**ECE 09495/09595**

**Assignment 4 Jacob Matteo**

**Github:** <https://github.com/jmatteo/Machine-Learning-Fall-20/tree/main/Assignments>

**Instructions**

1. Max Credit: 100 Points
2. All questions are from the Textbook – Dive into Deep Learning (<https://d2l.ai/>).
3. Submit a single PDF.
4. Please do not include code. Upload the code to your GitHub, share it publicly and add link in the assignment.

**Questions**

**Part – I**

1. Question 1: Read this blog (<http://jalammar.github.io/visualizing-neural-machine-translation-mechanics-of-seq2seq-models-with-attention/>) and write one-page summary on using ‘attention’ in Neural Machine Translation. (10 points)

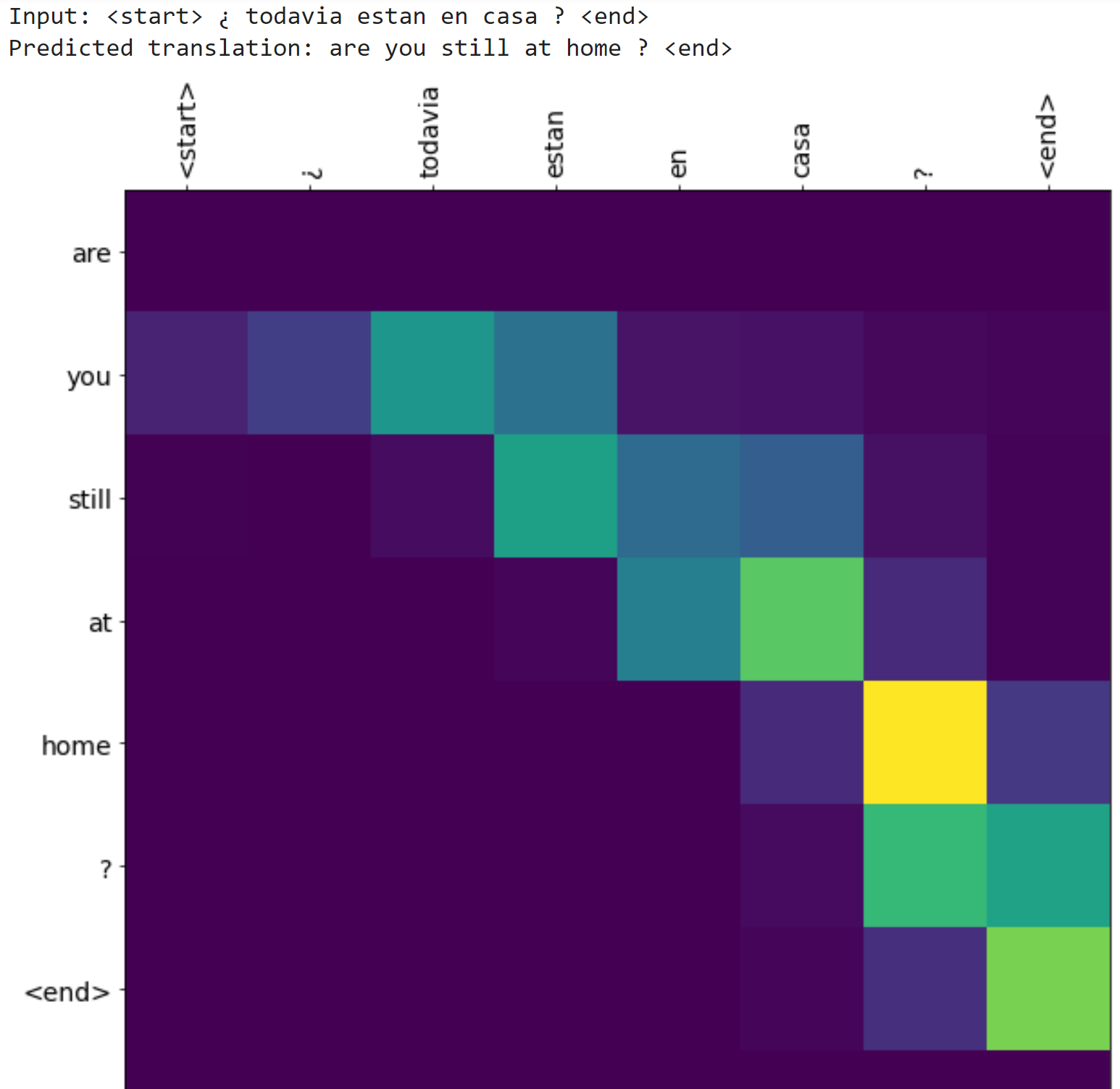
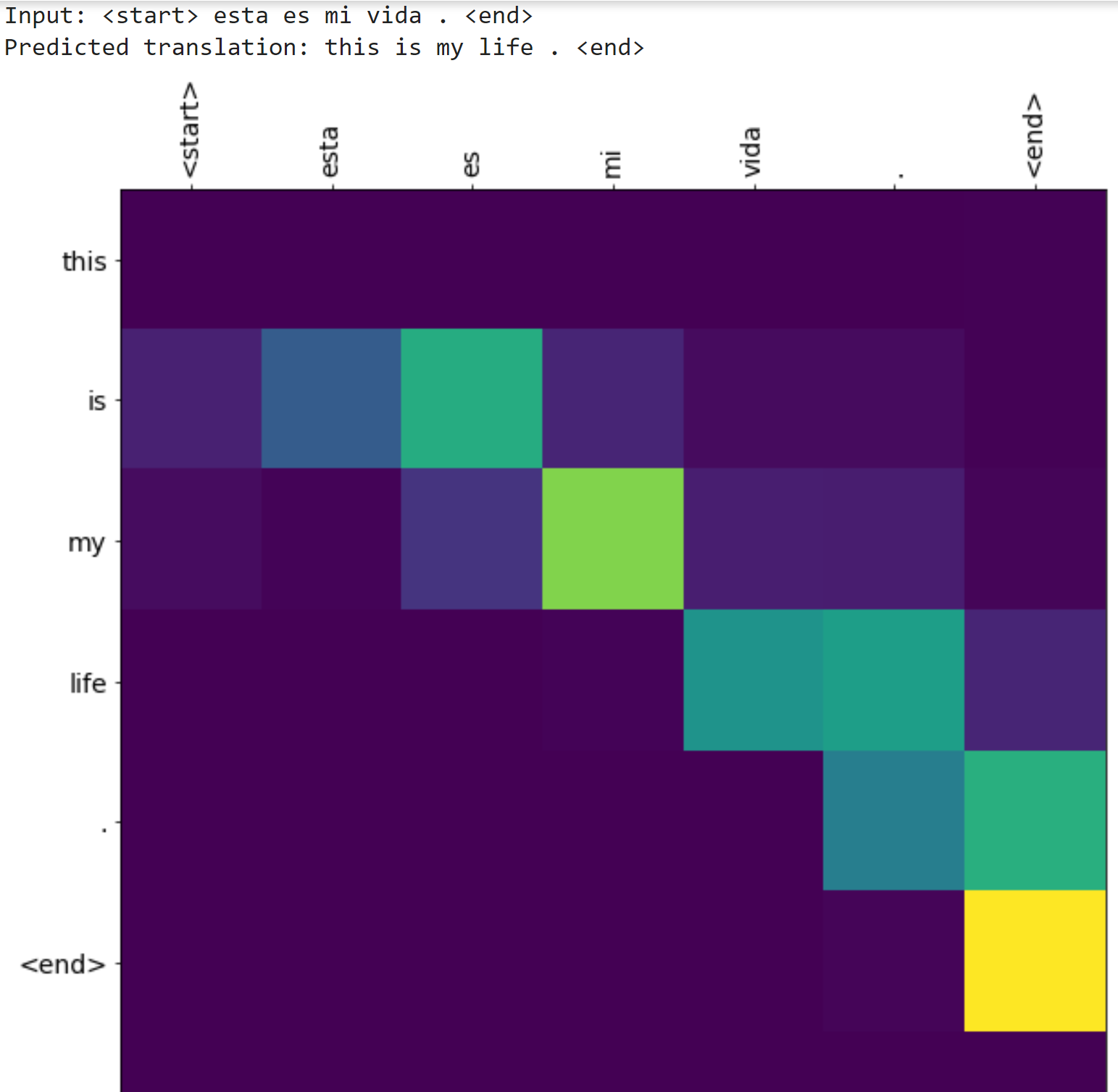
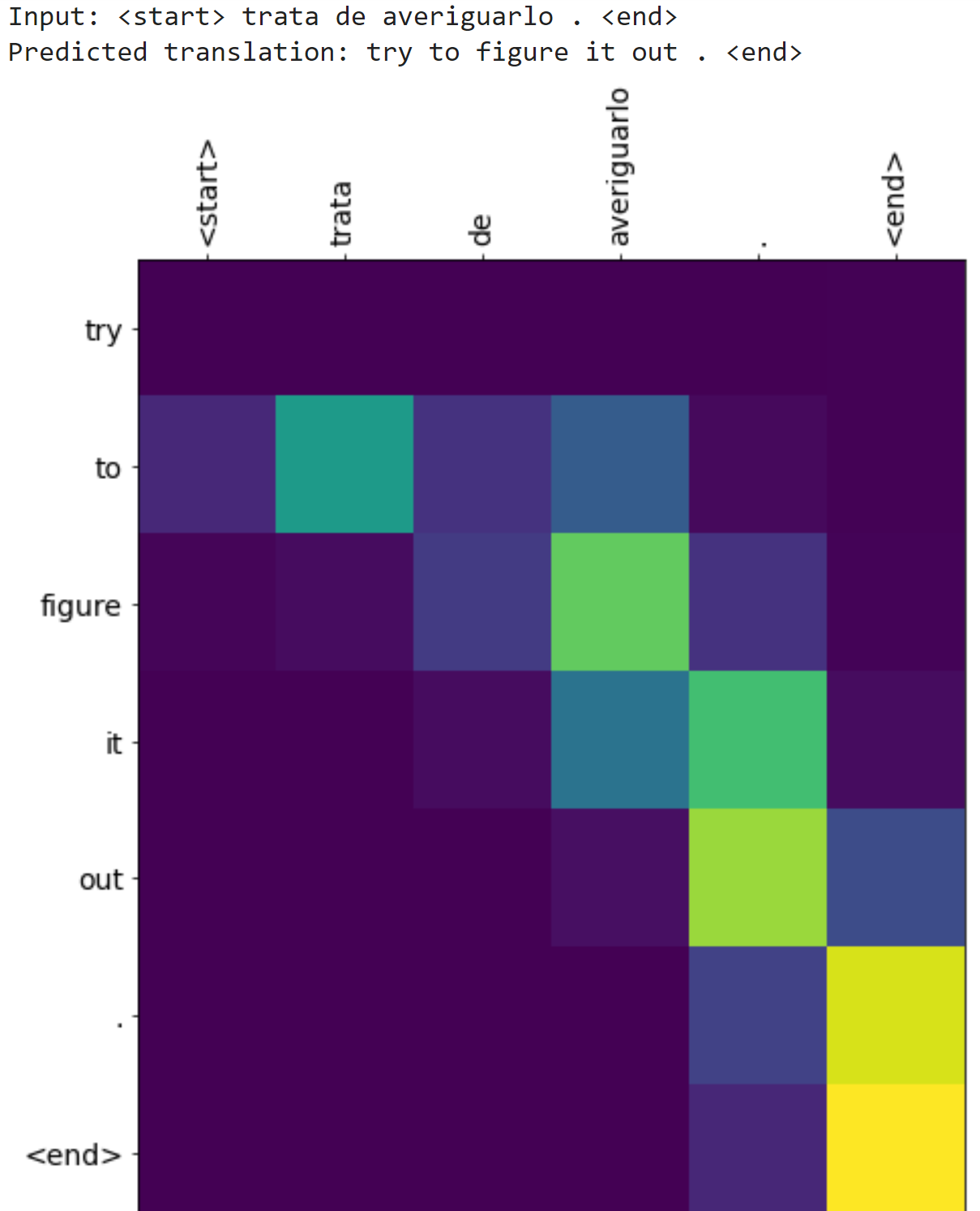
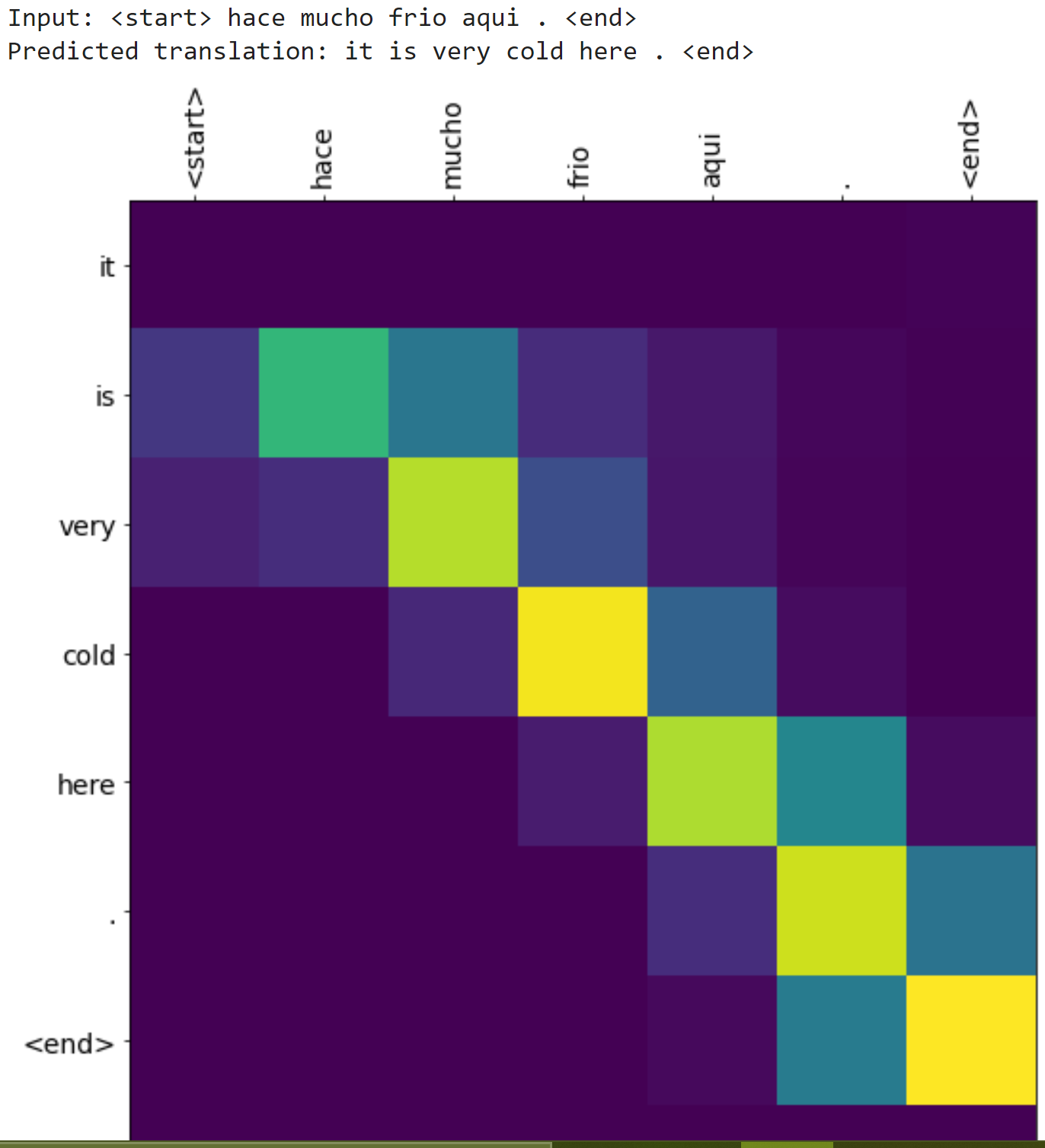
A big issue in machine learning is increasing the accuracy of the neural networks. Traditional neural networks take in small bits of data into an encoder block which turns the inputted information into usable data via a hidden state, sends the extracted hidden state to a decoder block which extracts the data from the encoded information, and then hands this bit of data to the rest of the network. How this works is each piece of data is encoded into a hidden state, and then this hidden state gets used in the next encoding to create a second hidden state, and this process then iterates. So, this process is repeated for each piece of information, one by one, to create each next hidden state, until all of the pieces of data are used and the final one hidden state is created. This last hidden state is then sent to the decoder to produce the output for the neural network. While this process works, it is not as accurate as it could be.

The fix to this lack of accuracy is the attention model. While slower to compute, the attention model increases the accuracy of neural networks. The attention model works by fundamentally changing how the encoder-decoder works. First, the encoder works similarly as before, but instead of sending just the final hidden state to the decoder, it sends every hidden state. Next, the decoder takes a scoring mechanism and softmaxes it, comparing the softmax’s score with each hidden state to weight the state. These weighed vectors are then lastly added together to create the final output, a context vector. Where in the traditional neural network, only the final hidden state is used in creating the final output, when using attention, each word must be compared to every hidden state. This is why it takes a much longer time and more compute power & storage to calculate a neural network that uses attention.

1. Build your own Spanish to English Translation System. Your reply should include description of the dataset, the model, its architecture, training details. Also, provide samples of the input and output. This link provides a tutorial in TensorFlow: <https://www.tensorflow.org/tutorials/text/nmt_with_attention>. (30 points)

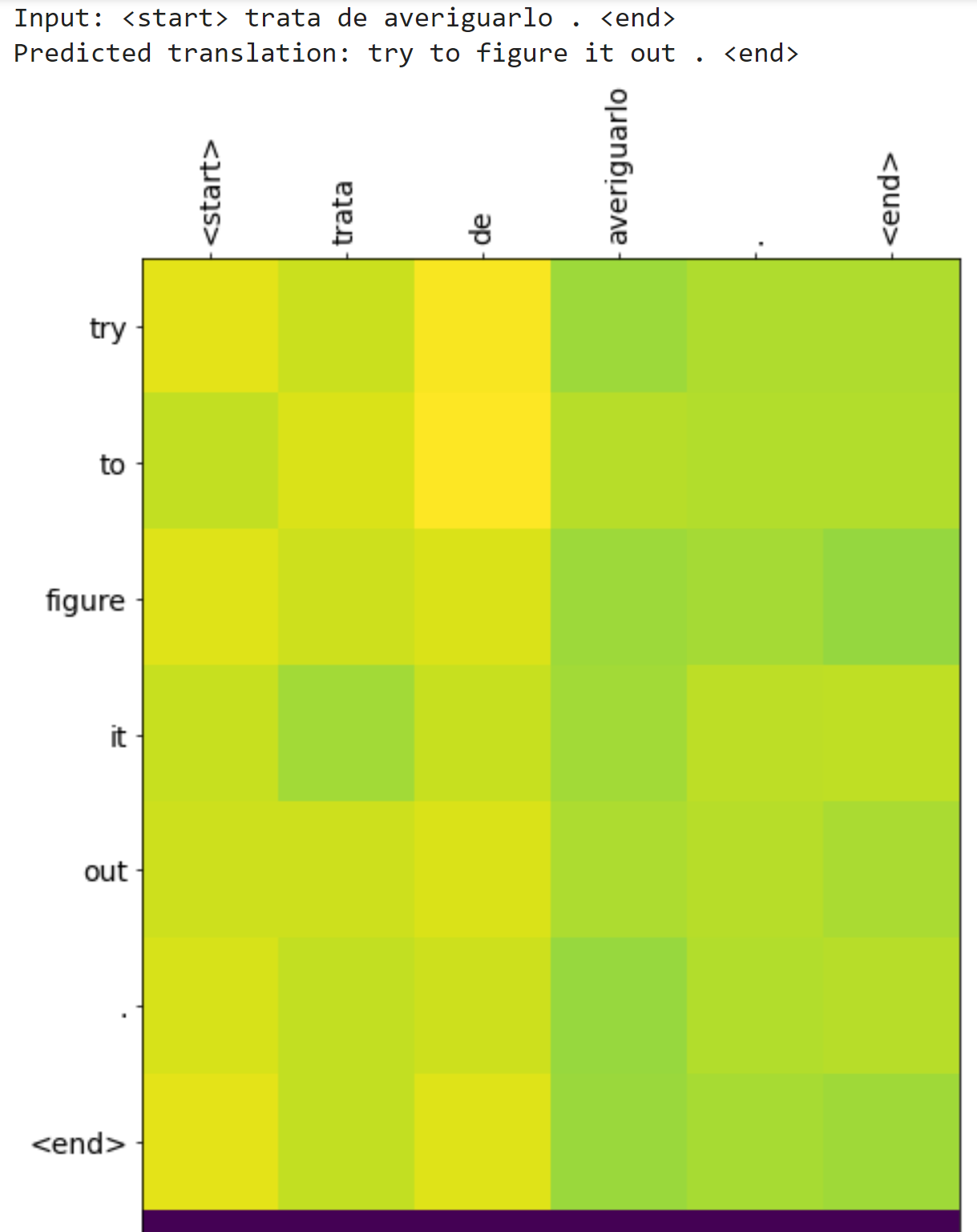
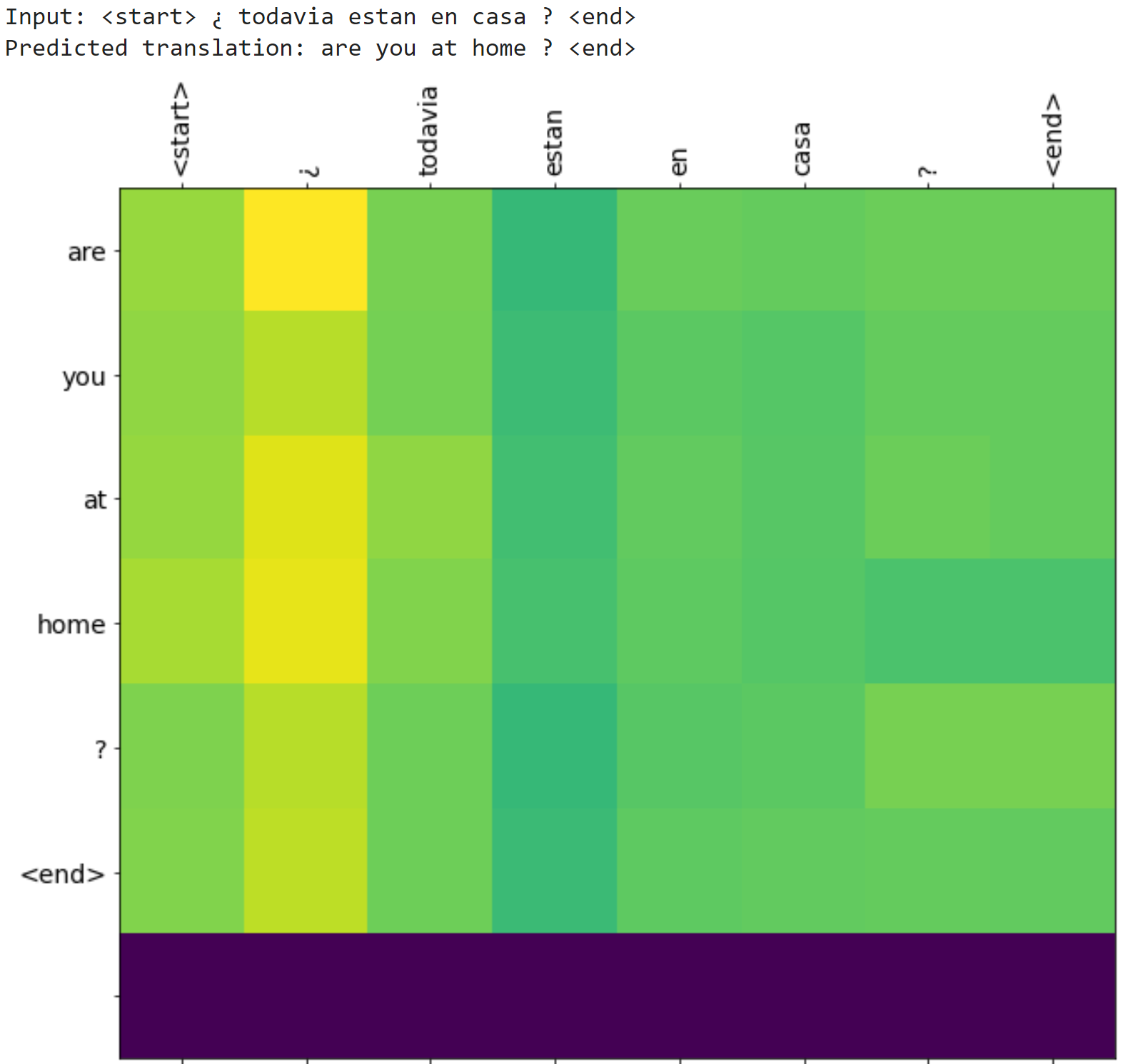
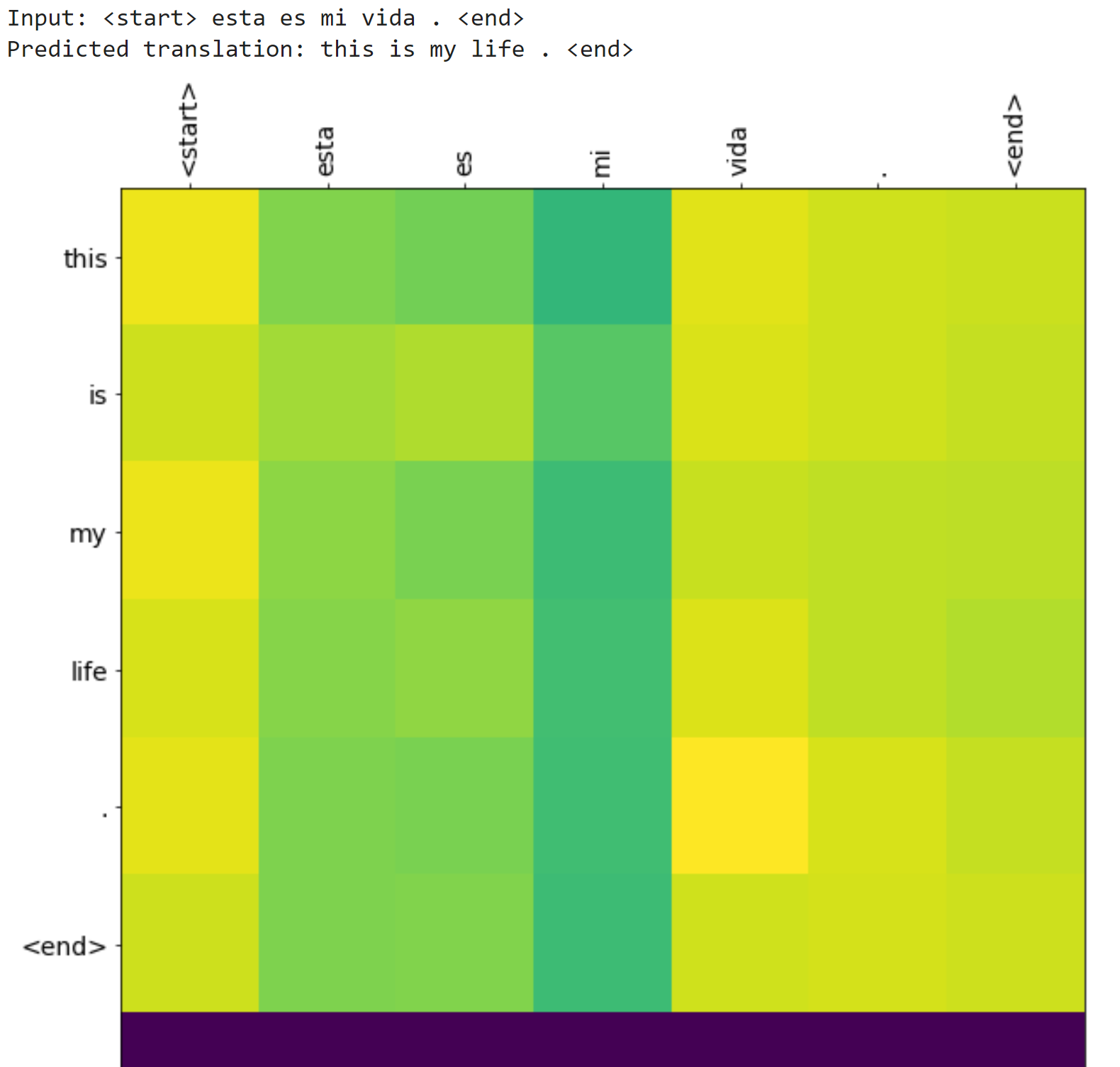
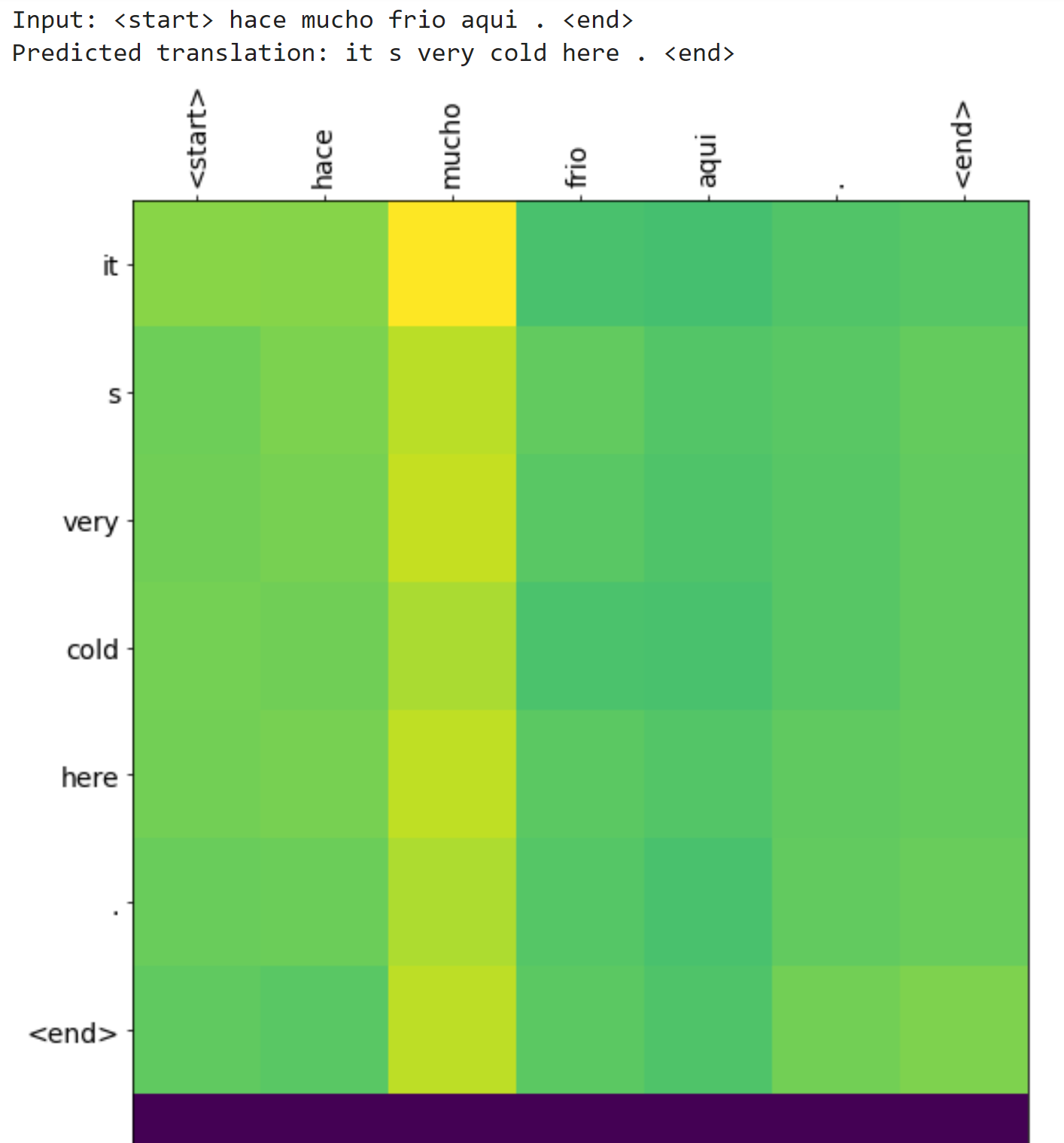
Link to code: <https://colab.research.google.com/drive/1RonirVaLnquOHyjL7DP1q6s4oZuvg_LV?usp=sharing>

The model used was a sequence to sequence (seq2seq) with an encoder and decoder model. It also uses attention. The ai translates Spanish to English and uses a dataset provided by manythings.org to train the ai with. The training used a max epoch of 10.



1. What happens when you remove ‘Attention’ in your Spanish to English Translation System? Give some example translations. (10 points)

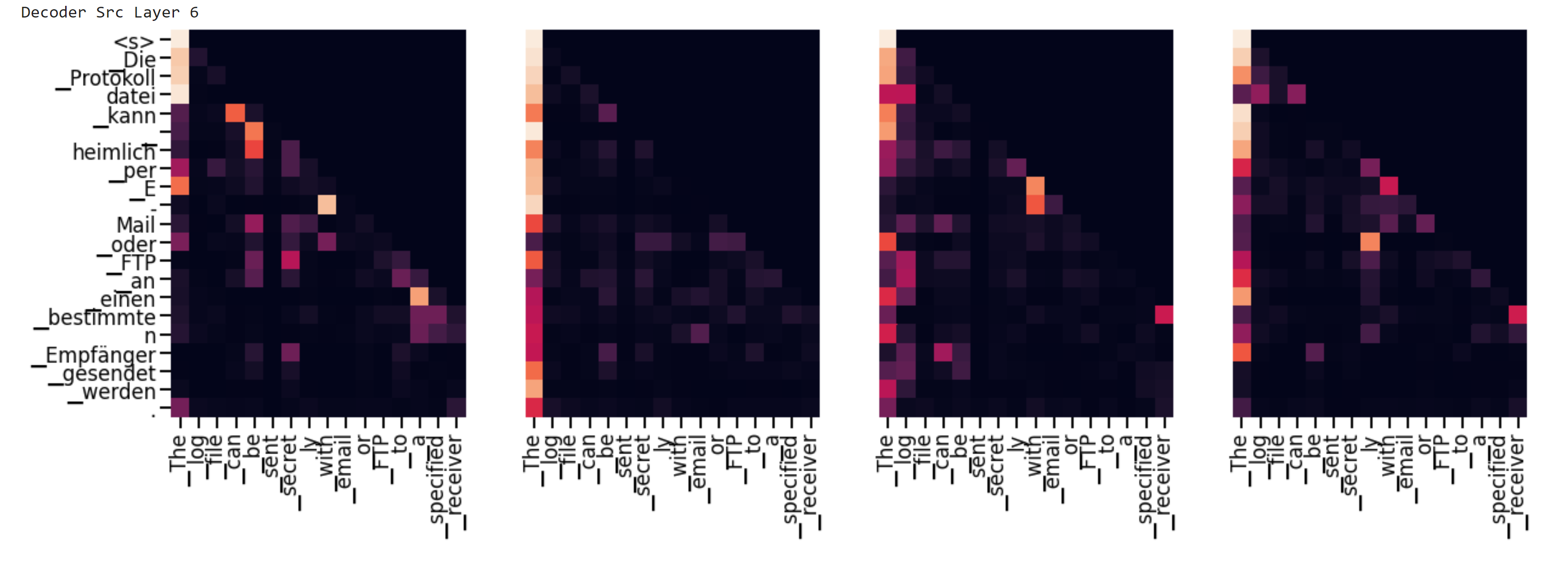
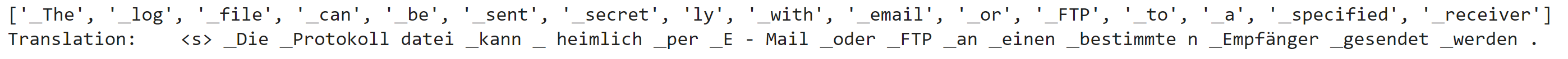
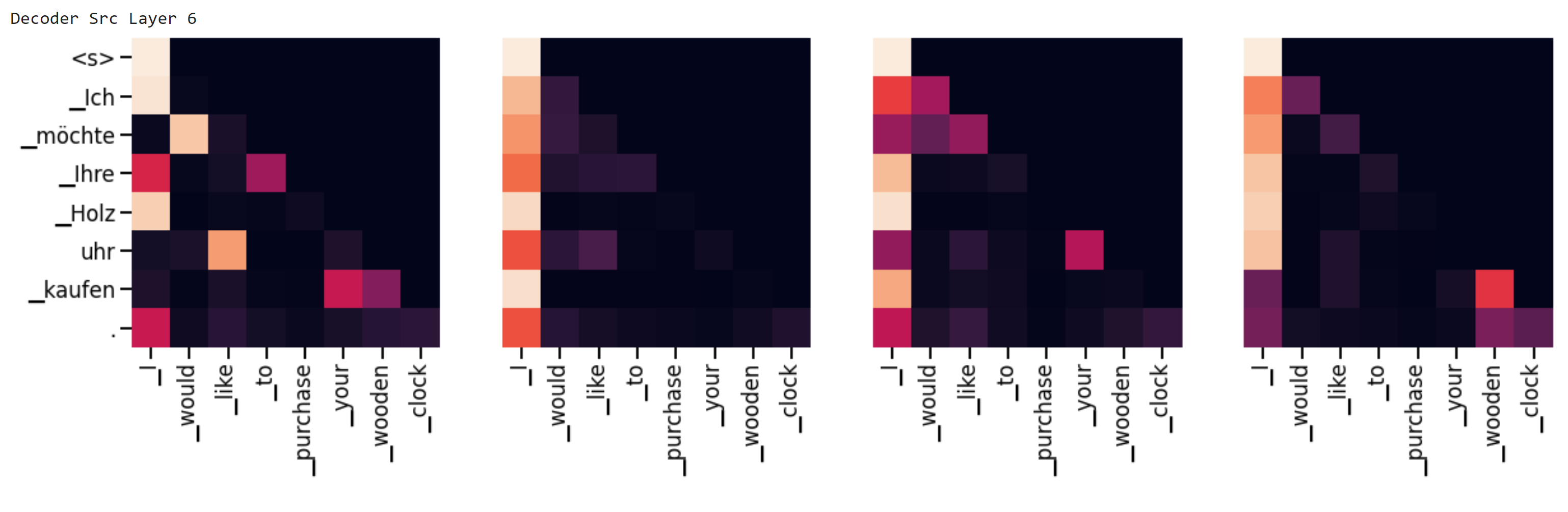
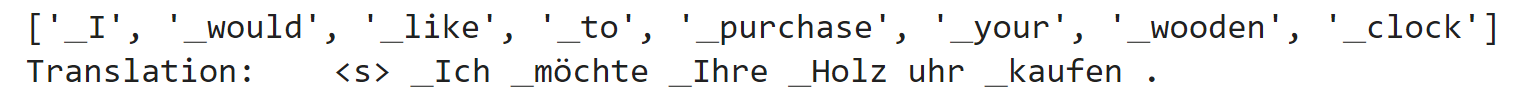
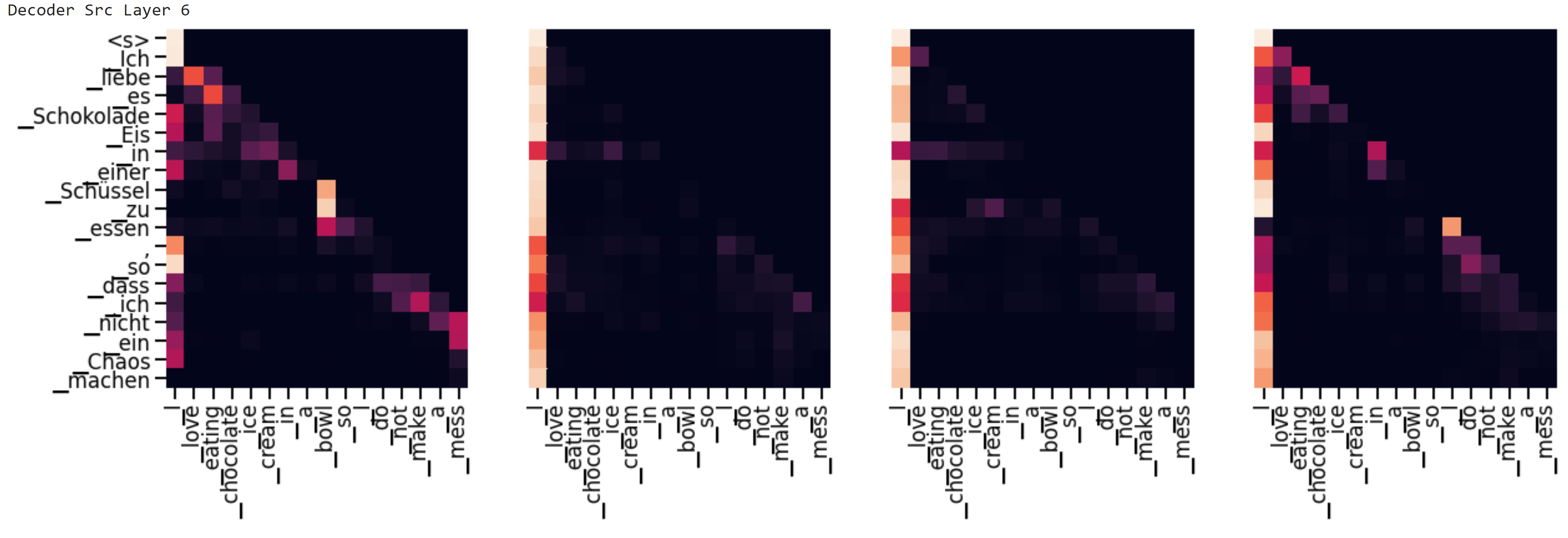
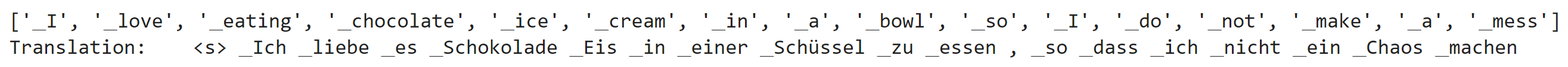
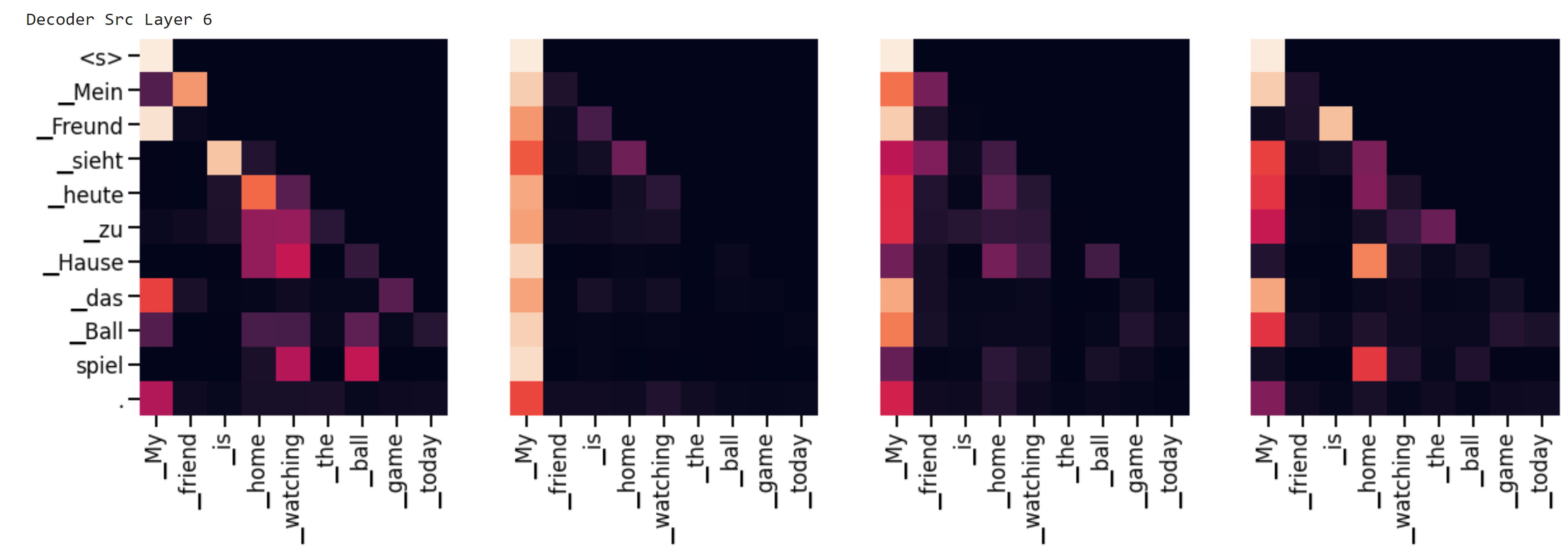
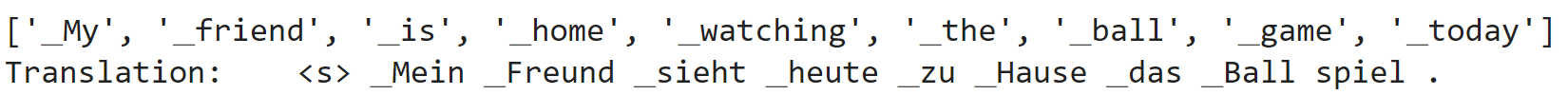
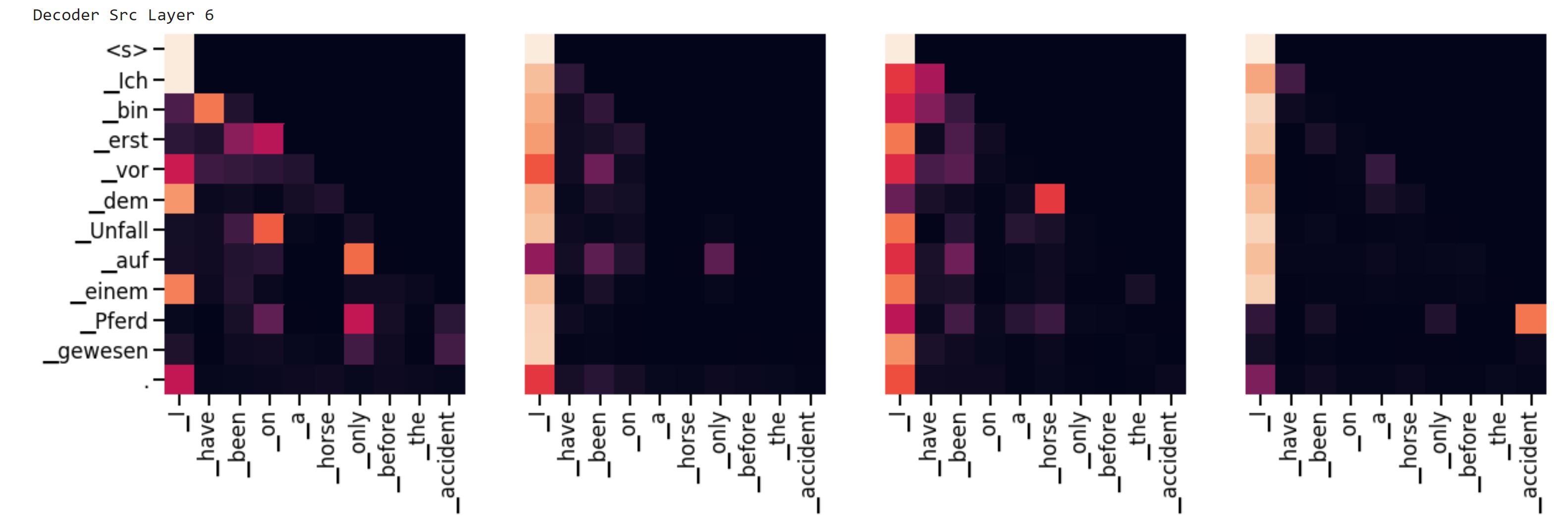
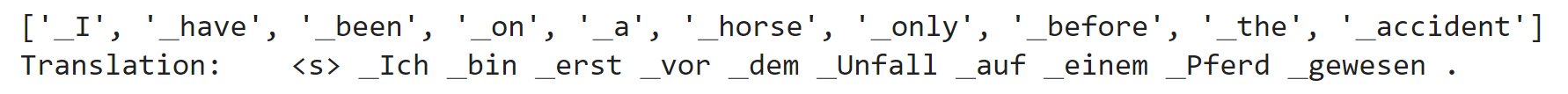
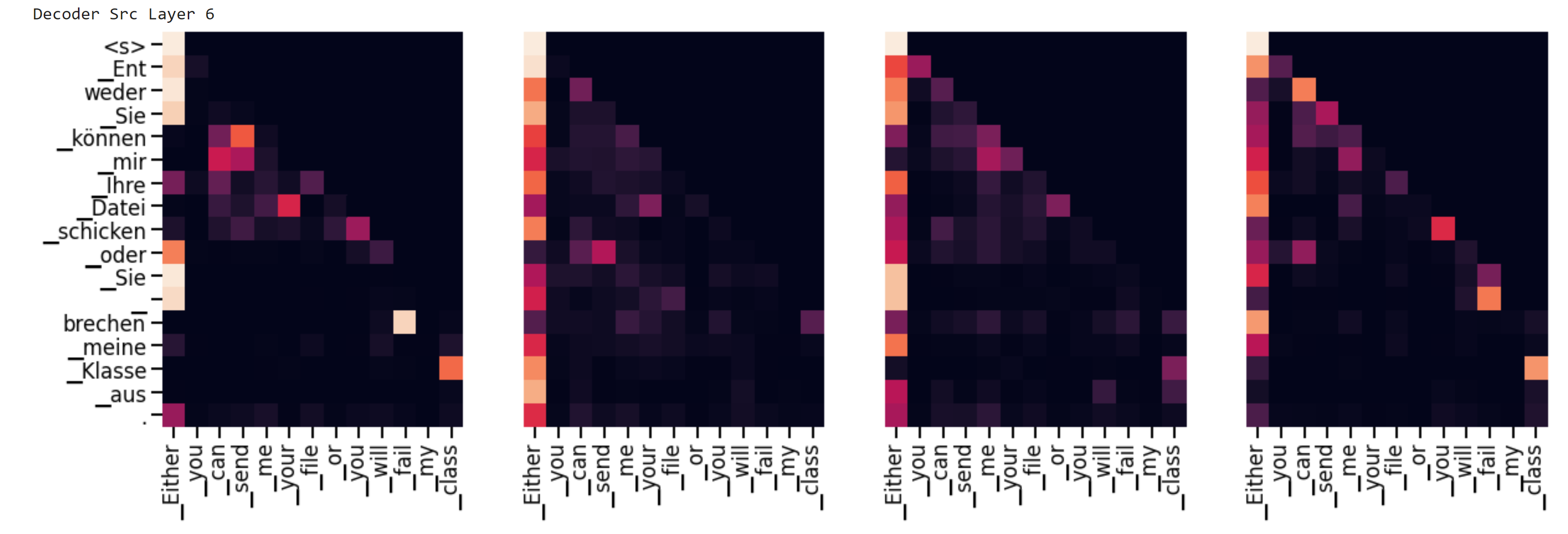
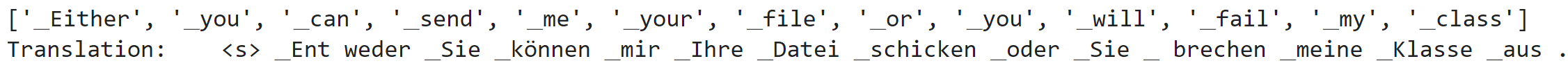
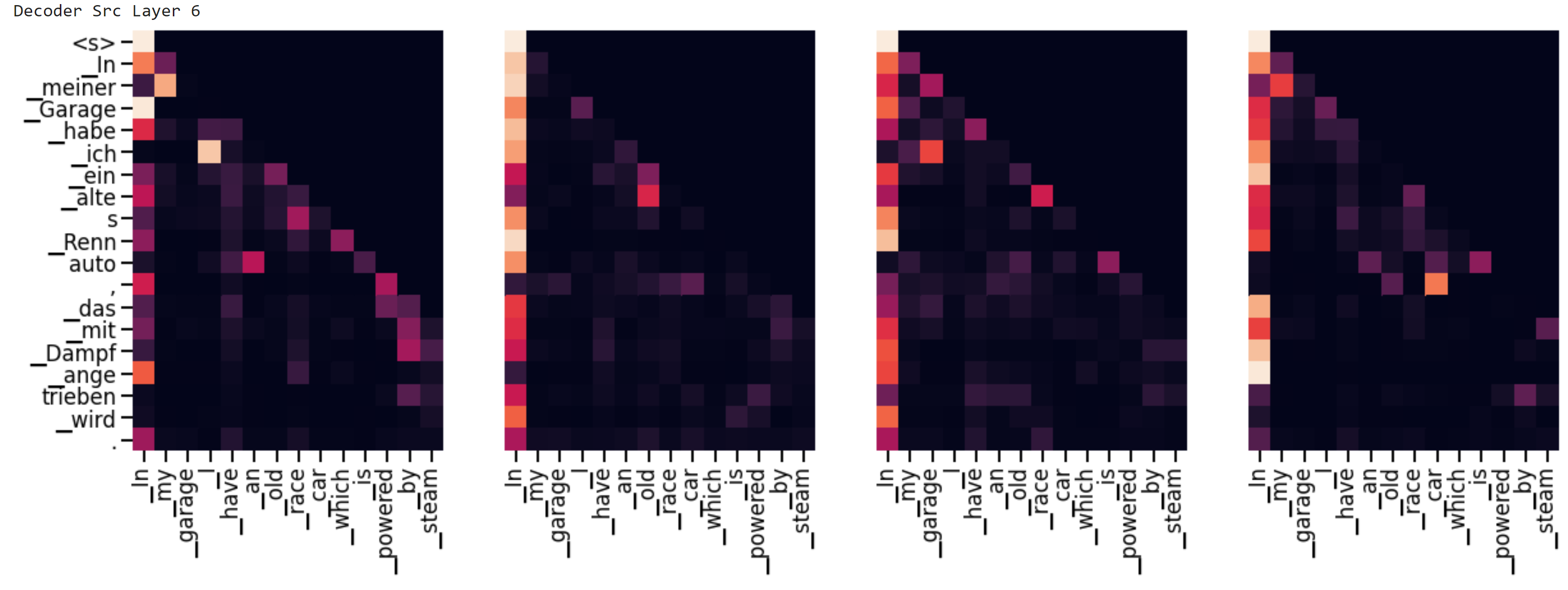
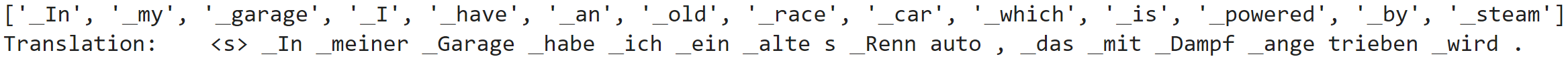
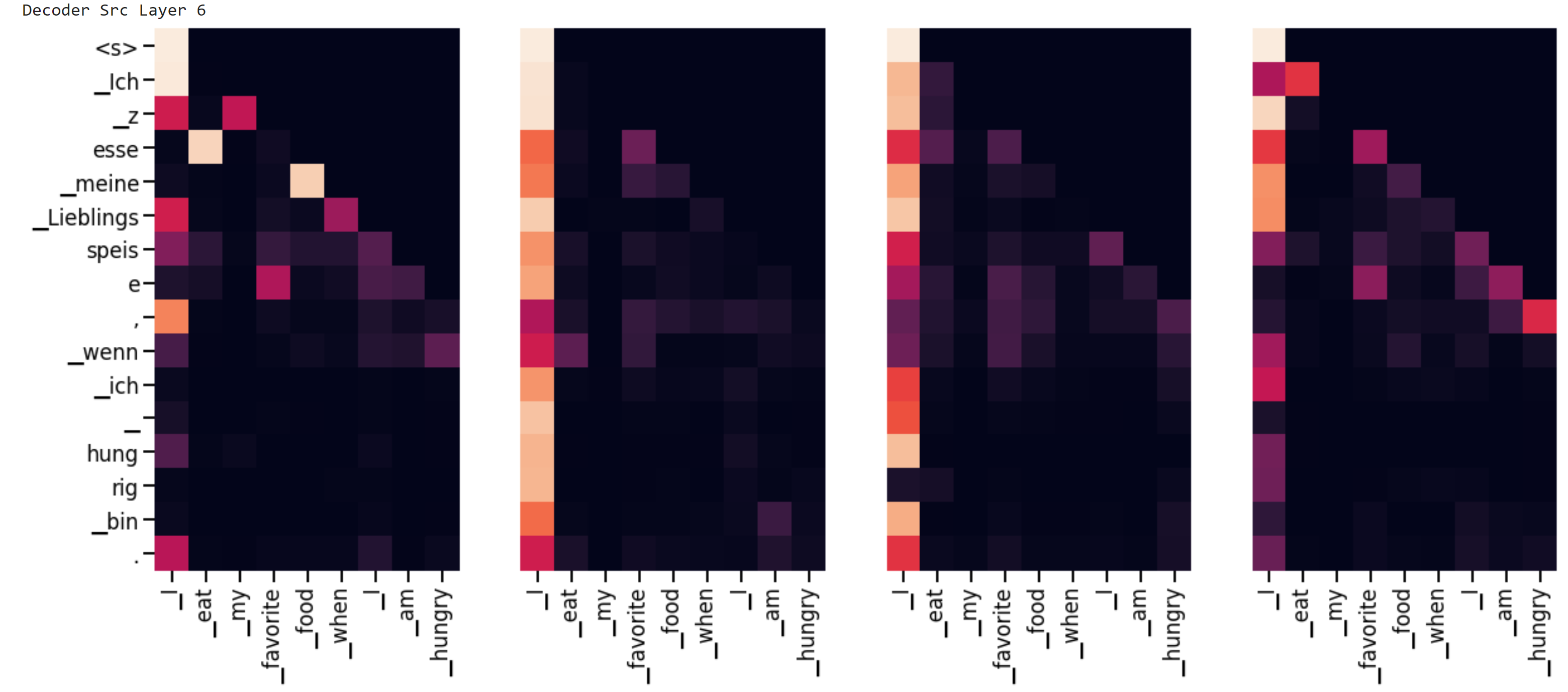
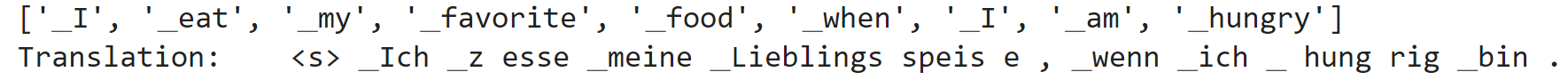
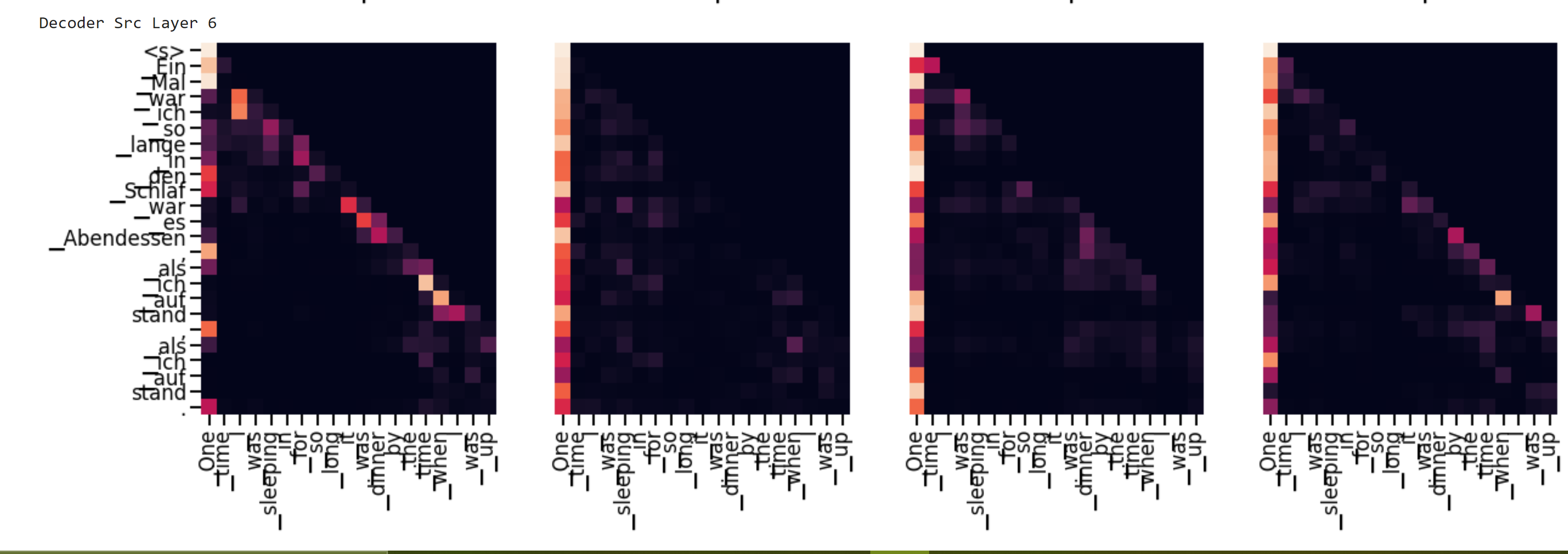
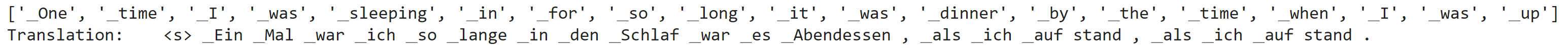
When the attention block is removed (in my code I removed the attention weights from the context vector calculations), the code behaved similarly in terms of run time and gave roughly the same answers, but based on the graphs, it seems as if the ai was less sure about it’s answers. The only changes were from “it is” to “it’s” in for “hace mucho frio aqui” line and from “are you still at home?” to “are you at home?” for the “¿todavia estan en casa?” line What was once a clear strip of color is now a sea of green uncertainty. For example:



**Part – II**

1. Follow the code from (<https://nlp.seas.harvard.edu/2018/04/03/attention.html>) and provide “attention visualizations” for 10 different cases of your choice. (See ‘attention visualization’ section in The Annotated Transformer tutorial for reference). (50 points)

<https://colab.research.google.com/drive/1qg5nIXVgNLzOZrPZOMkanU5yO8S8FIvj?usp=sharing>

* 1. 
  2. 
  3. 
  4. 
  5. 
  6. 
  7. 
  8. 
  9. 
  10. 