AI-based quiz for Intelligent Tutoring Systems

1st Pujan Mehta 2nd Ruturaj Gujar 3rd Sahil Jajodia 4th Jai Devani

ABSTRACT

We introduce a caption generation and refinement framework for Intelligent Tutoring Systems. Our approach first generates a sentence template with slot locations explicitly tied to specific image regions. These slots are then filled in by visual concepts identified in the regions by object detectors. The caption will be detailed and accurate. The idea is to use the model created as an attachment to a website that can be used by the teachers and students. The teachers can upload assignments and the AI-based model then takes the data in the form of images, analyses the given image and generates the MCQ quiz. The options for the quiz are then generated by our model using GloVe embedding. Using this approach, four options for each question are generated having one correct answer. The student is then graded by the model, based on the performance on the quiz. The web-app also allows students to register themselves and provides a detailed view of their performances on the homepage once logged in successfully. The teachers operating the site can easily view student performances of the overall subject.

I. PROBLEM DEFINITION

The steps towards the development of the model are Object Recognition followed by Caption Generation and Caption Refinement. Object Detection is concerned with identifying objects and giving each object a tag that describes the nature, properties and type of object. The goal of this step is to identify various objects present in the image such as stationary items, vehicles, animals, etc. A few popular techniques are Faster RCNN and YOLO. Next, we move in the area of generating captions for the images. This is the main heart of the project. It clubs the fields of computer vision and Natural Language Processing. In other words, Caption Generation is the process of generating textual description of an image. Image Captioning techniques such as Deep Visual Semantic, Show & Tell and Show, Attend & Tell are methods that can be used for it. Once the captions are generated we need to refine them. This is an essential step because the captions may not describe the image in a way as human as possible. Therefore, we use caption refinement techniques such as CBOW, SkipGram Model and GloVe method.

The front end of the website is to be designed using HTML, CSS and Bootstrap. Since these 3 provide us with the tools required for the website front end. The integration of the UI and model is to be done using the python framework called Django. The model is to be developed using Python language. Since, python is a powerful language for creating AI/ML based program because it has a large variety of libraries available for use. The datasets that are available for this purpose are the MS COCO, Flickr30k and Google Open Image Set.

II. LITERATURE SURVEY

A. *Datasets*

*1) **Flickr 30k**:* The Flickr30K dataset has become a standard benchmark for sentence-based image description. This paper presents Flickr30K Entities, which augments the 158k captions from Flickr30k with 244k coreference chains, linking mentions of the same entities across different captions for the same image, and associating them with 276k manually annotated bounding boxes. Such annotations are essential for continued progress in automatic image description and grounded language understanding. They enable us to define a new benchmark for localization of textual entity mentions in an image. We present a strong baseline for this task that combines an image-text embedding, detectors for common objects, a color classifier, and a bias towards selecting larger objects. While our baseline rivals in accuracy more complex state-of-the-art models, we show that its gains cannot be easily parlayed into improvements on such tasks as image-sentence retrieval, thus underlining the limitations of current methods and the need for further research.

*2) **MS COCO**:* COCO stands for Common Objects in Context. As hinted by the name, images in COCO dataset are taken from everyday scenes thus attaching "context" to the objects captured in the scenes. We can put an analogy to explain this further. Let's say we want to detect a person object in an image. A non-contextual, isolated image will be a close-up photograph of a person. Looking at the photograph, we can only tell that it is an image of a person. However, it will be challenging to describe the environment where the photograph was taken without having other supplementary images that capture not only the person but also the studio or surrounding scene. COCO was an initiative to collect natural images, the images that reflect everyday scene and provides contextual information. In everyday scenes, multiple objects can be found in the same image and each should be labeled as a different object and segmented properly. COCO dataset provides the labeling and segmentation of the objects in the images. A machine learning practitioner can take advantage of the labeled and segmented images to create a better performing object detection model.

The MS COCO dataset contains 91 common object categories with 82 of them having more than 5,000 labeled instances. In total the dataset has 2,500,000 labeled instances in 328,000 images. In contrast to the popular ImageNet dataset, COCO has fewer categories but more instances per category.

B. *Object Detection*

*1) **Faster RCNN**:* Faster RCNN is the modified version of Fast RCNN. The major difference between them is that Fast RCNN uses selective search for generating Regions of Interest, while Faster RCNN uses "Region Proposal Network", aka RPN. The first module is a deep fully convolutional network that proposes regions, and the second module is the Fast R-CNN detector that uses the proposed regions.
A Region Proposal Network (RPN) takes an image (of any size) as input and outputs a set of rectangular object proposals, each with an objectness score. To generate region proposals, we slide a small network over the convolutional feature map output by the last shared convolutional layer. This small network takes as input an n × n spatial window of the input convolutional feature map. Each sliding window is mapped to a lower-dimensional feature

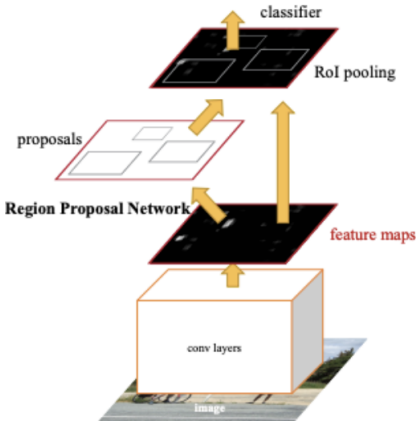

Fig. 1. Faster RCNN Architecture

The below steps are typically followed in a Faster RCNN approach:

- We take an image as input and pass it to the ConvNet which returns the feature map for that image.
- Region proposal network is applied on these feature maps. This returns the object proposals along with their objectness score.
- A RoI pooling layer is applied on these proposals to bring down all the proposals to the same size.
- Finally, the proposals are passed to a fully connected layer which has a softmax layer and a linear regression layer at its top, to classify and output the bounding boxes for objects.

The problems faced by Faster RCNN are:
All of the object detection algorithms we have discussed so far use regions to identify the objects. The network does not look at the complete image in one go, but focuses on parts of the image sequentially. This creates two complications:

- The algorithm requires many passes through a single image to extract all the objects.
- As there are different systems working one after the other, the performance of the systems further ahead depends on how the previous systems performed.

*2) **YOLO**:* Faster RCNN we saw above primarily use regions to localize the objects within the image. The network does not look at the entire image, only at the parts of the images which have a higher chance of containing an object.The YOLO framework (You Only Look Once) on the other hand, deals with object detection in a different way. It takes the whole image in a single instance and predicts the bounding box coordinates and class probabilities for these boxes. Prior detection systems repurpose classifiers or localizers to perform detection. They apply the model to an image at multiple locations and scales. High scoring regions of the image are considered detections. YOLO uses a totally different approach. It applies a single neural network to the full image. This network divides the image into regions and predicts bounding boxes and probabilities for each region. These bounding boxes are weighted by the predicted probabilities. **YOLOs biggest advantage is its superb speed** – it's incredibly fast and can process 45 frames per second. YOLO also understands generalized object representation.

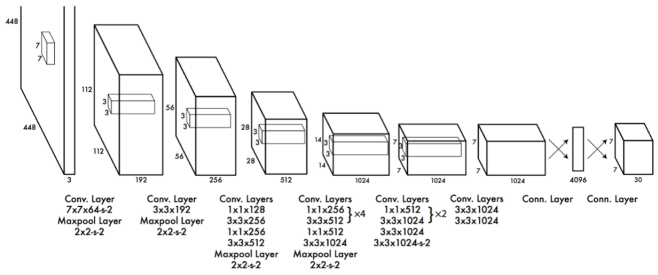

Fig. 2. YOLO Architecture

TABLE I
OBJECT DETECTION SUMMARY RESULTS

Algorithm	*MAP*	*FPS*
Faster RCNN	76.4	5
YOLOv2	78.6	40

C. *Image Captioning Method*

*1) **Deep Visual Semantic (RNN based) - CVPR 2015**:*
This paper uses Bidirectional RNN (BRNN) as a backbone for generating the captions. It has a structured objective function that combines the two modalities (Image and Sentence) through multimodal embedding. The authors have developed a deep neural network model that infers the latent alignment between segments of sentences and the region of the image that they describe.
The basic idea of this model is that it arranges the sentence fragments with the visual regions which have been described by the multimodal embeddings. On the architecture side, this

model uses Region Convolution Neural Networks (RCNN) to detect objects in the image. The RCNN is pretrained on ImageNet and finetuned on 200 classes. Additionally, it uses BRNN for computing word representations.
The RNN is trained to combine a word xt, the previous context $ht1$ to predict the next word yt. The optimizer used here is RMS Prop.

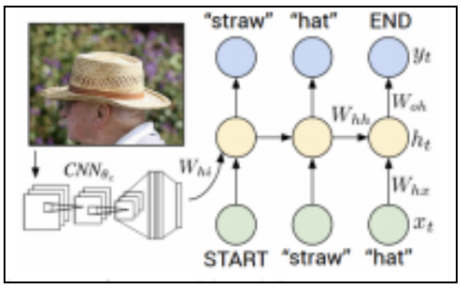

Fig. 3. Deep VS Architecture

*2) **Show, Attend and Tell (Attention Mechanism)**:* This paper uses the novel attention mechanism that learns to describe the content in images. The attention model has become very popular in other applications like machine translation. The paper describes approaches to caption generation that attempt to incorporate a form of attention with two variants: a "hard" attention mechanism and a "soft" attention mechanism. It also shows how one advantage of including attention is the ability to visualize what the model "sees".
Moving on to the optimization part, for the Flickr8k dataset RMSProp has been used, while for Flickr30k/MS COCO dataset the recently proposed Adam algorithm was employed. This paper uses an attention-based approach that gives state of the art performance on three benchmark datasets using the BLEU and METEOR metric. It also shows how the learned attention can be exploited to give more interpretability into the model's generation process and demonstrate that the learned alignments correspond very well to human intuition.

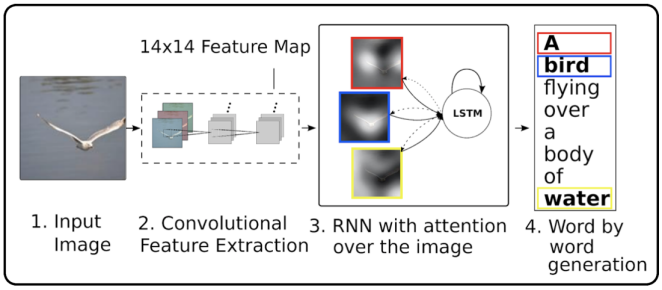

Fig. 4. Show Attend & Tell Architecture

D. Caption Refinement Techniques

*1) **Continuous Bag of Words Model (CBOW)**:* The CBOW model architecture tries to predict the current target word (the center word) based on the source context words (surrounding words). Considering a simple sentence, "the quick brown fox jumps over the lazy dog", this can be pairs of (context_window, target_word) where if we consider a context window of size 2, we have examples like ([quick, fox], brown), ([the, brown], quick), ([the, dog], lazy) and so on. Thus the model tries to predict the target_word based on the context_window words.

TABLE II
IMAGE CAPTIONING SUMMARY RESULTS

Paper	*Approach*	*Backbone*	*Meteor Score*
Deep VS	RNN	VGGNet	19.5
Show, Attend & Tell	LSTM	VGGNet	23.9

*2) **Global Vectors Model (GloVe)**:* GlobalVectors (GloVe) is a well-known model that learns vectors or words from their co-occurrence information. While word2vec is a predictive model - a feed-forward neural network that learns vectors to improve the predictive ability, GloVe is a count-based model. The statistics of word occurrences in a corpus is the primary source of information available to all unsupervised methods for learning word representations, and although many such methods now exist, the question still remains as to how meaning is generated from these statistics, and how the resulting word vectors might represent that meaning. In this section, we shed some light on this question. We use our insights to construct a new model for word representation which we call GloVe, for Global Vectors, because the global corpus statistics are captured directly by the model.

E. Intelligent Tutoring Systems (ITS)

ITSs are computer programs that use AI techniques to provide intelligent tutors that know what they teach, whom they teach, and how to teach. AI helps simulate human tutors in order to produce intelligent tutors. ITSs differ from other educational systems such as Computer-Aided Instruction (CAI). A CAI generally lacks the ability to monitor the learner's solution steps and provide instant help. For historical reasons, much of the research in the domain of educational software involving AI has been conducted under the name of Intelligent Computer-Aided Instruction (ICAI). In recent decades, the term ITS has often been used as a replacement for ICAI. The field of ITS is a combination of computer science, cognitive psychology, and educational research. The fact that ITS researchers use three different disciplines warrants important consideration regarding the major differences in research goals, terminologies, theoretical frameworks, and emphases among ITS researchers. Consequently, ITS researchers are required to have a good understanding of these three disciplines, resulting in competing demands.

III. PROPOSED ARCHITECTURE

Our approach reconciles classical slot filling approaches (that are generally better grounded in images) with modern neural captioning approaches (that are generally more natural

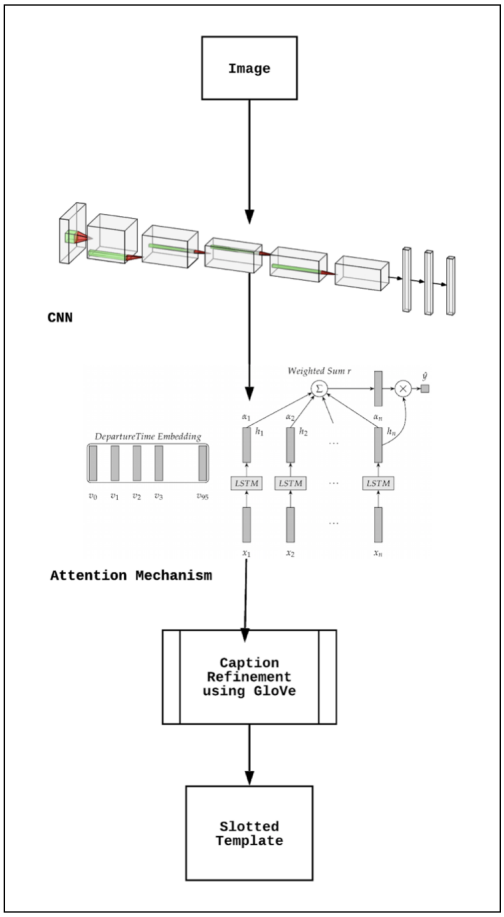

Fig. 5. Our Proposed Architecture

sounding and accurate). Our approach first generates a sentence 'template' with slot locations explicitly tied to specific image regions. These slots are then filled in by visual concepts identified in the regions by object detectors. It is a two-stage approach that first generates a hybrid template that contains a mix of (text) words and slots explicitly associated with image regions, and then fills in the slots with (text) words by recognizing the content in the corresponding image regions. The steps to generate a caption for a given image are as follows. First, we scan the entire image and form a set of visual or text words. This set of words basically lists each and every object encountered in the image. We make sure the entire image is scanned and there are no regions overlooked due to low quality or any other reason. Visual words are the objects that are encountered or nouns and textual words are the words that describe the properties using adverbs, prepositions, adjectives, etc. In the next step, we create a template using the textual words obtained previously and this template will have slots which are to be filled using the visual words. Before the final caption is ready we determine whether the visual word to be put at a given location in the template is singular or plural. After determining this we need to check for fine grained details. For example, the image might have a puppies in the scene. Here, the model just labels it as "dog". Here, we need to make it plural and the class should be set to puppy. This inturn returns a more detailed and accurate caption for the given image.

IV. Expected Outcome

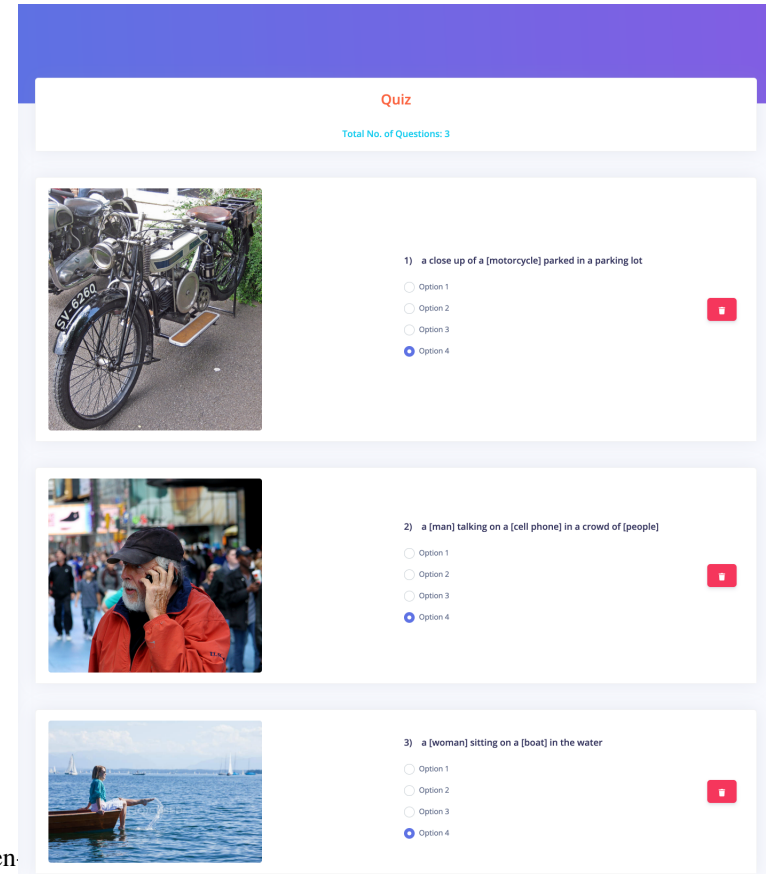

Fig. 6. Final Expected Outcome

References

[1] Jiasen Lu, Jianwei Yang, Dhruv Batra, Devi Parikh: Neural Baby Talk. In CVPR, 2018.
[2] Jeffrey Pennington, Richard Socher, Christopher D. Manning: GloVe: Global Vectors for Word Representation. In Computer Science Department, Stanford University, Stanford, CA 94305.
[3] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In NIPS, 2015
[4] Andrej Karpathy, Li Fei-Fei: Deep Visual-Semantic Alignments for Generating Image Descriptions. In CVPR, 2015
[5] Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, Yoshua Bengio: Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. arXiv Reprint
[6] Ali Alkhatlan, Jugal Kalita: Intelligent Tutoring Systems: A Comprehensive Historical Survey with Recent Developments. arXiv preprint arXiv:1812.09628, 2018.
[7] A. Das, H. Agrawal, C. L. Zitnick, D. Parikh, and D. Batra. Human Attention in Visual Question Answering: Do Humans and Deep Networks Look at the Same Regions? In EMNLP, 2016.