Automatic image captions using deep learning: Access in the Serbian language

SV61/2020 Jelena Miković

1 MOTIVATION

The motivation of the problem solved in the project lies in the need to enable effective recognition and describing the content of images in the Serbian language, which could have a wide practical application in different areas.

- Improving access to information: Automatically describing images allows users to get a quick and precise description of the content of the images in the Serbian language, which can significantly improve access information to people who prefer the Serbian language or have difficult access to visual contents.
- Development of accessible technologies: Implementation of such a system can contribute to development technologies that are accessible to visually impaired people, enabling them to obtain image descriptions via audio or text outputs.
- Improvement of image search and organization system: Automatic description of images can be useful for improving the search system and organization of images on the Internet or in local databases data, which would make it easier for users to find the desired images in the Serbian language.
- More efficient analysis of visual data: In business or research environments, automatic describing images can facilitate the analysis of large amounts of visual data, enabling faster and more effective decision-making based on visual information.

In short, solving the problem of automatic description of images in the Serbian language can have a wide scope practical application in improving access to information, developing accessible technologies, improvement of the search system and organization of images, as well as more efficient analysis of visual data.

2 RESEARCH QUESTIONS

Automatic description of images in Serbian aims to develop a system that can automatically generate descriptions of images in the Serbian language. Specifically, the problem is how to translate effectively visual information contained in the image into a textual description in the Serbian language. The input to the system is digital images in a format that is compatible with deep learning algorithms, while the output from the system is textual descriptions which accurately describe the content of the picture in the Serbian language.

This system will allow users to get a quick and accurate description of images in the Serbian language, which may have various applications in areas such as automated image labeling, helping people with with vision impairment, or improving the search and organization of large amounts of images.

The Flickr8k dataset is used to train and evaluate an automatic image description system. Consists of of 8,091 images, each with five captions describing the content of the image. The dataset provides a diverse set of images with multiple descriptions per image, making it suitable for training description generation models.

It is possible to download the dataset from the Kaggle site. Features are an image and 5 captions in English. Originally captions need to be translated with the help of an API or a ready-made model for translation.

3 RELATED WORK

3.1 EXISTING APPROACHES TO IMAGE CAPTIONING

3.1.1 CNN-RNN Architectures:

Many existing approaches utilize a combination of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). CNNs are employed as image feature extractors, capturing spatial information, while RNNs, typically LSTMs or GRUs, generate sequential text descriptions.

3.1.2 Transfer Learning with Pre-trained Models:

Transfer learning from pre-trained models like VGG, ResNet, or Inception has been widely adopted. These models are fine-tuned or used as feature extractors to obtain meaningful image representations, which are then fed into an RNN for caption generation.

3.1.3 Attention Mechanisms:

Attention mechanisms have been integrated into CNN-RNN architectures to improve caption quality. These mechanisms allow the model to focus on different parts of the image when generating each word, aligning the generated words more closely with the visual content.

3.1.4 Evaluation Metrics:

Evaluation metrics such as BLEU (Bilingual Evaluation Understudy), METEOR, ROUGE, and CIDEr are commonly used to assess the quality of generated captions by comparing them to human-written references. These metrics provide quantitative measures of caption accuracy and relevance.

3.1.5 Dataset Utilization:

Large-scale datasets like MSCOCO, Flickr30k, and Visual Genome are frequently used for training and evaluation. These datasets contain thousands of images paired with human-generated captions, enabling robust model training and evaluation across diverse visual and textual contexts.

3.2 COMPARISON WITH CURRENT METHODOLOGY

Feature Extraction: Similar to other approaches, your methodology uses a pre-trained VGG16 model to extract image features. This step ensures that the model captures meaningful visual information from images, which is crucial for generating accurate captions.

Sequence Processing: The use of LSTM for sequence generation aligns with standard practices in image captioning. LSTM networks are effective in learning sequential dependencies in textual data, allowing the model to generate coherent and contextually relevant captions.

Optimization Techniques: The exploration of different optimizers (SGD, Momentum, RMSprop, Adam) mirrors best practices in model optimization. This approach helps identify which optimizer configuration leads to the best performance in terms of caption quality as measured by BLEU score.

Error Analysis and Visualization: The discussion of error analysis and PCA visualization of image features demonstrates a comprehensive understanding of model performance and areas for improvement. This critical evaluation is essential for refining the model's architecture and training process.

4 METHODOLOGY

4.1 DATA PREPROCESSING

4.1.1 Reading and Cleaning Captions:

A text file containing image paths and corresponding Serbian captions is read into a pandas DataFrame. The captions are split into words and cleaned by removing punctuation and converting them to lowercase. Custom Serbian stopwords are removed to ensure only meaningful words are retained. Each cleaned caption is then formatted with 'startseq' at the beginning and 'endseq' at the end.

4.1.2 Feature Extraction Using VGG16:

The VGG16 model, pre-trained on ImageNet data, is loaded with its top layer removed to serve as a feature extractor. Images are resized to 224x224 pixels and preprocessed to match the input format required by VGG16. The extracted features from the images are then reduced to two dimensions using PCA for visualization.

4.2 MODEL TRAINING

4.2.1 Tokenization:

Captions are tokenized using the Keras Tokenizer, which converts words to unique integer indices and creates sequences of these integers. The maximum sequence length is determined based on the longest caption.

4.2.2 Creating Training Sequences:

For each caption, input-output pairs are created where the input is the image feature vector and the partial sequence of the caption, and the output is the next word in the sequence. These sequences are padded to ensure uniform length.

4.2.3 Splitting Data:

The dataset is split into training, validation, and test sets with a ratio of 70%, 15%, and 15%, respectively.

4.3 MODEL ARCHITECTURE

4.3.1 Image Feature Extractor:

A Dense layer with ReLU activation processes the image feature vectors followed by a Dropout layer to prevent overfitting.

4.3.2 Sequence Processor:

The tokenized caption sequences are passed through an Embedding layer followed by an LSTM layer. A Dropout layer is included to further prevent overfitting.

4.3.3 Combining Features and Sequence:

The outputs from the image feature extractor and the LSTM are combined using an Add layer. This combined representation is processed through another Dense layer to produce the final word predictions.

4.4 TRAINING AND EVALUATION

4.4.1 Training:

The model is trained using various optimizers (SGD and Adam). Each optimizer's performance is monitored, and the best model is saved based on validation loss.

4.4.2 Evaluation:

The models are evaluated using the BLEU score, which measures the quality of the generated captions by comparing them to the actual captions in the test set. The BLEU score is calculated for each optimizer to determine the best performing model.

5 DISCUSSION

5.1 EXPERIMENT SETUP

5.1.1 Testing Procedure:

The models were trained for 20 epochs with early stopping based on the validation loss. The training was conducted on a balanced dataset of image-caption pairs. Image features were precomputed using the VGG16 model, and captions were tokenized and padded for uniformity.

5.1.2 Evaluation Measure:

The BLEU score was used as the primary evaluation metric. It assesses how well the generated captions match the reference captions. The score ranges from 0 to 1, with higher scores indicating better performance.

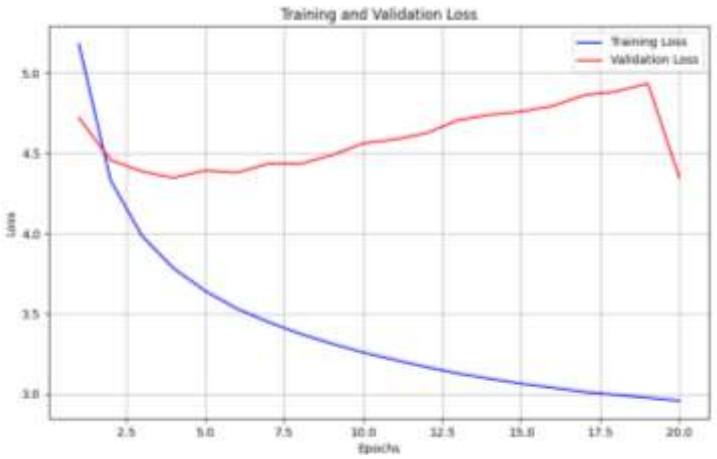
5.2 HYPERPARAMETER OPTIMIZATION

Various optimizers were tested to find the best configuration for the image captioning model. The learning rate and other parameters of the optimizers were tuned based on validation performance.

5.3 RESULTS

5.3.1 Model Performance:

The Adam optimizer achieved the highest BLEU score of 0.61 for some images, indicating it was the most effective at generating accurate captions. SG had some trouble with only achiving 0.35 as maximum. Gradient descent with momentum and RMSP were not trained in time and will be discussed in the final version of the code. The picture on the right is Adam's loss and val_loss.

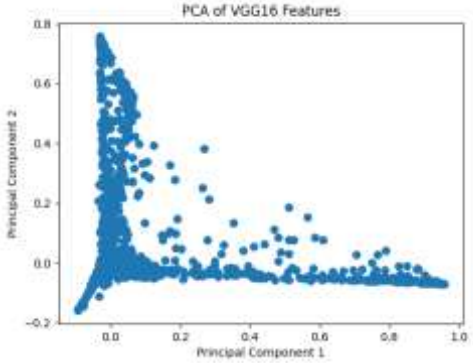


5.3.2 Error Analysis:

Common errors included generating repetitive words, missing key objects in images, or producing grammatically incorrect captions. These errors suggest the model could benefit from additional regularization techniques or more diverse training data.

5.3.3 Visualization:

PCA visualization of image features showed clear clustering, indicating the VGG16 model effectively captured relevant features. This clustering helps in understanding how well the model differentiates between different image classes.



6 REFERENCES

- https://medium.com/swlh/automatic-image-captioning-using-deep-learning-5e899c127387
- https://keras.io/api/applications/vgg/
- Soh, M. (2016). Learning CNN-LSTM architectures for image caption generation. Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep, 1.
- Wang, M., Song, L., Yang, X., & Luo, C. (2016, September). A parallel-fusion RNN-LSTM architecture for image caption generation. In *2016 IEEE international conference on image processing (ICIP)* (pp. 4448-4452). IEEE.
- https://towardsdatascience.com/gradient-descent-algorithm-a-deep-dive-cf04e8115f21
- https://towardsdatascience.com/gradient-descent-with-momentum-59420f626c8f
- https://deepai.org/machine-learning-glossary-and-terms/rmsprop
- https://medium.com/@nerdjock/deep-learning-course-lesson-7-4-adam-adaptive-momentestimation-e23434850bfc
- https://cloud.google.com/translate/automl/docs/evaluate