Knowledge Acquisition for Human-In-The-Loop Image Captioning

Ervine Zheng **Qi Yu** **Rui Li** **Pengcheng Shi** **Anne Haake**
Rochester Institute of Technology

Abstract

Image captioning offers a computational process to understand the semantics of images and convey them using descriptive language. However, automated captioning models may not always generate satisfactory captions due to the complex nature of the images and the quality/size of the training data. We propose an interactive captioning framework to improve machine-generated captions by keeping humans in the loop and performing an online-offline knowledge acquisition (KA) process. In particular, online KA accepts a list of keywords specified by human users and fuses them with the image features to generate a readable sentence that captures the semantics of the image. It leverages a multimodal conditioned caption completion mechanism to ensure the appearance of all user-input keywords in the generated caption. Offline KA further learns from the user inputs to update the model and benefits caption generation for unseen images in the future. It is built upon a Bayesian transformer architecture that dynamically allocates neural resources and supports uncertainty-aware model updates to mitigate overfitting. Our theoretical analysis also proves that Offline KA automatically selects the best model capacity to accommodate the newly acquired knowledge. Experiments on real-world data demonstrate the effectiveness of the proposed framework.

1 INTRODUCTION

Image captioning is the process of understanding image semantics and generating text descriptions. The former entails extracting features of entities, scenes, and their interactions from an image, while the latter involves integrating the extracted features and generating readable captions. Existing works usually focus on developing automated models, where images are the sole input to the model at the inference stage [Hossain et al., 2019, Al Sulaimi et al., 2021]. However, automated models may make errors, which leads to inaccurate captions. Besides, when processing images with multiple entities and complex scenes, the focus of automated models may deviate from the focus of a human user, resulting in unsatisfactory captions. In those cases, automated models do not allow users to control the caption generation. It hinders the broader applicability of these models, especially in critical domains (*e.g.,* medicine, security).

Interactive image captioning offers a solution to the above issues by involving humans in the loop. Specifically, users can provide a sequence of keywords (*i.e.,* informative words that capture the image's important semantics) as additional input to guide the caption generation process. Compared with fully automated captioning, interactive captioning is under-explored with only a few existing works [Zhang et al., 2017, Jia and Li, 2020, Huang et al., 2021]. We provide a high-level summary of existing approaches in Figure 1. Specifically, the bilateral caption generation [Zhang et al., 2017] approach predicts words before and after the user-input keywords. However, user inputs must be consecutive words, limiting the flexibility of user interaction. The contextualized keyword encoding [Huang et al., 2021] approach encodes user-input keywords as the loose guidance for sentence generation. However, it does not guarantee all user-input words appear in the generated caption. The double-ended keyword encoding [Jia and Li, 2020] approach allows a user to specify multiple words at the beginning and the end of a sentence and leverages a sequential decoding network to predict other words. However, words at the beginning of a sentence may be non-informative (*e.g.,*'a', 'the'). In addition, it cannot guarantee that the user-input words at the end will appear in the generated caption.

For human-in-the-loop image captioning, we identify two challenges: 1) Acquiring knowledge from users through effective and natural human-machine interactions. Ideally, an interactive model should allow users to control caption generation with minimum effort. However, existing approaches suffers several limitations, as discussed above. 2) Encoding such knowledge to improve the model's performance in fu-

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206. Copyright 2023 by the author(s).

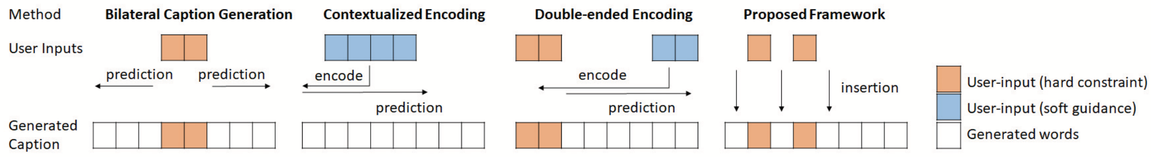


Figure 1: Illustration of interactive caption generation approaches given user-input words. Compared with existing methods (*e.g.,* Bilateral caption generation, Contextualized keyword encoding, Double-ended keyword encoding), the proposed framework allows users to specify keywords at any position, and user-input words are guaranteed to appear in the generated caption.

ture captioning tasks. Ideally, an interactive model should be able to learn from the users. However, existing approaches mainly focus on the current image for user interaction. The knowledge is not encoded into model parameters. Given new images, the model always requires the aid from users to generate satisfactory captions, which may incur an excessive burden.

To address those challenges, we propose a *knowledge acquisition framework for interactive captioning* that significantly improves the knowledge acquisition (KA) process from users. As shown in Figure 2, the whole process includes two stages: 1) *Online KA*, where the user interacts with the model by specifying several keywords, which are leveraged as additional inputs and fused with the image data to generate a complete caption. 2) *Offline KA*, where the keywords and generated captions are considered weak supervision to update model parameters. The updated model can generate better captions in the future even without aid from users and ultimately reduce users' burden. An illustrative example is provided in Figure 3.

For online KA, the proposed model introduces a multimodal conditioned caption completion (MC^3) mechanism, which leverages user-input keywords as a starting point and inserts other words to formulate a complete caption. The proposed MC^3 module also makes extensions to existing captioning methods to generate controllable captions. In prior works, user-input keywords are restricted to specific positions[Jia and Li, 2020], or treated as soft guidance rather than hard constraint[Huang et al., 2021], which limits the user's flexibility for knowledge sharing. In contrast, the proposed MC^3 module generates sentences through insertion operations, which guarantees user-input words to appear in the completed sentence as a hard constraint, and thus provides an effective way for users to control the caption generation process. In addition, the proposed model does not have positional restrictions on keywords so that users can focus on selecting the most informative keywords.

For offline KA, the keywords and generated captions are used to update the model. To support effective model updates, we propose an expandable Bayesian transformer as the building block, which introduces extensions to the conventional transformer architecture [Vaswani et al., 2017]. The conventional architecture uses the dot-product attention and feedforward layers to process the input sequence. The proposed Bayesian transformer introduces an Indian buffet process (IBP) mask to control the attended embeddings by the attention layer and the activated neurons in the feedforward layer. During offline KA, the model is updated to encode the knowledge from the user, and an IBP mask will dynamically allocate additional neural resources. In addition, the Bayesian transformer also supports uncertainty estimation, which is used to filter out the noises and mitigate overfitting during model updates. Therefore, the proposed framework can effectively learn from the user to enhance its capability of predicting captions in the future. Our theoretical analysis (See Theorem 1) shows that dynamic neural resource allocation optimally adjusts the model capacity to accommodate the newly acquired knowledge according to the Bayesian Information Criterion (BIC).

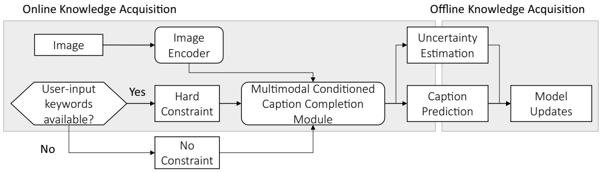

Online Knowledge Acquisition
Offline Knowledge Acquisition
Image
Image Encoder
Uncertainty Estimation
User-input keywords available?
Yes
Hard Constraint
Multimodal Conditioned Caption Completion Module
Caption Prediction
Model Updates
No
No Constraint

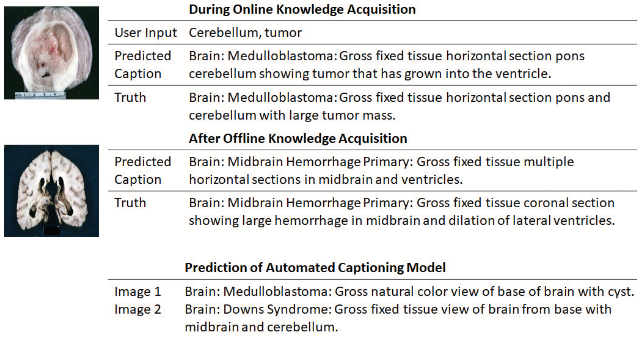
Figure 2: Schematic view of the proposed framework. For online KA, the model takes user-input keywords as hard constraints to generate captions (If keywords are unavailable, the model automatically predicts captions with no constraints). The model leverages offline KA to update parameters and encode user knowledge.

Our main contributions are: (i) a knowledge acquisition framework for human-in-the-loop image captioning, where knowledge queries are formulated as user-input keywords to support effective knowledge sharing; (ii) an online KA process that treats user-input words as a hard constraint in caption generation while providing flexibility for user interaction; (iii) an offline KA mechanism to effectively learn from users to improve the model performance in the future.

2 RELATED WORKS

We provide an overview of existing works that are most relevant to the proposed work.

Image captioning. Image captioning is an integrated task of computer vision and natural language processing, and existing models typically include a vision encoder and language



	During Online Knowledge Acquisition
User Input	Cerebellum, tumor
Predicted Caption	Brain: Medulloblastoma: Gross fixed tissue horizontal section pons cerebellum showing tumor that has grown into the ventricle.
Truth	Brain: Medulloblastoma: Gross fixed tissue horizontal section pons and cerebellum with large tumor mass.
	After Offline Knowledge Acquisition
Predicted Caption	Brain: Midbrain Hemorrhage Primary: Gross fixed tissue multiple horizontal sections in midbrain and ventricles.
Truth	Brain: Midbrain Hemorrhage Primary: Gross fixed tissue coronal section showing large hemorrhage in midbrain and dilation of lateral ventricles.
	Prediction of Automated Captioning Model
Image 1	Brain: Medulloblastoma: Gross natural color view of base of brain with cyst.
Image 2	Brain: Downs Syndrome: Gross fixed tissue view of brain from base with midbrain and cerebellum.

Figure 3: An illustrative example. Given an image, the user interacts with the model by providing several keywords, which are used to generate a complete caption via online KA. The results are used to update the model via offline KA. Given a new image, the updated model can generate better captions without the aid of users. In contrast, an automated model may generate less satisfactory captions.

model. In recent years, attention-based image captioning models are becoming popular because they consider the spatial aspects of the image and attend to different regions during the caption generation process. For vision encoder, existing works develops region-based attention [Ge et al., 2019, Cornia et al., 2018], graph-based attention [Yao et al., 2019, Shi et al., 2020], patch-based attention [Huang et al., 2019, Pan et al., 2020], etc. For language models, popular architectures include recurrent neural network [Huang et al., 2019, Zhu et al., 2020], transformer [Guo et al., 2020b, Luo et al., 2021], etc. Different from automated captioning, interactive captioning relies on multimodal text generation, where the image captioning is conditioned on both the image and the user inputs. Interactive image captioning is a relatively under-explored research topic. Existing works leverage keywords as additional input to generate captions with diverse contextual emphasis, which can be applied in an interactive setting. [Zhang et al., 2017] proposes a bilateral LSTM network to generate words before and after the user-input keyword. However, user input is restricted to adjacent words. To improve the flexibility of user interaction, [Jia and Li, 2020] allows a user to specify multiple words at the beginning and the end of a sentence and leverages an asynchronous bidirectional decoding network to complete the rest of the words. [Huang et al., 2021] encodes multiple keywords through a contextualized encoder as the loose guidance for sentence generation. However, they cannot guarantee all user-input words to appear in the generated caption. In recent years, constrained text generation models have been explored in natural language processing. Constrained Beam search [Anderson et al., 2017, Hokamp and Liu, 2017] typically leverages a finite-state machine to keep several states of enforcing the constraints over resulting output sequences. However, a large number of states may incur high computational costs. [Miao et al., 2019] involves a sampling mechanism where the constraints are placed in a template, and other words are added through sampling. However, the quality of generated caption is heavily dependent on the templates.

Transfer Learning. The offline knowledge acquisition involves model updates to encode the new knowledge learned from the user into model parameters. It is related to transfer learning, where a pre-trained model is finetuned so that it can perform better in specific tasks [Zhuang et al., 2020]. A widely-used baseline is finetuning an entire pre-trained model, but it requires keeping a new set of network weights for each task. For parameter-efficient model updates, one way is to sparsely update a small subset of parameters of the model [Guo et al., 2020a, Sung et al., 2021]. Another way is to add new parameters to the input or model [Karimi Mahabadi et al., 2021, Sung et al., 2022], and the Adapter [Houlsby et al., 2019] is a representative approach widely used in natural language processing and computer vision. The Adapter is a small residual module plugged into intermediate layers of the model to allow finetuning only a small set of parameters. However, for existing approaches, the user has to manually determine the size of additional neural resources allocated for finetuning. In contrast, the proposed method leverages an Indian buffet process (IBP) mask to dynamically allocate additional neural resources to encode new knowledge from users.

Knowledge acquisition. Knowledge acquisition is a process of acquiring information from human or other sources and formalizing information structure to perform certain tasks [Kidd, 2012]. For image captioning, [Zhou et al., 2019, Huang et al., 2020] propose to extract knowledge from external corpora and construct a graph network. [Chen et al., 2021a, Cao et al., 2020] propose to leverage linguistic knowledge transferred from the large-scale pretraining. In summary, prior works focus on KA from static external sources, while another important knowledge source (*i.e.*, human users) is overlooked. Our work aims to fill this gap.

Bayesian neural network In Bayesian neural network (BNN) [Goan and Fookes, 2020], the layer weights and network outputs are treated as the variables, and the goal of training is to find the marginal distributions that best fit the data. A drawback of the Bayesian setting is the increased trainable parameters. However, it should be noted that the image encoder typically uses pre-trained weights from off-the-shelf models. It does not have the Bayesian setting or additional parameters.

3 METHODOLOGY

The workflow of the proposed framework includes two stages: *online KA* and *offline KA*. We first summarize the problem formulation, and then provide details about the two stages. Main notations and the network architecture are given in Appendices A and C.

3.1 Problem Formulation

Given the input image denoted as I^{raw}, a user may choose to provide several keywords denoted as $Z = [z_1, z_2, ..., z_M]$. The interactive captioning model takes those keywords as additional inputs and generates a complete caption $\hat{Y}$. The above process is referred to as online KA. Moreover, the model should ideally learn from the user to perform better in the future. To this end, we also consider $\hat{Y}$ as weak supervision to update the model. After that, if a new image I^{new} in the same specialized domain is provided, the model should automatically generate satisfactory captions as $\hat{Y}^{new}$ even without aid from the user. The above process is referred to as offline KA.

3.2 Online Knowledge Acquisition

We consider encoder-decoder, a typical architecture for image captioning models with an encoder to extract image features and a decoder to generate textual descriptions. For the encoder, we leverage a pre-trained network as the image feature extractor to generate a three-dimensional feature map I, which is a compact representation of the input image I^{raw}.

$$I = \texttt{Encoder}(I^{raw}) \tag{1}$$

For the rest of the paper, we let $f(\cdot)$ denote the transformation through a feed-forward layer, $f^{\text{ReLU}}(\cdot)$ for additional ReLU activation, $f^{\text{softmax}}(.)$ for softmax activation, $f^{\text{Em}}(\cdot)$ for embedding layer, and $f^{\text{Ln}}(\cdot)$ for layer normalization.

We use an Bayesian transformer as the backbone of the decoder. Specifically, each word x_n in the current incomplete caption and its position n are transformed into an embedding vector $\mathbf{b}_n = f^{\text{Em}}(x_n) + f^{\text{Em}}(n)$.

$\mathbf{b}_n$ is then processed by the expandable Bayesian transformer for decoding. The proposed framework dynamically allocates neural resources to encode ever-growing knowledge learned from users. To this end, we leverage the expandable embedding and the expandable feedforward layer. Intuitively, the expandable embeddings are learnable embedding vectors of pseudo tokens, which contain additional information not captured by actual word tokens. The model can attend to both the pseudo tokens' embeddings and actual words' embeddings. The pseudo tokens are organized as a two-dimensional matrix S. Each column of S is considered a pseudo token that the model can selectively attend to, and a column-wise mask $\boldsymbol{v}$ is deployed to control whether a specific column (*i.e.,* pseudo token) can be attended. The output is

$$\beta_{r,c} = S_{r,c} v_c \tag{2}$$

where (r, c) are row and column indices. The expandable embeddings have a Bayesian setting. Every entry of S has a Gaussian prior:

$$S_{r,c} \sim \mathcal{N}(\mu_0, \sigma_0) \tag{3}$$

and the mask $\boldsymbol{v}$ has an Indian buffet process (IBP) prior [Ghahramani and Griffiths, 2006]

$$u_c \sim \text{Beta}(a_0, b_0),\ \pi_c = \prod\nolimits_{c'=1}^{c} u_{c'},\ v_c \sim \text{Bern}(\pi_c) \tag{4}$$

For simplicity, Eq (4) can be re-written as

$$v_c \sim \text{IBP}(a_0, b_0) \tag{5}$$

Intuitively, if $v_c = 1$, the model's attention to column c is enabled, and if $v_c = 0$ the attention is disabled. The expandable embedding offers a few benefits: 1) It allows encoding additional information not captured in actual words' embeddings; 2) It encourages only a few columns to be enabled for attention at the beginning, but more columns to be enabled when the model acquires knowledge, and thus essentially allocates more neural resources. We then transform $\mathbf{b}$ and $\boldsymbol{\beta}$ into query, key, and value variables:

$$\begin{aligned} \mathbf{c}^Q &= W^Q \mathbf{b}, \quad \mathbf{c}^K = W^K \text{concat}[\mathbf{b}, \boldsymbol{\beta}], \\ \mathbf{c}^V &= W^V \text{concat}[\mathbf{b}, \boldsymbol{\beta}] \end{aligned} \tag{6}$$

where $\{W^Q, W^K, W^V\}$ are weight matrices, and concat$[\cdot, \cdot]$ denotes concatenation. A dot-product self attention with normalization is applied in a similar way to the conventional transformer.

$$\begin{aligned} \tilde{\mathbf{o}}_n &= \text{selfAttn}(\text{concat}[\mathbf{b}, \boldsymbol{\beta}]) \\ &= f^{\text{Ln}}(\text{softmax}(\frac{(\mathbf{c}_n^Q)^\top \mathbf{c}^K}{\sqrt{d}})\mathbf{c}^V + \mathbf{b}_n) \end{aligned} \tag{7}$$

where d is the length of vector. Intuitively, Eqs (6) integrates the semantic information of a word with other words and the expandable embeddings through the attention from $\mathbf{c}^Q$ to $\mathbf{c}^V$ and $\mathbf{c}^K$. $\tilde{\mathbf{o}}_n$ is then processed by an expandable feedforward layer f^{ex}. The expandable feedforward layer also introduces a Bayesian setting. Denote the output of dot-product attention as $\tilde{o}_n$, the layer's weight matrix as W and column-wise mask as $\boldsymbol{\zeta}$, the output is calculated as

$$\begin{aligned} & f(\tilde{\mathbf{o}}_n) = (W \circ \boldsymbol{\zeta})\tilde{\mathbf{o}}_n \\ & \text{where } W_{r,c} \sim N(\theta_0, s_0), \quad \zeta_c \sim \text{IBP}(\alpha_0, \beta_0) \end{aligned} \tag{8}$$

where $\circ$ denotes column-wise multiplication. Again, $\boldsymbol{\zeta}$ controls the dynamic neural resource allocation. If $\zeta_c = 0$, the c-th column of the weight matrix W is deactivated and makes no contribution to the output. As more knowledge is acquired, additional columns will be activated through model updates. The output $\mathbf{o}_n$ of the expandable Bayesian transformer is calculated as

$$\mathbf{o}_n = f^{\text{Ln}}(\tilde{\mathbf{o}}_n(1 + f(\tilde{\mathbf{o}}_n))) \tag{9}$$

Multimodal Conditioned Caption Completion (MC^3) In an interactive setting, the user provides keywords to guide the model to generate captions. Ideally, user-input

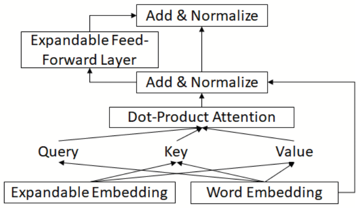
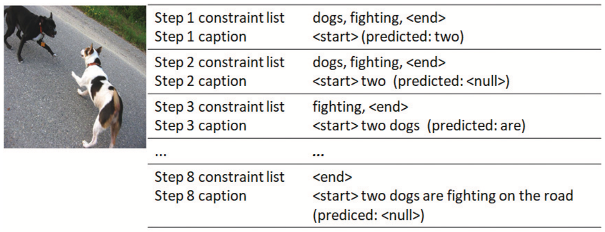


Figure 4: The expandable Bayesian transformer



Step 1 constraint list Step 1 caption	dogs, fighting, <end> <start> (predicted: two)
Step 2 constraint list Step 2 caption	dogs, fighting, <end> <start> two (predicted: <null>)
Step 3 constraint list Step 3 caption	fighting, <end> <start> two dogs (predicted: are)
...	...
Step 8 constraint list Step 8 caption	<end> <start> two dogs are fighting on the road (prediced: <null>)

Figure 5: An example of caption completion

keywords shall appear in the generated captions. We design the MC^3 module with insertion operations to achieve this goal. It takes user-input keywords as hard constraints and keeps inserting words until the output is a complete caption. Specifically, during caption generation, we maintain two sequences: the incomplete caption and the constraint list. Two special tokens are introduced as '<start>' and '<end>', denoting the start and end of a caption. The initial incomplete caption contains only the '<start>' token. The initial constraint list contains all user-input keywords, and it is then appended by the '<end>' token for convenience. The model sequentially predicts the next word to be inserted between the current incomplete caption and the first word in the constraint list. We also introduce another special token '<null>', which indicates no actual words to be inserted into the slot. Once the model predicts '<null>', the first word in the constraint list is added to the incomplete caption. Since this word as a constraint is satisfied, it is removed from the constraint list. The model keeps predicting the words to be inserted until the constraint list is empty. An illustrative example of insertion-based caption completion is provided in Figure 5.

Denote the incomplete caption at the current iteration as X and the constraint list as $Z = [z_1, ..., z_M]$ where z_m is the m-th keyword. The inserted word $\hat{y}$ is predicted by

$$\hat{y} = \mathtt{MC}^3(I, X, Z) \tag{10}$$

Eq. (10) summarizes the process at a high level, and the details are provided as follows. First, each word x_n in the current incomplete caption, along with its position n is transformed into an embedding vector $\mathbf{b}_n$. We use the Bayesian transformer as the backbone for decoding.

$$\mathbf{o_n} = \text{selfAttn}(\mathbf{b}_n) \tag{11}$$

Second, multimodal information (user-input keywords and the image) are integrated by cross-attention.

$$\begin{aligned} \boldsymbol{\omega}_n &= \text{crossAttn}(\mathbf{o}_n, \tilde{Z}), \\ \text{where } \tilde{Z} &= \text{concat}[f^{\text{Em}}(Z), I] \end{aligned} \tag{12}$$

Intuitively, $\tilde{Z}$ manifests all the information from user-input keywords Z and the image I. The calculation of crossAttn$(\mathbf{o}, \tilde{Z})$ is generally similar to self-attention, but the keys, values are calculated using $\tilde{Z}$. The model predicts words based on probability

$$p(y_n|Y) = f^{\text{softmax}}(f^{\text{ReLU}}(\boldsymbol{\omega}_n)) \tag{13}$$

Training of MC^3. We use public image-captioning datasets that contain images and ground-truth captions and extract keywords to prepare (image, keywords, caption) tuple to train MC^3. The proposed model uses the Bayesian network and let $\mathbf{\Phi} = \{\boldsymbol{W}, \boldsymbol{S}, \boldsymbol{v}, \boldsymbol{\zeta}\}$ denote the trainable weights. The training of Bayesian layers aims to optimize the posterior distribution of the weights, denoted as $q(\mathbf{\Phi})$. During the forward pass, the weights of layer i are randomly sampled from $q(\boldsymbol{\phi}_i)$, and the corresponding operation is performed to map the layer's input to output. During back-propagation, the gradients with respect to the posterior parameters are propagated.

The training of the model requires the inference of posterior parameters for the corresponding layers. Since the exact inference of the posterior is usually intractable, a variational distribution q is introduced to approximate those posteriors. Let Y denote the ground-truth captions, and Z denote the ordered keywords extracted from Y. For variational inference, $q(\mathbf{\Phi})$ is factorized as

$$\begin{aligned} q(\mathbf{\Phi}) = &\prod\nolimits_i q(S^i) \prod\nolimits_{i'} q(W^{i'}) \\ &\prod\nolimits_j q(u^j) q(v^j|u^j) \prod\nolimits_{j'} q(\epsilon^{j'}) q(\zeta^{j'}|\epsilon^{j'}) \end{aligned} \tag{14}$$

The overall objective function is defined as the negative evidence lower bound

$$\begin{aligned} L^{\text{train}} =& KL[q(\mathbf{\Phi})||p(\mathbf{\Phi})] - E_{q(\mathbf{\Phi})}[\ln p(Y|\mathbf{\Phi}, Z, X)] \\ &- E_{q(\mathbf{\Phi})}[\ln p(Z|\mathbf{\Phi}, X)] \end{aligned} \tag{15}$$

where the first term is the Kullback–Leibler divergence that quantifies the difference between the prior and posterior distribution, and the other terms are the expectation of the log-likelihood empirically estimated via Monte-Carlo sampling. During the forward pass, the Beta distribution is relaxed as the Kumaraswamy distribution [Kumaraswamy, 1980] and the Bernoulli distribution is relaxed as the concrete Bernoulli distribution [Maddison et al., 2016]. $KL[q(\mathbf{\Phi})||p(\mathbf{\Phi})]$ can be expanded as the sum of Kullback–Leibler divergence terms of $\{\boldsymbol{W}, \boldsymbol{S}, \boldsymbol{v}, \boldsymbol{\zeta}\}$. The term $KL(q(S^i)||p(S^i))$ can be expanded as

$$
\begin{aligned}
&KL(q(S^i)||p(S^i)) \\
&= \sum_{r,c}[\ln\sigma_0 - \ln\sigma^i_{r,c} + \frac{(\sigma^i_{r,c})^2 + (\mu^i_{r,c} - \mu_0)^2}{2(\sigma_0)^2} - \frac{1}{2}]
\end{aligned}
\tag{16}
$$

where $(\mu^i_{r,c}, \sigma^i_{r,c})$ are the posterior parameters. The term $KL(q(u^j)||p(u^j))$ can be expanded as

$$
\begin{aligned}
&KL(q(u^j)||p(u^j)) \\
&= \sum_c \ln\frac{B(a_0, b_0)}{B(a^j_c, b^j_c)} + (a^j_c - a_0)\psi(a^j_c) + (b^j_c - b_0)\psi(b^j_c) \\
&\quad + (a_0 + b_0 - a^j_c - b^j_c)\psi(a^j_c + b^j_c)
\end{aligned}
\tag{17}
$$

where B and ψ denote beta and digamma functions, respectively, and (a^j_c, b^j_c) are the posterior parameters. The term $KL(q(v^j|u^j)||p(v^j|u^j)$ can be expanded as

$$
\begin{aligned}
&KL(q(v^j|u^j)||p(v^j|u^j) = \sum_c \kappa^j_c(\ln\kappa^j_c - \ln\pi^j_c) \\
&\quad + (1 - \ln\kappa^j_c)(\ln(1 - \ln\kappa^j_c) - \ln(1 - \ln\pi^j_c))
\end{aligned}
\tag{18}
$$

where κ^j_c is the posterior parameter. Other terms can be expanded in the same way.

3.3 Off-Line Knowledge Acquisition

It would be ideal if the model could learn from the user and perform better in the future. To this end, we introduce offline KA to perform uncertainty-aware model updates. There are several issues to be addressed. 1) Although the user provides keywords for the corresponding images during online KA, no ground truth captions are provided. 2) The images that receive user interaction are typically on a small scale, which incurs the challenge of retraining the model while mitigating overfitting.

To address the first challenge, we leverage the keywords and the predicted captions as weak supervision. However, predicted captions are noisy, and the model is not equally confident in predicting each word in the caption. To this end, we leverage uncertainty estimation and down-weigh the predicted words with high uncertainty during retraining. The Bayesian architecture provides a natural way to quantify uncertainty. At inference, we sample network weights from the posterior distribution $q(\mathbf{\Phi})$ to generate multiple Monte-Carlo (MC) samples of predictions. The uncertainty of the n-th predicted word is evaluated using the entropy:

$$
u_n = -(\bar{\boldsymbol{\xi}}_n)^\top \ln\bar{\boldsymbol{\xi}}_n, \text{ where } \bar{\boldsymbol{\xi}}_{\boldsymbol{n}} = 1/J \times \sum\nolimits_{j=1}^{J} \mathbf{p}^j(\hat{y})
\tag{19}
$$

where $\mathbf{p}^j(\cdot)$ denotes the predicted probability from one MC sample, and $\bar{\boldsymbol{\xi}}_n$ is the averaged probability vector, which can be considered as uncertainty-augmented probability. If the predicted probability vector is consistently far from some one-hot vector over multiple MC samples, it implies a high u_n. In a different situation, if the model provides inconsistent predictions over multiple MC samples, it also results in a high u_n, indicating the model is uncertain about its predictions due to inconsistency.

In addition, an image can usually be described in multiple ways. To encourage diversification, we leverage multiple caption candidates generated by the Beam search for model updates. (Beam search is an algorithm to generate a sentence word by word while keeping a fixed beam of candidate sentences at each step) Specifically, in the Beam search at online KA, we can record multiple predicted captions in the Beam as $\{\hat{Y}_1, \hat{Y}_2, ...\hat{Y}_B\}$ where B is the Beam size.

To address the second challenge, we propose to leverage the Bayesian architecture to preserve the existing knowledge learned from pre-training and encode the new knowledge learned from the user. Recall that during the pre-training, the model learns the posterior distribution of MC^3, denoted as $q(\mathbf{\Phi})$. During offline KA, $q(\mathbf{\Phi})$ is considered as the prior knowledge (*i.e.,* prior distribution), and our goal is to infer the updated posterior distribution $q(\mathbf{\Phi}^{new})$ while using $q(\mathbf{\Phi})$ as regularization. In addition, the expandable Bayesian transformer allows new neural resources to be allocated to the expandable embeddings and the expandable feedforward layers. The loss function used for retraining is

$$
\begin{aligned}
L =& KL[q(\mathbf{\Phi}^{new})||q(\mathbf{\Phi})] \\
& - E_{q(\mathbf{\Phi})}\left[\sum_{b,n}\frac{\lambda_{b,n}\ln p(\hat{y}_{b,n}|\mathbf{\Phi}, X, Z)}{NB}\right] \\
\lambda_{b,n} &= 1 - u_{b,n}/\ln V
\end{aligned}
\tag{20}
$$

where KL denotes KL divergence. $\hat{y}_{b,n}$ is the of n-th word in b-th candidate caption, $u_{b,n}$ is the corresponding entropy, and V is the size of vocabulary. $\lambda_{b,n}$ is the weighting factor with $\lambda_{b,n} \to 1$ for confident predictions and $\lambda_{b,n} \to 0$ for uncertain predictions. N is the length of a caption.

Automated Prediction Without user interaction, the proposed framework can make automated caption generation. In this case, the constraint list only contains '<end>' token (no user-input keywords). Therefore, a complete caption can be predicted by the proposed MC^3 in the same way.

3.4 Theoretical Analysis

The proposed framework leverages the Indian buffet process to dynamically allocate neural resources to make the model learn from users. The mask variable controls whether a neuron is activated or not. Our theoretical result below shows that when optimizing the loss in Eq. (20), we also select the best model based on the Bayesian Information Criterion (BIC).

Theorem 1. *Denote W^* as the optimized network parameters, $\boldsymbol{\zeta}$ and $\boldsymbol{v}$ as the corresponding mask variables of* *the expandable feedforward layer and expandable embedding. Optimizing the evidence lower bound is equivalent to* $BIC = \ln p(y|\boldsymbol{\zeta}, \boldsymbol{v}, W^*) - \frac{1}{2}(A_f + A_e)\ln D$*, where* D *is the number of data instances and* A_f *and* A_e *are the number of activated neurons in expandable feedforward layer and expandable embedding.*

Proof Sketch. If the mask variables are close to zero, the corresponding neurons do not affect the prediction and can be pruned. Therefore, we gather all activated neurons from the expandable feedforward layer and the expandable embedding, and flatten as θ. The unnormalized log posterior can be approximated at an optimized θ^* as

$$\ln[p(y|\theta)p(\theta)] \approx \ln[p(y|\theta^*)p(\theta^*)] + (\theta - \theta^*)\nabla_\theta + \frac{1}{2}(\theta - \theta^*)^\top H_\theta(\theta - \theta^*) \tag{21}$$

where ∇_θ and H_θ are the gradient and the Heissian. We then calculate $p(y)$, the likelihood with respect to the model as

$$p(y) = \int p(y|\theta)p(\theta)\mathrm{d}\theta \approx p(y|\theta^*)p(\theta^*)\frac{2\pi^{\frac{|\theta^*|}{2}}}{|H_\theta|^{\frac{1}{2}}} \tag{22}$$

Assume the observations are independent and identically distributed, using the weak law of large numbers, $|H_\theta| = N^{A_e+A_f}|F_\theta|$ where $|F_\theta|$ is the Fisher information matrix for a single observation. With the above results, we show that

$$\begin{aligned}\ln p(y) =& \ln p(y|\theta^*) + \ln p(\theta^*) \\ &+ \frac{1}{2}(A_e + A_f)\left(\ln\frac{2\pi}{D}\right) - \ln|F_\theta| \\ \approx& \ln p(y|\theta^*) - \frac{1}{2}(A_e + A_f)\ln D\end{aligned} \tag{23}$$

□

The detailed proof is provided in Appendix B. In addition, the proposed framework provides several favorable properties when compared with existing constrained sentence generation methods (*e.g.,* faster inference than constrained Beam search). The detailed discussion is provided in Appendix D.

4 EXPERIMENTS

We evaluate the proposed method on natural and medical image captioning datasets, including MS-COCO [Young et al., 2014] and PEIR dataset[Library, 2022]. The former is for natural image captioning, while the latter is for medical image captioning. We use EfficientNet [Tan and Le, 2019] to extract image features. Based on image categories, we split the dataset in a way following [Del Chiaro et al., 2020] into sub-groups 'animals', 'transportation', and 'sports'. For PEIR, images are split into groups 'cardiovascular', 'respiratory' and 'nervous'. For each group, only one-third is used for pre-training, and the rest are used for testing. In this setting, automated captioning models are unlikely to perform well because the model is not trained with sufficient data for each task, and it would be beneficial to involve user interactions.

The proposed framework involves two stages, online and offline KA. 1) Online KA: We use the keyword extraction criteria discussed in the next paragraph and select 1-5 keywords from each caption in training set for model pre-training. For each group in the test set, we randomly selected 50 images for user interactions. A group of college students participated as volunteers in evaluating model predictions. Users are instructed to type 1-5 keywords relevant to the image into a textbox. Then the model predicts complete captions, which are compared with the ground-truth captions to evaluate how the model leverages user-input keywords to generate better captions. We set the Beam size for Beam search to 4 and record the generated captions and uncertainty. 2) Offline KA. The generated captions and uncertainty information are used for model updates. After the model is updated, the remaining images from each group are used for testing. We make the model generate automated caption predictions and compare them with the ground truth to evaluate how the model learns from users to improve its performance on unseen images.

For hyperparameters, we follow the convention of BNN [Goan and Fookes, 2020] by setting $\mu_0 = \theta_0 = 0$, $\sigma_0 = s_0 = 1$, $a_0 = \alpha_0 = 1, , b_0 = \beta_0 = 1$. The word embedding layer is plugged in with the GLOVE embedding [Pennington et al., 2014] and the corresponding $d = 300$. The proposed method and baselines are trained with Intel Core i7-3820 CPU and NVIDIA GeForce RTX2070 GPU. The proposed framework needs to be trained before online knowledge acquisition. We use public image-captioning datasets that contain images and ground-truth captions and extract keywords to prepare (image, keywords, ground-truth caption) tuple to train the model. The keywords are considered the most informative words that capture the semantics of the image. The criteria for extracting keywords from ground-truth captions are: 1) part-of-speech, which is considered an essential factor in quantifying the importance of words. Usually, nouns and verbs are considered more important than others, such as pronouns. Therefore, we require the selected keywords to be nouns or verbs. 2) the frequency of a specific word occurring in the captions. We require the frequency to be smaller than a preset threshold (0.1) to remove stopwords. It should be noted that the ground-truth keywords for medical images are provided in the PEIR dataset. Therefore, for the PEIR dataset, we directly use the ground-truth keywords for model training. Additional details (*e.g.,* user interface, training-evaluation protocol) are provided in Appendix E.

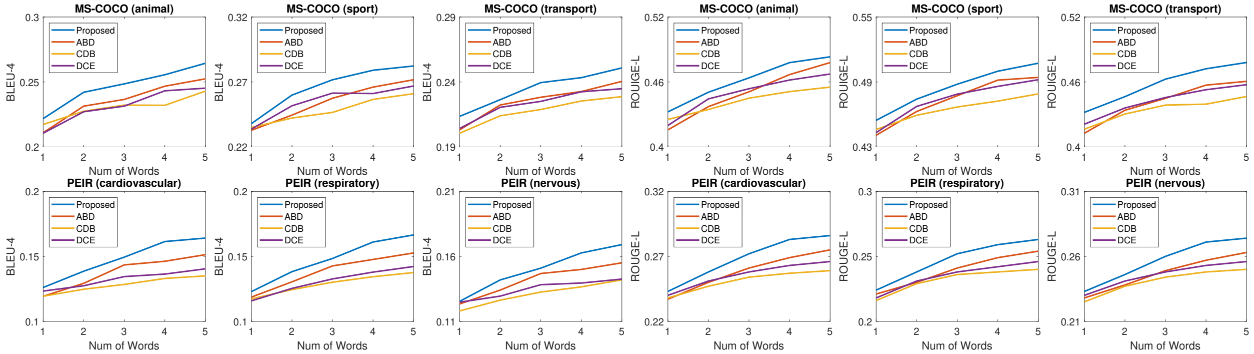
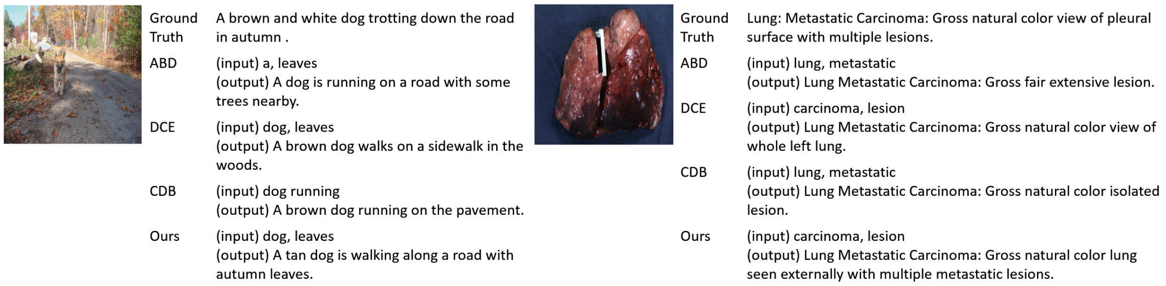


Figure 6: Quantitative comparison with different annotation budget (online knowledge acquisition)



Ground Truth	A brown and white dog trotting down the road in autumn .
ABD	(input) a, leaves (output) A dog is running on a road with some trees nearby.
DCE	(input) dog, leaves (output) A brown dog walks on a sidewalk in the woods.
CDB	(input) dog running (output) A brown dog running on the pavement.
Ours	(input) dog, leaves (output) A tan dog is walking along a road with autumn leaves.

Ground Truth	Lung: Metastatic Carcinoma: Gross natural color view of pleural surface with multiple lesions.
ABD	(input) lung, metastatic (output) Lung Metastatic Carcinoma: Gross fair extensive lesion.
DCE	(input) carcinoma, lesion (output) Lung Metastatic Carcinoma: Gross natural color view of whole left lung.
CDB	(input) lung, metastatic (output) Lung Metastatic Carcinoma: Gross natural color isolated lesion.
Ours	(input) carcinoma, lesion (output) Lung Metastatic Carcinoma: Gross natural color lung seen externally with multiple metastatic lesions.

Figure 7: Illustrative examples of generated captions based on user inputs (online KA)

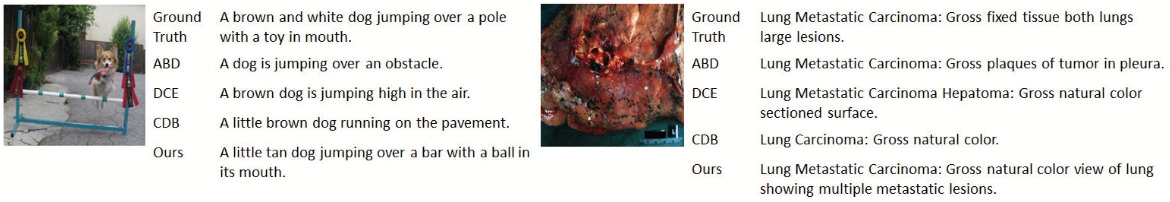
Baselines and evaluation metrics. We compare with baselines including ABD [Jia and Li, 2020], DCE [Huang et al., 2021] and CDB [Zhang et al., 2017]. ABD is a method based on double-ended keyword encoding. It allows a user to specify multiple words at the beginning and the end of a sentence and uses an asynchronous bidirectional decoding network to generate other words. CDB is a baseline based on bilateral caption generation. It leverages a bilateral LSTM network to generate words before and after the keywords. DCE is a method based on contextualized keyword encoding. It leverages an LSTM-based multimodal input encoder to process keywords as soft guidance and feed it into another LSTM for sentence generation. For evaluating the effectiveness of an interactive system, one important aspect is the quality of generated results given a limited annotation budget of users. In the context of interactive captioning, the **annotation budget** is the number of words specified by users. The numbers of user-input keywords are 1 to 5, and we report the average quality of generated captions. For offline KA, it should be noted that all the above captioning baselines do not involve model updates. Therefore, we extend those baselines by integrating with representative transfer learning methods, including fully Finetune [Tsimpoukelli et al., 2021] and Adapter [Chen et al., 2021b] (as discussed in Related Work section). For evaluation metrics, we use BLEU-4, METEOR, and ROUGE-L. Those metrics measure the similarity of a predicted text against one or more ground truth texts mainly based on n-gram matchings. Generally speaking, BLEU captures modified n-gram precision, ROUGE-L captures the longest common subsequence, and METEOR attempts to balance both modified precision and recall. A detailed explanation of the metrics can be found at [Hossain et al., 2019].

Results. We first present the results for online KA. Quantitative comparisons are provided in Figure 6. The proposed framework outperforms other baselines. The results can be intuitively explained by the difference in model design. CDB only allows consecutive words to be specified by the user, which may be insufficient to generate captions in the desired form by the user. DCE does not enforce a hard constraint of user-input words in the generated captions, and thus the user has limited control over the caption generation process. ABD first encodes user-input words at the end of a sentence through backward LSTM, then generates the complete caption with a forward LSTM with user-input words at the beginning of the sentence. However, backward LSTM provides an encoding rather than enforcing a hard constraint. In addition, a caption may start from non-informative words, which spend the budget but provide little information to the subsequent generation process. Qualitative examples are provided in Figure 7. (Note that ABD requires keywords to be at the beginning and the end of a sentence, and CDB requires keywords to be consecutive) In general, the proposed method leverages user-input keywords more effectively.

We then present the experiment results for offline KA. Quantitative comparisons are provided in Table 1, where all interactive captioning baselines are integrated with the Adapter method. We also include additional baselines, including the constrained Beam search method (CBS) and the Point-

Table 1: Quantitative comparison for automated caption predictions after offline KA

PEIR	Cardiovascular			Respiratory			Nervous		
	BLEU	ROUGE	METEOR	BLEU	ROUGE	METEOR	BLEU	ROUGE	METEOR
ABD+Adapter	0.144	0.242	0.148	0.132	0.232	0.139	0.125	0.237	0.151
CDB+Adapter	0.130	0.237	0.136	0.123	0.224	0.134	0.123	0.234	0.137
DCE+Adapter	0.135	0.251	0.152	0.127	0.228	0.137	0.128	0.240	0.140
Pointing+Adapter	0.140	0.228	0.130	0.121	0.215	0.128	0.119	0.225	0.134
CBS+Adapter	0.126	0.209	0.115	0.119	0.197	0.115	0.117	0.204	0.129
Proposed	0.152	0.268	0.170	0.138	0.249	0.153	0.140	0.252	0.160
MS-COCO	**Animal**			**Transport**			**Sport**		
	BLEU	ROUGE	METEOR	BLEU	ROUGE	METEOR	BLEU	ROUGE	METEOR
ABD+Adapter	0.145	0.357	0.168	0.148	0.362	0.167	0.169	0.375	0.173
CDB+Adapter	0.134	0.342	0.147	0.139	0.346	0.153	0.140	0.351	0.158
DCE+Adapter	0.145	0.346	0.159	0.155	0.349	0.159	0.152	0.362	0.166
Pointing+Adapter	0.132	0.327	0.139	0.137	0.295	0.149	0.129	0.328	0.149
CBS+Adapter	0.125	0.305	0.129	0.124	0.281	0.136	0.115	0.295	0.134
Proposed	0.157	0.390	0.184	0.168	0.387	0.191	0.183	0.384	0.185



Ground Truth: A brown and white dog jumping over a pole with a toy in mouth.

ABD: A dog is jumping over an obstacle.

DCE: A brown dog is jumping high in the air.

CDB: A little brown dog running on the pavement.

Ours: A little tan dog jumping over a bar with a ball in its mouth.

Ground Truth: Lung Metastatic Carcinoma: Gross fixed tissue both lungs large lesions.

ABD: Lung Metastatic Carcinoma: Gross plaques of tumor in pleura.

DCE: Lung Metastatic Carcinoma Hepatoma: Gross natural color sectioned surface.

CDB: Lung Carcinoma: Gross natural color.

Ours: Lung Metastatic Carcinoma: Gross natural color view of lung showing multiple metastatic lesions.

Figure 8: Illustrative examples of automatically generated captions after model updates (offline KA)

ing method. Those methods were not originally proposed for interactive captioning but can be applied in an offline KA setting. Specifically, the pointing method leverages a pointing mechanism on the output layer of the network to directly modify the predicted probability. Constrained Beam search converts the keyword constraints into many states, each maintaining a Beam of candidate captions during caption generation. The proposed framework outperforms other baselines. A possible reason is that the proposed framework is a unified model with a more efficient way of preserving existing knowledge and encoding new knowledge. In contrast, the baselines may still be prone to overfitting and thus forget prior knowledge during model updates. Qualitative examples are provided in Figure 8.

Ablation study. For the ablation study, we note that the proposed framework leverages the expandable embedding and the expandable feedforward layer to encode the knowledge from the user to improve model performance in the future. Alternative design choices include using only the expandable embedding, only the expandable feedforward layer, or none of them. For those alternative choices, we report the results for offline knowledge acquisition in Figure 9, which indicates that both components contribute to better performance. We also provide a qualitative analysis of activated neurons. Once the model is pre-trained on the PEIR dataset, we plot the activated neurons of different layers (*i.e.,* expandable embeddings or expandable feedforward layer) in Figure 10 where each row corresponds to a layer, and each column corresponds to a neuron. A dark grid indicates that the neuron is activated. After the model is updated for offline KA for cardiovascular images, we plot the activated neurons, which shows more neurons are activated to encode the new knowledge from the user.

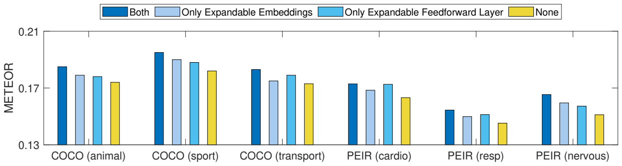


Figure 9: Ablation study on alternative architecture

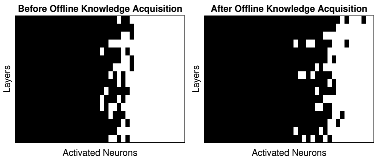


Figure 10: Visualization of activated neurons before (left) and after offline knowledge acquisition (right).

5 CONCLUSION

In this paper, we propose a knowledge acquisition framework for interactive image captioning by acquiring knowledge from user-input keywords to improve the quality of predicted captions. A multimodal conditioned caption completion module is developed to fuse the image data with user-input keywords and ensure all keywords appear in the updated image caption. Moreover, the framework effectively encodes the newly acquired knowledge through dynamic neural resource allocation to benefit future caption predictions even without aid from users. The proposed model can be potentially applied to human-in-the-loop image captioning in specialized domains (*e.g.,* medicine), where accurate technical concepts and text descriptions are difficult to generate by conventional captioning algorithms.

Acknowledgements

This research was partially supported by NSF IIS award IIS-1814450 and ONR award N00014-18-1-2875. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the official views of any funding agency. We would like to thank the anonymous reviewers for reviewing the manuscript.

References

Mousa Al Sulaimi, Imtiaz Ahmad, and Mohammad Jeragh. Deep image captioning survey: A resource availability perspective. In *2021 29th Conference of Open Innovations Association (FRUCT)*, pages 3–13. IEEE, 2021.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Guided open vocabulary image captioning with constrained beam search. In *EMNLP*, 2017.

Jize Cao, Zhe Gan, Yu Cheng, Licheng Yu, Yen-Chun Chen, and Jingjing Liu. Behind the scene: Revealing the secrets of pre-trained vision-and-language models. In *European Conference on Computer Vision*, pages 565–580. Springer, 2020.

Jun Chen, Han Guo, Kai Yi, Boyang Li, and Mohamed Elhoseiny. Visualgpt: Data-efficient image captioning by balancing visual input and linguistic knowledge from pretraining. 2021a.

Xianyu Chen, Ming Jiang, and Qi Zhao. Self-distillation for few-shot image captioning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 545–555, 2021b.

Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. Paying more attention to saliency: Image captioning with saliency and context attention. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(2):1–21, 2018.

Riccardo Del Chiaro, Bartłomiej Twardowski, Andrew D Bagdanov, and Joost Van de Weijer. Ratt: Recurrent attention to transient tasks for continual image captioning. *arXiv preprint arXiv:2007.06271*, 2020.

Hongwei Ge, Zehang Yan, Kai Zhang, Mingde Zhao, and Liang Sun. Exploring overall contextual information for image captioning in human-like cognitive style. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1754–1763, 2019.

Zoubin Ghahramani and Thomas L Griffiths. Infinite latent feature models and the indian buffet process. In *Advances in neural information processing systems*, pages 475–482, 2006.

Ethan Goan and Clinton Fookes. Bayesian neural networks: An introduction and survey. In *Case Studies in Applied Bayesian Data Science*, pages 45–87. Springer, 2020.

Demi Guo, Alexander M Rush, and Yoon Kim. Parameter-efficient transfer learning with diff pruning. *arXiv preprint arXiv:2012.07463*, 2020a.

Longteng Guo, Jing Liu, Xinxin Zhu, Peng Yao, Shichen Lu, and Hanqing Lu. Normalized and geometry-aware self-attention network for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10327–10336, 2020b.

Chris Hokamp and Qun Liu. Lexically constrained decoding for sequence generation using grid beam search. *arXiv preprint arXiv:1704.07138*, 2017.

MD Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CsUR)*, 51(6):1–36, 2019.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019.

Feicheng Huang, Zhixin Li, Shengjia Chen, Canlong Zhang, and Huifang Ma. Image captioning with internal and external knowledge. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 535–544, 2020.

Jia-Hong Huang, Ting-Wei Wu, Chao-Han Huck Yang, and Marcel Worring. Deep context-encoding network for retinal image captioning. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 3762–3766. IEEE, 2021.

Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. Attention on attention for image captioning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4634–4643, 2019.

Zhengxiong Jia and Xirong Li. icap: Interactive image captioning with predictive text. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, pages 428–435, 2020.

Rabeeh Karimi Mahabadi, Sebastian Ruder, Mostafa Dehghani, and James Henderson. Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. *arXiv e-prints*, pages arXiv–2106, 2021.

Alison Kidd. *Knowledge acquisition for expert systems: A practical handbook*. Springer Science & Business Media, 2012.

Ponnambalam Kumaraswamy. A generalized probability density function for double-bounded random processes. *Journal of hydrology*, 46(1-2):79–88, 1980.

PEIR Digital Library. Peir digital library, 2022. URL https://peir.path.uab.edu/library/.

Yunpeng Luo, Jiayi Ji, Xiaoshuai Sun, Liujuan Cao, Yongjian Wu, Feiyue Huang, Chia-Wen Lin, and Rongrong Ji. Dual-level collaborative transformer for image captioning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 2286–2293, 2021.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Ning Miao, Hao Zhou, Lili Mou, Rui Yan, and Lei Li. Cgmh: Constrained sentence generation by metropolis-hastings sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6834–6842, 2019.

Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10971–10980, 2020.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

Zhan Shi, Xu Zhou, Xipeng Qiu, and Xiaodan Zhu. Improving image captioning with better use of captions. *arXiv preprint arXiv:2006.11807*, 2020.

Yi-Lin Sung, Varun Nair, and Colin A Raffel. Training neural networks with fixed sparse masks. *Advances in Neural Information Processing Systems*, 34:24193–24205, 2021.

Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5227–5237, 2022.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Hierarchy parsing for image captioning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2621–2629, 2019.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

Xiaodan Zhang, Shengfeng He, Xinhang Song, Pengxu Wei, Shuqiang Jiang, Qixiang Ye, Jianbin Jiao, and Rynson WH Lau. Keyword-driven image captioning via context-dependent bilateral lstm. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 781–786. IEEE, 2017.

Yimin Zhou, Yiwei Sun, and Vasant Honavar. Improving image captioning by leveraging knowledge graphs. In *2019 IEEE winter conference on applications of computer vision (WACV)*, pages 283–293. IEEE, 2019.

Xinxin Zhu, Weining Wang, Longteng Guo, and Jing Liu. Autocaption: Image captioning with neural architecture search. *arXiv preprint arXiv:2012.09742*, 2020.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020.

Appendix

Organization of Appendix. In this Appendix, we first summarize the main notations used throughout the paper in Section A. We provide the proof to Theorem 1 in Section B. We then visualize the architecture and provide the pseudo code for the knowledge acquisition process in Section C. We discuss several favorable properties of the proposed framework when compared with existing constrained sentence generation methods in Section D. We provide additional experiment results in Section E. We discuss broader impacts and limitations in Section F. The link to the source code is provided in Section G.

A Summary of Main Notations

Table 2: Summary of Main Notations

I	image feature map extracted by image encoder
Y	ground-truth caption
$\hat{y}_n$	predicted word at position n
X	incomplete caption
x_n	word at position n in X
Z	constraint list of user-input keywords
z_m	m-th keyword in Z
$\mathbf{b}_n$	the embedding vector of word token at position n
$\mathbf{b}$	the embedding vectors of all word tokens in the sequence
$\mathbf{c}_n^Q$,$\mathbf{c}_n^K$,$\mathbf{c}_n^V$	the query, key and value vectors for token at position n
$\mathbf{c}^Q$,$\mathbf{c}^K$,$\mathbf{c}^V$	the query, key and value vectors for all tokens in the sequence
$\mathbf{o}_n$	the output of transformer block with self attention at position n
$\mathbf{v}$	mask vector of expandable embeddings
ζ	mask vector of expandable feedforward layer
S	pseudo token embedding matrix
β	the pseudo tokens embedding matrix multiplied by Indian buffet process mask
W	weight matrix of expandable feedforward layer
$\tilde{Z}$	concatenation of keyword embeddings and image feature vectors
$\boldsymbol{\omega}_n$	the output of transformer block with cross attention at position n
Φ	the set of trainable parameters
$u_{b,n}$	uncertainty estimation for prediction of n-th word in b-th candidate caption
$\bar{\boldsymbol{\xi}}_{\boldsymbol{n}}$	the averaged probability vector at position n
B	the size of beam for Beam search
A_e	the number of activated neurons in expandable embeddings
A_f	the number of activated neurons in expandable feedforward layer

B Proof of Theorem 1

Proof. We first consider the expandable feedforward layer. Denote the set of mask variables that are close to zero as $v_0^{A_f}$ and the corresponding layer weights as $W_0^{A_f}$. Also denote non-zero mask variables as v^{A_f} and the corresponding layer weights as W^{A_f}. Given the input to the expandable feedforward layer x^{in}, The output x^{out} can be expressed as

$$x^{\text{out}} = ([v^{A_f}; v_0^{A_f}] \odot [W^{A_f}; W_0^{A_f}])^\top x^{\text{in}} = (v^{A_f} \odot W^{A_f})^\top x^{\text{in}} \tag{24}$$

where $\odot$ denotes the operation of applying mask to the weight matrix. We then consider expandable embedding and the self attention. Denote the set of mask variables that are close to zero as $\zeta_0^{A_e}$ and the corresponding embedding as $\beta_0^{A_e}$. Also denote non-zero mask variables as ζ^{A_e} and the corresponding embedding as β^{A_e}, and the embedding of actual token as b. The logit s_{mn} before softmax in self attention can be expressed as

$$\begin{aligned} &c^Q = W^Q\mathbf{b}, \quad c^K = W^K[\mathbf{b}; \zeta^{A_e}\beta^{A_e}; \zeta_0^{A_e}\beta_0^{A_e}], \quad c^V = W^V[\mathbf{b}; \zeta^{A_e}\beta^{A_e}; \zeta_0^{A_e}\beta_0^{A_e}], \\ &x^{\text{out}} = f^{\text{Ln}}[\text{softmax}(\frac{(c^Q)^T c^K}{\sqrt{d}})\mathbf{c}^V + \mathbf{b}] = f^{\text{Ln}}[\text{softmax}(\frac{(W^Q\mathbf{b})^\top W^K[\mathbf{b}; \zeta^{A_e}\beta^{A_e}]}{\sqrt{d}})W^V[\mathbf{b}; \zeta^{A_e}\beta^{A_e}] + \mathbf{b}] \end{aligned} \tag{25}$$

Eqs. (24) and (25) shows that if the mask variable is zero, the corresponding neuron does not affect the prediction and can be pruned. Therefore, we gather all activated neurons from expandable feedforward layer and the expandable embedding,

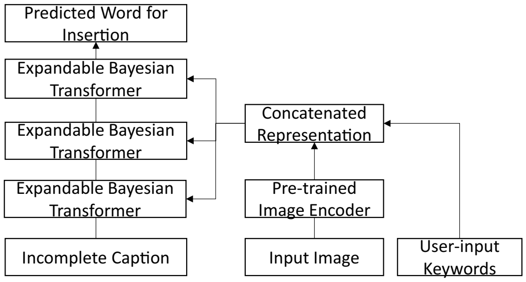

Predicted Word for Insertion
Expandable Bayesian Transformer
Expandable Bayesian Transformer
Expandable Bayesian Transformer
Incomplete Caption
Concatenated Representation
Pre-trained Image Encoder
Input Image
User-input Keywords

Figure 11: Architecture of the proposed framework

and denote them as θ. Using Taylor's expansion, we approximate the log unnormalized posterior as

$$f = \ln[p(y|\theta)p(\theta)] \approx \ln[p(y|\theta^*)p(\theta^*)] + (\theta - \theta^*)\nabla_\theta f + \frac{1}{2}(\theta - \theta^*)^\top H_\theta(\theta - \theta^*) \tag{26}$$

where ∇_θ and H_θ are the gradient and the Heissian. We then calculate $p(y)$, the likelihood with respect to model M as

$$\begin{aligned} p(y) &= \int p(y|\theta)p(\theta)d\theta \approx \int \exp[f + (\theta - \theta^*)\nabla_\theta f + \frac{1}{2}(\theta - \theta^*)^\top H_\theta(\theta - \theta^*)]d\theta \\ &\approx p(y|\theta^*)p(\theta^*) \int \exp[\frac{1}{2}(\theta - \theta^*)^\top H_\theta(\theta - \theta^*)]d\theta \approx p(y|\theta^*)p(\theta^*)\frac{2\pi^{\frac{|\theta^*|}{2}}}{|H_\theta|^{\frac{1}{2}}} \end{aligned} \tag{27}$$

The above derivation uses the fact that the gradient is zero because θ^* is the optimized parameter, and H_θ is negative definite at θ^*.

For simplification, $p(\theta)$ is assumed an non-informative prior with constant density. Then the Heissian can be shown as

$$H_{ij} = \frac{\partial^2}{\partial\theta_i\partial\theta_j}\ln p(y|\theta) = \frac{\partial^2}{\partial\theta_i\partial\theta_j}\sum_d \ln p(x_d|\theta) \tag{28}$$

Assume the observations are independent and identically distributed, using the weak law of large numbers,

$$H_{ij} = \frac{\partial^2}{\partial\theta_i\partial\theta_j}\frac{1}{D}\sum_d Dp(x_d|\theta) \approx \frac{\partial^2}{\partial\theta_i\partial\theta_j}E[D\ln p(x_d|\theta)], \quad |H_\theta| \quad = N^{A_e+A_f}|F_\theta| \tag{29}$$

where $|F_\theta|$ is the Fisher information matrix for a single observation. With the above results, we show that

$$\ln p(y) = \ln p(y|\theta^*) + \ln p(\theta^*) + \frac{1}{2}(A_e + A_f)(\ln 2\pi - \ln D) - \ln|F_\theta| \approx \ln p(y|\theta^*) - \frac{1}{2}(A_e + A_f)\ln D \tag{30}$$

Note that the terms $\ln|F_\theta|$ and $\ln p(\theta^*)$ are negligible compared to other terms when the number of data instances D is large. Since Eq. (30) follows the form of Bayesian Information Criterion (BIC), we prove that when optimizing the loss in Eq. (20), we also select the best model based on the Bayesian Information Criterion. □

C Network Architecture and the Workflow

The proposed framework consists of the pre-trained image feature extractor (*i.e.,* EfficientNet [Tan and Le, 2019] architecture), and the multimodal conditional caption completion module with trainable layers. The architecture is summarized in Figure 11. The whole process is summarized in Algorithm 1.

D Comparison with Constrained Beam Search

The constrained Beam search is a sentence generation method that satisfies keyword constraints. Specifically, it leverages a sequential language model to generate sentence word by word. To enforce keyword constraints, the constrained Beam search maintains many states. Each state corresponds to a subset of keywords and stores a Beam of candidate sentences during sentence generation. The candidate sentences in the state that corresponds to the complete set of keywords satisfy all the constraints. And practically, we can pick the top-K candidate sentence from that state with the highest joint probability of all words as the model output. An illustrative example of the states for constrained Beam search is provided in Figure 12. The constrained Beam search method was not originally proposed for human-in-the-loop image captioning, but it can be applied to the human-in-the-loop setting.

Algorithm 1 Key stages of the proposed framework

```
Given parameters $\mathbf{\Phi}$ and beam size $B$;
Given an image $I^{raw}$;
Online Knowledge Acquisition:
User specifies a list of keywords to guide caption generation;
Initialize incomplete caption with '<start>' token;
Construct a constraint list of user-input keywords and append '<end>' token;
Generate feature map $I$ via Eq.(1);
while Constraint list $Z$ is not empty do
    Calculate embedding $\mathbf{b}_n$ of incomplete caption;
    Calculate $\tilde{\mathbf{o}}_n$ using self-attention via Eq.(11);
    Calculate $\mathbf{o}_n$ using cross-attention via Eq.(12);
    Predict word for insertion $\hat{y}$ via Eq.(13), estimate uncertainty via Eq.(19);
    if $\hat{y}$ is '<null>' token then
        Move word $z_1$ from constraint list to the current incomplete caption;
    else
        Add $\hat{y}$ to the current incomplete caption;
    end if
end while
Keep $B$ candidate captions sequence using Beam search and record corresponding uncertainty;
Off-line knowledge acquisition:
Collect candidate captions sequence and uncertainty estimation during online knowledge acquisition stage;
Reset the constraint list $Z$ to empty and append '<end>' token;
Calculate weighting factor $\lambda_{b,n}$ via Eq.(20);
Calculate $\hat{y}$ through feed-forward similar to online knowledge acquisition;
Back-propagate to update trainable parameters via Eq.(20);
```

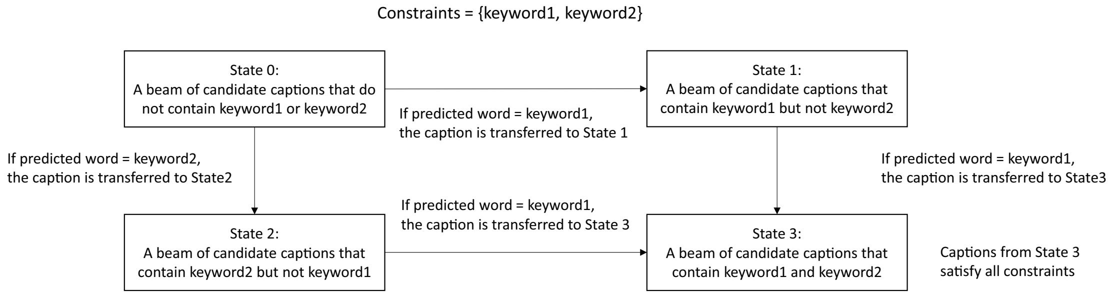

Constraints = {keyword1, keyword2}
State 0:
A beam of candidate captions that do not contain keyword1 or keyword2
If predicted word = keyword1, the caption is transferred to State 1
State 1:
A beam of candidate captions that contain keyword1 but not keyword2
If predicted word = keyword2, the caption is transferred to State2
If predicted word = keyword1, the caption is transferred to State3
If predicted word = keyword1, the caption is transferred to State 3
State 2:
A beam of candidate captions that contain keyword2 but not keyword1
State 3:
A beam of candidate captions that contain keyword1 and keyword2
Captions from State 3 satisfy all constraints

Figure 12: Example of constrained Beam search (assume there are two user-input keywords).

The proposed framework provides several favorable properties when compared with the constrained Beam search. First, the proposed framework considers keywords as the additional input and the constraint, while the constrained Beam search considers keywords as only the constraint. Therefore, when selecting top-K candidate sentences with the highest joint probability, the selection criteria are different. Specifically, the joint probability evaluated by the proposed method is

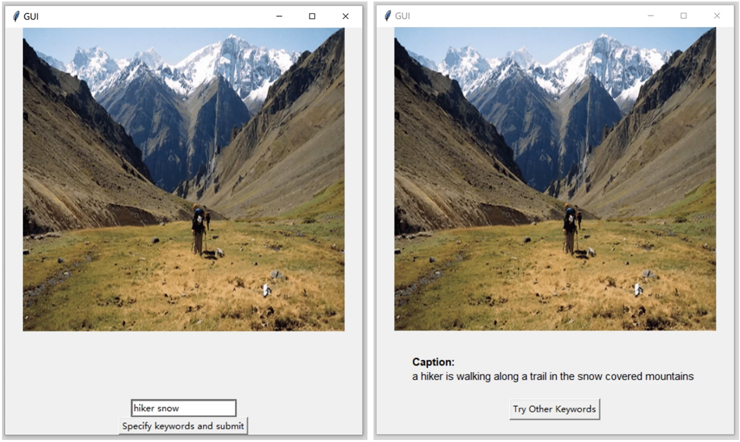
$$p(\hat{Y}|I,Z) = \prod_n p(\hat{y}_n|\hat{y}_{1:n-1}, I, Z)$$

where $\hat{Y}$ is the predicted sentence, $\hat{y}_n$ is the predicted word at position n, I is image features, and Z is the set of user-input keywords. Naturally, user-input keywords manifest useful information, and the caption generation should be conditioned on both the keywords and image features.

However, the joint probability evaluated by constrained Beam search is

$$p(\hat{Y}|I) = \prod_{n} p(\hat{y}_n|\hat{y}_{1:n-1}, I)$$

It does not leverage the information from keywords, which is a weakness. The proposed framework offers faster inference than constrained Beam search. In constrained Beam search, the number of states grows exponentially with respect to the number of keywords. Denote L as the maximum length of sentence, B as the Beam sizes, M as the number of keywords, and V as the size of the vocabulary. The complexity of constrained Beam search is $O(2^M BLV)$ while the proposed caption generation with Beam search is $O(BLV)$. For constraint Beam search, each state is represented by a binary vector of length M. If the i-th entry is 0, it indicates the i-th keyword is missing in all the candidate captions in this state. If the j-th entry is 0, it indicates the j-th keyword exists in all the candidate captions in this state. There are a total of 2^M states. During inference, any candidate caption start from state $s_0 = [0, 0, ..., 0]$. Once a keyword is generated during the process, the candidate caption is transferred to the corresponding state. To get the complete captions that contain all keywords, constraint Beam search focus on the last state $s_{2^M-1} = [1, 1, ..., 1]$. Each state keeps B candidate captions, and when predicting each word, the model evaluates the probability of all words in the vocabulary. Therefore, the complexity is $O(2^M BLV)$. In contrast, the proposed model does not need to maintain 2^M states, and thus the complexity is $O(BLV)$.



hiker snow

Caption:
a hiker is walking along a trail in the snow covered mountains

Figure 13: Illustrative example of user interface for collecting input (left) and showing output (right)

E Additional Experiment Details, Results, and Ablation Study

The proposed framework involves two stages, online and offline knowledge acquisition. As discussed in the main paper, in each stage, the proposed framework and the baselines are trained and evaluated. A high-level summary of the training-evaluation protocol is provided below.

Step	Online Knowledge Acquisition
1	Pre-train model using paired image-caption data from the training set.
2	For each category from the testing set, randomly select 50 images for user interaction.
3	Collect user-input keywords and predict complete captions.
4	Evaluate the quality of predicted captions using ground-truth captions.
5	Record keywords and predicted captions for offline knowledge acquisition.
	Offline Knowledge Acquisition
6	Update the model using the data prepared in Step 5.
7	Use the remaining images from each category to predict captions without user interaction.
8	Evaluate the quality of predicted captions using ground-truth captions.

For user interaction, the image is displayed in a pop-up window, and the user can type the keywords into the textbox. An illustration of the user interface is provided in Figure 13.

We include the results for online knowledge acquisition in Table 3. A statistical testing of METEOR score is conducted on the proposed method versus the second-best method across testing cases. For the 3 subgroups in COCO dataset, p-values are 0.008, 0.020, 0.017, which are considered significant with $\alpha = 0.05$. For PEIR, p-values are 0.025, 0.032, 0.040. Before online knowledge acquisition (KA) (corresponding to no user inputs), all methods achieve similar results, as reported in Table 4.

Table 3: Quantitative comparison for online KA

PEIR	**Cardiovascular**			**Respiratory**			**Nervous**		
	BLEU	ROUGE	METEOR	BLEU	ROUGE	METEOR	BLEU	ROUGE	METEOR
ABD	0.142	0.257	0.173	0.141	0.240	0.141	0.145	0.246	0.165
CDB	0.128	0.247	0.156	0.136	0.236	0.127	0.131	0.239	0.148
DCE	0.135	0.253	0.161	0.139	0.238	0.130	0.135	0.245	0.151
Pointing	0.130	0.251	0.158	0.129	0.230	0.126	0.119	0.235	0.137
CBS	0.122	0.245	0.150	0.125	0.211	0.115	0.108	0.219	0.131
Proposed	0.151	0.273	0.181	0.149	0.253	0.152	0.150	0.259	0.172
MS-COCO	**Animal**			**Transport**			**Sport**		
	BLEU	ROUGE	METEOR	BLEU	ROUGE	METEOR	BLEU	ROUGE	METEOR
ABD	0.229	0.438	0.234	0.224	0.434	0.232	0.254	0.473	0.261
CDB	0.224	0.428	0.219	0.210	0.421	0.217	0.239	0.461	0.227
DCE	0.225	0.441	0.231	0.219	0.437	0.225	0.258	0.476	0.235
Pointing	0.211	0.419	0.225	0.207	0.426	0.199	0.221	0.415	0.234
CBS	0.168	0.374	0.171	0.188	0.381	0.193	0.205	0.409	0.201
Proposed	0.245	0.465	0.248	0.238	0.462	0.246	0.275	0.491	0.272

We then present the experiment results for offline KA to evaluate how the model learns from user interactions to improve its performance in the future. In other words, we make the retrained model generate automated predictions for images on the hold-out test set. Quantitative comparisons are provided in Table 1. The proposed framework outperforms other baselines. A possible reason is that the proposed framework is a unified model with a more efficient way of preserving existing knowledge and encoding new knowledge.

Table 4: Quantitative comparison with no keywords

	ABD	CDB	CDE	Proposed	ABD	CDB	CDE	Proposed	ABD	CDB	CDE	Proposed
PEIR	Cardiovascular				Respiratory				Nervous			
	0.124	0.127	0.119	0.129	0.104	0.111	0.108	0.115	0.121	0.119	0.122	0.126
MS-COCO	Animal				Transport				Sport			
	0.152	0.157	0.148	0.160	0.161	0.155	0.152	0.164	0.168	0.171	0.173	0.175

Table 5: User Evaluation (percentage of generated captions where the proposed method is preferred over the baselines)

Model	Online KA			Offline KA		
	Animal	Transportation	Sport	Animal	Transportation	Sport
Proposed vs ABD	76.0%	84.0%	64.0%	64.1%	67.1%	56.2%
Proposed vs CDB	82.0%	88.0%	78.0%	73.4%	79.6%	76.5%
Proposed vs DCE	66.0%	78.0%	72.0%	65.6%	64.0%	59.3%

We also performed an evaluation based on human judgments of the quality of generated captions. For the natural image dataset, we generate captions for the testing images after online and offline knowledge acquisition, and present the generated captions by the proposed method and baselines to the user. Users were asked to select the caption that best describes the image semantics. We calculate the percentage of users who chose the proposed method over baselines, as shown in Table 5.

F Broader Impact and Limitations

The proposed framework involves humans in the loop for image captioning tasks, where users specify a sequence of keywords (*i.e.,* informative words that capture the image's important semantics) as additional input to guide the caption generation process. The model can be potentially applied to specialized domains (*e.g.,* medicine), where accurate technical concepts and descriptions are difficult to generate by automated captioning. Since the proposed framework is interactive, the model may be misguided if the user provides irrelevant or wrong keywords. One possible solution to this issue is to make the model evaluate the quality of the user-specified keyword and alert the user to double-check the keywords when necessary.

G Link to the Source Code

The source code is available at https://github.com/ritmininglab/KA-HITP.