Tables 3 and 4 for Hindi and Japanese respectively). The human evaluation results correlate with automated evaluation as shown in Tables 1 and 2. Ladhak et al. (2020) reported cross-lingual ATS score with same data for four different languages. The R-L score for four languages are 34.06, 37.09, 31.67 and 32.33. We obtain R-L scores of 27.22 and 33.49 for Hindi and Japanese respectively, which shows that the few-shot performance of ZmBART is acceptable.

4.6 Distractor Generation (DG)

The final task to judge ZmBART's performance is Distractor Generation (DG). It is the task of generating incorrect options (also known as distractors) from reading comprehension MCQ. The generated distractors should be in the context with the question but shouldn't be semantically equivalent to the answer. Formally, *for given passage, question and answer triplet, generate a long, coherent, and grammatically correct wrong option*. Considering the fact that for a given triplet there can be many incorrect options that are completely different from each other, the problem is even more challenging. We use English DG dataset from (Maurya and Desarkar, 2020) which consists of approx 135k/17k/17k (train/validation/test) split. We were unable to find a suitable dataset in Japanese language. For Hindi language we created a dataset called ***HiDG***[4] of 1k/1k/5k split. Similar to QG, to create input for ZmBART we concatenate the answer, question and passage in the same order and separate them with special token <s>.

To generate HiDG, we first extracted <passage, question, answer> triplets from English SQuAD 1.1 with atleast 150 tokens in the triplet. We generate distractors for these examples using model proposed by Maurya and Desarkar (2020). The distractors were translated to Hindi using Google Translator service. The translated distractors were manually verified or corrected (if necessary) by human annotators.

The evaluation of the task is challenging because: 1) there can be more then one correct distractors. Automated evaluation metrics may not able to capture this aspect as only one ground truth distractor is available and 2) it may possible that the generated distractor is semantically similar to answer with high lexical overlap with reference distractor in those situation lexical match based metrics are not suitable. To evaluate the DG task we mainly rely on BERTScore and manual evaluation. Towards this effort we consider higher number of DG samples for manual evaluation. Results from Tables 1 and 3 indicate the superiority of ZmBART over the baseline models for this task.

To summarize, we have performed experiments for 14 different task-setup combinations involving low resource languages. With four tasks in Hindi and three tasks in Japanese, and each task in zero shot and few shot setup, we provide detailed comparative evaluation for the tasks. The tasks are of different natures, and each task offers its own unique challenge. We critically analyze the performances to show the robustness and the range

[4] Implementation, dataset, pre-trained checkpoints and ZmBART generated text are available at `https://github.com/kaushal0494/ZmBART`

of applicability for the proposed ZmBART framework. We use fairseq library (Ott et al., 2019) for all the implementation and experiments. The implementation details are included in supplementary.

5 Results Analysis and Ablation Study

In this section, we provide further analysis of the experimental results. We also perform ablation studies to understand the impacts of the different modeling decisions made in designing the framework.

•**Supervised Training Results:** Table 5 shows the comparative results of fine-tuned mBART with and without auxiliary task on task-specific supervised English data. We observe that there is no significant performance degradation of ZmBART over original mBART model with pure supervised training. Even, the auxiliary task helps in achieving slight improvement over the original mBART performance in most setups. This concludes that ZmBART can be adopted as replacement of original mBART model with additional functionalities.

Task	Setting	BL	R-1	R-2	R-L	BS
NHG	W/ Aux-Task	15.9	43.22	21.33	40.88	90.13
	W/O Aux-Task	15.9	43.15	21.25	40.77	90.13
QG	W/ Aux-Task	20.6	53.20	26.53	51.37	92.18
	W/O Aux-Task	21.4	52.66	26.63	51.25	92.41
ATS	W/ Aux-Task	16.0	40.01	18.11	38.29	90.20
	W/O Aux-Task	15.8	39.52	18.00	37.91	90.10
DG	W/ Aux-Task	10.3	31.76	14.89	31.18	89.33
	W/O Aux-Task	10.0	31.87	14.59	31.30	89.42

Table 5: Automated evaluation results of mBART on task-specific supervised English dataset (with and without Auxiliary Task)

•**Effect of Auxiliary Task:** Table 6 includes the results with and without auxiliary task of ZmBART for ATS and QG tasks in zero-shot setting. It can be inferred that without the auxiliary task, lexical match based scores are poor because the decoder generates code-mixed outputs. We see that the BERTScore is still reasonable without auxiliary task owing to the multilingual mBART embedding. However, generation of the data in appropriate language is enabled only after inclusion of the auxiliary task. The auxiliary task contributes in two ways: it enables zero-shot generation and improves the mBART multilingual latent space even more as indicated by the improved BERTScore.

Model	Abstractive TS			Question Generation		
Metrics	**R-1**	**R-2**	**R-3**	**BL**	**R-L**	**BS**
Hindi Language						
ZmBART w/o Aux	4.34	0.10	3.19	0.9	16.64	70.72
ZmBART with Aux	21.27	5.30	17.64	4.4	26.51	74.19
Japanese Language						
ZmBART w/o Aux	6.80	0.11	5.30	6.7	33.07	70.35
ZmBART with Aux	36.60	15.26	29.89	18.8	38.74	75.91

Table 6: Zero-shot results of ZmBART with and without auxiliary task for Hindi and Japanese

With these results we now want to understand *whether the auxiliary task is able to generalize across multiple tasks, or favors specific tasks.* Among the tasks considered in this work, we see that generation of meaningful summaries/headlines require understanding/abstracting of input text which is unlikely to be obtained by repeating sentences from input passages, as done in the auxiliary task. ZmBART achieves good zero-shot/few-shot/supervised results (Tables 1-5) on ATS and NHG over strong baselines. The generated headlines and summaries were found to be mostly abstractive, they don't contain large continuous sequences from input text. As described in Sections 4.4 and 4.6, Question Generation and Distractor Generation are more challenging tasks and have objectives vastly different from the auxiliary task's objective. Even for these tasks, decent evaluation scores (Tables 1-5) and improvements over the baselines across the languages considered indicate that the solutions are not spurious. Incorporation of auxiliary task improves the performance of diverse downstream tasks on real benchmark datasets, and does not favor any specific task or dataset.

• **Approaches to avoid Catastrophic Forgetting:** We use two approaches to address the catastrophic forgetting problem, (a) Freezing model components and (b) optimized regularization (Van de Ven and Tolias, 2019). Tables 7 and 8 show the automated evaluation results with different approaches used to deal with the catastrophic forgetting problem. It can be noted that the proposed modelling setup (i.e., ZmBART) gives best results.

• **Effect of Architecture on Few-shot Training:** In this set-up we experiment with few-shot training with mBART (directly fine-tuned on task-specific supervised English data) and ZmBART (trained with auxiliary task and fine-tuned with English data). The results are presented in Table 9. We find that ZmBART does better than mBART in corresponding setups. Moreover, although freezing the decoder layer and word embeddings helps in zero-shot setting, it is natural and useful to unfreeze them during few shot training.

• **Few-shot performance with Supervised data:** Figures 3 and 4 show the trends of few-shot training of ZmBART with respect to supervised Hindi and Japanese training data for ATS

Setup	Setting-Details	BL(hi/ja)	R-L(hi/ja)	BS(hi/ja)
Model Components	Freeze word embedding (WE)	2.5/13.6	21.55/31.99	72.02/73.18
	Freeze WE + subset of Encoder & Decoder layers	2.9/15.3	22.62/36.60	72.24/72.98
	Freeze WE + Encoder layers	2.2/13.8	19.69/36.91	69.73/72.97
	Freeze WE + Decoder layers (ZmBART)	**4.4/18.8**	**26.51/38.74**	**74.19/75.91**
Regularized Optimization	Elastic Weight Consolidation (EWC)	2.1/11.6	18.21/29.47	68.36/72.91

Table 7: Evaluation scores for different modeling approaches to avoid catastrophic-forgetting for QG Task

Setup	Setting-Details	R-1(hi/ja)	R-2(hi/ja)	R-L(hi/ja)
Model Components	Freeze word embedding (WE)	13.02/26.07	05.67/03.96	12.45/17.62
	Freeze WE + subset of Encoder & Decoder layers	14.27/25.72	06.70/03.21	13.76/18.28
	Freeze WE + Encoder layers	09.81/22.67	04.10/02.38	09.66/13.68
	Freeze WE + Decoder layers (ZmBART)	**34.94/35.25**	**19.38/09.24**	**32.74/19.92**
Regularized Optimization	Elastic Weight Consolidation (EWC)	12.01/22.16	05.43/03.11	11.22/16.31

Table 8: Evaluation scores for different modeling approaches to avoid catastrophic-forgetting for NHG Task

Model	NHG			QG		
Metrics	R-1	R-2	R-3	BL	R-L	BS
mBART+WE	50.61	34.32	49.01	6.1	31.20	77.01
mBART	51.49	35.04	49.64	7.1	32.96	77.61
ZmBART+WE	51.81	35.04	50.07	6.9	32.82	77.40
ZmBART	**52.37**	**35.52**	**50.50**	**7.9**	**34.49**	**78.39**

Table 9: Hindi language few-shot results for different architectural setups. *WE* indicates that word embeddings and decoder layer parameters are frozen

and QG tasks respectively. We observe that with a small number of supervised examples (e.g. 100) the model achieves decent few-shot performance. We found the trends for different tasks to be similar. The improvement in model performance tends to be minimal after 1000 examples.

6 Conclusion

In this paper, we propose a novel unsupervised framework (ZmBART) for cross-lingual transfer and generation. The framework transfers supervision from HRL to LRLs which enables zero-shot language generation. The framework does not use any direct or pseudo-parallel data. ZmBART is directly applied to multiple generation tasks and languages. The model includes a carefully designed auxiliary task that further improved the multilingual embedding space, and helped to initialize encoder-decoder weights to enable zero shot language generation. We performed experiments in three languages and 18 task-setup combinations: four supervised tasks in English, four tasks in Hindi (each with zero-shot and few-shot), and three tasks in Japanese (each with zero-shot and few-shot). Except zero-shot question generation tasks, for all other tasks involving LRLs, the proposed model generated good quality results as validated by automated and manual evaluation measures. In future we want to extend this work by adding multiple other languages and tasks, and also explore other choices of auxiliary tasks for better model transfer.

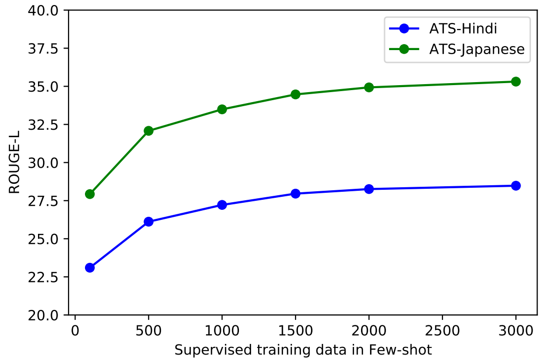


Figure 3: ZmBART model few-shot performance with supervised Hindi/Japanese data for ATS task

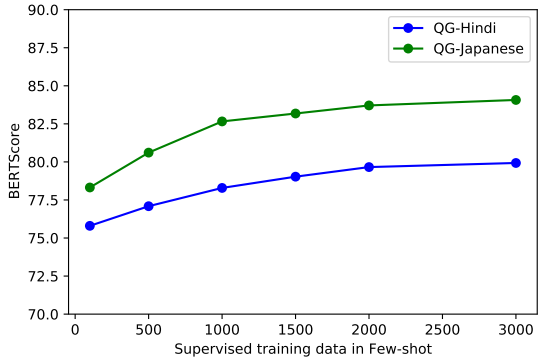


Figure 4: ZmBART model few-shot performance with supervised Hindi/Japanese data for QG task

Acknowledgments

We thank the support from Nvidia AI Technology Center (NVAITC) towards the requirements of computing power and compute infrastructure. We thank the human annotators for human evaluation and the anonymous reviewers for their constructive feedback.

References

Zewen Chi, Li Dong, Furu Wei, Wenhui Wang, Xian-Ling Mao, and Heyan Huang. 2020. Cross-lingual natural language generation via pre-training. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7570–7577.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A Benchmark for Information-Seeking Question Answering in Typologically Diverse Languages. In *Transactions of the Association of Computational Linguistics*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Xiangyu Duan, Mingming Yin, Min Zhang, Boxing Chen, and Weihua Luo. 2019. Zero-shot cross-lingual abstractive sentence summarization through teaching generation and attention. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3162–3172, Florence, Italy. Association for Computational Linguistics.

Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O.K. Li. 2019. Improved zero-shot neural machine translation via ignoring spurious correlations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1258–1268, Florence, Italy. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Kango Iwama and Yoshinobu Kano. 2019. Multiple news headlines generation using page metadata. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 101–105, Tokyo, Japan. Association for Computational Linguistics.

Vishwajeet Kumar, Nitish Joshi, Arijit Mukherjee, Ganesh Ramakrishnan, and Preethi Jyothi. 2019. Cross-lingual training for automatic question generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4863–4872, Florence, Italy. Association for Computational Linguistics.

Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A new benchmark dataset for cross-lingual abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4034–4048, Online. Association for Computational Linguistics.

Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida I. Wang, and Luke Zettlemoyer. 2020a. Pre-training via paraphrasing. *CoRR*, abs/2006.15020.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020b. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020c. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020d. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online. Association for Computational Linguistics.

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.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Kaushal Kumar Maurya and Maunendra Sankar Desarkar. 2020. Learning to distract: A hierarchical multi-decoder network for automated generation of long distractors for multiple-choice questions for reading comprehension. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1115–1124.

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

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Shi-qi Shen, Yun Chen, Cheng Yang, Zhi-yuan Liu, and Mao-song Sun. 2018. Zero-shot cross-lingual neural headline generation. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 26(12):2319–2327.

Norio Takahashi, Tomohide Shibata, Daisuke Kawahara, and Sadao Kurohashi. 2019. Machine comprehension improves domain-specific Japanese predicate-argument structure analysis. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 98–104, Hong Kong, China. Association for Computational Linguistics.

Gido M Van de Ven and Andreas S Tolias. 2019. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*.

Xiaojun Wan, Huiying Li, and Jianguo Xiao. 2010. Cross-language document summarization based on machine translation quality prediction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 917–926, Uppsala, Sweden. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3901–3910, Brussels, Belgium. Association for Computational Linguistics.

7 Supplementary Materials

7.1 Implementation Details:

We use a standard sequence-to-sequence Transformer architecture with 12 layers (each 16 heads) for encoder and decoder. The model has a dimension of 1024 (approx 680M parameters). Additional layer-normalization was used with both the encoder and decoder. We found FP16 precision stabilized the training. We trained all the models on 4 Nvidia V100 GPUs (32GB). Similar to mBART we use the Adam optimizer ($\epsilon = 1e$-6, $\beta_2 = 0.98$) and linear learning rate decay scheduling. The training started with a dropout value 0.3 and was later reduced to 0.2 after 20k steps and 0 after 40k steps. The loss function was cross-entropy label smoothing loss. 2500 warm-up steps and $3e$-5 learning rate were used. The model selection was done based on validation data likelihood. We use beam-search with beam size 5 in the decoding for all the tasks. We loaded mBARTCC25 pre-trained checkpoint weights and further pre-train/fine-tune model on task-specific data with teacher forcing method.

The above set of parameters are used for all the target tasks as well as the auxiliary task. We process different batch sizes of input for different tasks. We use 2048, 3000, 4096, 2048, and 5000 tokens per GPU for ATS, DG, QG, auxiliary, and NHG tasks, respectively. We use shared Byte Pair Encoding (BPE) vocabulary from sentencepiece tokenizer of size 250k. We use 34k/1k/1k (train/validation/test) data-points for auxiliary language (approx 11333 from each languages). We train the mBART model with the auxiliary task around 10k steps. Training time for the auxiliary task is around 2-3 hours. The fine-tuning times for TS, QG, NHG, and DG were around 4-5, 1-2, 1-2, and 2-3 hours. We observe a longer fine-tuning time for ATS because of long passages. We selected the best model based on loss and perplexity on the validation datasets. We checked with early-stopping and other checkpoints, which resulted in poor performance.

7.2 Evaluation Metric and Tokenizer Details:

For Automated evaluation, we use sacreBLEU implementation, ROUGE-L, and BERTScore. For ATS and NHG tasks, ROUGE-1, ROUGE-2, and ROUGE-L are used. We explicitly use community-adopted language specific-tokenizers. Links for language-specific tokenizers are given below:

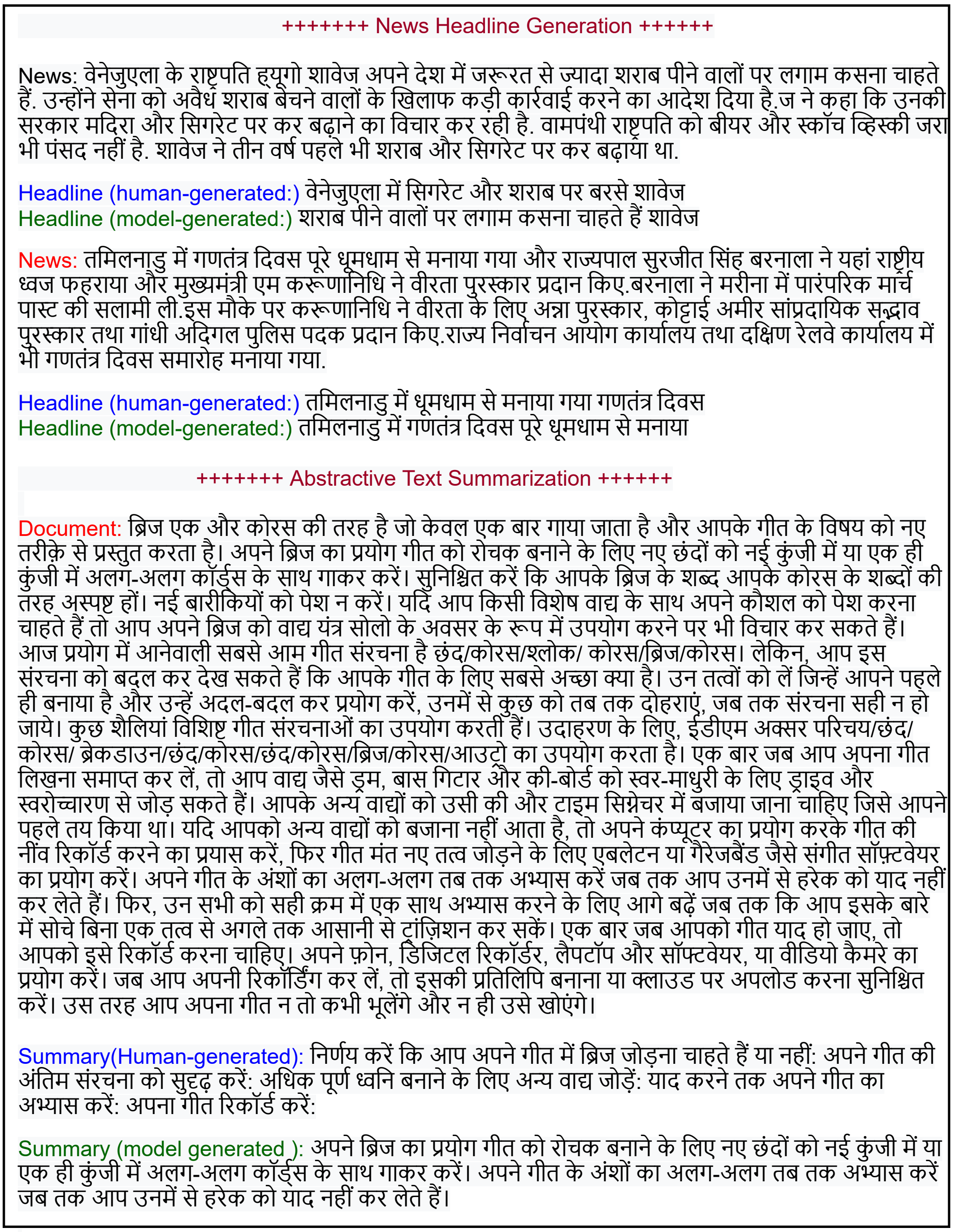
- **English:** Default sacreBLEU tokenizer i.e, https://github.com/mjpost/sacrebleu
- **Hindi:** https://anoopkunchukuttan.github.io/indic_nlp_library/
- **Japanese:** http://www.phontron.com/kytea/

Links of publicly available implementations of automated evaluation metrics which we use directly in this work:

- **BLEU:** https://github.com/mjpost/sacrebleu
- **ROUGE:** https://github.com/pltrdy/files2rouge
- **BERTScore:** https://github.com/Tiiiger/bert_score

7.3 Few Zero-shot Generated outputs from ZmBART:

In the next few figures, we present sample outputs generated by the model in zero-shot setups, for Hindi and Japanese languages.



+++++++ News Headline Generation ++++++

News: वेनेजुएला के राष्ट्रपति ह्यूगो शावेज अपने देश में जरूरत से ज्यादा शराब पीने वालों पर लगाम कसना चाहते हैं. उन्होंने सेना को अवैध शराब बेचने वालों के खिलाफ कड़ी कार्रवाई करने का आदेश दिया है.ज ने कहा कि उनकी सरकार मदिरा और सिगरेट पर कर बढ़ाने का विचार कर रही है. वामपंथी राष्ट्रपति को बीयर और स्कॉच व्हिस्की जरा भी पंसद नहीं है. शावेज ने तीन वर्ष पहले भी शराब और सिगरेट पर कर बढ़ाया था.

Headline (human-generated:) वेनेजुएला में सिगरेट और शराब पर बरसे शावेज
Headline (model-generated:) शराब पीने वालों पर लगाम कसना चाहते हैं शावेज

News: तमिलनाडु में गणतंत्र दिवस पूरे धूमधाम से मनाया गया और राज्यपाल सुरजीत सिंह बरनाला ने यहां राष्ट्रीय ध्वज फहराया और मुख्यमंत्री एम करूणानिधि ने वीरता पुरस्कार प्रदान किए.बरनाला ने मरीना में पारंपरिक मार्च पास्ट की सलामी ली.इस मौके पर करूणानिधि ने वीरता के लिए अन्ना पुरस्कार, कोट्टाई अमीर सांप्रदायिक सद्भाव पुरस्कार तथा गांधी अदिगल पुलिस पदक प्रदान किए.राज्य निर्वाचन आयोग कार्यालय तथा दक्षिण रेलवे कार्यालय में भी गणतंत्र दिवस समारोह मनाया गया.

Headline (human-generated:) तमिलनाडु में धूमधाम से मनाया गया गणतंत्र दिवस
Headline (model-generated:) तमिलनाडु में गणतंत्र दिवस पूरे धूमधाम से मनाया

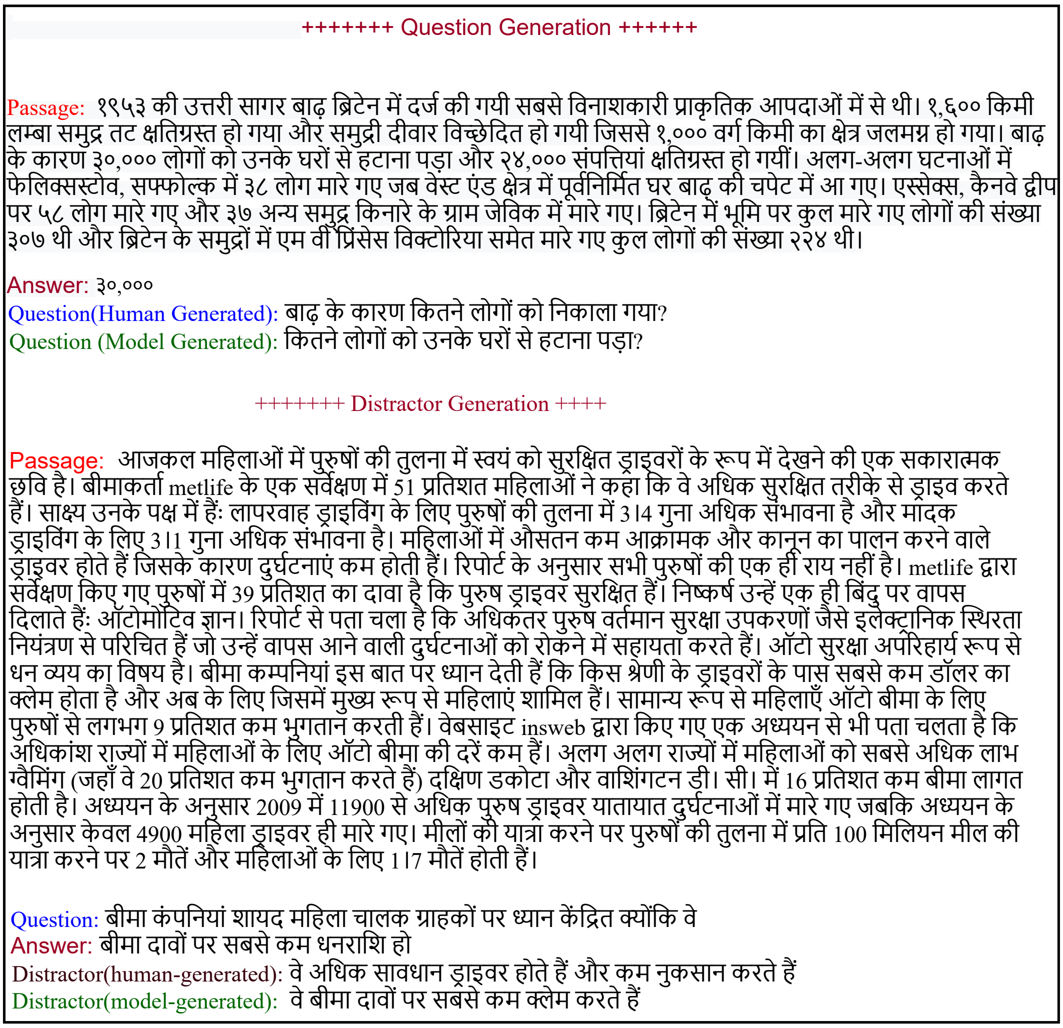
+++++++ Abstractive Text Summarization ++++++

Document: ब्रिज एक और कोरस की तरह है जो केवल एक बार गाया जाता है और आपके गीत के विषय को नए तरीक़े से प्रस्तुत करता है। अपने ब्रिज का प्रयोग गीत को रोचक बनाने के लिए नए छंदों को नई कुंजी में या एक ही कुंजी में अलग-अलग कॉर्ड्स के साथ गाकर करें। सुनिश्चित करें कि आपके ब्रिज के शब्द आपके कोरस के शब्दों की तरह अस्पष्ट हों। नई बारीकियों को पेश न करें। यदि आप किसी विशेष वाद्य के साथ अपने कौशल को पेश करना चाहते हैं तो आप अपने ब्रिज को वाद्य यंत्र सोलो के अवसर के रूप में उपयोग करने पर भी विचार कर सकते हैं। आज प्रयोग में आनेवाली सबसे आम गीत संरचना है छंद/कोरस/श्लोक/ कोरस/ब्रिज/कोरस। लेकिन, आप इस संरचना को बदल कर देख सकते हैं कि आपके गीत के लिए सबसे अच्छा क्या है। उन तत्वों को लें जिन्हें आपने पहले ही बनाया है और उन्हें अदल-बदल कर प्रयोग करें, उनमें से कुछ को तब तक दोहराएं, जब तक संरचना सही न हो जाये। कुछ शैलियां विशिष्ट गीत संरचनाओं का उपयोग करती हैं। उदाहरण के लिए, ईडीएम अक्सर परिचय/छंद/कोरस/ ब्रेकडाउन/छंद/कोरस/छंद/कोरस/ब्रिज/कोरस/आउट्रो का उपयोग करता है। एक बार जब आप अपना गीत लिखना समाप्त कर लें, तो आप वाद्य जैसे ड्रम, बास गिटार और की-बोर्ड को स्वर-माधुरी के लिए ड्राइव और स्वरोच्चारण से जोड़ सकते हैं। आपके अन्य वाद्यों को उसी की और टाइम सिग्नेचर में बजाया जाना चाहिए जिसे आपने पहले तय किया था। यदि आपको अन्य वाद्यों को बजाना नहीं आता है, तो अपने कंप्यूटर का प्रयोग करके गीत की नींव रिकॉर्ड करने का प्रयास करें, फिर गीत मंत नए तत्व जोड़ने के लिए एबलेटन या गैरेजबैंड जैसे संगीत सॉफ़्टवेयर का प्रयोग करें। अपने गीत के अंशों का अलग-अलग तब तक अभ्यास करें जब तक आप उनमें से हरेक को याद नहीं कर लेते हैं। फिर, उन सभी को सही क्रम में एक साथ अभ्यास करने के लिए आगे बढ़ें जब तक कि आप इसके बारे में सोचे बिना एक तत्व से अगले तक आसानी से ट्रांज़िशन कर सकें। एक बार जब आपको गीत याद हो जाए, तो आपको इसे रिकॉर्ड करना चाहिए। अपने फ़ोन, डिजिटल रिकॉर्डर, लैपटॉप और सॉफ्टवेयर, या वीडियो कैमरे का प्रयोग करें। जब आप अपनी रिकॉर्डिंग कर लें, तो इसकी प्रतिलिपि बनाना या क्लाउड पर अपलोड करना सुनिश्चित करें। उस तरह आप अपना गीत न तो कभी भूलेंगे और न ही उसे खोएंगे।

Summary(Human-generated): निर्णय करें कि आप अपने गीत में ब्रिज जोड़ना चाहते हैं या नहीं: अपने गीत की अंतिम संरचना को सुदृढ़ करें: अधिक पूर्ण ध्वनि बनाने के लिए अन्य वाद्य जोड़ें: याद करने तक अपने गीत का अभ्यास करें: अपना गीत रिकॉर्ड करें:

Summary (model generated): अपने ब्रिज का प्रयोग गीत को रोचक बनाने के लिए नए छंदों को नई कुंजी में या एक ही कुंजी में अलग-अलग कॉर्ड्स के साथ गाकर करें। अपने गीत के अंशों का अलग-अलग तब तक अभ्यास करें जब तक आप उनमें से हरेक को याद नहीं कर लेते हैं।

Figure 5: Sample outputs for zero-short NHG and ATS in Hindi language



+++++++ Question Generation ++++++

Passage: १९५३ की उत्तरी सागर बाढ़ ब्रिटेन में दर्ज की गयी सबसे विनाशकारी प्राकृतिक आपदाओं में से थी। १,६०० किमी लम्बा समुद्र तट क्षतिग्रस्त हो गया और समुद्री दीवार विच्छेदित हो गयी जिससे १,००० वर्ग किमी का क्षेत्र जलमग्न हो गया। बाढ़ के कारण ३०,००० लोगों को उनके घरों से हटाना पड़ा और २४,००० संपत्तियां क्षतिग्रस्त हो गयीं। अलग-अलग घटनाओं में फेलिक्सस्टोव, सफ्फोल्क में ३८ लोग मारे गए जब वेस्ट एंड क्षेत्र में पूर्वनिर्मित घर बाढ़ की चपेट में आ गए। एस्सेक्स, कैनवे द्वीप पर ५८ लोग मारे गए और ३७ अन्य समुद्र किनारे के ग्राम जेविक में मारे गए। ब्रिटेन में भूमि पर कुल मारे गए लोगों की संख्या ३०७ थी और ब्रिटेन के समुद्रों में एम वी प्रिंसेस विक्टोरिया समेत मारे गए कुल लोगों की संख्या २२४ थी।

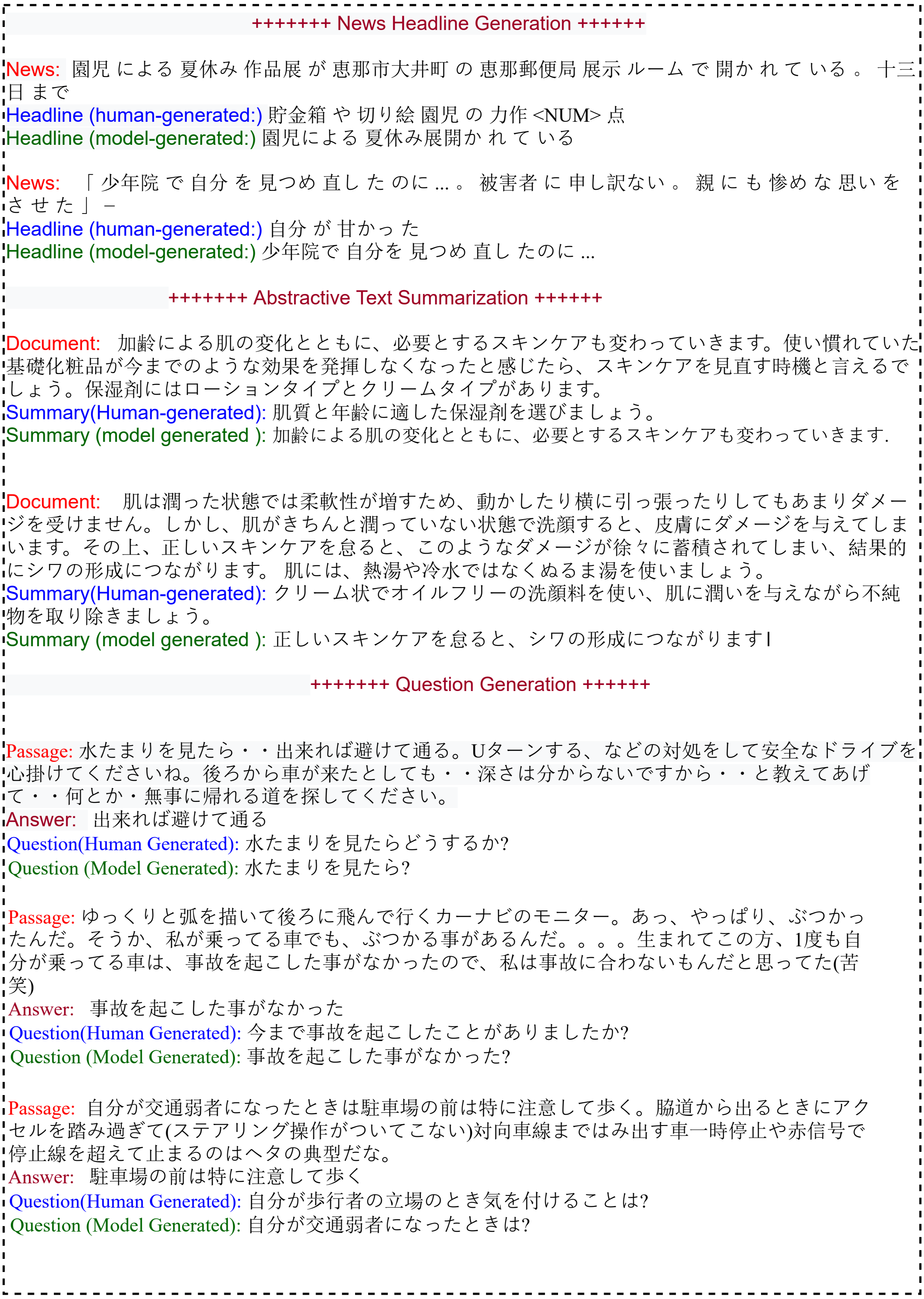
Answer: ३०,०००
Question(Human Generated): बाढ़ के कारण कितने लोगों को निकाला गया?
Question (Model Generated): कितने लोगों को उनके घरों से हटाना पड़ा?

+++++++ Distractor Generation ++++

Passage: आजकल महिलाओं में पुरुषों की तुलना में स्वयं को सुरक्षित ड्राइवरों के रूप में देखने की एक सकारात्मक छवि है। बीमाकर्ता metlife के एक सर्वेक्षण में 51 प्रतिशत महिलाओं ने कहा कि वे अधिक सुरक्षित तरीके से ड्राइव करते हैं। साक्ष्य उनके पक्ष में हैंः लापरवाह ड्राइविंग के लिए पुरुषों की तुलना में 3।4 गुना अधिक संभावना है और मादक ड्राइविंग के लिए 3।1 गुना अधिक संभावना है। महिलाओं में औसतन कम आक्रामक और कानून का पालन करने वाले ड्राइवर होते हैं जिसके कारण दुर्घटनाएं कम होती हैं। रिपोर्ट के अनुसार सभी पुरुषों की एक ही राय नहीं है। metlife द्वारा सर्वेक्षण किए गए पुरुषों में 39 प्रतिशत का दावा है कि पुरुष ड्राइवर सुरक्षित हैं। निष्कर्ष उन्हें एक ही बिंदु पर वापस दिलाते हैंः ऑटोमोटिव ज्ञान। रिपोर्ट से पता चला है कि अधिकतर पुरुष वर्तमान सुरक्षा उपकरणों जैसे इलेक्ट्रानिक स्थिरता नियंत्रण से परिचित हैं जो उन्हें वापस आने वाली दुर्घटनाओं को रोकने में सहायता करते हैं। ऑटो सुरक्षा अपरिहार्य रूप से धन व्यय का विषय है। बीमा कम्पनियां इस बात पर ध्यान देती हैं कि किस श्रेणी के ड्राइवरों के पास सबसे कम डॉलर का क्लेम होता है और अब के लिए जिसमें मुख्य रूप से महिलाएं शामिल हैं। सामान्य रूप से महिलाएँ ऑटो बीमा के लिए पुरुषों से लगभग 9 प्रतिशत कम भुगतान करती हैं। वेबसाइट insweb द्वारा किए गए एक अध्ययन से भी पता चलता है कि अधिकांश राज्यों में महिलाओं के लिए ऑटो बीमा की दरें कम हैं। अलग अलग राज्यों में महिलाओं को सबसे अधिक लाभ ग्वैमिंग (जहाँ वे 20 प्रतिशत कम भुगतान करते हैं) दक्षिण डकोटा और वाशिंगटन डी। सी। में 16 प्रतिशत कम बीमा लागत होती है। अध्ययन के अनुसार 2009 में 11900 से अधिक पुरुष ड्राइवर यातायात दुर्घटनाओं में मारे गए जबकि अध्ययन के अनुसार केवल 4900 महिला ड्राइवर ही मारे गए। मीलों की यात्रा करने पर पुरुषों की तुलना में प्रति 100 मिलियन मील की यात्रा करने पर 2 मौतें और महिलाओं के लिए 1।7 मौतें होती हैं।

Question: बीमा कंपनियां शायद महिला चालक ग्राहकों पर ध्यान केंद्रित क्योंकि वे
Answer: बीमा दावों पर सबसे कम धनराशि हो
Distractor(human-generated): वे अधिक सावधान ड्राइवर होते हैं और कम नुकसान करते हैं
Distractor(model-generated): वे बीमा दावों पर सबसे कम क्लेम करते हैं

Figure 6: Sample outputs for zero-short QG and DG in Hindi language



+++++++ News Headline Generation ++++++

News: 園児 による 夏休み 作品展 が 恵那市大井町 の 恵那郵便局 展示 ルーム で 開か れ て いる 。 十三日 まで
Headline (human-generated:) 貯金箱 や 切り絵 園児 の 力作 <NUM> 点
Headline (model-generated:) 園児による 夏休み展開か れ て いる

News: 「 少年院 で 自分 を 見つめ 直し た のに ... 。 被害者 に 申し訳ない 。 親 に も 惨め な 思い を さ せ た 」 –
Headline (human-generated:) 自分 が 甘かっ た
Headline (model-generated:) 少年院で 自分を 見つめ 直し たのに ...

+++++++ Abstractive Text Summarization ++++++

Document: 加齢による肌の変化とともに、必要とするスキンケアも変わっていきます。使い慣れていた基礎化粧品が今までのような効果を発揮しなくなったと感じたら、スキンケアを見直す時機と言えるでしょう。保湿剤にはローションタイプとクリームタイプがあります。
Summary(Human-generated): 肌質と年齢に適した保湿剤を選びましょう。
Summary (model generated): 加齢による肌の変化とともに、必要とするスキンケアも変わっていきます.

Document: 肌は潤った状態では柔軟性が増すため、動かしたり横に引っ張ったりしてもあまりダメージを受けません。しかし、肌がきちんと潤っていない状態で洗顔すると、皮膚にダメージを与えてしまいます。その上、正しいスキンケアを怠ると、このようなダメージが徐々に蓄積されてしまい、結果的にシワの形成につながります。 肌には、熱湯や冷水ではなくぬるま湯を使いましょう。
Summary(Human-generated): クリーム状でオイルフリーの洗顔料を使い、肌に潤いを与えながら不純物を取り除きましょう。
Summary (model generated): 正しいスキンケアを怠ると、シワの形成につながります I

+++++++ Question Generation ++++++

Passage: 水たまりを見たら・・出来れば避けて通る。Uターンする、などの対処をして安全なドライブを心掛けてくださいね。後ろから車が来たとしても・・深さは分からないですから・・と教えてあげて・・何とか・無事に帰れる道を探してください。
Answer: 出来れば避けて通る
Question(Human Generated): 水たまりを見たらどうするか?
Question (Model Generated): 水たまりを見たら?

Passage: ゆっくりと弧を描いて後ろに飛んで行くカーナビのモニター。あっ、やっぱり、ぶつかったんだ。そうか、私が乗ってる車でも、ぶつかる事があるんだ。。。。生まれてこの方、1度も自分が乗ってる車は、事故を起こした事がなかったので、私は事故に合わないもんだと思ってた(苦笑)
Answer: 事故を起こした事がなかった
Question(Human Generated): 今まで事故を起こしたことがありましたか?
Question (Model Generated): 事故を起こした事がなかった?

Passage: 自分が交通弱者になったときは駐車場の前は特に注意して歩く。脇道から出るときにアクセルを踏み過ぎて(ステアリング操作がついてこない)対向車線まではみ出す車一時停止や赤信号で停止線を超えて止まるのはヘタの典型だな。
Answer: 駐車場の前は特に注意して歩く
Question(Human Generated): 自分が歩行者の立場のとき気を付けることは?
Question (Model Generated): 自分が交通弱者になったときは?

Figure 7: Sample outputs for zero-shot NHG, ATS and QG in Japanese language