Arabic Image Captioning using Pre-training of Deep Bidirectional Transformers

Anonymous ACL submission

Abstract

Image captioning is the process of automatically generating a textual description of an image. It has a wide range of applications, such as effective image search, auto archiving and even helping visually impaired people to see. English image captioning has seen a lot of development lately, while Arabic image captioning is lagging behind. In this paper, we developed and evaluated several Arabic image captioning models with well-established metrics on a public image captioning benchmark. We initialized all models with transformers pre-trained on different Arabic corpora. After initialization, we fine-tuned them with image-caption pairs using a learning method called OSCAR. OSCAR uses object tags detected in images as anchor points to significantly ease the learning of image-text semantic alignments. In relation to the image captioning benchmark, our best performing model scored 0.39, 0.25, 0.15 and 0.092 with BLEU-1,2,3,4 respectively, an improvement over previously published scores of 0.33, 0.19, 0.11 and 0.057. Beside additional evaluation metrics, we complemented our scores with human evaluation on a sample of our output. Our experiments showed that training image captioning models with Arabic captions and English object tags is a working approach, but that a pure Arabic dataset, with Arabic object tags, would be preferable.

1 Introduction

The amount of available digital images has increased enormously and captions help us understand and interpret them. While manual captioning is a tedious task, automatic image captioning uses algorithms to extract meaningful information about the content of an image and generate a human-readable sentence from this information.

State-of-the-art automatic image captioning networks are today trained on English corpora. The resulting captions could then be translated into Arabic using a neural machine translation (NMT) model. However, ElJundi et al. (2020) showed the necessity of an end-to-end Arabic image captioning system, which eliminates sources of error that may come from the unique sentence structure and complex morphology of the Arabic language.

Attai and Elnagar (2020), in a survey on the current state of Arabic image captioning systems, conclude that research conducted for Arabic image captioning is very scarce and that it can mainly be attributed to the lack of publicly available datasets. They also stress that few Arabic image captioning research projects utilized attention mechanisms to focus on the important parts of the image. Such attention mechanisms shall contribute to the caption generation process and give better results.

In their survey, Attai and Elnagar did not mention the transformer architecture as proposed by Vaswani et al. (2017), which is solely based on attention mechanisms. Moreover, transformers in natural language models are gaining more popularity as these models create new state-of-the-art results on different benchmarks, including the OSCAR English image captioning model (Li et al., 2020). This system uses object tags detected in images as anchor points to significantly ease the learning of image-text semantic alignments.

To the best of our knowledge, no transformer-based model for Arabic image captioning had been put to the test. In this paper, we describe an approach to switch the language models of OSCAR with pre-trained Arabic and multilingual ones, then train them on public Arabic benchmark datasets.

The main contributions of this work can be summarized as follows: (*i*) We evaluate transformer-based Arabic image captioning and compare our results to previous ones. (*ii*) In relation to the public image captioning benchmark, one of our best performing models scored 0.39, 0.25, 0.15 and 0.092 with BLEU-1,2,3,4 respectively, an improvement over previously published scores of 0.33, 0.19, 0.11 and 0.057. (*iii*) We show that training image cap-

tioning models with Arabic captions and English object tags is a working approach, but that a pure Arabic dataset, with Arabic object tags, is preferable.

2 Related Work

In this section, we summarize recent developments in English image captioning and comment on the current state of Arabic image captioning.

2.1 English Image Captioning

Attention is a technique in neural networks that mimics cognitive attention, and has shown great success in image captioning models ever since Xu et al. (2015) introduced an attention-based model that automatically learns to describe the contents of images. You et al. (2016) developed an algorithm that learns to selectively attend to semantic concept candidates and combine them with hidden states and outputs of recurrent neural networks. Huang et al. (2019) take the attention concept one step further in their work, where they propose an "Attention on Attention" (AoA) module, which extends the conventional attention mechanisms to determine the relevance between attention results and queries.

State-of-the-art image captioning today is based on transformers, an architecture that builds solely on attention mechanisms. Zhou et al. (2019) presented a unified vision-language pre-training (VLP) model which can be fine-tuned for both image captioning and visual question answering (VQA) tasks. Li et al. (2020) presented a new learning method OSCAR (**O**bject-**S**emanti**c**s **A**ligned P**r**e-training), and showed that learning of cross-modal representations can be significantly improved by introducing object tags detected in images. These object tags are used as "anchor points" during training to ease the learning of semantic alignments between images and texts. Zhang et al. (2021) studied improved visual representations, dubbed VinVL, and utilized an upgraded approach, dubbed OSCAR+, to pre-train transformer-based VL fusion models. They then fine-tuned the models on various VL benchmarks and created new state-of-the-art results on seven public benchmarks, including image captioning on the COCO Caption benchmark (see Section 3.1). VinVL has since its release been surpassed by other VLP models, for example LEMON (**L**arg**E**-scale i**M**age capti**ON**er) (Hu et al., 2021) which studies the scaling behavior of VLP for image captioning.

By the time of this work, VinVL was the state of the art and in this paper we utilized OSCAR with VinVL on Arabic image captioning.

2.2 Arabic Image Captioning

Arabic image captioning (AIC) introduces additional challenges compared to English captioning. In a survey on the state of AIC, Attai and Elnagar (2020) conclude that research conducted for Arabic image captioning is very scarce and that it can mainly be attributed to the lack of publicly available datasets. The Arabic language is also known for its morphological complexity, and a variety of dialects, which makes it harder to process.

Jindal leveraged the heavy influence of root-words to generate captions of an image directly in Arabic using root-word based recurrent neural networks (Jindal, 2017, 2018). They also reported the first BLEU score for direct Arabic caption generation, from experimental results on datasets from various Middle Eastern newspaper websites and the Flickr8k dataset (see Section 3.2).

Al-muzaini et al. (2018) developed a generative merge model for Arabic image captioning based on a deep RNN-LSTM and a CNN model. They used crowd sourcing to translate samples from two image captioning benchmarks: MS COCO and the Flickr8k dataset. They used a relatively small training set (2400 images) from an unpublished dataset. To reduce the risk of overfitting, ElJundi et al. (2020) developed an annotated dataset for Arabic image captioning (Flickr8k), which, as of today, remains the only public benchmark for AIC. They also developed a base model for AIC that relies on text translation from English image captions and compared it to an end-to-end model that directly transcribes images into Arabic text.

None of the works mentioned above utilized attention mechanisms in their proposed models. Afyouni et al. (2021) developed a hybrid object-based, attention-driven image captioning model. They performed a comprehensive set of experiments using popular metrics and multilingual semantic sentence similarity techniques to assess the lexical and semantic accuracy of generated captions.

Out of all the works from above, only ElJundi et al. (2020) have made their dataset publicly available, and is therefore the only work we can directly compare our models with.

When finishing this work, we discovered a Mas-

ter's thesis, contemporaneous to our work by Sabri (2021). Though not a refereed publication, the author built neural network architectures which include techniques not previously explored in the Arabic image captioning literature, such as transformers. This approach yielded better results over the benchmark published by ElJundi et al. (2020).

3 Datasets

For this work, we mainly used two public datasets for image captioning: Microsoft COCO and Flickr8k. We describe them in detail now.

3.1 Microsoft COCO

Microsoft Common Objects in Context (COCO) (Lin et al., 2014) is a dataset consisting of 123,287 images including object detection, segmentation, and five captions per image (616,435 captions in total). As its name suggests, the COCO dataset contains complex everyday scenes with common objects in their natural context.

For comparison, we adopted the widely used Karpathy split of COCO (Karpathy and Fei-Fei, 2015), i.e. 113,287 train images, 5,000 validation images and 5,000 test images. We used 414,113 pre-translated captions over 82,783 training images with the Advanced Google Translate API[1], dubbed Arabic-COCO. Figure 1a shows an example of an image from the train split with its five English captions and five Arabic captions. For the Arabic speaking reader, note the error in the second machine translated, where the phrase ركوب الأمواج "ride a wave", should be replaced with its present tense يركب الموج "riding a wave".

Sabri (2021) showed that, out of a random sampled subset of 150 captions from Arabic-COCO, 46% of the translations were unintelligible. Based on this finding, we considered the captions to be noisy, which is why we did not create a validation and testing set out of Arabic-COCO.

3.2 Flickr8k

The Flickr8k dataset (Hodosh et al., 2013) consists of 8,092 images. Each image in this dataset is associated with five different captions that describe the entities and events depicted in the image. They were collected via a crowdsourcing marketplace (Amazon Mechanical Turk) with a total of 40,460 captions.

[1] https://github.com/canesee-project/Arabic-COCO

Human translations into Arabic of both the COCO and Flickr8k datasets have been done before. For example, Al-muzaini et al. (2018) built an Arabic dataset based on these two English benchmark datasets. Most of them are not public, therefore we used Arabic Flickr8k by ElJundi et al. (2020). Arabic Flickr8k is split into 6,000 train images, 1,000 validation images, and 1,000 test images, all with three Arabic captions each.

The translation to Arabic was performed by ElJundi et al. in two steps, first by using the Google Translate API and then by validating captions with professional Arabic translators. Finally, they chose the top three translated captions out of five for each image, which makes 24,000 captions in total. Figure 1b shows an example of an image from the train split with its three original English captions and three verified Arabic captions. Note that even though verified, the quality of these Arabic captions is sometimes questionable. For example, the second caption in Figure 1b is رجل أسود, which incorrectly translates to "black man".

Table 1 shows the complete list of image caption datasets used in this report.

Table 1: Statistics for the Arabic-COCO and Flickr8k translated by ElJundi et al. (2020).

Datasets	Train		Validation		Test	
	#Imgs	#Caps	#Imgs	#Caps	#Imgs	#Caps
Arabic-COCO	82,783	414,113	-	-	-	-
Flickr8k	6,000	18,000	1,000	3,000	1,000	3,000
TOTAL	88,783	432,113	1, 000	3,000	1,000	3,000

4 Methodology

As methodology, we used a two-step pipeline, as shown in Figure 2:

1. Extract region features and object tags from an image through a convolutional neural network (CNN) encoder.

2. Generate a sentence from the region features and object tags through a language model, in our case a pre-trained transformer.

As a learning method for our IC model, we used OSCAR (Li et al., 2020) and to evaluate our results, we used well-establish metrics for IC. The following subsections describe these steps in detail.

A young boy surfing in low waves.
A young boy is standing on a surfboard and riding a wave.
A surfer rides his surf board on some very small waves.
A young boy is standing on a surfboard in the water.
A young boy is standing on a surfboard in the ocean.

صبي صغير يتزلج على الأمواج المنخفضة.
صبي صغير يقف على لوح ركوب الأمواج وركوب الأمواج.
راكب أمواج يركب لوح الأمواج على بعض الأمواج الصغيرة جدًا.
صبي صغير يقف على لوح تزلج على الماء في الماء.
صبي صغير يقف على لوح ركوب الأمواج في المحيط.

(a) COCO

A longhaired man surfing a large wave.
A man in black on a surfboard riding a wave.
A man surfing in the ocean.

رجل طويل الشعر يتزلج موجة كبيرة
رجل أسود على لوح ركوب الأمواج يركب موجة
رجل يمارس رياضة ركوب الأمواج في المحيط

(b) Flickr8k

Figure 1: Caption annotations in English and Arabic for an image sample from the (a) COCO dataset and the (b) Flickr8k dataset.

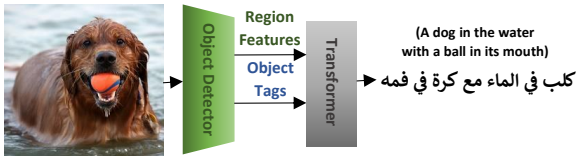


Figure 2: An overview of our methodology.

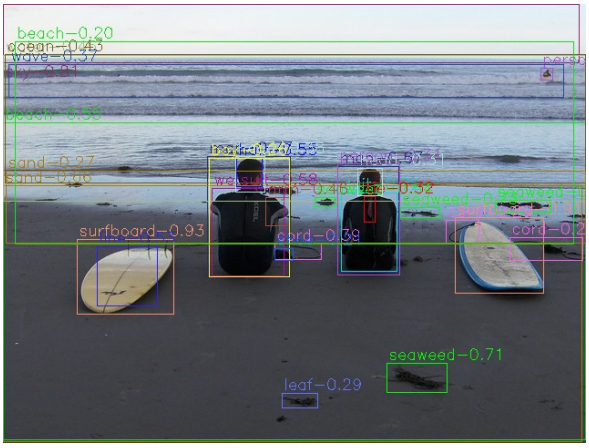


Figure 3: Object detection on an image from the COCO dataset using the X152-C4 architecture. The set of detected object tags are (`Arm, Beach, Boy, Cord, Hair, Head, Leaf, Line, Man, Ocean, Person, Sand, Seaweed, Sky, Suit, Surfboard, Tie, Water, Wave, Wetsuit`).

4.1 Image Feature Extraction and Object Tag Detection

For feature extraction, Zhang et al. (2021) trained a large-scale object and attribute detection model based on the ResNeXt-152 C4 architecture (Xie et al., 2016), shortened as X152-C4. ResNeXt is named after and adopts the ResNet strategy, a residual learning framework designed to ease the training of networks that are substantially deeper than those used previously (He et al., 2016). For this work, we utilized X152-C4 for feature extraction, pre-trained on 2.49 million unique images, including the COCO dataset. Figure 3 shows an example of object detection with the X152-C4 model. For each detected object, an image region vector is generated, which represents the vector input to the last linear classification layer.

4.2 The Transformer and BERT

The transformer architecture builds solely on attention mechanisms and was first proposed by Vaswani et al. (2017). The transformer has proved superior in sequence-to-sequence modeling, and the key lies in the possibility to capture the relationships between each word in a sequence with every other word.

Proposed by Devlin et al. (2019), BERT showed that pre-trained representations reduced the need for many heavily-engineered task-specific architectures. In other words, by pre-training general language representations, BERT was the first fine-tuning based representation model that achieved

state-of-the-art performance on a large group of sentence-level tasks, outperforming many task-specific architectures.

The release of BERT preceded many other BERT-based language models trained on different corpora from different languages, and will be the main base for our image captioning model. The following paragraphs describe the models used in this work and Table 2 shows the different models configurations for comparison.

mBERT. mBert, short for Multilingual BERT, was pre-trained with the multilingual Wikipedia dataset that consists of the top 104 most common languages (Devlin et al., 2018), including Arabic. In this comparison, we used the `bert-base-multilingual-uncased`[2] version of mBERT from HuggingFace.

AraBERT. AraBERT (Antoun et al., 2020) achieved state-of-the-art performance on most tested Arabic NLP tasks. The models were trained on news articles manually scraped from Arabic news websites and several publicly available large Arabic corpora. One of the corpora is named OSCAR (**O**pen **S**uper-large **C**rawled **A**ggregated Co**r**pus), not to be confused with the image captioning model OSCAR (**O**bject-**S**emanti**c**s **A**ligned P**r**e-training). There are several versions of AraBERT available. We used the `bert-base-arabertv02`[3] configuration in this work.

ArabicBERT. ArabicBERT (Safaya et al., 2020) was the first pre-trained BERT model for Arabic when it was released. It was originally pre-trained as an approach to solve a sub-task of the Multilingual Offensive Language Identification shared task (OffensEval 2020). We used the `bert-base-arabic`[4] configuration in this project.

GigaBERT. GigaBERT (Lan et al., 2020) is a set of models pre-trained as a bilingual BERT and designed specifically for Arabic NLP and English-to-Arabic zero-shot transfer learning. Their best model significantly outperforms mBERT and AraBERT on some supervised and zero-shot transfer settings. The training dataset consists of a dump of Arabic Wikipedia, an Arabic version of OSCAR and the Gigaword corpus, which consists of over 13 million news articles. We used the `GigaBERT-v4-Arabic-and-English`[5] configuration in this work.

4.3 The OSCAR Learning Method

The vanilla $\text{BERT}_{\text{BASE}}$ cannot handle image region features as input. As a learning method, we used OSCAR (Li et al., 2020), which achieves state-of-the-art results on six well-established vision-language understanding and generation tasks, including image captioning.

Previous pre-training methods concatenate image region features and text features as input and then use self-attention to learn image-text semantics in a brute force manner. OSCAR uses object tags detected in images as anchor points to ease the alignment of image region and word embeddings. The method is motivated by the observation that the salient objects in an image can be accurately detected by modern object detectors and that these objects are often mentioned in the caption.

The original OSCAR paper adapts the pre-trained models to seven downstream VL tasks. For IC fine-turning, they processed the input samples to triples consisting of image region features, captions, and object tags. They then randomly masked out 15% of the caption tokens and use the corresponding output representations to perform classification and predict the token ids, similar to the masked token loss used by BERT.

We used the caption inference procedure described by Li et al. (2020). They first initialize the caption generation by feeding in a `[MASK]` token and sampling a token from the vocabulary based on the likelihood of the output. Next, the `[MASK]` token in the previous input sequence is replaced with the sampled token and a new `[MASK]` is appended for the next word prediction. The generation process terminates when the model outputs the `[STOP]` token. We used the same beam search with a beam size of 5.

4.4 Evaluation Metrics

We compared the system performances with evaluation metrics used in machine translation, like BLEU-1,2,3,4 (Papineni et al., 2002), ROUGE-L (Lin, 2004), and METEOR (Banerjee and Lavie, 2005), but also image caption specific metrics[3],

[2] https://huggingface.co/bert-base-multilingual-uncased
[3] https://huggingface.co/aubmindlab/bert-base-arabertv02
[4] https://huggingface.co/asafaya/bert-base-arabic
[5] https://huggingface.co/lanwuwei/GigaBERT-v4-Arabic-and-English
[3] https://github.com/tylin/coco-caption

Table 2: Configuration comparisons for mBert, AraBERT, ArabicBERT, and GigaBERT.

Models	Training Data		Vocabulary			Configuration	
	source	#tokens (all/ar)	tokenization	size (all/ar)	cased	size	#parameters
mBERT	Wiki	21.9B/153M	WordPiece	110k/5k	no	base	172M
AraBERT	Wiki, Oscar, News articles	2.5B/2.5B	SentencePiece	64k/58k	no	base	136M
ArabicBERT	Wiki, Oscar	unknown	WordPiece	32k/28k	no	base	111M
GigaBERT	Wiki, Oscar, Gigaword	10.4B/4.3B	WordPiece	50k/26k	no	base	125M

like CIDEr (Vedantam et al., 2014) and SPICE (Anderson et al., 2016). For comparisons of semantic meaning, we utilized the transformer-based Multilingual Universal Sentence Encoder[4] (MUSE) (Yang et al., 2020) and angular similarity. Specifically, Eq. 1 gives the angular similarity S_θ between two vector embeddings $\boldsymbol{v}$ and $\boldsymbol{u}$.

$$S_\theta = 1 - \arccos\left(\frac{\boldsymbol{v}\cdot\boldsymbol{u}}{\|\boldsymbol{v}\|\,\|\boldsymbol{u}\|}\right)/\pi \qquad (1)$$

This way of evaluating captions is similar to the technique proposed by Afyouni et al. (2021).

To verify the quality of the candidate captions, we complement our results with human evaluation. For this task, native Arab speaking experts evaluated a sample of the candidate captions generated across the proposed models. We followed the guidelines of the Transparent Human Benchmark (THUMB), a human evaluation protocol proposed by Kasai et al. (2021). The authors base their evaluations on two main scores (*precision* and *recall*) and three types of penalties (*fluency*, *conciseness*, and *inclusive language*).

Precision measures how precise the caption is given the image, while recall measures how much of the salient information (e.g., objects, attributes, and relations) from the image is covered by the caption. Both scores are assessed in the scale of 1–5. The overall score is computed by averaging precision and recall and deducting penalty points, with a maximum deduction of 0.5. Kasai et al. (2021) found most captions from modern neural network models were highly fluent and concise. Since precision and recall covers the context of an image, in our work the penalty will be purely based on grammar and semantics errors. For example, consider the candidate caption:

فتاة تتارجح بمضرب بيسبول على كرة

"Girl swinging a baseball bat on a ball"

[4]https://tfhub.dev/google/universal-sentence-encoder-multilingual-large/3

Although the verb "swinging" is literally translated to تتارجح, it does not convey the meaning of the image in Arabic. It should be correctly translated to تضرب "hits" instead, giving the caption 0.5 penalty points.

5 Evaluation

5.1 Preprocessing

Before training the models, we ran all of the images through the X152-C4 object detector for extraction of region features and object tags. Since all of the image features and object tag labels are made available for the Karpathy split of the COCO dataset by Li et al. (2020), only Flickr8k images had to be inferred. We then split the Flickr8k image features and object tags into train, validation, and test images following ElJundi et al. (2020).

To train models on Arabic captions and Arabic object tag *labels*, we simply translated English labels directly with the Google Translate API. A 10% sample of the 1,114 object tags translations detected in the Flickr8k dataset were validated by two native Arab speaking experts on a scale of 1-3 (1: incorrect, 2: partly correct, 3: correct). The annotators gave the sample a mean score of 2.76 and 2.62 with a pairwise Cohen kappa coefficient of 0.43 (moderate agreement).

5.2 Experimental Setup

We initialized the captioning model with various Arabic-specific BERT configurations. In order to select the best models, we carried out two experiments considering the multi/bilingual aspects and the learning curve of the fitting procedure:

1. Evaluation of two multilingual models both trained on
   (a) Arabic captions and Arabic labels
   (b) Arabic captions and English labels

   We carried out this experiment mainly for comparing the object labels ability to affect the final image-text alignment.

2. Evaluation of the learning curve for three different models, respectively trained on 50%, 75% and 100% of a dataset. From the results, we can tell if the validation loss decreases with the amount of data or if some adjustment have to be made to the models, for example with a hyper parameter grid search. Out of the trained models, we chose the two most accurate ones as candidates for large scale training.

After we picked two candidate models, we made a third and final experiment:

3. Do large scale training on the candidate models on datasets of different size. Evaluate the models both with automatic and human metrics and compare the results with previous models.

We carried out the first two experiments on Google Colab GPU:s (1 P100 GPU with 16 GB memory). We carried out the final large scale experiments on a workstation (1 GV100 GPU with 32 GB memory) and a high performance computer (HPC) system (8 K80 GPU:s with 12 GB memory each).

For all the experiments above, we saved training and validation loss values at every epoch, while model checkpoints were saved every 5 epochs. All the experiments used the AdamW optimizer and a linearly decaying learning rate according to the recipe described in OSCAR (Li et al., 2020). Exact model hyper parameters for each experiment are shown in the Appendix A section.

5.3 Experimental Results

English vs Arabic labels. Table 3 shows the final evaluation scores for all models. Our first experiments show that both approaches, training on English and Arabic object labels, work in principle. Already at this stage, GigaBERT trained on English labels outperformed previous reported BLEU-1,2,3,4 scores with 0.0123, 0.0144, 0.0190, 0.0167 respectively. However, note that these scores were obtained from the val-split, and not the final test-split. We think that the reason to why GigaBERT with English labels outperforms Arabic labels is that the quality of the original English labels, in combination with GigaBERT's English pretraining, is much better than its machine translated counterpart. mBert is only trained on Wikipedia (Devlin et al., 2018), while GigaBERT is trained on the Gigaword corpus in addition to Wikipedia and web crawl data. This is how we explain GigaBERT's better performance. Moreover, the vocabulary of GigaBERT (21k English tokens vs 26k Arabic tokens) is richer and more balanced than the vocabulary of mBERT (53k English tokens vs 5k Arabic tokens), see Table 2.

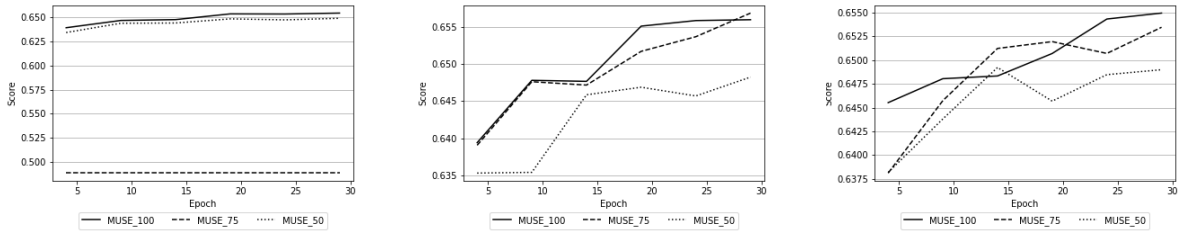
Table 3: Evaluation scores (evaluation on epoch 30) for the trained models. The best scoring models are marked in bold for each evaluation metric.

Model	Labels	BLEU-4	ROUGE-L	METEOR	CIDEr	SPICE
GigaBERT	English	**0.074**	**0.29**	0.3	**0.33**	**0.037**
	Arabic	0.062	**0.29**	**0.31**	0.31	**0.037**
mBert	English	0.058	0.28	0.30	0.29	0.031
	Arabic	0.067	**0.29**	0.30	0.31	0.033

Learning Curve. We evaluated all of the models from the learning curve experiment with MUSE to investigate the correlation between semantic scores and an increased amount of data. The evaluation over training time is shown in Figure 4 for AraBERT, ArabicBERT, and GigaBERT. In general, more data increased evaluation scores. One notable thing is that the final score of GigaBERT trained on 75% of data outperformed 100%, but Figure 4b shows that the 100% curve is generally higher than the 75% curve. This finding suggests that the average MUSE score has a high variance. Note that GigaBERT trained on 100% of Flickr8k is identical to the model trained on Arabic labels in the previous experiment.

In the case of AraBERT, the 75% MUSE curve is way lower than the 100% and 50% curves, but the 100% loss curve is still higher than the 50% one. The unstable training results of AraBERT suggest that the selected learning rate is too large. We performed learning rate grid search on AraBERT and GigaBERT on the interval $\eta \in [1e^{-5}, 7e^{-5}]$ to minimize validation loss, and found an optimum at $\eta = 3e^{-5}$.

Large Scale Training. Table 4 presents the final test scores (BLEU-1,2,3,4, ROUGE-L, METEOR, CIDEr and MUSE) of a selection of our models, and models previously proposed by Jindal (2018), Al-muzaini et al. (2018), Afyouni et al. (2021) and ElJundi et al. (2020). Out of the previous works, only the model by ElJundi et al. (2020) is tested on the same Flickr8k test set as ours. We were unable to obtain the splits from the other studies, and have no data regarding on how their splits may differ from ours. The difference between their model scores and our are quite large in some cases. On



(a) AraBERT MUSE scores (b) GigaBERT MUSE scores (c) ArabicBERT MUSE scores

Figure 4: MUSE evaluation scores over all epochs for (a) AraBERT, (b) GigaBERT and (c) ArabicBERT.

Table 4: Our model scores compared to previous models. The highest scores on our test-split are marked in bold. Of all the previous ones, only the model by ElJundi et al. (2020) uses the same test-split as us. Other test-splits are unknown.

Model	Test set	B1	B2	B3	B4	ROUGE-L	METEOR	CIDEr	MUSE
Jindal (2018)	Flickr8k	0.658	0.559	0.404	0.223	-	0.201	-	-
Al-muzaini et al. (2018)	COCO & Flickr8k	0.462	0.260	0.190	0.080	-	-	-	-
Afyouni et al. (2021)	COCO	0.649	0.413	0.241	0.136	0.470	0.408	-	0.78
ElJundi et al. (2020)	Flickr8k	0.332	0.193	0.105	0.057	-	-	-	-
AraBERT32-Flickr8k		**0.391**	**0.246**	0.150	0.092	0.331	0.314	0.415	**0.671**
AraBERT32-COCO		0.365	0.221	0.129	0.0715	0.310	**0.317**	0.36	0.669
AraBERT256-Flickr8k	Flickr8k	0.387	0.244	**0.151**	**0.093**	**0.334**	0.312	**0.428**	0.668
GigaBERT32-Flickr8k		0.386	0.241	0.144	0.0827	0.331	0.315	0.403	0.669
GigaBERT32-COCO		0.36	0.215	0.124	0.0708	0.308	0.311	0.344	0.668
	Δ	**0.059** ↑	**0.053** ↑	**0.046** ↑	**0.036** ↑				

(a) (b) (c) (d)

Figure 5: Human evaluation of four candidate captions produced by AraBERT32-COCO: two accurate candidate captions (a) and (b), and two inaccurate candidate captions (c) and (d). Each candidate caption is accompanied by the reference caption from the Flickr8k test-split with the most MUSE similarity and a THUMB score.

possible explanation could be that our BERT-based approach differs from previous LSTM approaches, which can achieve significantly higher results than a BERT-based model for a small dataset on NLP tasks (Ezen-Can, 2020).

All of our models are named after the scheme *modelBatchSize-dataset*, where *model* is our initialization model, *BatchSize* is the training batch size and *dataset* is the dataset trained on. For example, one of our best performing models was initialized on AraBERT and trained with a batch size of 32 on Flickr8k. Therefore, we named the model AraBERT32-Flickr8k. AraBERT32-Flickr8k outperforms the model by ElJundi et al. (2020) on all BLEU scores, and most remarkably on BLEU-4, where we see a 61.4% increase. We chose to drop the SPICE scores from the table because of the evaluation scripts incompatibility with the Arabic language.

We complemented Table 4 with human evaluations on a sample of the dataset according to the guidelines of THUMB (Kasai et al., 2021). Figure 5 shows four generated captions from AraBERT32-COCO with images and human evaluations. All of

the evaluations were made by two experts in Arabic language.

In general, the human evaluations show accurate results. In Figure 5a, the candidate caption:

رجل يرك دراجة ترابية فوق تلة صخرية
"Man riding a dirt bike on a rocky hill"

is nearly perfect. It is almost identical to the reference caption:

رجل يركب دراجة ترابية فوق بعض الصخور
"Man riding a dirt bike over some rocks",

and only differs in the last phrase.

Not all results were accurate. Looking at Figure 5c, the first row shows the candidate caption

مجموعة من الناس يتسلقون على ظهر شاحنة
"Group of people climbing on the back of a truck",

while the closest reference caption مدينة ملاهي translates to "Amusement park". Though the candidate sentence is fluent and grammatically correct, it appears to be random in the context of the image. This shows how the models in these examples fail to identify objects in the image and correctly describe a scene.

A potential source of error for the incorrect image-text alignment could be noise in the machine translated data input, i.e. *"garbage in, garbage out"*. For example, the publicly available Arabic-COCO used is purely machine translated and has to be verified by humans before employed in testing. The justification to why we still use machine-translated data is that we rely on the BERT-based language models to handle the grammar and syntax, while we count on the machine-translation model to correctly translate salient objects. The failure to do so leads to errors in learning image-text semantic alignments. For example, in our dataset, mistranslated object labels can be found. Some nouns are mistranslated into their homophone counterparts: "light" (*noun*) to خفيفة (*adjective*, bright; well-lighted), "block" (*noun*) to منع (*adjective*, to obstruct, or prevent someone or something) and so on. Li et al. (2020) showed that OSCAR learning curves for fine-tuning with object tags converge significantly faster than the methods without tags. In other words, high quality labels are crucial in image-text alignment for VL-pretrained models.

For the complete table with scores for all trained models, see Appendix B.

6 Conclusion

This work focused on Arabic image captioning using pre-trained bidirectional transformers. We can draw many conclusions from it.

The special challenge in Arabic image captioning is, not regarding the lack of well-annotated datasets, the morphological complexity of the Arabic language which makes it harder to process. With our work, we showed that it is possible to achieve state-of-the-art results with a minimal pre-processing scheme and by adapting English captioning models to other languages through public dataset benchmarks.

Furthermore, we achieved results better than the previous work on the Flickr8k dataset by ElJundi et al. (2020). Our experiments also show that both approaches, training on English and Arabic object labels, work in principle. In addition, we proposed working configurations and heuristics for hyper parameters in future experimentation on our proposed models. Therefore, our models provide a new baseline for the AIC community.

Further work in the field should be to verify all machine translated Arabic labels by humans before further training on the datasets. This task should not be too expensive since there are only 1,114 object tags translations detected in the Flickr8k dataset, and 253 additional object tags in Arabic-COCO. This could greatly improve training. Secondly, the lack of qualitative Arabic data should be solved by translation and verification of all COCO captions, and then making the resulting dataset publicly available. As a suggestion, one could follow a crowd sourcing procedure as described by Almuzaini et al. (2018), which includes some of the instructions that were used in the creation of COCO captions, and additional instructions specific to the Arabic language. This would create a new benchmark Arabic captioning dataset that we could train and test our models on.

Finally, we hope that our work will be useful for future Arabic image captioning models, and that it will spur more contributions to the field in the closest future.

References

Imad Afyouni, Imtinan Azhara, and Ashraf Elnagar. 2021. AraCap: A hybrid deep learning architecture for Arabic Image Captioning. In *ACLing 2021: 5th International Conference on AI in Computational Linguistics*.

Huda A. Al-muzaini, Tasniem N. Al-yahya, and Hafida Benhidour. 2018. Automatic arabic image captioning using rnn-lstm-based language model and cnn. *International Journal of Advanced Computer Science and Applications*, 9(6).

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *Computer Vision – ECCV 2016*, pages 382–398, Cham. Springer International Publishing.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Anfal Attai and Ashraf Elnagar. 2020. A survey on arabic image captioning systems using deep learning models. In *14th International Conference on Innovations in Information Technology (IIT)*, pages 114–119.

Satanjeev Banerjee and Alon Lavie. 2005. 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, Ann Arbor, Michigan. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Multilingual bert readme. https://github.com/google-research/bert/blob/master/multilingual.md. [Online; accessed 6 Feb. 2022].

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Obeida ElJundi, Mohamad Dhaybi, Kotaiba Mokadam, Hazem Hajj, and Daniel Asmar. 2020. Resources and end-to-end neural network models for arabic image captioning. In *15th International Conference on Computer Vision Theory and Applications*.

Aysu Ezen-Can. 2020. A Comparison of LSTM and BERT for Small Corpus. *ArXiv*, abs/2009.05451.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Micah Hodosh, Peter Young, and Julia Hockenmaier. 2013. Framing image description as a ranking task: Data, models and evaluation metrics. In *24th International Joint Conference on Artificial Intelligence*.

Xiaowei Hu, Zhe Gan, Jianfeng Wang, Zhengyuan Yang, Zicheng Liu, Yumao Lu, and Lijuan Wang. 2021. Scaling up vision-language pre-training for image captioning. *CoRR*, abs/2111.12233.

Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. 2019. Attention on Attention for Image Captioning. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Vasu Jindal. 2017. A deep learning approach for arabic caption generation using roots-words. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*.

Vasu Jindal. 2018. Generating image captions in Arabic using root-word based recurrent neural networks and deep neural networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 144–151, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jungo Kasai, Keisuke Sakaguchi, Lavinia Dunagan, Jacob Morrison, Ronan Le Bras, Yejin Choi, and Noah A. Smith. 2021. Transparent human evaluation for image captioning. *CoRR*, abs/2111.08940.

Wuwei Lan, Yang Chen, Wei Xu, and Alan Ritter. 2020. An empirical study of pre-trained transformers for Arabic information extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4727–4734, Online. Association for Computational Linguistics.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. 2020. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Computer Vision – ECCV 2020*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Sabri Monaf Sabri. 2021. Arabic Image Captioning using Deep Learning with Attention. Master's thesis, University of Georgia.

Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2054–2059, Barcelona (online). International Committee for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Conference on Neural Information Processing Systems (NIPS)*.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2014. Cider: Consensus-based image description evaluation. *CoRR*, abs/1411.5726.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2016. Aggregated residual transformations for deep neural networks. *arXiv preprint arXiv:1611.05431*.

Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning (PMLR)*.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5579–5588.

Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J. Corso, and Jianfeng Gao. 2019. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

A Experiment Hyperparameters

English vs Arabic labels. All experiments were trained and validated with the Flickr8k train- respective val-split. Table 5 shows the exact hyperparameters for the experiments.

Learning curve. All experiments were validated with the Flickr8k val-split and trained on Arabic labels. Table 6 shows the exact hyperparameters for the experiments. Grid search optimization was made on AraBERT and GigaBERT in the interval $\eta \in [1e^{-5}, 7e^{-5}]$ and a step size of $1e^{-5}$.

Large scale. All experiments were validated and tested with the Flickr8k test- respective val-split, and trained on Arabic labels. Table 7 shows the exact hyperparameters for the experiments.

B Complementary Results

Table 8 shows scores for all models trained during the last experiment.

Table 5: Hyperparameters used for the English vs Arabic labels experiments.

Model	Train	Object labels	Learning rate	Batch size	#Epochs
GigaBERT	Flickr8k	eng/ar	1e-4	32	30
mBERT	Flickr8k	eng/ar	1e-4	32	30

Table 6: Hyperparameters and datasets used for the learning curve experiments.

Model	Train	% of dataset	Learning rate	Batch size	#Epochs
AraBERT	Flickr8k	50/75/100	1e-4	32	30
Arabic-BERT	Flickr8k	50/75/100	1e-4	32	30
GigaBERT	Flickr8k	50/75/100	1e-4	32	30

Table 7: Hyperparameters and datasets used for the large scale experiments.

Model	Train	Object labels	Learning rate	Batch size	#Epochs
AraBERT	Flickr8k	ar	3e-5	32	30
	Arabic-COCO	ar	5e-5	32	50
	Arabic-COCO+Flickr8k	ar	3e-5	32	50
	Flickr8k	ar	5e-5	256	30
	Arabic-COCO	ar	9e-5	256	50
	Arabic-COCO+Flickr8k	ar	9e-5	256	50
GigaBERT	Flickr8k	eng	3e-5	32	30
	Arabic-COCO	eng	3e-5	32	50
	Arabic-COCO+Flickr8k	eng	3e-5	32	50
	Flickr8k	eng	9e-5	265	30
	Arabic-COCO	eng	9e-5	265	50
	Arabic-COCO+Flickr8k	eng	9e-5	256	50

Table 8: Our model scores compared to previous models. The highest scores on our test-split are marked in bold. Of all the previous ones, only the model by ElJundi et al. (2020) uses the same test-split as us. Other test-splits are unknown.

Model	Test set	B1	B2	B3	B4	ROUGE-L	METEOR	CIDEr	MUSE
Jindal (2018)	Flickr8k	0.658	0.559	0.404	0.223	-	0.201	-	-
Al-muzaini et al. (2018)	COCO & Flickr8k	0.462	0.260	0.190	0.080	-	-	-	-
Afyouni et al. (2021)	COCO	0.649	0.413	0.241	0.136	0.470	0.408	-	0.78
ElJundi et al. (2020)	Flickr8k	0.332	0.193	0.105	0.057	-	-	-	-
AraBERT32-Flickr8k	Flickr8k	**0.391**	**0.246**	0.150	0.092	0.331	0.314	0.415	**0.671**
AraBERT32-COCO		0.365	0.221	0.129	0.0715	0.31	**0.317**	0.36	0.669
AraBERT32-COCO+Flickr8k		0.358	0.216	0.127	0.0715	0.317	0.316	0.364	0.661
AraBERT256-Flickr8k		0.387	0.244	**0.151**	**0.093**	**0.334**	0.312	**0.428**	0.668
AraBERT256-COCO		0.355	0.211	0.122	0.069	0.303	0.313	0.335	0.665
AraBERT256-COCO+Flickr8k		0.339	0.204	0.12	0.0686	0.302	0.31	0.339	0.655
GigaBERT32-Flickr8k		0.386	0.241	0.144	0.0827	0.331	0.315	0.403	0.669
GigaBERT32-COCO		0.36	0.215	0.124	0.0708	0.308	0.311	0.344	0.668
GigaBERT32-COCO+Flickr8k		0.362	0.216	0.127	0.0675	0.312	0.308	0.359	0.661
GigaBERT265-Flickr8k		0.376	0.235	0.141	0.0803	0.322	0.313	0.385	0.664
GigaBERT265-COCO		0.339	0.198	0.113	0.062	0.287	0.306	0.312	0.662
GigaBERT265-COCO+Flickr8k		0.365	0.217	0.128	0.0705	0.315	0.309	0.373	0.662
	Δ	**0.059** ↑	**0.053** ↑	**0.046** ↑	**0.036** ↑				