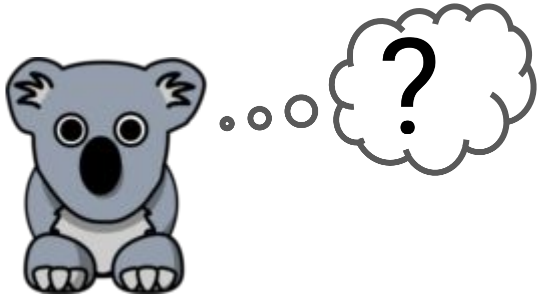
NMT Code Quiz

Could you answer these questions for your NMT toolkit of choice?



1. Training

You have successfully installed Joey NMT and written a configuration file `config.yaml`.

Which command would you use to start training a model with this configuration?

1. Training

You have successfully installed Joey NMT and written a configuration file `config.yaml`.

Which command would you use to start training a model with this configuration?

```
python3 -m joeynmt train config.yaml
```

2. Saving

How do you know your model was saved during training?

☐ Check in the validation report whether there's any line ending with "*".

☐ Check the training log if it says it saved checkpoints.

☐ Check if there are any `*.ckpt` files in the model directory.

☐ The model always gets saved during training.

2. Saving

How do you know your model was saved during training?

____

- ✅ Check in the validation report whether there's any line ending with "*".
- ✅ Check the training log if it says it saved checkpoints.
- ✅ Check if there are any `*.ckpt` files in the model directory.
- ☐ The model always gets saved during training.

3. Testing

When using Joey NMT in test mode, can you specify the checkpoint for testing anywhere outside the configuration file?

☐ Yes

☐ No

3. Testing

When using Joey NMT in test mode, can you specify the checkpoint for testing anywhere outside the configuration file?

[✓ Yes]

☐ No

4. Parameters

How many parameters does the model specified in `configs/default.yaml` have in total? This includes all parameter weights and biases of the model, including e.g. the embeddings. Hint: Joey NMT computes it for you.

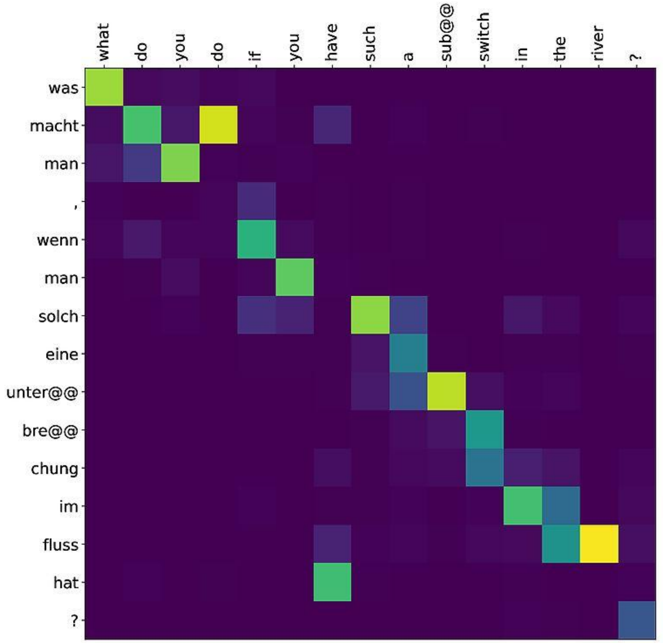
4. Parameters

How many parameters does the model specified in `configs/default.yaml` have in total? This includes all parameter weights and biases of the model, including e.g. the embeddings. Hint: Joey NMT computes it for you.

66,376

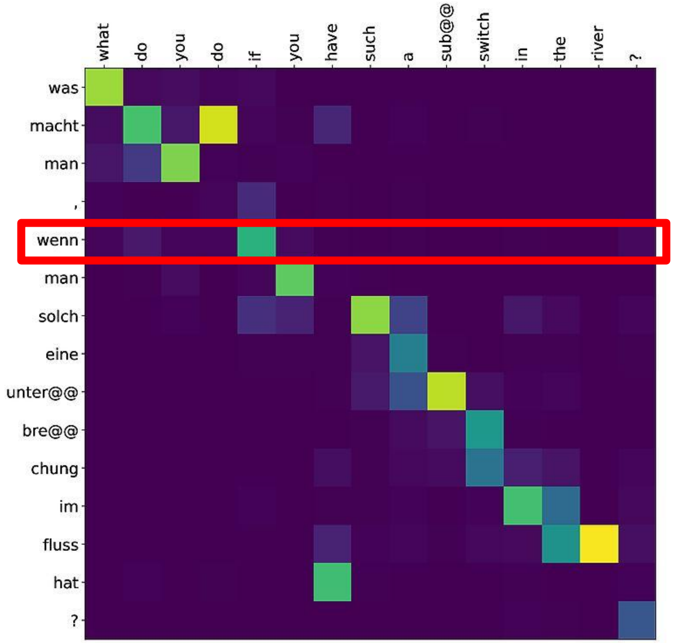
5. Attention

Which source token receives most attention when generating the target word "if"?



5. Attention

Which source token receives most attention when generating the target word "if"?



6. Speed

How do you find out how fast your model trains (including validations)?

6. Speed

How do you find out how fast your model trains (including validations)?

The number of tokens per second is logged and reported in the log file.

7. Pre-processing

What kind of pre-processing does Joey NMT do for you? (if specified)

☐ splitting into sub-word units (BPEs)
☐ data filtering by source/target length ratio
☐ data filtering by source and target length
☐ tokenization
☐ lowercasing

7. Pre-processing

What kind of pre-processing does Joey NMT do for you? (if specified)

- ☐ splitting into sub-word units (BPEs)
- ☐ data filtering by source/target length ratio
- ☑ data filtering by source and target length
- ☐ tokenization
- ☑ lowercasing

8. Post-processing

What kind of post-processing does Joey NMT do for you? (if specified)

- ☐ recasing
- ☐ detokenization
- ☐ subword merging ("un-BPE-ing")
- ☐ delemmatization

8. Post-processing

What kind of post-processing does Joey NMT do for you? (if specified)

- ☐ recasing
- ☐ de-tokenization
- ✓ subword merging ("un-BPE-ing")
- ☐ de-lemmatization

9. Checkpoints

In a debugging scenario, you don’t want to store checkpoints for your current model. There’s a line that you can add to your configuration file to make the model not save any checkpoints during training. What is this line?

9. Checkpoints

In a debugging scenario, you don’t want to store checkpoints for your current model. There’s a line that you can add to your configuration file to make the model not save any checkpoints during training. What is this line?

```
keep_last_ckpts: 0
```

10. Model size

Change the following model configuration to use three encoder layers.

```
encoder:
    rnn_type: "lstm"
    embeddings:
    embedding_dim: 16
    hidden_size: 64
    bidirectional: True
```

Which line would you have to add?

10. Model size

Change the following model configuration to use three encoder layers.

```
encoder:
    rnn_type: "lstm"
    embeddings:
    embedding_dim: 16
    hidden_size: 64
    bidirectional: True
```

Which line would you have to add?

```
    num_layers: 3
```

11. Data path

Which line would you have to add to the data configuration below to use my `home/my_dir/my_data.en` as test input file? Hint: mind the file ending.

```
data:
    src: "en"
    trg: "fr"
    train: "test/data/reverse/train"
    dev: "test/data/reverse/dev”
    level: "word"
    lowercase: False
    max_sent_length: 25
```

11. Data path

Which line would you have to add to the data configuration below to use my `home/my_dir/my_data.en` as test input file? Hint: mind the file ending.

```
data:
    src: "en"
    trg: "fr"
    train: "test/data/reverse/train"
    dev: "test/data/reverse/dev”
    level: "word"
    lowercase: False
    max_sent_length: 25
```

```
test: "my_home/my_dir/my_data"
```

12. Training hyperparameters

Modify the following config such that it uses a constant learning rate of 0.02.

```
training:
    optimizer: "adam"
    learning_rate: 0.001
    clip_grad_norm: 1.0
    batch_size: 10
    scheduling: "plateau"
    patience: 5
    decrease_factor: 0.5
    early_stopping_metric: "eval_metric"
    epochs: 6
    validation_freq: 1000
    logging_freq: 100
    model_dir: "reverse_model"
    max_output_length: 30
```

☐

12. Training hyperparameters

Modify the following config such that it uses a constant learning rate of 0.02.

```
training:
    optimizer: "adam"
    learning_rate: 0.02
    clip_grad_norm: 1.0
    batch_size: 10
    ~~scheduling: "plateau"~~
    patience: 5
    decrease_factor: 0.5
    early_stopping_metric: "eval_metric"
    epochs: 6
    validation_freq: 1000
    logging_freq: 100
    model_dir: "reverse_model"
    max_output_length: 30
```

13. Vocabulary generation

When the vocabulary is extracted from training data, we keep only the `src_voc_limit / trg_voc_limit` most frequent tokens that occur at least `src_min_freq` / `trg_min_freq` times in the training data.

For our example, the vocabulary limit is 15, while the minimum frequency is 3. After counting the tokens in the training data and filtering by minimum frequency, we have the following counts:

i: 22	you: 14	and: 9	,: 9	to: 7	if: 6	joey: 5	’t: 5	anymore: 5	’re: 4
scared: 4	be: 4	angry: 4	but: 3	it: 3	out: 3	don: 3	get: 3	oh: 3	the: 3

Which of those tokens would not end up in the vocabulary, according to Joey NMT ’s vocabulary building?

☐ it ☐ the ☐ oh ☐ get

☐ don ☐ but ☐ out

13. Vocabulary generation

When the vocabulary is extracted from training data, we keep only the `src_voc_limit` / `trg_voc_limit` most frequent tokens that occur at least `src_min_freq` / `trg_min_freq` times in the training data.

For our example, the vocabulary limit is 15, while the minimum frequency is 3. After counting the tokens in the training data and filtering by minimum frequency, we have the following counts:

i: 22	you: 14	and: 9	,: 9	to: 7	if: 6	joey: 5	't: 5	anymore: 5	're: 4
scared: 4	be: 4	angry: 4	but: 3	it: 3	out: 3	don: 3	get: 3	oh: 3	the: 3

Which of those tokens would not end up in the vocabulary, according to Joey NMT 's vocabulary building?

[✓] it [✓] the [✓] oh [✓] get

☐ don ☐ but [✓] out

14. Special tokens

Which is the default token used for marking the end-of-sequence position in Joey NMT ?

e.g. <end> or [EOS]?

14. Special tokens

Which is the default token used for marking the end-of-sequence position in Joey NMT ?

e.g. <end> or [EOS]?

</s>

15. Data iterators

Training and validation data are treated differently in Joey NMT - but in which ways?

For example, if you choose “sorting”, it means that validation and training data are handled differently with respect to sorting - one gets sorted and the other doesn’t.

- ☐ Shuffling
- ☐ Filtering
- ☐ Tokenization
- ☐ Embedding
- ☐ Sorting

15. Data iterators

Training and validation data are treated differently in Joey NMT - but in which ways?

For example, if you choose “sorting”, it means that validation and training data are handled differently with respect to sorting - one gets sorted and the other doesn’t.

- ✅ Shuffling
- ✅ Filtering
- ☐ Tokenization
- ☐ Embedding
- ✅ Sorting

16. Training loop

Where is the training for-loop over epochs defined? Paste the line in the textbox below. (Not the line number)

16. Training loop

Where is the training for-loop over epochs defined? Paste the line in the textbox below. (Not the line number)

```
for epoch_no in range(self.epochs):       (in training.py)
```

17. End of training

When does training end? (Assuming there are no technical problems like memory errors etc.)

We refer to settings in the configuration file, e.g. learning rate.

- ☐ When the minimum learning rate (`learning_rate_min`) has been reached.
- ☐ When the maximum validation scores has been reached.
- ☐ Just after `keep_last_ckpts` checkpoints have been saved.
- ☐ When Joey NMT gets tired.
- ☐ When all epochs (epochs) have been completed.
- ☐ When you interrupt the training process with Ctrl+C.

17. End of training

When does training end? (Assuming there are no technical problems like memory errors etc.)

We refer to settings in the configuration file, e.g. learning rate.

- [x] When the minimum learning rate (`learning_rate_min`) has been reached.
- [ ] When the maximum validation scores has been reached.
- [ ] Just after `keep_last_ckpts` checkpoints have been saved.
- [ ] When Joey NMT gets tired.
- [x] When all epochs (epochs) have been completed.
- [x] When you interrupt the training process with Ctrl+C.

18. Model

What does `model.forward()` return?

18. Model

What does `model.forward()` return?

- decoder outputs
- decoder last hidden state
- attention probabilities
- attentional vectors

19. Initialization

How are forget gates of LSTMs initialized by default?

- ☐ All ones
- ☐ Random normal initialization
- ☐ Random uniform initialization
- ☐ All zeros
- ☐ Xavier initialization

19. Initialization

How are forget gates of LSTMs initialized by default?

☑ All ones
☐ Random normal initialization
☐ Random uniform initialization
☐ All zeros
☐ Xavier initialization

20. Embeddings

In the configuration we can ”freeze” the embeddings, so that they are not (further) trained:

```
embeddings:
    embedding_dim: 16
    freeze: True
```

Where does the freezing happen in JoeyNMT’s code? Please give the freezing function’s name.

20. Embeddings

In the configuration we can ”freeze” the embeddings, so that they are not (further) trained:

```
embeddings:
    embedding_dim: 16
    freeze: True
```

Where does the freezing happen in JoeyNMT’s code? Please give the freezing function’s name.

```
freeze_params
```

21. Bidirectional

How are forward and backward states combined for a bidirectional recurrent encoder? Choose the correct tensor operation:

- ☐ `torch.add`
- ☐ `torch.cat`
- ☐ `torch.addbmm`
- ☐ `torch.sum`
- ☐ `torch.mul`
- ☐ `torch.pow`

21. Bidirectional

How are forward and backward states combined for a bidirectional recurrent encoder? Choose the correct tensor operation:

- ☐ `torch.add`
- [✓ `torch.cat`]
- ☐ `torch.addbmm`
- ☐ `torch.sum`
- ☐ `torch.mul`
- ☐ `torch.pow`

22. Bridge

What’s the name of the function that connects encoder and decoder by computing the initial decoder state given the last encoder state?

- ☐ `bridge layer`
- ☐ `BahdanauAttention`
- ☐ `init decoder hidden`
- ☐ `bridge layer`
- ☐ `_bridge`
- ☐ `init_decoder_hidden`
- ☐ `init_hidden`
- ☐ `_init_hidden`
- ☐ `bridge`
- ☐ `LuongAttention`
- ☐ `attend`
- ☐ `forward step`

22. Bridge

What’s the name of the function that connects encoder and decoder by computing the initial decoder state given the last encoder state?

- ☐ `bridge layer`
- ☐ `BahdanauAttention`
- ☐ `init decoder hidden`
- ☐ `bridge layer`
- ☐ `_bridge`
- ☐ `init_decoder_hidden`
- ☐ `init_hidden`
- [✓ `_init_hidden`]
- ☐ `bridge`
- ☐ `LuongAttention`
- ☐ `attend`
- ☐ `forward step`

23. Loss computation

Find the place where the batch loss is computed (comparing model outputs with targets), and paste the statement below. e.g.

```
train_batch_loss = my_loss_function(outputs, targets)
```

23. Loss computation

Find the place where the batch loss is computed (comparing model outputs with targets), and paste the statement below. e.g.
`train_batch_loss = my_loss_function(outputs, targets)`

```
batch_loss = loss_function(
    input=log_probs.contiguous().view(
        -1, log_probs.size(-1)),
    target=batch.trg.contiguous().view(-1))
```

24. Batch

During training, the Batch object in JoeyNMT holds the reference sequence in trg for computing the loss and in trg input for feeding it into the decoder.

What’s the difference between those two tensors? (`batch.trg` vs. `batch.trg_input`)

☐ <s> is prepended to the first, otherwise no difference
☐ </s> is appended to the first and <s> is prepended to the second
☐ <s> is prepended to the second, otherwise no difference
☐ <s> is appended to the first and </s> is prepended to the second
☐ </s> is appended to the first, otherwise no difference

24. Batch

During training, the Batch object in JoeyNMT holds the reference sequence in trg for computing the loss and in trg input for feeding it into the decoder.

What’s the difference between those two tensors? (`batch.trg` vs. `batch.trg_input`)

____

- ☐ <s> is prepended to the first, otherwise no difference
- [✓ </s> is appended to the first and <s> is prepended to the second]
- ☐ <s> is prepended to the second, otherwise no difference
- ☐ <s> is appended to the first and </s> is prepended to the second
- ☐ </s> is appended to the first, otherwise no difference

25. Inference algorithm

Where in the code is the decision made whether to decode greedily or with beam search? Paste the line below.

Hint: it’s an if-statement.

```
if beam size == 0:
```

25. Inference algorithm

Where in the code is the decision made whether to decode greedily or with beam search? Paste the line below.

Hint: it’s an if-statement.

```
if beam size == 0:
```

26. Validation score computation

Find the place where the validation score (here `BLEU`, `eval metric: bleu`) is computed and paste the statement below.

26. Validation score computation

Find the place where the validation score (here BLEU, `eval_metric: bleu`) is computed and paste the statement below.

```
current_valid_score = bleu(valid_hypotheses,
                           valid_references)
```

27. BLEU computation

Which library is used for BLEU score computation?

27. BLEU computation

Which library is used for BLEU score computation?

```
sacreBLEU
```

28. Optimizers

Let’s say you have invented a new optimizer and implemented it in PyTorch as `torch.optim.MagicOptimizer`. Now you want to use it in JoeyNMT by setting optimizer: magic in the configuration file. Which of JoeyNMT’s Python files would you have to add the following lines to?

```
elif optimizer_name == "magic": # new awesome optimizer
    optimizer=torch.optim.MagicOptimizer(
        parameters, weight_decay=weight_decay,
         lr=learning_rate)
```

28. Optimizers

Let’s say you have invented a new optimizer and implemented it in PyTorch as `torch.optim.MagicOptimizer`. Now you want to use it in JoeyNMT by setting optimizer: magic in the configuration file. Which of JoeyNMT’s Python files would you have to add the following lines to?

```
elif optimizer_name == "magic": # new awesome optimizer
    optimizer=torch.optim.MagicOptimizer(
        parameters, weight_decay=weight_decay,
         lr=learning_rate)
```

____

`builders.py`

29. Attention

For Bahdanau attention, find the line where the attention scores for a decoder hidden state are computed (before masking).

29. Attention

For Bahdanau attention, find the line where the attention scores for a decoder hidden state are computed (before masking).

```
scores = self.energy_layer( torch.tanh(self.proj_query +
                                       self.proj_keys))
```

30. Plotting

You want to use a different colormap for attention visualization, namely the one called “binary”. Give the line of JoeyNMT’s code that is responsible for plotting the attention, modified to use the new colormap.

30. Plotting

You want to use a different colormap for attention visualization, namely the one called “binary”. Give the line of JoeyNMT’s code that is responsible for plotting the attention, modified to use the new colormap.

```
plt.imshow(scores, cmap=’binary’, aspect=’equal’,
                origin=’upper’, vmin=0., vmax=1.)
```