Deep Active Learning for Image Captioning in Remote Sensing

Project Report

WS 22/23
Mentored by Genc Hoxha and Lars Möllenbrok

Lorenz Hufe
Lara Wallburg

March 2, 2023

Contents

List of Figures

1 Introduction

Image captioning is the task of describing an input image with one or more sentences. As the predicted sentence is to describe not only visible objects, but also relationships between and attributes of such objects, image captioning is, not only structurally, different from mere image classification and can capture significantly more of its semantic content. Deep learning based systems for image captioning generally consist of an encoder and a decoder module. The encoder receives as its input an image and learns to extract its features. The extracted feature vector is then used as an input for the decoder, which learns to translate the latent vector into a natural language sentence. As the spatial resolution of sensors increases, remote sensing images get more detailed and contain more information which might better be captured by an image captioning system instead of an image classification system. This is one of the reasons why image captioning in remote sensing can be of interest. [1]

Active learning is a framework for supervised machine learning which can be used in cases where a lot of raw data might be available, but labeling it is costly. This makes it interesting for image captioning, as captions usually need to be manually added by one or more professionals. Active learning is an iterative process in which a model is initially trained on a small subset L of all available data. Afterwards, previously unlabeled samples are annotated and added to L, which is then used in a second training cycle. It is desirable that the newly annotated datapoints are those which are most informative for the training process. In our work, we will focus on the selection criteria which aim to choose such most informative datapoints. We will train a remote sensing image captioning system (RSIC) using an active learning framework with different criteria in place.

In the past decade, many machine learning approaches to the image captioning task have been proposed. What most solutions have in common is the usage of an image encoder for feature extraction followed by a language model for sentence generation as in the RSIC systems proposed by [3] and [1]. [4] describes the common usage of convolutional neural networks such as ResNet and VGGNet as encoders and a high prevalence of LSTM based decoders, for example in [5]. More recently, transformer based architectures have become of interest for encoding and/or decoding due to their prior successes in image classification and text translation [6] [7]. Active learning in image captioning has been described in [1], where uncertainty and diversity based criteria are used for sampling. [8] suggests that diversity criteria come short in image captioning as diversity in images doesn't account for textual information, while diversity in text is hard to caption due to the sequential structure of the output sentence. [9] highlights that uncertainty sampling can be problematic in deep

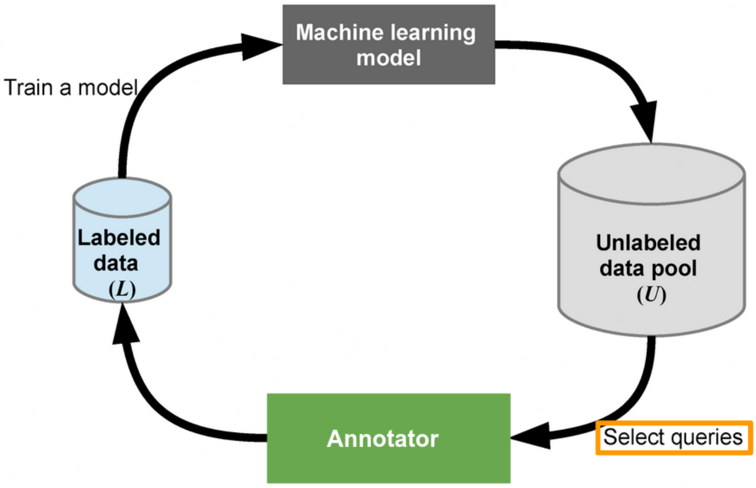


Figure 1: An active learning cycle

Source: [2] (modified)

active learning because deep neural networks tend to be overly confident in their predictions.

2 Methodology

Uncertainty based sampling relies on the assumption that samples which are located close to a decision boundary are hard for the model to predict. The argument here is that difficult samples are particularly informative for the training process as they might contain features which the model does not capture already. The selection of those is based on the uncertainty of the model, which is estimated using different approaches, two of which we chose to investigate. The approaches have in common that they are calculated based on the logits of the final prediction.

The simplest algorithm is to pick those samples where the probability of the returned prediction is lowest. We call this least confidence sampling.

The second algorithm consists of calculating the difference between the prediction probability and the second most likely prediction probability. Those samples which are lowest in difference get added to the training set. This margin of confidence approach assumes that those are the samples where the decision wasn't clear but torn between different results.

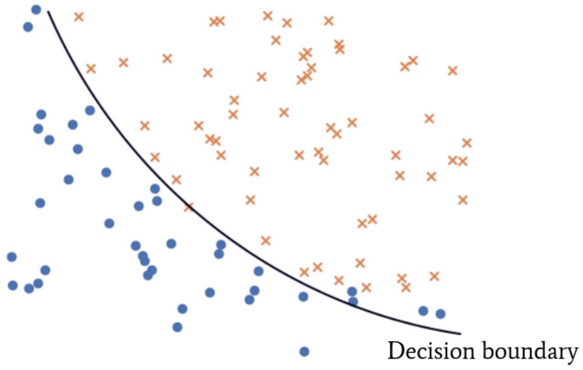


Figure 2: Decision boundary of a binary classification problem

Diversity based sampling is a sampling paradigm which assumes that the performance of a model improves when it is trained on a diverse dataset. This is best understood imagining the opposite: In a multiclass problem, a model which was only trained on one class would perform worse than one trained on all classes as it is likely to overfit on the class it has seen. Furthermore, it is assumed that the training would quickly stagnate as adding more datapoints of the same class is unlikely to introduce any new information to the model. In a real world scenario, such classes are of course not explicitly given for an unlabeled dataset. Diversity based sampling therefore relies on finding implicit classes in the form of underlying structures and correlations of a dataset which can be discovered through clustering of the data. Clusters are formed based on features extracted from the data, which, in multimodal problems like image captioning, can be obtained from the image itself, or from the generated text. For image based clustering, the feature space made out of the latent vectors directly generated by the image encoder is clustered. For text based clustering, features are to be extracted from the generated caption through a pretrained language model, for which we used the BERT model [10]. In both cases, the number of clusters is equal to the number of samples to be labeled, and one sample per cluster is labeled.

Uncertainty and diversity based sampling can be fused by applying the methods consecutively. Finding the most uncertain samples first, then clustering them and selecting one sample per cluster can be desirable because it combines the information contained in both approaches and reduces the costs of the computationally intensive clustering significantly. For a desired number of new samples S, we selected 4S least certain datapoints, clustered them into S clusters, and then randomly selected one datapoint from each cluster. Applying clustering first and then selecting the least certain datapoints per cluster doesn't offer the advantage of complexity reduction but is still interesting to implement in order to learn more about the behaviour of the proposed active learning strategies.

As the aim of our project was to compare different active learning criteria, and literature on active learning in image captioning is quite sparse, we made a broad attempt at gaining an understanding the influence of those criteria. The following is a list of eight basic experiments we ran.

1. Baseline: Random selection
   Simulating an active learning process by randomly sampling from the unlabeled dataset serves as a baseline for the smapling critera.
2. Upper bound: Full dataset
   Training on the full dataset without sampling any data gives an upper bound of the performance which can be achieved.
3. Least confidence
4. Margin of confidence
5. Image diversity
6. Text diversity
7. Fusion: Margin of confidence and image diversity
8. Fusion: Image diversity and margin of confidence

3 Setup

3.1 Image Captioning System Architecture

The image captioning system used in this study consists of a Vision Encoder and a Text Decoder. The Vision Encoder is implemented using a Vision Transformer called ViT-16-Base [11], which was pretrained on the ImageNet 21-k dataset. The Text Decoder is implemented using GPT-2 [12], which was pretrained on the WebText dataset. The encoder and decoder were also pretrained together on the Coco-Captions [13] dataset.

This architecture is available on huggingface.co under `https://huggingface.co/nlpconnect/vit-gpt2-image-captioning`.

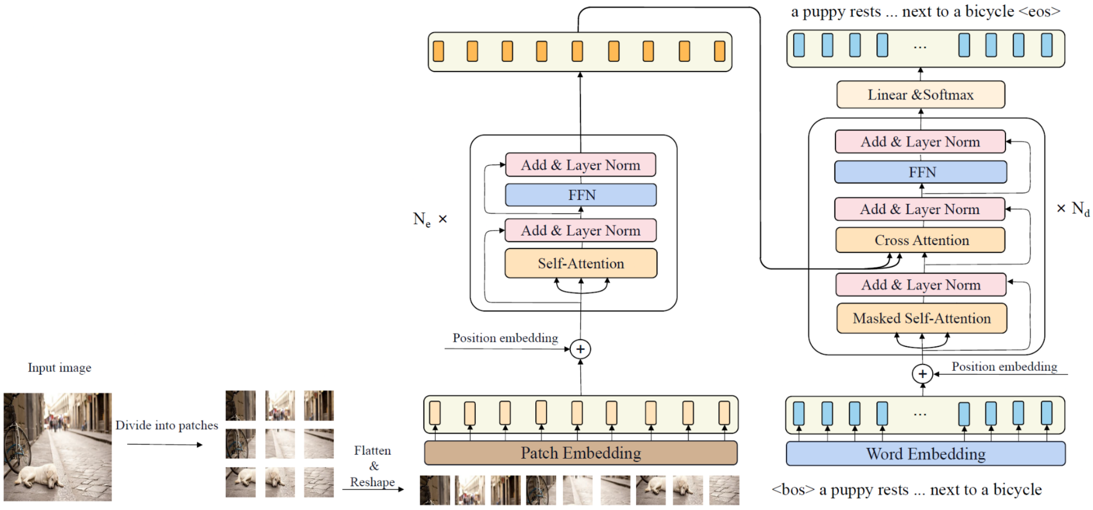


Figure 3: Encoder-decoder pipeline

The image is processed by a Vision Encoder, which extracts its features, and then fed into the Text Decoder, which generates the natural language sentence. The encoder and decoder are pretrained together on the Coco-Captions dataset.

3.2 Hyperparameters

In the first active learning cycle, we used 5% of the dataset for training. Running a total of nine cycles, adding another 5% of the initial dataset at each cycle except the last, we progressed to training on 45% of the data in the last cycle. In each cycle, the model weights were reset to their pretrained weights and then optimized for ten epochs. We used a constant learning rate of 0.0001 and a batch size of 12.

Hyperparameter	**Value**
Loss Function	Cross Entropy Loss
Learning Rate	0.0001
Batch Size	12
Number of Cycles	9
Inital Dataset Size	5%
Dataset Increase per Cycle	5%

3.3 Dataset

We used the NWPU-Captions dataset, to our knowledge the largest dataset for remote sensing image captioning. [3]. With 45 scene classes, each containing 700 RGB images, the total number of images is 31500. Each image is of size 256x256px, with spatial resolutions ranging from less than 30 m/px up to 0.2 m/px. The images were drawn from the NWPU-RESISC45 dataset [14], which in turn used Google Earth as its source. Through manual annotation by seven experts, each image has been assigned five different descriptive English sentences, which makes for a total number of 157500 sentences in the dataset. The dataset is already split into training, validation and test subsets which include 25200, 3150 and 3150 images, respectively.

3.4 Metrics

We evaluated our model based on three metrics which are common for machine translation evaluation. Each metric is computed by comparison of a generated caption and a set of reference sentences. The respective scores are calculated for each reference and only the best is returned.

The BLEU-4 score [15] is a precision-based metric for calculating the similarity between machine-generated sentences and reference sentences. It measures the overlap between 4-grams (sequences of four words) in the machine-generated text and the reference text, and includes a penalty term to discourage the generation of long sentences.

In contrast, the METEOR score [16] is a more complex metric that takes into account precision, recall, and alignment between the machine-generated text and the reference text. It employs a combination of unigram, bigram, and trigram matches, and also includes several other features such as stemming, synonymy, and paraphrasing. The Meteor score is widely used in machine translation and other natural language processing tasks as an alternative or complementary metric to BLEU.

The ROUGE-L metric [17] is based on the longest common subsequence (LCS) between the machine-generated text and the reference text. The LCS is defined as the longest sequence of words that appears in both texts in the same order, with possible gaps between words. ROUGE-L measures the overlap between the LCS in the machine-generated text and the LCS in the reference text, and normalizes it by the length of the reference text.

ROUGE-L is one of the several ROUGE metrics that are commonly used for evaluating the performance of text summarization and machine translation systems. The ROUGE metrics employ different variants of n-gram matching and longest common subsequence measures, and have been shown to correlate well with human judgements of text similarity and quality.

4 Results

4.1 Quantitative Evaluation

The following figures show the metrics which were achieved on the validation set. Each with the percentage of labeled training data which was used at each cycle, ranging from 5% to 45%, represented on the x-axis. Each figure includes the baseline scores.

4 shows the random baseline as well as the upper limit as explained in our methodology. The METEOR, ROUGE-L and BLEU scores of 0.76, 0.8 and 0.55 could not be achieved by training on randomly sampled 45% or less of the data. At 40%, the performance of the baseline was best.

5 shows the performance of the uncertainty based sampling, which we implemented based on the minimum confidence and minimum margin of confidence. Minimum confidence sampling shows to consistently perform slightly worse than the baseline. The margin of confidence seems to be a more successful uncertainty measure, but the performance is similar to the random baseline. 6 shows the evaluation of our two proposed diversity criteria. The influence of using text or image features is not perceptible in the validation scores, and neither one is successful in beating the baseline. While all previous criteria showed performances similar to the baseline, the fusion of the two most successful variations, margin on confidence sampling and image diversity sampling, stays far behind in terms of all evaluated metrics.

In addition to evaluating our model on the validation set as in 4, we used the test set to evaluate the model trained on 100% of the data. In comparison to MLCA-Net [3],

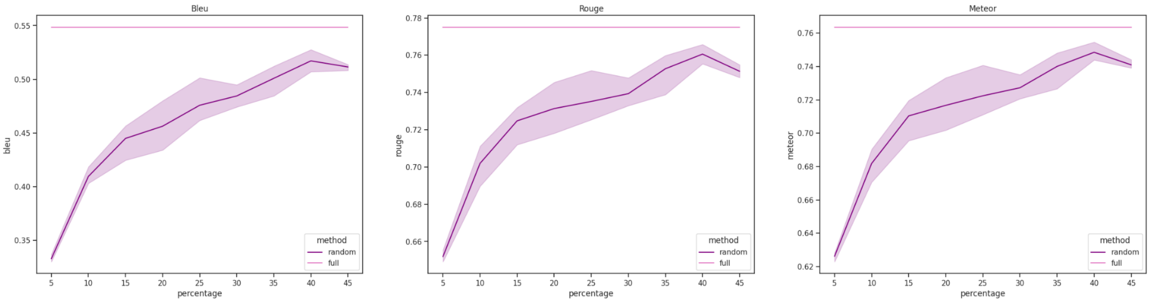


Figure 4: Evaluation of random sampling and full dataset training

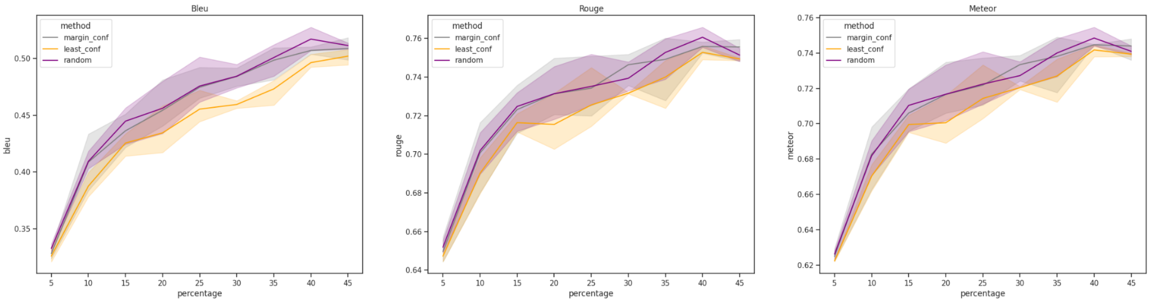


Figure 5: Evaluation of least confidence and margin of confidence based sampling

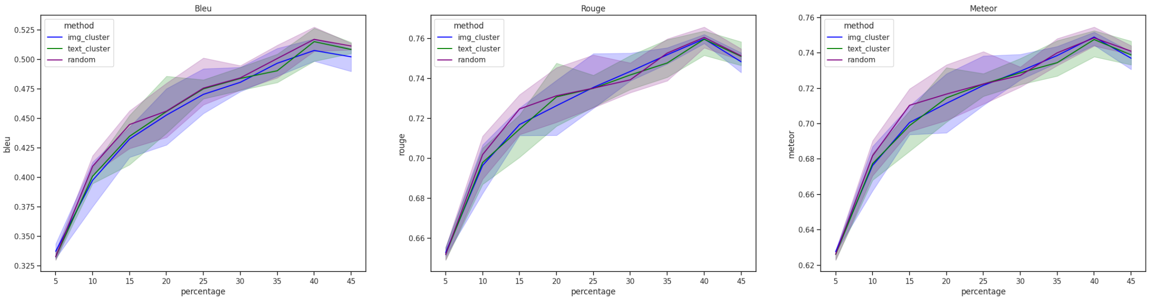


Figure 6: Evaluation of image diversity and text diversity based sampling

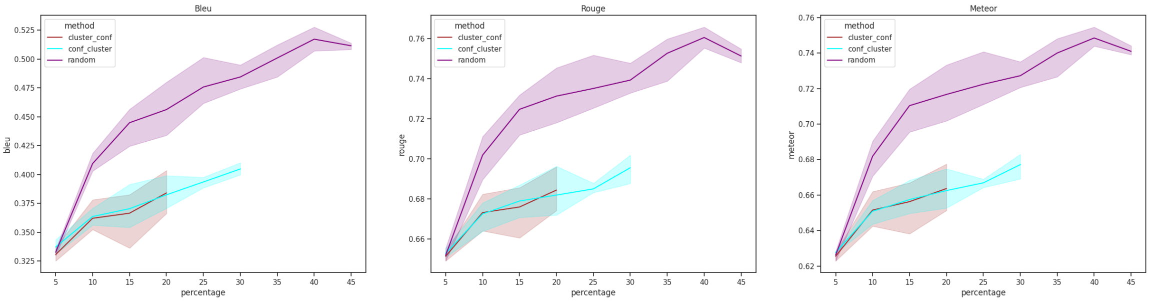


Figure 7: Evaluation of fused sampling methods

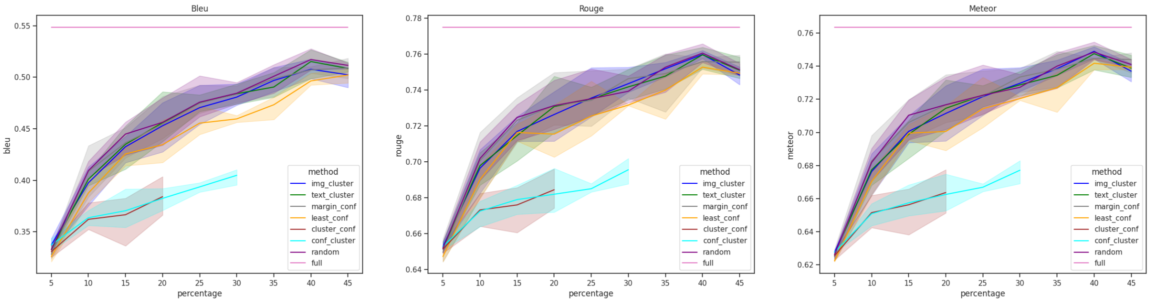


Figure 8: Comparison of all criteria

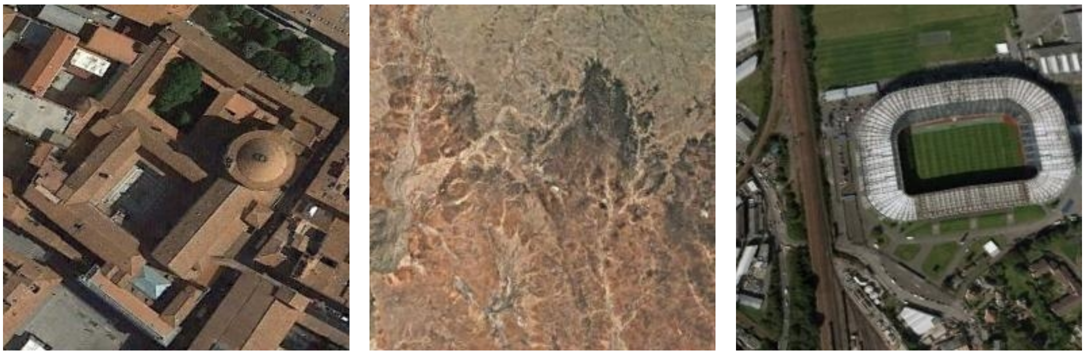
to our knowledge the only captioning system trained on tested on the NWPU-Captions dataset, our model achieved higher scores for each considered metric.

	BLEU-4	Meteor	ROUGE-L
MLCA-Net [3]	0.478	0.337	0.601
Ours	**0.55**	**0.76**	**0.73**

Table 1: Test results after training on 100% of data.

4.2 Qualitative Evaluation

For a qualitative evaluation of the captioning, we selected some example images from the validation set on which metrics similar to those achieved by margin of confidence or diversity based sampling were achieved.



(a) The church with a tower pointed is the rest of the church has some orange sloping roofs.

(b) The desert has yellow soil and some black hills. the desert terrain is rugged.

(c) There stadium stadium has dense residential areas on three sides.

Figure 9: Input images with their predicted captions

5 Discussion

5.1 Confidence Criteria

As our model outputs sentences instead of single decision items, the calculation of confidence was non-trivial here and took some trial and error. In a single-label classifier,

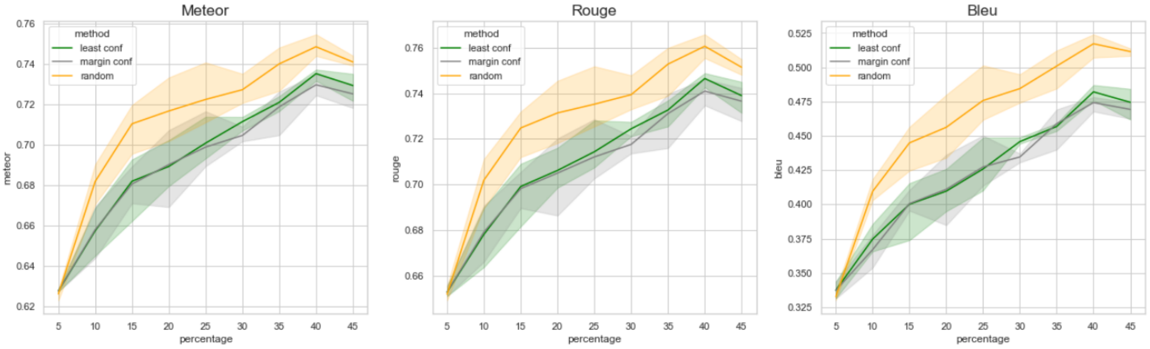


Figure 10: Evaluation of least confidence sampling based on the sentence mean

for example, the confidence estimation is merely based on the probability (or margin of probability) with which the label was emitted. As in our case, each word of the output sentence is emitted with a certain probability, our first approach was to calculate the confidence of each word C(W) and then average all word confidences to get a sentence confidence C(S).

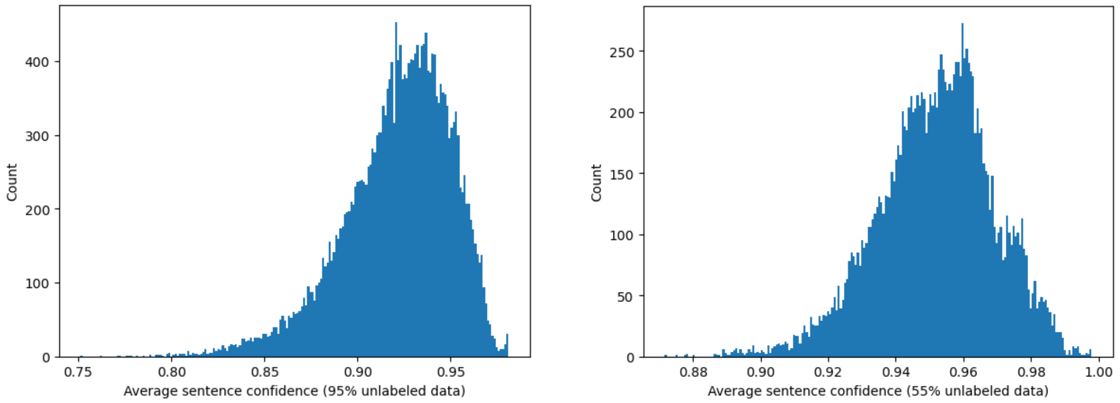
$$C_{\text{mean}}(S) = \frac{\sum_{i=1}^{N} C(W_i)}{N}, \text{ with } S = \{W_1, W_2, ..., W_N\} \tag{5.1.1}$$

As can be seen in 10, this resulted in a model performance which was worse than the performance when sampling randomly. To investigate this, we extracted two histograms showing how often each confidence score was produced. Even after just one cycle of training on 5% of data, all confidences were already surprisingly high 11a. The distribution didn't significantly change when training on 45% data 11b, even though one might expect the model to gain confidence by further training. This behavior can be explained by 12 which shows the average confidence per word in a sequence.

We obtained the data by inference on the validation set on the fully trained model. The first words of a sentence are emitted with a lower confidence of higher variance, while the word confidences increase towards the end of the sentence. Due to the recurrent sentence generation, words are strongly depended on previous words, which means that the prediction of the first words influences the sentence the most and is usually less confident. As a proof of concept, we implemented a naive method in which we only considered the lowest word confidence per sentence.

$$C_{\text{min}}(S) = min(C(W_1), C(W_2), ..., C(W_N)) \tag{5.1.2}$$

Considering the sentence minimum instead of the sentence average confidence makes for a noticeable difference in emitted confidence histograms. Rather than making



(a) After one cycle

(b) After seven cycles

Figure 11: Distribution of average sentence confidences

predictions with high confidence which changes little, 13a shows that after the first training cycle, the minimum confidences are relatively low and broadly distributed between 0.1 and 0.6. After the last training epoch **??**, the histogram gets much more narrow, showing a distribution closely around 0.4. This shift implies a strong correlation between training progress and model confidence. As expected, this improved the performance significantly, but didn't cause a performance which could beat our random baseline 5. In order to caption more of the sentence structure, we propose a third algorithm based on a weighted sentence mean, where the weight f_i of each word confidence depends on the position in the text and its associated variance of confidences. We hope that this could be an active learning criterion which performs better than the baseline.

$$C_{\text{weight}}(S) = \frac{\sum_{i=1}^{N} f_i * C(W_i)}{N} \tag{5.1.3}$$

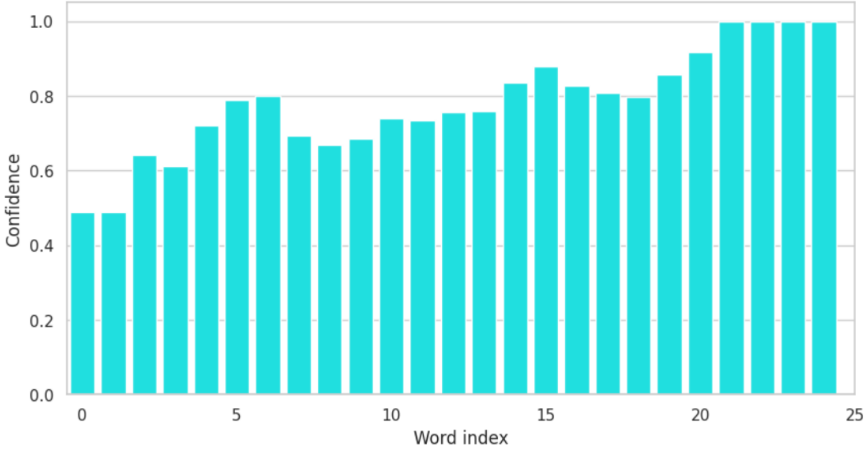
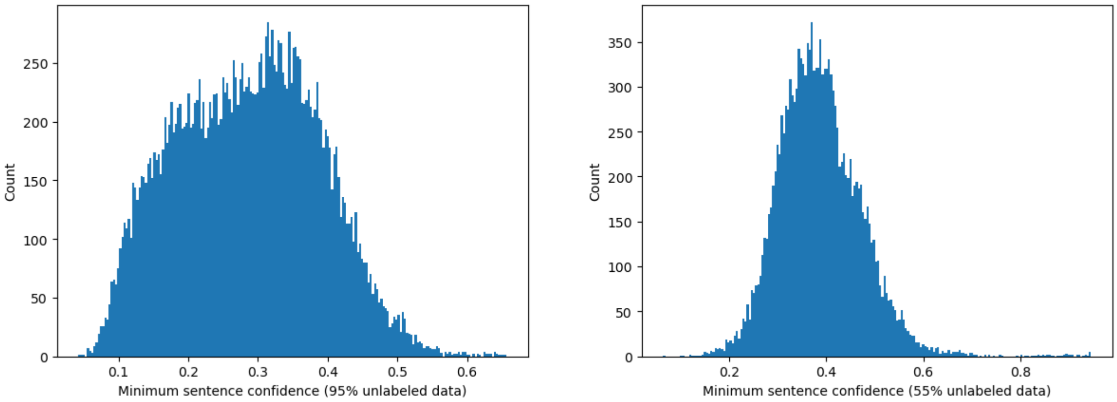


Figure 12: Average confidence of a trained model at each word index



(a) After one cycle

(b) After seven cycles

Figure 13: Distribution of minimum sentence confidences

5.2 Diversity Critera

Our findings showed that a diversity-based selection strategy was not necessarily better than random selection. We identified two phenomena that could explain this. Our first hypothesis is that the latent spaces used in our project were not semantically rich enough to ensure that k-means clustering could find hidden classes. Secondly, random selection is likely to already cover a diverse range of samples across the dataset.

To gain more insight into the latent spaces used, we employed UMAP [18], a dimensionality reduction method, to plot the latent spaces in a two-dimensional scatter plot (see figure 14). The plot revealed that semantically similar generated captions were clustered closely together in the BERT latent space. Upon inspecting the generated clusters, we found that most clusters contained mostly samples of one class. We concluded that the algorithm works as intended, but simply does not yield better performance than random selection.

We argue that random selection itself is already diverse enough for active learning in image captioning. By selecting 5% of 31,500 images, we selected 1,575 unlabeled images from 45 different scene classes. Since each of these 45 different scene classes contained the same amount of images, random selection is self-regulating towards a diverse selection. Selecting one sample of a class makes it less likely to sample this class in the next selection, thus random selection slightly favors diversity.

As a lower bound, we modeled the random selection strategy using a uniform distribution to draw a sample x. For instance, the probability of selecting class **airport** is $p(x = \text{airport}) = \frac{1}{45}$, implying that the probability of not drawing class **airport** is $p(x \neq \text{airport}) = \frac{44}{45}$. Therefore, the probability of not selecting any samples of class **airport** in one selection cycle would be $(\frac{44}{45})^{1575} = 4.249 \cdot e^{-16}$. This probability is roughly equivalent to winning the lottery (6 out of 49) twice in a row.[1]. Hence, random selection in our setup will very likely draw a sample from every class, making it a reasonably diverse selection strategy.

However, it would be interesting to test the diversity criterion when fewer samples are selected in every cycle. For instance, the probability that one class is not selected in random sampling with a selection size of 64 would be $(\frac{44}{45})^{64} = 0.237$, which is much more reasonable than our chosen selection size.

5.3 Fusion of Criteria

The results from the fused criteria stayed below expectations throughout the course of our studies. The results presented in the final presentation were indeed caused by an bug in the code. But even after running the experiments again the results stayed significantly below the random baseline.

In an attempt to understand the reason for the bad performance of the combined methods we analyzed the number of selected images per class and found a pattern of image selection correlated to the lexicographic ordering of the scene class names 15.

We have found the implementation detail which could be causing this error, but due to the long run times we unfortunately could not test our hypothesis. Preliminary

[1] The probability of winning the lottery 6 out of 49 twice in a row is $5.114 \cdot e^{-15}$

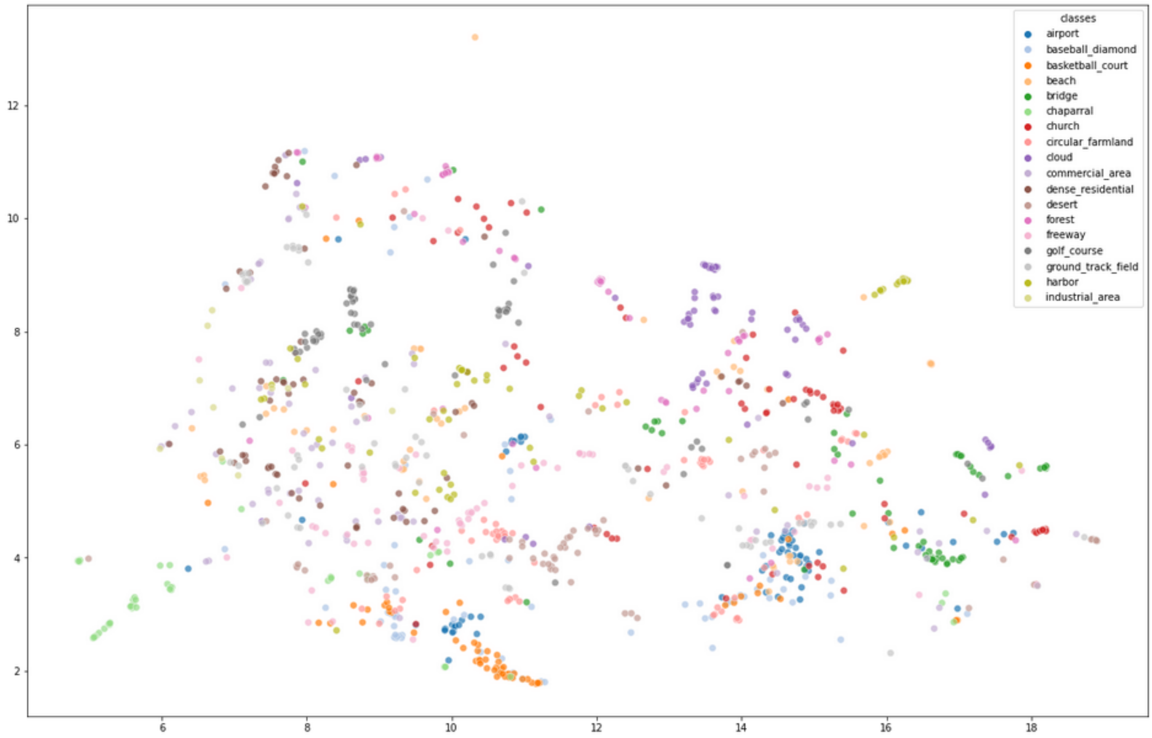


Figure 14: Latent space of BERT model

This figure shows a cluster of data points in the latent space of the BERT language model. Each data point represents an image and its associated class. The BERT embedding of generated captions show meaningfull relation, in a way, that samples from within a class are close to each other

results of the running experiments suggest slightly better performance than shown in the results section, but no improvement over the random baseline.

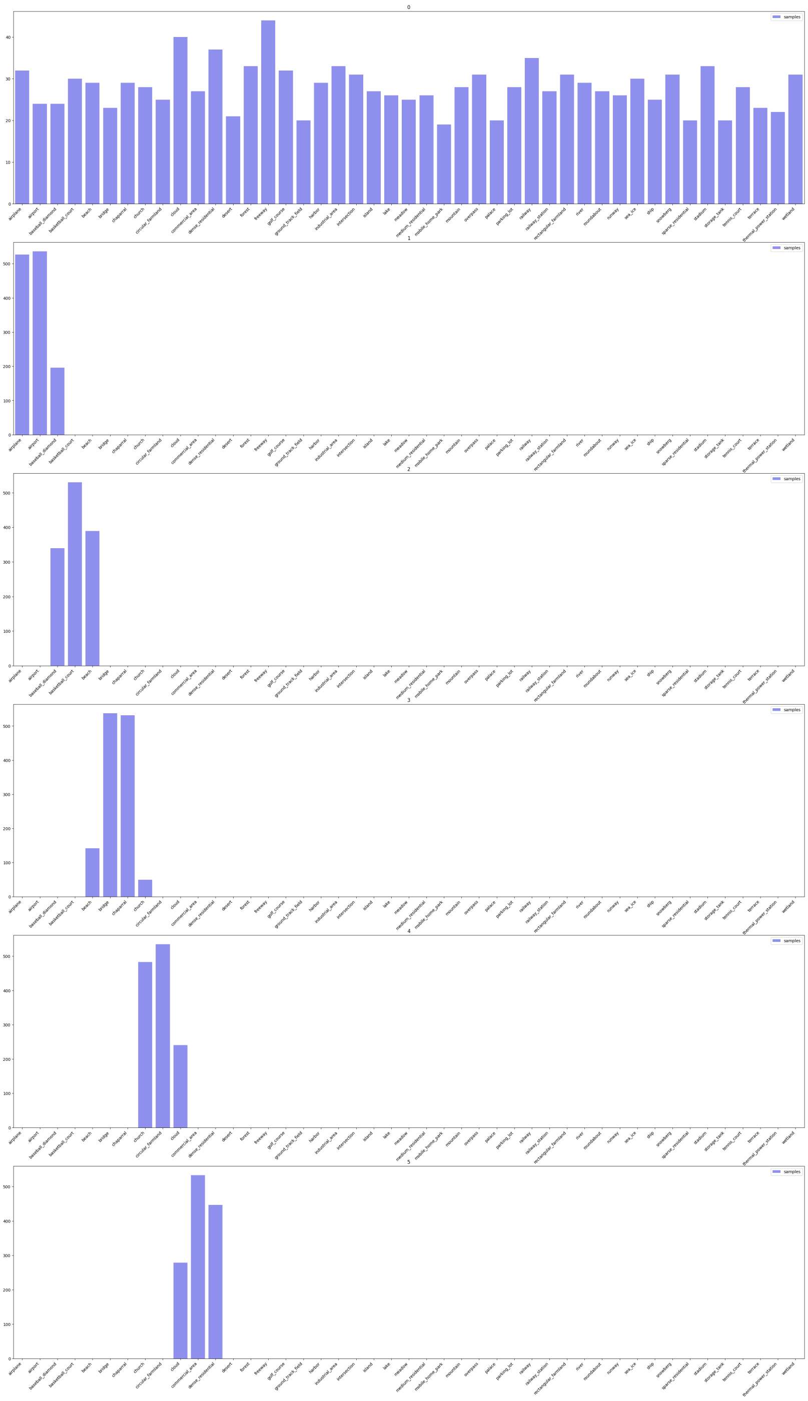


Figure 15: Samples were selected in lexicographic ordering, most likely due to a bug in the implementation

5.4 Full Dataset

While our primary objective was to compare various active learning criteria, training our captioning system on the full dataset without active learning also yielded some interesting insights. Notably, without hyperparameter tuning, our model performed better than MLCU-Net, the state-of-the-art model for image captioning on NWPU-Captions. Our chosen pipeline has two key benefits which may have helped achieve a high performance. Firstly, recent research has shown that transformer based architectures regularly outperform convolutional networks such as MLCU-Net. Secondly, we were able to benefit from the pretrained weights of our system, where not only the encoder and the decoder where pretrained, but also their concatenation. This already made the parameters of our networks well-suited for image captioning and consequently, we were able to achieve very good results after only ten epochs of training. 16 shows the distribution of meteor scores across the 45 scene classes. The plot suggests that our captioning system is quite robust to the content of the input image.

To further explore our selected metrics, we plotted the distribution of evaluation scores 17. While the ROUGE and Meteor scores show a distribution which corresponds well to the model performance, we noticed that the BLEU scores are distributed quite broadly and show a high count of 0-values despite the generally good model performance. Being based on the precision of 4-grams in a sentence and only allowing exact word matches, even small changes can result in a BLEU score of 0. As an example, one might consider a reference and a predicted sequence of seven words, where all words except for the fourth match. In this case, the score for each 4-gram would be zero, even though the prediction might still be close to the reference or even have the same meaning. Because of the low robustness of BLEU-4, and for its lack of paralleling human perception, we conclude that it should not be used alone for the evaluation of image captioning systems and favor more stable and flexible metrics such as ROUGE and METEOR which both have been described as correlating well with human judgement [17] [16].

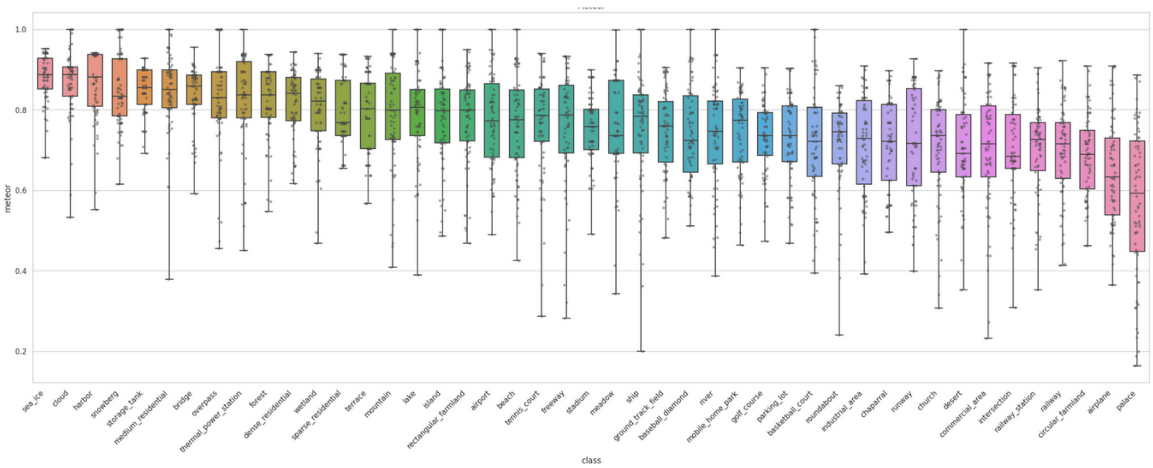


Figure 16: Meteor scores by scene class

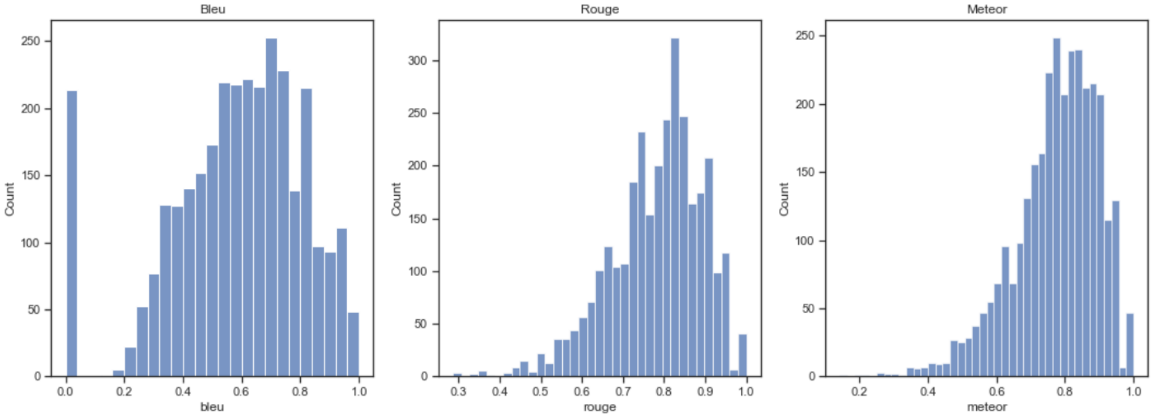


Figure 17: Distribution of evaluation scores

5.5 Future Work

Since we were unable to beat the random selection baseline with our chosen selection criteria, we propose further approaches that could improve the performance of selection by diversity or uncertainty. To improve uncertainty based sampling, we propose implementing a weighted average based uncertainty criterion as described in 5.1. We suspect that averaging uncertainty values across a sentence and weighting each value according to its position in the sentence would improve performance as this approach takes into account the recurrent logic on which a sentence is predicted in a GPT2 network. To render diversity based sampling more effective, we propose to directly utilize and enforce the assumption that classes which don't perform well in terms of evaluation metrics should be sampled more frequently, an approach which so far is only followed implicitly. As such metrics obviously can't be computed for the unlabeled data, our proposal is to cluster the the whole dataset (that is, the labeled and unlabeled training data) and assign the METEOR score of the labeled datapoint closest to an unlabeled datapoint to this unlabeled datapoint. Our third suggestion is to take into account the activations of the model. If the activation for the prediction of unlabeled data is different from previously seen activation, we would assume that the datapoint diverges from other datapoints and is therefore handled differently by the model. Labeling such points would follow a mixture of the uncertainty and diversity paradigms. In our pretrained network, we expect that it will be sufficient to consider the activations which happen at the edge between the encoder and decoder.

Besides implementing new sampling methods, we think it would be of interest to experimentally test our assumption that random sampling performed well in our methods due to the balanced classes in the dataset. Training on a dataset with artificially injected class imbalance could offer insights in how the performance of different sampling methods corresponds to the distribution of the dataset. Futhermore it would be interesting to see, how the selection strategies behave, if only a small batch of images are labeled at each cycle.

Lastly, we want to highlight again the promising results we were able to score using the full dataset without an active learning framework, even without any hyperparameter tuning. Tuning hyperparameters such as the arbitrarily chosen learning rate could offer further improvements, while the validity and robustness of the pipeline can be assessed by testing it on further datasets. The success of such work could make for a interesting candidate for a remote sensing image captioning system.

5.6 Conclusion

To sample the most informative images in our active RSIC system, we tested four different single sampling criteria, two of which were uncertainty based and two of which were diversity based. We were not able to find a difference in training performance between using those criteria or random sampling and proposed adaptions which could be able to overcome the limitations of those criteria, such as taking into account word order for uncertainty calculation, or combining a clustering algorithm with a performance measure. The fusion of two criteria worked unexpectedly badly and needs to be further investigated in the future. Our transformer based pipeline achieved better results than the RSIC system which was previously proposed for the NWPU-Captions dataset.

References

[1] Genc Hoxha, Andrea Munari, and Farid Melgani. A new active image captioning fusion strategy. In *2022 IEEE Mediterranean and Middle-East Geoscience and Remote Sensing Symposium (M2GARSS)*, pages 1–4. IEEE, 2022.

[2] Akram M. Radwan. Human active learning. *https://www.intechopen.com/chapters/63962 (accessed 11.12.2022).*

[3] Qimin Cheng, Haiyan Huang, Yuan Xu, Yuzhuo Zhou, Huanying Li, and Zhongyuan Wang. Nwpu-captions dataset and mlca-net for remote sensing image captioning. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–19, 2022.

[4] MD. Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. A comprehensive survey of deep learning for image captioning. *ACM Comput. Surv.*, 51(6), feb 2019.

[5] Qi Wu, Chunhua Shen, Peng Wang, Anthony Dick, and Anton van den Hengel. Image captioning and visual question answering based on attributes and external knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1367–1381, 2018.

[6] Chenyang Liu, Rui Zhao, and Zhenwei Shi. Remote-sensing image captioning based on multilayer aggregated transformer. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022.

[7] Sen He, Wentong Liao, Hamed R. Tavakoli, Michael Yang, Bodo Rosenhahn, and Nicolas Pugeault. Image captioning through image transformer. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, November 2020.

[8] Beichen Zhang, Liang Li, Li Su, Shuhui Wang, Jincan Deng, Zheng-Jun Zha, and Qingming Huang. Structural semantic adversarial active learning for image captioning. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM ’20, page 1112–1121, New York, NY, USA, 2020. Association for Computing Machinery.

[9] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020.

[12] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[13] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

[14] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *CoRR*, abs/1703.00121, 2017.

[15] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[16] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

[17] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[18] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.