Vision Transformers for Optical Music Recognition of Monophonic Scores

Christo Hristov
Stanford University
christoh@stanford.edu

Maddox de Bretteville
Stanford University
maddoxd@stanford.edu

Abstract

Optimal Music Recognition (OMR) is a specialized domain within computer vision focused on the interpretation and conversion of sheet music images into usable musical notation. Current OMR methodologies primarily leverage Convolutional Recurrent Neural Networks (CRNNs) or Convolutional Neural Networks (CNNs) combined with transformer encoder-decoder architectures to generate sequences of musical symbols from input images.

However, with the advent of the Vision Transformer (ViT), as introduced in "An Image is Worth 16x16 Words" , recent research indicates that minimizing the implicit bias in deep learning models can enhance interpretative accuracy. Therefore, we explore a purely transformer-based approach to OMR, employing a pretrained ViT alongside a transformer decoder to generate the desired musical symbol sequences.

Additionally, we incorporated an explicitly defined semantic musical vocabulary tailored for the transformer encoder-decoder model. Despite encountering technical challenges that prevented the complete training of our model, we are confident that this approach represents the future of efficient and accurate musical symbol recognition from sheet music images.

1. Introduction

In recent years, the accessibility of extensive digital music score collections has greatly benefitted both professional musicians and amateurs. Platforms like IMSLP and various library initiatives provide access to vast repositories, making previously hard-to-find printed materials readily available. However, while digitization facilitates easy copying and distribution and offers durability advantages over physical copies, many music applications are constrained to symbolically encoded scores.

Notation software, computer-assisted composition tools, and digital musicology systems primarily focus on computationally-encoded symbols like notes and bar-lines rather than pixels from digitized images.

The scientific musicological domain stands to gain significantly from music encoded in symbolic formats such as MEI or MusicXML, allowing for scalability in real-world scenarios. While initiatives like OpenScore and KernScores aim to bridge the gap between digitized images and encoded music content, manual transcription remains impractical due to its time and resource-intensive nature. Thus, assisted or automatic transcription systems like Optical Music Recognition (OMR) are deemed necessary.

Despite the potential of OMR, its reliability currently falls short of optical character or speech recognition technologies. Recent advances in machine learning, particularly Deep Learning (DL), offer promise. Past approaches to OMR typically involve manually segmenting sheet music into smaller components for individual processing, a cumbersome process fraught with inductive biases.

The advent of Deep Learning practices, notably Convolutional Neural Networks (CNNs), has enabled the possibility of an end-to-end approach to transcribing photos of sheet music. Current approaches utilize a Convolutional Recurrent Neural Network (CRNN), employing a CNN to extract features from the image and feeding columns of the outputted feature map as inputs to a Recurrent Neural Network (RNN). Given the sequential and patterned nature of music, it is believed that sequence models such as Long Short-Term Memory (LSTM) networks can effectively interpret and predict musical scores. However, the development of transformers, which effectively capture long-range dependencies, presents an opportunity to improve OMR tasks.

While transformers have demonstrated superior performance over LSTMs in sequence modeling, their ability to capture relevant features of an image and how they compare to CNNs require further investigation. Transformers introduce fewer inductive biases into a model compared to the stricter CNN architecture, suggesting that a Transformer encoder may better optimize data and capture more relevant features. This study aims to explore the functionality of a full Transformer encoder and decoder model for image-to-sequence tasks, particularly in the context of OMR with monophonic scores. Such a model has not yet been implemented, but promising results in smaller tasks like OMR

may pave the way for a transition away from CNNs.

In our proposed model, we take an image of a monophonic (single instrument, single melody) incipit and pass it through Google's base ViT encoder pre-trained on ImageNet-21k to produce encoded outputs which is the last hidden state of the pre-trained ViT. This output is then passed into the decoder which then generates the target sequence in a explicitly defined semantic vocabulary using both cross attention with the encoder and self-attention based on the already generated output. When the output is generated, we compare this with the target sequence of notes to generate the loss and train and evaluate the model.

2. Related Works

Before the emergence of Deep Learning and its application to the field of OMR, primary methods inmvolved multiple stages of image preprocessing and various computational methods that do not allow for the kind of learning possible with modern deep learning methods [6]. The most relevant paper to what we are exploring is "End=to-End Neural Optical Music Recognition of Monophonic Scores" by Calvo-Zaragoza et al. [2], whose dataset and results we will be attempting to reproduce with a different model architecture discussed below. The paper utilizes a CNN to extract the symbols from the image and then utilizes a RNN to help with the sequential aspect of translating sheet music. The paper also utilizes the Connectionist Temporal Classification loss function which was not necessary for our model given our choice of model architecture. The paper required CTC because of the way the CRNN works with the given input data (an image of a line of music) and the semantic sequence it is going to predict. Because the image was divided into sections for the CRNN, the CTC was required since some sections would generate blanks, not necessarily actual notes. Given that this paper was published in 2018, there have been newer methods that we believe will improve on the results accomplished by Calvo-Zaragoza et al.

Of equal importance to the original Calvo-Zaragoza paper that provides the dataset for our study are a number of additional research papers and work relevant to our architecture. An integral base for our work is the "Attention is all you need" paper by Vaswani et al [11]. By leveraging attention mechanisms, the Transformer achieves state-of-the-art results on various tasks, and we utilize the Decoder architecture from the original Vaswani paper with some slight modifications for our model. In addition to the original Transformer and attention mechanism, Vision Transformers (ViTs) have been shown to perform very well on image classification when used on image patches [10]. This was proposed in "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale" by Dosovitskiy et al. which utilized 16x16 patches of images trained on ImageNet and JFT [3]. This paper produced results that outperformed ResNet-152x4. This paper is important because the CRNN in the Calvo-Zaragoza paper could potentially be outperformed by the use of Vision Transformers (ViT) that achieve excellent results compared to state-of-the-art CNNs while requiring substantially fewer computational resources to train.

In addition to these base models that are building blocks for our final model. There are a number of similar models that utilize ViT or transformer-based sequence translation to tackle similar tasks to that of OMR. Optical Character Recognition is a related task to OMR where there has been a fair amount of published research, one main study being "TrOCR: Transformer-Based Optical Character Recognition with Pre-trained Models" by Li et al. [1]. In addition to TrOCR, there is "Dtrocr: Decoder-only transformer for optical character recognition," paper by Masato Fujitake which differs from our encoder-decoder based approach, but presents a potential solution to the OMR task built off of the prior TrOCR paper[4]. There are non-OCR approaches to understanding document images like an invoice which is fairly similar to the task of understanding musical scores. One approach is "OCR-free Document Understanding Transformer" which uses a more simple Transformer-based approach utilizing cross-entropy as the loss function [5]. Interestingly, Ríos-Vila et al. in "On the Use of Transformerse for End-to-End Optical Music Recognition" concluded that transformers were not as suited for OMR tasks as CNNs, but they did not used pretrained ViTs in their model, which we believe could show large improvements [9]. More recently in 2024, Ríos-Vila et al., have explored the use of an encoder decoder model they call a Sheet music Transformer to work on OMR beyond monophonic transcription to deal with more complex sheet music [7]. They expanded on this model to work on a full page of sheet music, employing various methods to analyze an entire page [8]. These models differ both because they focus on non-monophonic scores and utilize a CNN as a decoder with CTC loss, whereas our model uses a pre-trained ViT encoder and normal cross-entropy loss. The similarities between these studies and our model lie in the same Transformer decoder to produce the output sequence. Given the pre-trained ViT we will utilize and the demonstrated success by Ríos-Vila et al. with an encoder-decoder based structure, we have reason to believe that our proposed architecture can be successful.

3. Methods

To encode the image, we employed a ViT Base 16 model pretrained on ImageNet-21k (14 million images, 21,843 classes). We opted for the base model over the larger version due to its significantly fewer parameters, allowing the combined encoder-decoder model to train for more iterations within the same timeframe. More iterations enable

the non-pretrained decoder model to learn more accurate weights. As demonstrated in the original ViT paper, the base model performs comparably to the large model when pretrained on ImageNet-21k, with the increased complexity of the larger model only being beneficial with larger datasets. The ViT must be pretrained to produce relevant results, as transformers lack inductive bias, meaning they require extensive training to encode relevant features. The pretrained model divides the input image into 16x16 pixel patches, which are then projected into 768-dimensional embeddings through a learned projection layer. These patch embeddings are sequentially arranged and concatenated with 1D positional embeddings to produce a sequence of (1, 768) embeddings.

During training, we worked with images of varying dimensions that were not square. The original ViT was pretrained with 224x224 images, and thus learned positional encodings specific to this dimension. The transformer architecture's parameters for attention weights and linear projections depend on 16x16 patch sizes. However, the pretrained model should theoretically be invariant to input sequence length, as shown in the original paper where the model is fine-tuned on images with different dimensions than the original 224x224. The first challenge was interpolating these 224x224 positional encodings to our target image dimensions. Our images were not square and varied in size, making interpolation less accurate. We decided to scale all images to a fixed size of 128x800 and interpolate the pretrained positional encodings to this dimension. While we could have replaced these pretrained encodings with new ones for 128x800 images, we believed that with our limited data, new encodings would not optimize as well as the pretrained ones. The ViT encoder produces 768-dimensional embeddings for each image patch, theoretically encoding relevant features.

For the Transformer decoder, we utilized the architecture from the "Attention is All You Need" paper. This consists of 6 layers with 8 parallel attention heads for both masked self-attention and cross-attention. We first tokenized the new semantic vocabulary and introduced a learnable embedding layer that produced 512-dimensional vectors for each token. This learned embedding was concatenated with sinusoidal positional encodings and passed into the transformer decoder layers. Each layer used masked self-attention and cross-attention with feed-forward networks as implemented in the paper. This architecture allowed our model to capture more information than the original CRNN. The CRNN took columns from the CNN output, using past columns rather than previously predicted symbols to predict the sequence. By utilizing both masked self-attention and cross-attention, the decoder attended to the entire image and the generated sequence, improving the model's ability to capture both image and musical patterns. In the original paper, both the

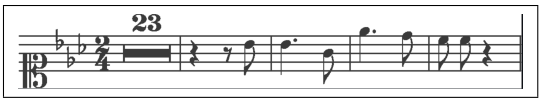

Figure 1. This is an example of one of the images of the training set. This specific example is 220016918-1_2_1. The data comes with an mei, mid, agnostic and semantic version of this stave.

encoder and decoder use a hidden size of 512. Here, the transformer decoder has a hidden size of 768, so we modified the key and value matrices in the multi-head cross-attention to project from 768 dimensions to 64, resulting in 512-dimensional embeddings when concatenated with the other 8 heads. We performed this projection within the attention matrices to preserve as much information from the image as possible. After the decoder layers, we included a final linear projection from the hidden size to the vocabulary size.

To evaluate the loss, we used Cross-Entropy loss instead of the original CTC loss used in the CRNN paper. The CRNN implementation created a frame for every column of the featurized image, potentially resulting in frames not corresponding to symbols. To address this mismatch, the CRNN included blank symbols in the vocabulary and used CTC loss to emphasize differences in predicted and true sequences. However, by using a transformer decoder, every frame produces a symbol, eliminating this mismatch. Thus, we could directly compare our predicted sequence with the true sequence using cross-entropy loss, which also masked padded tokens to prevent them from contributing to the model's training.

4. Dataset and Features

The dataset which we are using is the PrIMuS Dataset which stands for "Printed Images of Music Staves" which can be accessed here http://grfia.dlsi.ua.es/primus/. PrIMuS contains 87,678 real-music incipits (an incipit is a sequence of notes, typically the first ones, used for identifying a melody or musical work), each one represented by five files: the Plaine and Easie code source, an image with the rendered score, the musical symbolic representation of the incipit both in Music Encoding Initiative format (MEI) and in a simplified encoding (semantic encoding), and a sequence containing the graphical symbols shown in the score with their position in the staff without any musical meaning (agnostic encoding).

In the training of our model, we use 78,755 examples to train on, and then evaluate on 8,923 examples for the test set. As described earlier, each file contains different representations of the same data, but we only use the image and the semantic representation of the data for the purpose of our model. We elect to only use the semantic representation, because it is most fitting for the transformer

clef-C1 keySignature-EbM timeSignature-2/4 multirest-23 barline rest-quarter rest-eighth note-Bb4_eighth barline note-Bb4_quarter. note-G4_eighth barline note-Eb5_quarter. note-D5_eighth barline note-C5_eighth note-C5_eighth rest-quarter barline .

Figure 2. This is the semantic representation of the stave shown in Figure 1. This is the sequence that the model would be trying to predict from the given image. This semantic representation occurs in sequences which we believe a transformer is capable of learning due to the attention mechanism. We tokenized the vocabulary for this prediction of the note sequence.

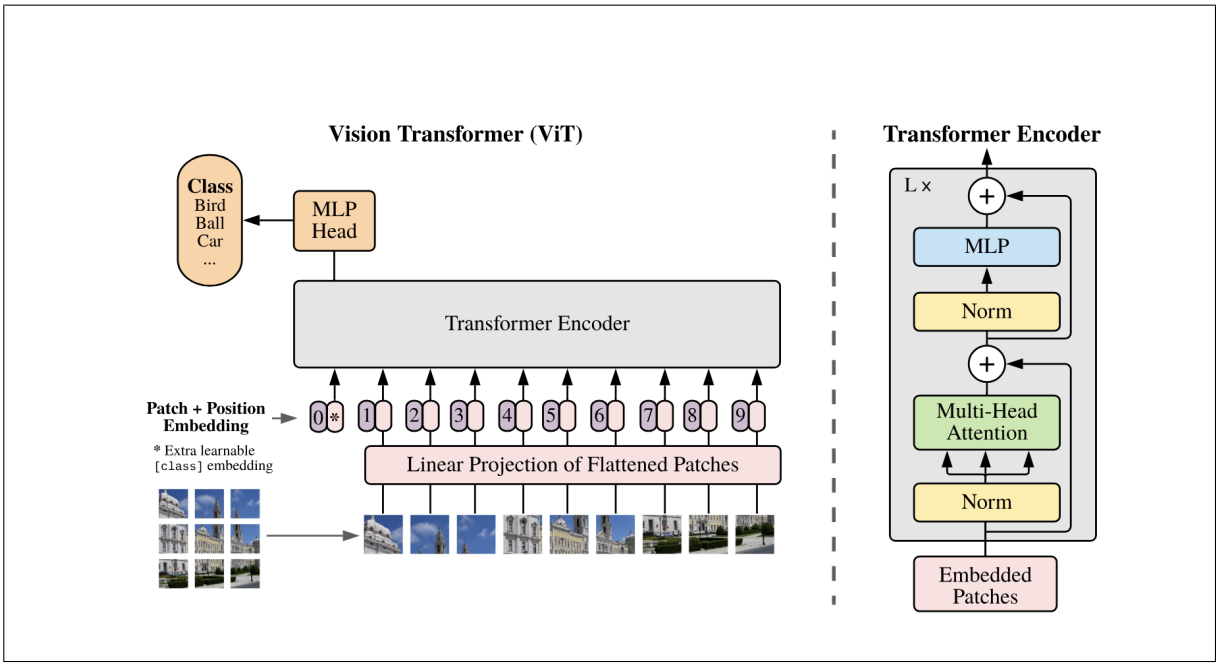

Figure 3. This photo is taken from "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale" by Dosovitskiy et al., which is the model used as our encoder [3]. First, an image is split into fixed-size patches, linearly embedded, position embeddings are added, and then the resulting sequence of vectors is fed into a standard Transformer encoder. The learnable "classification token" is added to the sequence so that the model can perform classification.

decoder, which seeks to capture patterns and relationships between symbols in the outputted sequence. These patterns are more likely to occur in the semantic decoding, which captures semantic musical information which inherently encodes musical patterns. These musical patters may be less aparent in the agnostic representation, which captures more physical features of the dataset. Additionally, there is a converted from the semantic representation to the MIDI file which meant we could also audibly compare the results of our model to the results of the CRNN. As is custom with a Transformer architecture, we tokenized the vocabulary which consists of 1,781 notes for the semantic representation of the data. We also added and tokenized a <START>, <END>, and <PAD> token. The <START> and <END> token are necessary for the training of the transformer decoder. The original CRNN took image columns as frames and therefore was not concerned with predicting the end of a sequence. The CRNN simply produced one symbol per frame. However, to train the transformer decoder, we need a <START> token, from which the transformer can autoregressively begin predicting. The <END> token is necessary to evaluate the predicted sequence. When testing, the sequence terminates with the <END> token, and this embedding as well as its relation with the rest of the vocabulary must be learned. The padded token is included as our input sequence ids are padded to a constant size to vectorize the learning process.

The preprocessing of images was another crucial decision in the training of the model. The ViT was pretrained on 224 x 224 images. However, the PrIMuS dataset included images in a more rectangular aspect ratio with varying dimensions. Scaling these images to the 224 x 224 pretrained input size would mean patches would include more than one note, which we believed would detract from the decoder's ability to attend meaningfully to the encoded patches. In

order to vectorize the training process, images within the same batch would have to be scaled to the same size. In the original paper, this is done with image padding. While image padding would not affect the attention mechanisms withing the transformer, it would skew the information provided by the positional encodings. These positional encodings seek to capture relativity within an image, and so would not translate well to images with padding or irrelevant information. Esentially, the positional encodings would be unable to differentiate between padded and non padded regions of the image, and so could possibly capture positional info about padding. Thus, we elected to just scale all images to the same 128 x 800 size. This size ensured that after seperating the image into 16 x 16 patches, no patch was likely to contain more than one note. The images were additionally inputted in a greyscale format, meaning they only had one channel.

The pretrained model required images with three channels. To account for this mismatch without introducing new information, we simply extended this one channel value into three channels. Each channel was then normalized with a mean of (0.5, 0.5, 0.5) and a standard deviation of (0.5, 0.5, 0.5).

5. Experiments/Results/Discussion

Our primary concern was computational time. The combined Transformer encoder and decoder model comprises approximately 167 million parameters. For comparison, the base transformer model used in "Attention Is All You Need" contains 65 million parameters and required approximately 12 hours to train with 100,000 training steps. Using this as a general benchmark, given the similarity in attention mechanisms, we aimed for around 80,000 training steps. We chose a batch size of 16, consistent with the CRNN paper, to balance computational efficiency and gradient update effectiveness. With a training set of 78,755 examples, we planned to run the model for approximately 20 epochs to achieve 80,000 training steps.

We adopted the same optimizer configurations as in the "Attention Is All You Need" paper, which were proven optimal for transformer models. This involved using an Adam Optimizer with a varied learning rate that increased to 2e-5 after 5,000 warm-up steps before gradually decreasing. A similar optimizer was used for pretraining the ViT model. The varied learning rate helped mitigate large initial gradient jumps due to early training losses and allowed for more precise adjustments as the model approached convergence. The recommended number of warm-up steps is 5-10 percent of the total training steps, which equated to approximately 5,000 steps in our case.

The original Calvo-Zaragoza paper employs two performance measures: Sequence Error Rate (the ratio of incorrectly predicted sequences) and Symbol Error Rate (the average number of elementary editing operations needed to produce the reference sequence from the predicted sequence). We adopted the Sequence Error Rate as our evaluation metric, assessing it on a smaller validation set comprising 10 percent of the training set every five epochs. Predicted sequences were autoregressively generated with the ¡START¿ token removed after generation. The model with the best validation accuracy was saved for evaluation on the test set.

Unfortunately, we were unable to fully train the model. Despite multiple sources indicating that the pretrained ViT could be fine-tuned on different image sizes, and the original paper supporting this adaptation, the ViT model we downloaded from Hugging Face required input images to be 224x224 pixels. This requirement potentially eliminated meaningful information encoded from the music sequence, as patches contained more than one music note and images were significantly distorted. Ideally, each embedded patch would include less than a note, allowing the attention mechanisms to attend to multiple patches as necessary.

Additionally, we used Google Colab to train our model with a GPU, which required downloading all training files onto Colab. The training files were organized in a manner where smaller directories within the parent directory contained all necessary information for one image. When the parent directory was downloaded onto Colab, these subdirectories lost their contents, preventing the model from accessing the necessary images and labels. Locally, with global file paths, the model functioned correctly, producing outputs for batches and propagating gradients. This suggests that our model can run if the files are correctly downloaded onto Colab.

6. Conclusion/Future Work

While we were unable to produce results from the model we designed, we still believe that there is potential for a pre-trained ViT encoder and Transformer decoder model to achieve valuable results on the PrIMuS dataset. All prior work has either used a CNN encoder or a untrained ViT which we believe is limiting the performance of those prior models. The viability of a ViT for the encoder task in OMR does have the capability of being a valuable solution to OMR.

It is unfortunate that we were unable to produce any results despite having a seemingly functional model. Given the unique dataset and task, there was a lot of code that went into preprocessing and organizing this data to match with the new model and loss function. Additionally, I would be interested in discovering how to adapt the pretrained ViT to be finetuned on our new image size. That could mean using the Git Hub of the ViT directly and explicitly downloading the pretrained weights, instead of downloading the model from Hugging Face.

It would be interesting to see if our combine model would indeed perform better with the number of training examples provided. That would allude to the ability for ViT to indeed capture semantically meaningful information from images, and possibly overtake CNN's in image to sequence generation. Additionally, with more time, it would be interesting to compare the ViT large model with the ViT base model, to see if there were significant differences with relatively little finetuning data.

As for future work, it would be interesting to explore the use of a ViT decoder Transformer encoder model like the one we proposed but on an entire sheet such as the one proposed by Ríos-Vila et al., [8]. Despite the end result of our model, we do believe that our proposed model has serious potential as a solution to the OMR task and more broadly the image to sequence task.

7. Acknowledgments

We utilized a number of public codebases during for the code of our project.

- We used this as a base for our OMR model, especially the primus.py file and the dataset. https://github.com/OMR-Research/tf-end-to-end
- We used the "Attention is all you need" github repo for the decoder of our model. https://github.com/jadore801120/attention-is-all-you-need-pytorch/tree/master/transformer
- This is the pre-trained ViT from google that we used as the encoder¿ https://huggingface.co/google/vit-base-patch16-224-in21k

Christo worked on writing the functions to process and format the data for training and testing. He additionally worked on implementing the ViT Transformer and combining it with the decoder. Maddox was responsible for writing the Transformer decoder model and determining the experiment details. The writing of the paper was split evenly.

References

[1] Trocr: Transformer-based optical character recognition with pre-trained models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):13094–13102, Jun. 2023.

[2] J. Calvo-Zaragoza and D. Rizo. End-to-end neural optical music recognition of monophonic scores. *Applied Sciences*, 8(4), 2018.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[4] M. Fujitake. Dtrocr: Decoder-only transformer for optical character recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8025–8035, 2024.

[5] G. Kim, T. Hong, M. Yim, J. Nam, J. Park, J. Yim, W. Hwang, S. Yun, D. Han, and S. Park. Ocr-free document understanding transformer. In *European Conference on Computer Vision*, pages 498–517. Springer, 2022.

[6] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. Marçal, C. Guedes, and J. Cardoso. Optical music recognition: State-of-the-art and open issues. *International Journal of Multimedia Information Retrieval*, 1, 10 2012.

[7] A. Ríos-Vila, J. Calvo-Zaragoza, and T. Paquet. Sheet music transformer: End-to-end optical music recognition beyond monophonic transcription. *arXiv preprint arXiv:2402.07596*, 2024.

[8] A. Ríos-Vila, J. Calvo-Zaragoza, D. Rizo, and T. Paquet. Sheet music transformer++: End-to-end full-page optical music recognition for pianoform sheet music. *arXiv preprint arXiv:2405.12105*, 2024.

[9] A. Ríos-Vila, J. M. Iñesta, and J. Calvo-Zaragoza. On the use of transformers for end-to-end optical music recognition. In A. J. Pinho, P. Georgieva, L. F. Teixeira, and J. A. Sánchez, editors, *Pattern Recognition and Image Analysis*, pages 470–481, Cham, 2022. Springer International Publishing.

[10] A. Steiner, A. Kolesnikov, X. Zhai, R. Wightman, J. Uszkoreit, and L. Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.