# AIMH at SemEval-2021 Task 6: Multimodal Classification Using an Ensemble of Transformer Models

**Nicola Messina** **Fabrizio Falchi** **Claudio Gennaro** **Giuseppe Amato**
ISTI-CNR, via G. Moruzzi 1, 56124 Pisa, Italy
{name.surname}@isti.cnr.it

## Abstract

This paper describes the system used by the AIMH Team to approach the SemEval Task 6. We propose an approach that relies on an architecture based on the transformer model to process multimodal content (text and images) in memes. Our architecture, called DVTT (Double Visual Textual Transformer), approaches Subtasks 1 and 3 of Task 6 as multi-label classification problems, where the text and/or images of the meme are processed, and the probabilities of the presence of each possible persuasion technique are returned as a result. DVTT uses two complete networks of transformers that work on text and images that are mutually conditioned. One of the two modalities acts as the main one and the second one intervenes to enrich the first one, thus obtaining two distinct ways of operation. The two transformers outputs are merged by averaging the inferred probabilities for each possible label, and the overall network is trained end-to-end with a binary cross-entropy loss.

## 1 Introduction

Social networks play a critical role in our society. Nowadays, most of the ideas, thoughts, and political beliefs are shared through the internet using social platforms like Twitter, Facebook, or Instagram. Although these online services enable information to be spread efficiently and effectively, it is non-trivial to understand if the shared contents are free of subtle meanings altering people's judgment abilities.

Among all the types of content living in a social network, memes acquire a significant role. Memes are small yet effective units of information able to spread cultural ideas, symbols, or practices and usually exist under the form of pictures, possibly with overlayed text. Memes are created so that they can propagate rapidly and reach a large number of users; for this reason, they are one of the most popular types of content used in an online disinformation campaign, influencing the users through several rhetorical and psychological techniques, such as causal oversimplification, name-calling, or smear. The automatic detection of these memes and the disinformation techniques they are possibly employing is a challenging yet crucial task for the proper management of a social network.

In the last few years, machine learning and deep learning have defined remarkable milestones in automatic content extraction and reasoning from multimedia data. All these breakthroughs acquire a fundamental role in large-scale analysis of multimedia content from social networks.

In this work, we tackle the problem of recognizing which kind of disinformation technique is used to forge memes for a disinformation campaign. In particular, we propose an architecture based on the well-established transformer architecture model (Vaswani et al., 2017) for processing both the textual and visual inputs from the meme. This architecture, which we call DVTT (Double Visual Textual Transformer), comprises two full transformer networks working respectively on images and texts; however, each of these transformers is conditioned on the other modality. We consider this task as a multi-label classification problem, where text and/or images from the meme are processed, and probabilities of presence of each possible persuasion technique are returned as a result.

In this paper, we tackle subtasks 1 and 3 of the *SemEval 2021 Task 6* challenge [1] (Dimitrov et al., 2021). Subtask 1 consists of identifying which of 20 possible persuasion techniques are used in it given only the textual content; subtask 3 is very similar to subtask 1, but both textual and visual contents of the meme are used, and there are 22 possible persuasion techniques. Our proposed models

[1] https://propaganda.math.unipd.it/semeval2021task6/index.html

could reach the 5th position for subtask 1 and the 4th position for subtask 3 on the publicly available leaderboard. The code for replicating our results is available on GitHub[2].

## 2 Background

Recently, machine learning, and deep learning in particular, defined astonishing milestones in automatic content extraction and reasoning from multimedia data. In particular, concerning joint visual and textual analysis, many state-of-the-art approaches succeeded in tasks like visual question answering (Hu et al., 2017; Anderson et al., 2018; Teney et al., 2017), image captioning (Zhou et al., 2020; Rennie et al., 2017; Huang et al., 2019; Cornia et al., 2019), and image-text matching (Chen et al., 2019; Lu et al., 2019; Faghri et al., 2018; Lee et al., 2018; Messina et al., 2020), often using structured reasoning using graph networks and graph convolutions.

In the last few years, a graph-network related model, the transformer network (Vaswani et al., 2017), acquired increasing attention on the joint processing of images and texts. Many works, inspired by the BERT model (Devlin et al., 2019), obtained remarkable results on word region alignments, visual-question answering, and image-text matching using transformer encoders (Lu et al., 2019; Qi et al., 2020; Su et al., 2020).

Recently, the authors in (Carion et al., 2020) used the full transformer stack to construct a powerful object detector, demonstrating these models' potential in pure visual contexts.

Given the enormous flexibility of the transformer architecture, in this work, we consider images and texts respectively as sets and sequences of vectors, and we ask the transformer to process them to produce probabilities of presence of each possible persuasion technique.

## 3 System Overview

In this section, we first give a brief overview of the Transformer architecture on which our proposal is based; then, we present in detail our system proposals for solving subtasks 1 and 3.

### 3.1 Review of the Transformer Architecture

In the original transformer formulation for language translation, the source sequence is processed using the *transformer encoder* model, which creates a suitable set of contextualized vectors encoding the input sequence. Using the representations created by the encoder, the *transformer decoder* module is trained to predict the words for the target sentence one at a time. During the decoding process, the decoder is conditioned, at each time step, by the vectors generated by the encoder.

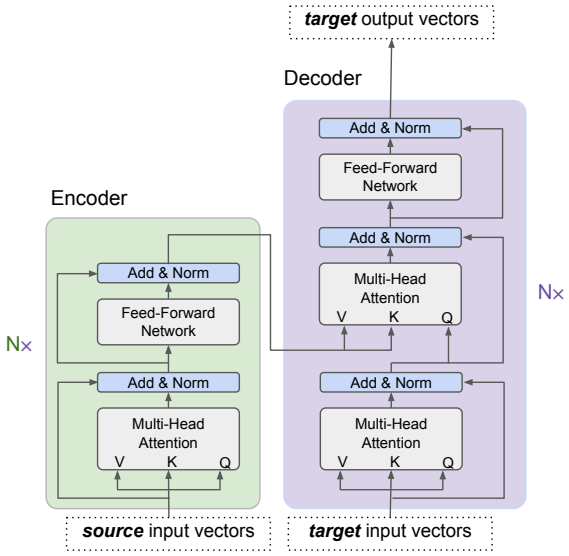



Figure 1: The transformer network. Encoder and Decoder modules are replicated N times.

Both the encoder and the decoder modules leverage the power of the multi-head attention mechanism. This mechanism transforms every word representation from a target sentence to a new representation space conditioned on the words from a source sentence.

The multi-head attention associates to the source sequence vectors $\{\boldsymbol{s_i}\}$ a key $K_i$ and to the target vectors $\{\boldsymbol{t_j}\}$ a query $Q_j$ and a value $V_j$; the target values are transformed using the scaled dot-product attention as follows:

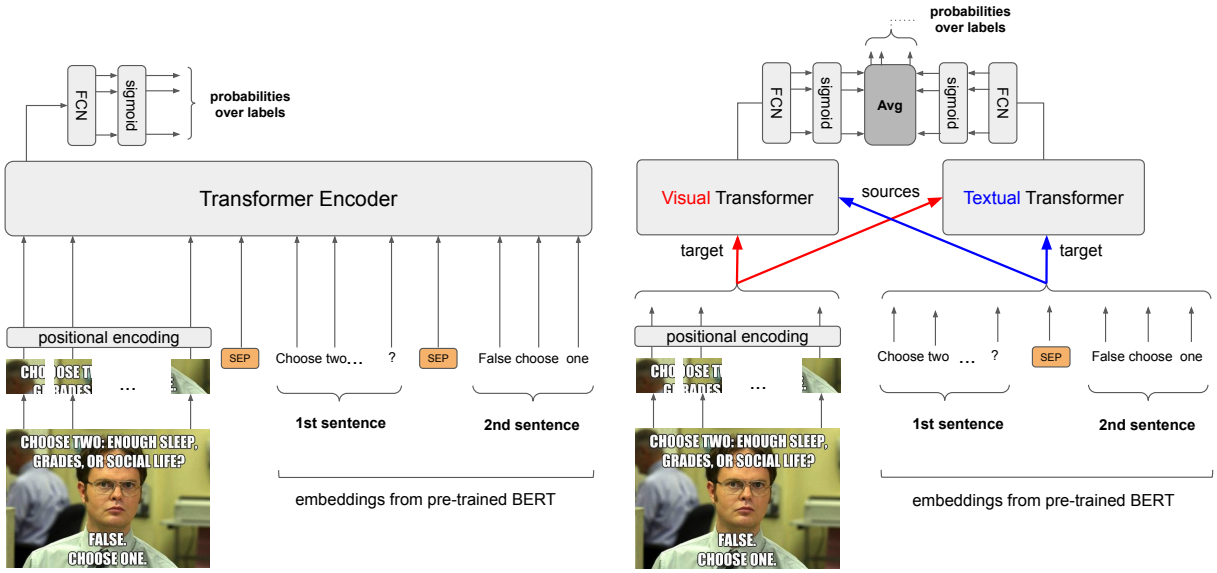
$$\text{Att}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \quad (1)$$

This is the core of the multi-head attention mechanism, which is the fundamental building block in the transformer architecture for both the encoder and the decoder modules (Figure 1).

### 3.2 A Transformer Encoder Baseline

Although the transformer architecture was originally designed to handle sentences (sequences of words), the model in itself can effectively process an arbitrary set of vectors possibly coming from

[2] https://github.com/mesnico/MemePersuasionDetection




(a) VTTE (b) DVTT

Figure 2: The proposed architectures: (a) the Visual-Textual Transformer Encoder model (VTTE) used as a baseline, and (b) the novel Double Visual Textual Transformer model (DVTT). The shown meme is taken from https://engineermemes.blogspot.com, and it is licensed under the Creative Common license.

other modalities (e.g., different chunks of an image).

For this reason, many works (Lu et al., 2019; Qi et al., 2020; Su et al., 2020) recently proposed architectures based on the transformer encoder model to jointly reason on images and texts for solving tasks like visual question answering or image-text matching.

Inspired by these works, we defined a baseline for inferring probabilities over the possible persuasion techniques by feeding images and texts to a transformer encoder, using the first output token as the input to the multi-label classifier head. The transformer encoder visual and textual input features are pre-extracted respectively from a CNN and a pre-trained BERT model, as explained in Section 4. Like in BERT, where different sentences are encoded by separating them using a special token, the textual and the visual inputs are separated by the SEP embedding. An overview of this approach is presented in Figure 2a. We refer to this baseline as *VTTE* (Visual-Textual Transformer Encoder).

### 3.3 Double Visual-Textual Transformer

In this work, instead, we propose an architecture that can exploit the full Transformer architecture to jointly reason on visual and texts and producing label probabilities as output. We call this model *DVTT* (Double Visual Textual Transformer), and it is outlined in Figure 2b. DVTT is composed of two different transformer networks able to process visual and textual inputs concurrently; the important aspect of DVTT is that each transformer is conditioned on the other modality so that it is possible for the whole architecture to jointly reason on the two modalities following two different paths: in the first, the text is the key aspect, and images integrate the reasoning performed on the text; conversely, in the second, the images are the primary modality and the text intervene to enrich the visual features.

For each of the two transformers, the final head is a multi-classification head constructed on the first token of the output sequence. In particular, a linear layer outputs the logits over each possible persuasion technique, and the final softmax operator converts logits into probabilities, exactly like in the VTTE baseline model. The two transformers outputs are merged by averaging the inferred probabilities for each possible label, and the overall network is trained end-to-end with a binary cross-entropy loss.

## 4 Experiments

We used the data provided by the *SemEval 2021 Task 6* challenge organizers to train and validate our model. Although we mainly concentrated on subtask 3 (images + texts), we also tackled subtask 1, which is essentially equivalent to subtask 3, except that only the text is available.

**Dataset** The provided dataset comprises 687 memes for training, 63 memes for validating on the so-called development set, and 200 memes for the final testing. All the memes carry textual captions written in English. Note that, in the end, we were allowed to use the annotations for the development set, so we had at our disposal a total of 750 annotated memes to use for the training and validation phases. The annotations consist of a list of persuasion techniques for every meme. In subtask 1 there are 20 possible persuasion techniques and 22 in subtask 3.

**Metrics** The official metrics for computing the model performance are the Micro-$F_1$ and Macro-$F_1$ scores;

The $F_1$-score is defined as the harmonic mean of precision and recall:

$$F_1 = \frac{2}{\text{recall}^{-1} + \text{precision}^{-1}} \tag{2}$$

The $F_1$-score gives values in the interval [0, 1], hence it is often a good way of summarizing the performance of binary classifiers.

The difference between Micro-$F_1$ and Macro-$F_1$ scores lies in the way *precision* and *recall* are computed: in Micro-$F_1$, they are computed from all the true positives, false positives, and false negatives over all the labels; for this reason, Micro-$F_1$ gives each sample the same weight, thus giving more emphasis to the most frequent labels. On the other hand, Macro-$F_1$ is computed as the mean value among the $F_1$-scores computed on the different labels: Macro-$F_1 = \frac{1}{N}\sum_1^N F_1^i$, where $N$ is the number of labels and $F_1^i$ is the $F_1$-score computed among the samples having label $i$. In this case, all the classes contribute equally regardless of how often they appear in the dataset.

**Model Setup** For subtask 3, we used the proposed DVTT model (Figure 2b). We used a learning rate of $5 \cdot 10^{-5}$ and a batch size of 8. We trained the models for 40 epochs in all the experiments, decreasing the learning rate after 30 epochs to $5 \cdot 10^{-6}$. The transformer is composed of 4 encoder layers and 4 decoder layers, with 1024-dimensional feed-forward networks for producing queries, keys, and values.

As a baseline for subtask 3, we used the VTTE architecture (shown in Figure 2a), composed of a 4-layer transformer encoder module, with a multi-label classification head on top, exactly like the one in DVTT. For subtask 1, instead, we used the VTTE architecture (Figure 2a) with the same setup used for the subtask 3 baseline, except that the visual input is not fed to the network.

**Features Extraction** For all the conducted experiments, we obtained suitable visual and textual features from pre-trained state-of-the-art networks. Concerning images, we re-scaled them to $256 \times 256$, and we took a $224 \times 224$ crop (a random crop during training and a center crop during inference). We also normalized the images using the pixels mean and standard deviation computed on the whole dataset.

In order to input an image to the transformer, we had to encode it as a set of features. We used a ResNet50 pre-trained on image classification, as it is characterized by a good performance at low computational costs compared to deeper backbones; we down-sampled the features maps from the last convolutional layer to a $7 \times 7$ spatial grid of 2048-dimensional features. The resulting flattened 49 visual features were then augmented with their spatial positions by appending the normalized coordinates of the chunk to the 2048-dimensional visual feature. Another possibility consisted of using visual features extracted from state-of-the-art object detectors, like Faster-RCNN. However, images carried in memes are not homogeneous: they show possible stacked scenes and overlayed text, making it very difficult for an object detector to identify the most critical regions.

Concerning text processing, we used a pre-trained BERT model (Devlin et al., 2019) for extracting word embeddings. BERT embeddings are trained on some generic language processing tasks such as sentence prediction or sentence classification and demonstrated state-of-the-art results in many downstream tasks. Every meme can carry one or more sentences, encoded in the same string and separated by "\n\n". For this reason, during the string tokenization phase, we simply replaced "\n\n" with the SEP token. In the basic DVTT model, we trained only the transformer models, leaving the feature extractor fixed. In the Experiments section, we also report the results from a fine-tuning of the feature extractors.

**Validation** The test-set annotations were hidden to the participants, so the model should be validated using a split of the available annotated data. Given that the available annotated memes are relatively


**Predicted:** Flag-waving, Name calling/Labeling, Smears, Transfer

**GT:** Smears, Appeal to (Strong) Emotions, Name calling/Labeling, Flag-waving, Transfer

**Predicted:** Appeal to fear/prejudice, Exaggeration/Minimisation, Loaded Language, Name calling/Labeling, Smears

**GT:** Glittering generalities (Virtue), Loaded Language, Name Calling/Labeling, Smears, Transfer, Whataboutism

Figure 3: Example of predictions from the DVTT model for subtask 3. In green, the true positives labels; in red, the false positives labels. Images obey to the Creative Common license and they are searched on the Bing image-search engine using "free to modify, share and use" license filtering.

| **Model** | Macro-$F_1$ | Micro-$F_1$ |
|---|---|---|
| VTTE (Baseline) | 0.327 | 0.596 |
| DVTT | 0.336 | **0.601** |
| DVTT - Balanced | 0.300 | 0.489 |
| DVTT - Finetuned | **0.341** | 0.592 |
| DVTT - 2 layers | 0.310 | 0.596 |
| DVTT - 6 layers | 0.325 | 0.583 |

Table 1: Ablation results on subtask 3

| **Model** | Macro-$F_1$ | Micro-$F_1$ |
|---|---|---|
| VTTE | 0.372 | 0.566 |
| VTTE - Balanced | 0.361 | 0.490 |
| VTTE - Finetuned | 0.386 | **0.581** |
| VTTE - 2 layers | 0.365 | 0.565 |
| VTTE - 6 layers | **0.389** | 0.569 |

Table 2: Ablation results on subtask 1. Note that VTTE in this case does not receive the images in input.

few, we validated our model using cross-validation. In particular, we split the training data into six different folds, training six different models by using five out of six data folds and validating them using the remaining fold. We selected the model having the best sum of Micro-$F_1$ and Macro-$F_1$ scores on the validation fold. All the performance measures reported in the Results section are an average of the metrics from this 6-fold validation procedure.

For participating in the final competition on the test set, we prepared an ensemble model composed of all the six trained models, and we produced the final probabilities by soft-voting.

We used a final binary-classification threshold of 0.3 over the label probabilities.

## 5 Results

Concerning subtask 3, we studied the performance of our DVTT model by comparing the $F_1$-scores against the VTTE baseline; furthermore, we tried also to train the model using a balanced sampling of the labels and to fine-tune the feature extractors (BERT and the ResNet-50), using a learning rate of $1/10$ with respect to the one used for training the transformer models. Using a lower learning rate during the fine-tuning process is a common procedure to avoid model overfitting. We also report the results of slightly different variants of the DVTT model obtained by increasing and decreasing the number of the transformer's encoder / decoder layers: the base architecture contains four layers; we also experimented with two and six. The results of these experiments are reported in Table 1.

For subtask 1, instead, we used the VTTE model without visual input, trying out the same experiments performed for subtask 3. In this case, when varying the number of layers, we only considered the transformer encoder ones (there is no decoder in the VTTE model). The ablation results on subtask 1 are reported in Table 2.

## 6 Discussion

Looking at the subtask 3 results in Table 1, we can notice that the proposed DVTT model can achieve slightly better results than the VTTE baseline. In particular, the DVTT with fine-tuned BERT and ResNet50 modules achieve the best results on the Macro-$F_1$ metric. Also, the choice of using four encoder and decoder layers seems to lead to the best compromise on both the metrics. Concerning the results of subtask 1 in Table 2, fine-tuning the BERT model is even in this case a good choice. Fine-tuning the feature extractors, in fact, enables the model to slightly adjust the weights of the backbones pre-trained on generic tasks to align them to the specific domain.

Figure 3 reports some examples of predictions from our model for subtask 3. We evidenced in green the true positives and in red the false positives. The model can correctly identify most of the persuasion techniques. However, there are cases where it is probably necessary to access more contextual information to solve the most complex labels. For example, in the second meme from the left, the model outputs the label *Exaggeration/Minimisation* probably due to the presence of vague quantities (*Killed thousands of innocents*). It would be necessary to access external data to effectively reason on the complex common sense and historical facts hidden behind these complex memes.

## 7 Conclusions

In this work, we proposed transformer-based models for tackling subtasks 1 and 3 of the SemEval-2021 Task 6, concerning the identification of persuasion techniques in memes. In particular, for subtask 3 which involves both images and texts from the meme, we proposed a Double Visual Textual Transformer (DVTT) model. This model uses the full power of the transformer architecture; it demonstrated better results than the baseline, which is composed of a single transformer encoder module fed with both images and text. On the public leaderboard, we reached 4th place on subtask 3. Using the baseline model, which can process text alone without images, we also tackled subtask 1, reaching 5th place on the public leaderboard.

In the future, we plan to improve our visual feature extraction pipeline, using face expression detection and classification and possibly employing ad-hoc trained object detectors suitable for meme images. Also, it would be interesting to study the effective reasoning abilities of the proposed models, by leveraging the attention mechanisms of the transformer, possibly integrating the data with a knowledge base of historical facts that helps to create a more suitable context.

## Acknowledgments

This work was partially supported by "Intelligenza Artificiale per il Monitoraggio Visuale dei Siti Culturali" (AI4CHSites) CNR4C program, CUP B15J19001040004, by the AI4EU project, funded by the EC (H2020 - Contract n. 825619), and AI4Media under GA 951911.

## References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proc. of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Uniter: Learning universal image-text representations. *arXiv preprint arXiv:1909.11740*.

Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. 2019. Show, control and tell: A framework for generating controllable and grounded captions. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 8307–8316.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT 2019*, pages 4171–4186. Association for Computational Linguistics.

Dimiter Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. 2021. Task 6 at semeval-2021: Detection of persuasion techniques in texts and images. In *In Proceedings of the 15th International Workshop on Semantic Evaluation*.

Fartash Faghri, David J. Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2018. VSE++: improving visual-semantic embeddings with hard negatives. In *BMVC 2018*, page 12. BMVA Press.

Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. 2017. Learning to reason: End-to-end module networks for visual question answering. In *Proc. of the IEEE International Conference on Computer Vision*, pages 804–813.

Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. 2019. Attention on attention for image captioning. In *Proc. of the IEEE International Conference on Computer Vision*, pages 4634–4643.

Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. 2018. Stacked cross attention for image-text matching. In *Proc. of the European Conference on Computer Vision (ECCV)*, pages 201–216.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, pages 13–23.

Nicola Messina, Fabrizio Falchi, Andrea Esuli, and Giuseppe Amato. 2020. Transformer reasoning network for image-text matching and retrieval. In *International Conference on Pattern Recognition (ICPR) 2020 (Accepted)*.

Di Qi, Lin Su, Jia Song, Edward Cui, Taroon Bharti, and Arun Sacheti. 2020. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data. *arXiv preprint arXiv:2001.07966*.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. Vl-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*.

Damien Teney, Lingqiao Liu, and Anton van Den Hengel. 2017. Graph-structured representations for visual question answering. In *Proc. of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *AAAI*, pages 13041–13049.