Submission for Deep Learning Exercise

Team: shallow_learning_group
Students: Batuhan Karaca

December 19, 2023

Warmup

Attention on Input

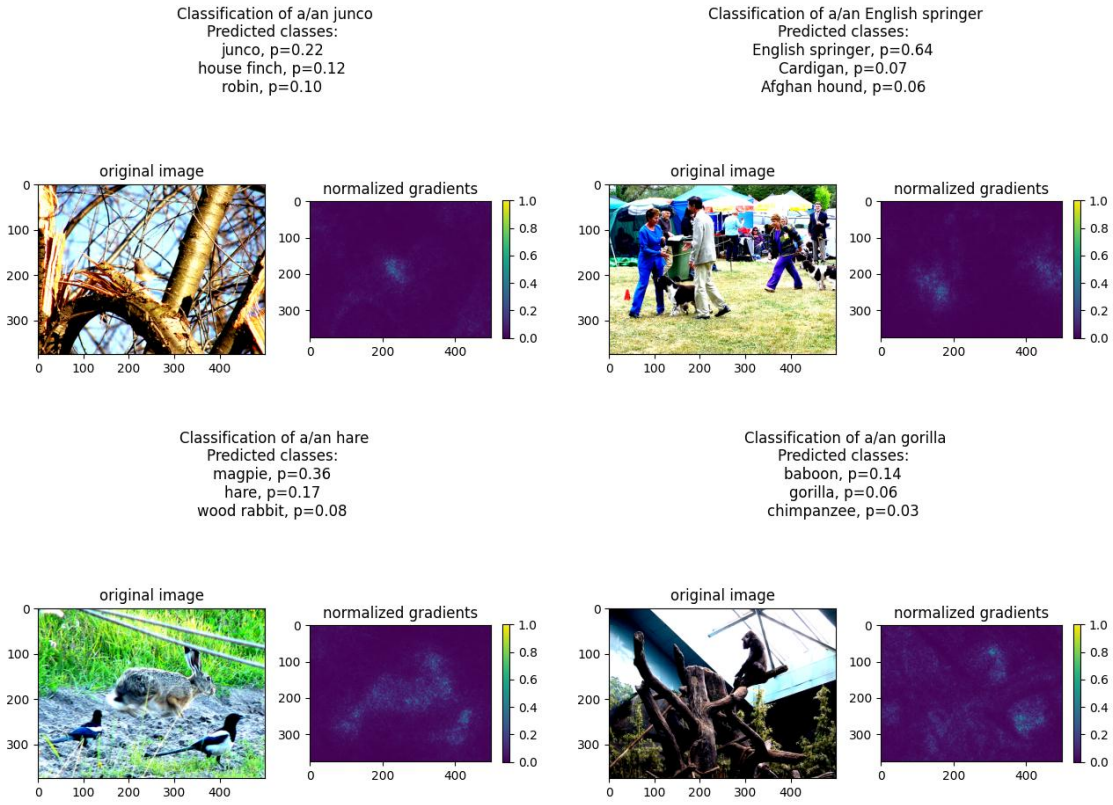

Figure 1: Sample images with their predictions and normalized gradients

As seen in figure 1, the network pays more attention (markings on the gradients), to the inputs (i.e. pixels) which are more relevant to the classes that have highest confidence(s). For example, the gradients mark only the locations of the junco (as it is the only instance relevant), both the hare and two birds, and the dogs.

When the network is highly confident, their gradients are are confident as well. For instance the image of monkeys is more noisy than the image of dogs even though both images have plenty amount of similar, or

real objects (compared to the other two images). The network may be less confident for monkeys because the image is lacking some contrast in some parts, creating *illusions* of monkeys.

Adversarial Examples

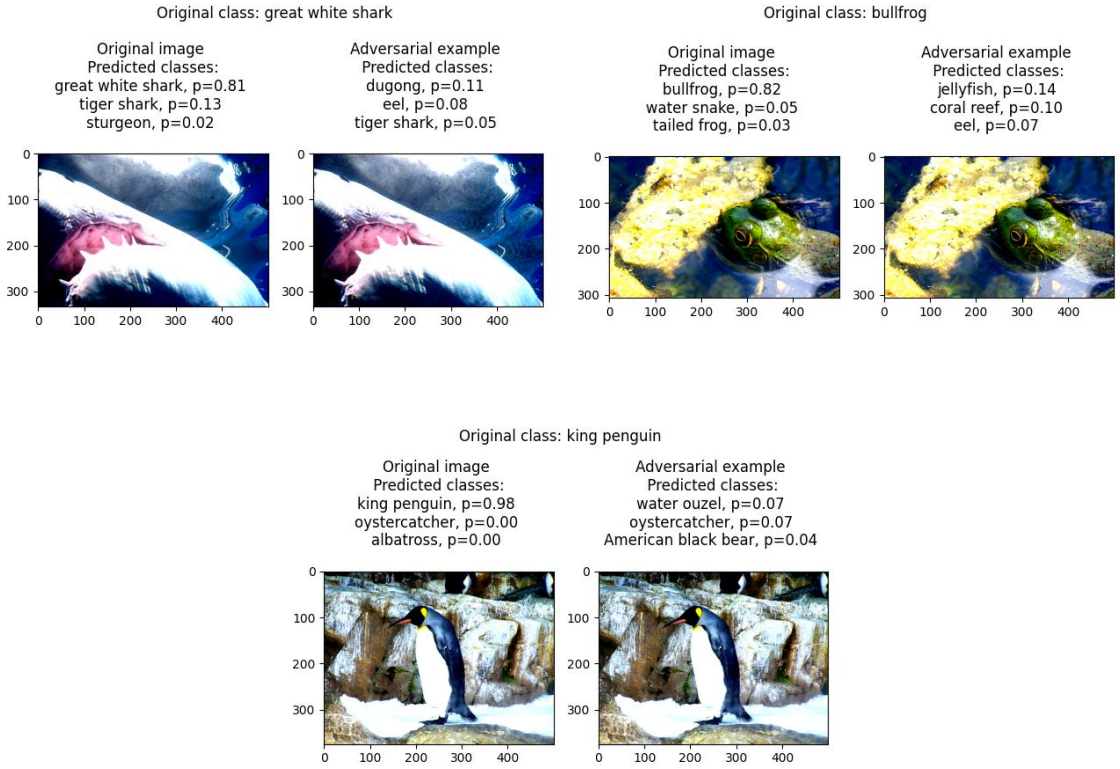

Figure 2: Sample images with their noisy versions

Learning to Count using Explicit Attention

As verified from the output of `plot_attention.py`, the sample input is [0, 1, 0, 1, 1, 2, 0, 2, 0, 0].

Visualizing Explicit Attention

Row	Number 0	Number 1	Number 2
Output-0	0.958	0.862	0.007
Output-1	1	almost 1	almost 0
Output-2	0.413	0.296	0.586

Table 1: Attention weights corresponding to the numbers

Figure 3, visualizes the attention weights, whereas Table 1 shows the corresponding values for each number. As seen in the figure, for the first two queries, the network attends to number 0. For the last query, it attends to number 2. Note that network attends to numbers 0 and 1 with large margin, whereas it attends to number 2 with smaller margin. This inequality

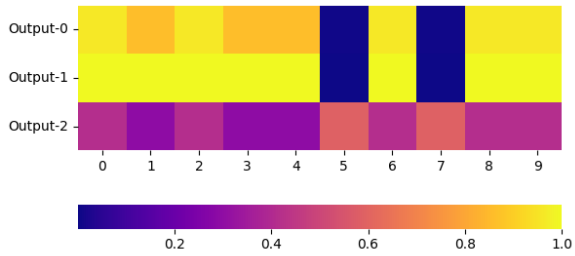

Figure 3: Attention weights when sigmoid is used during attention step on sample input

Model with Softmax

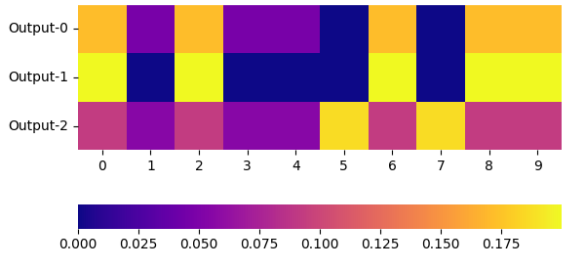

Figure 4: Attention weights when softmax is used during attention step on sample input

As seen in figure 4, softmax function maps the values to a smaller range, unlike sigmoid. Therefore, the overall network should output values with lower confidence, and hence the accuracy may decrease (indeed it does with around 0.58 instead of 1.00 of sigmoid on test data).

Neural Machine Translation

Note that because German has composite words, (i.e. Videoform for `--line=7`) made up of multiple words (expanding German vocabulary almost infinitely), the corresponding word in input became unknown. As seen in figure 5, the attention weights are mostly diagonal (not necessarily for encoder-decoder layers). This means that the diagonal words are the most similar. This could be due to both languages having similar order, with few exceptions. One exception is that in German language, in the sentences with modal verbs (i.e. *werden/going to*) the main verb (i.e. *erzählen/tell*) comes at the end, whereas in English language it comes after the modal verb.

No head was able to capture the relation between the articles (i.e. *the* and *das*), and related them to other

words. This could be due to the fact that the articles have no meanings by themselves.

The preposition *to* does not have the meaning of *towards* (or *zu*/*nach* in German). It is used in the phrase *going to* to mean a metaphorical move *to* future. For this sentence, separating *going* and *to* may not be a good idea as *to* has lost its real meaning in the phrase. Therefore, the network could not capture a meaningful relation for it.

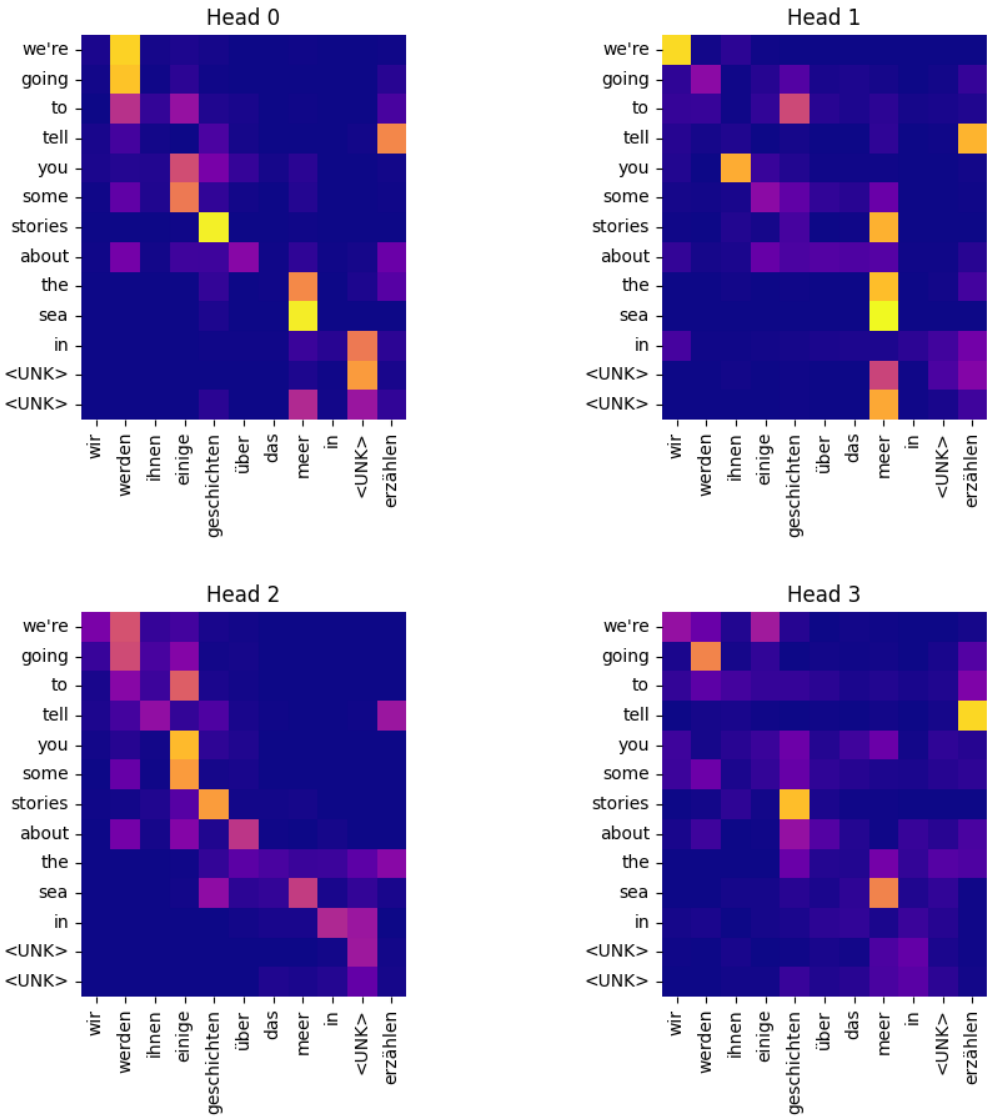

Figure 5: Attention weights of the last encoder-decoder attention layer