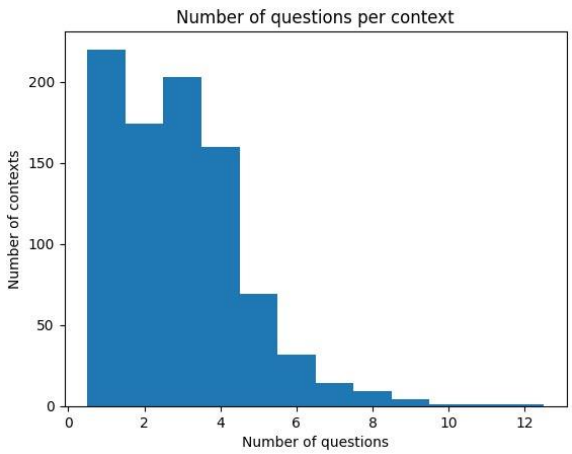
Assignment 5

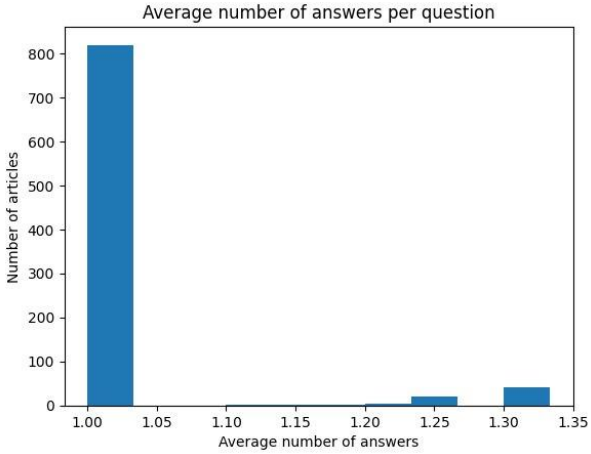
Task 2: Generative open-book Question Answering

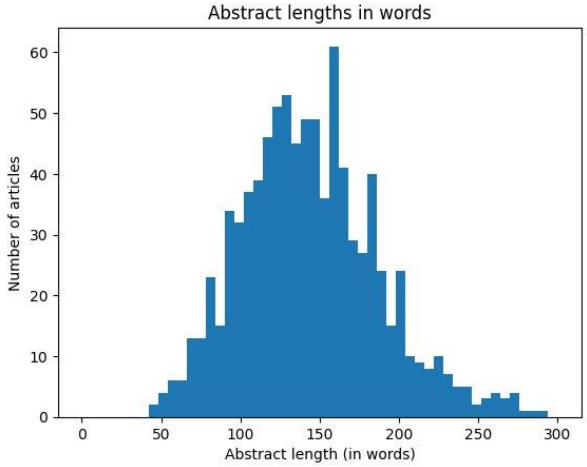
Group Members: Jerome Stephan, Magnus Menger, Paul Chevelev

Research Journey: Our research journey was quite nice. We really liked to get into the task, investigating the data and then after training the model, seeing live how good it trains. Our prediction where actually quite alright, which surprised us a lot. We really felt like researchers.

Question 2: Here is our EDA for the train split:

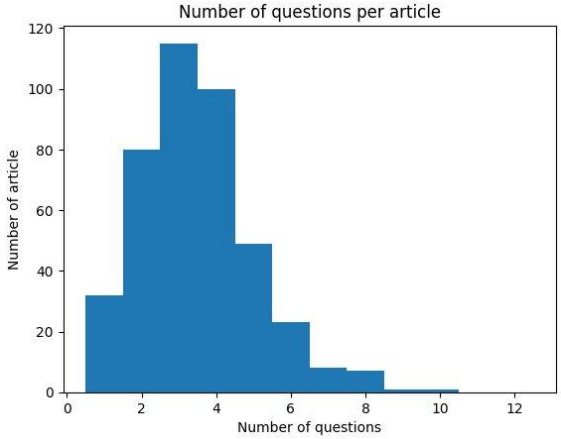


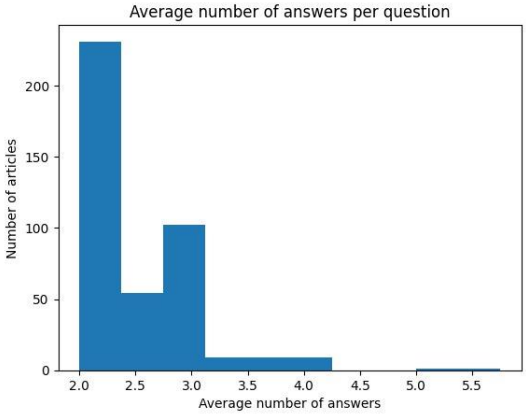


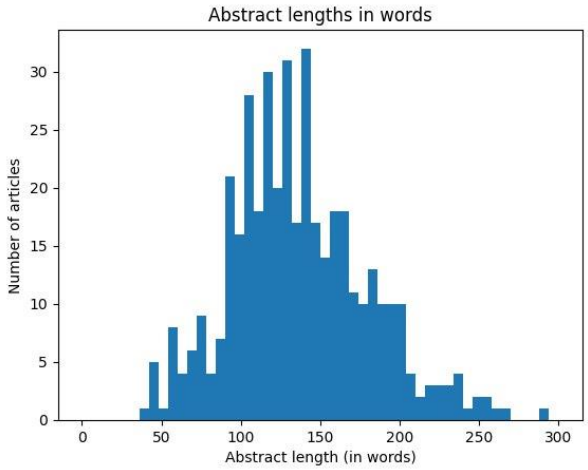


Train:
Number of answers: 2675
Number of free form answers: 622
Number of extractive answers: 1363
Number of unanswerable questions: 281
Average number of answers per question: 1.031623602005399

Question 3: Here is our EDA for the test split:







Test:
Number of answers: 3554
Number of free form answers: 878
Number of extractive answers: 1817
Number of unanswerable questions: 366
Average number of answers per question: 2.44934527911785

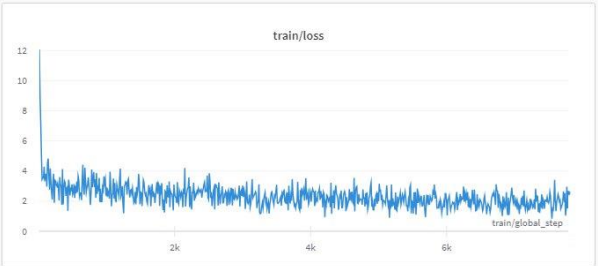
Question 6: Yes, it seems like a lot of the sequences will be truncated since half of the abstracts are already at 128+ length and then we still have the questions.

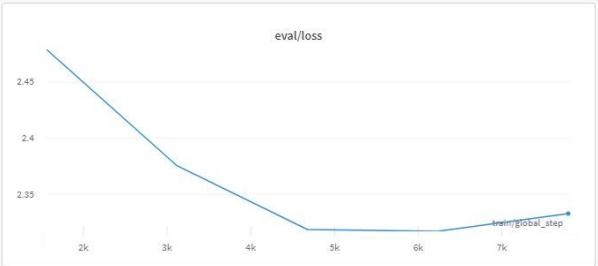
Question 10: Our setup for this task is the following:

```
training_args = Seq2SeqTrainingArguments(
    per_device_train_batch_size=1,
    per_device_eval_batch_size=1,
    learning_rate=5e-4,
    weight_decay=0.001,
    # prediction_loss_only=True,
    lr_scheduler_type="cosine",
    output_dir="./results_scratch",
    evaluation_strategy="epoch",
    save_strategy="epoch",
    load_best_model_at_end=True,
    seed=SEED,
    num_train_epochs=5,
    logging_dir='./logs',
    report_to="wandb",
    logging_steps=10,
)
```

With model = T5ForConditionalGeneration.from_pretrained("google/t5-efficient-mini").

Our losses look like this on wandb:





Question 11: We looked at one of the questions, and we got this:

```
Question: What languages do they evaluate their methods on?
Context: Rumour detection is hard because the most accurate systems operate retrospectively,
only recognising rumours once they have collected repeated signals. By then the rumours might
have already spread and caused harm. We introduce a new category of features based on
novelty, tailored to detect rumours early on. To compensate for the absence of repeated
signals, we make use of news wire as an additional data source. Unconfirmed (novel)
information with respect to the news articles is considered as an indication of rumours.
Additionally we introduce pseudo feedback, which assumes that documents that are similar to
previous rumours, are more likely to also be a rumour. Comparison with other real-time
approaches shows that novelty based features in conjunction with pseudo feedback perform
significantly better, when detecting rumours instantly after their publication.
Generated Answer: English
Actual Answer: Chinese
```

The question expected a specific language as answer, so we were very surprised when we saw that we actually got a language as answer. So, this output makes a lot of sense, however the answer was still wrong since we expected Chinese and got English. The model seems to have some understanding of the questions however it lacks finetuning. More training or maybe more context could help with this.