Report #1

Work under data & model types and architectures

Data:

As a source of toxic sentences it was decided to use "default" dataset, but with small preparation: choosing the references with a toxicity score higher than the one of its translation. In total about 319K reference-translation pairs were chosen to be used as training data and hence training dataset was constructed.

Structure:

It was a main part of the "preparation" phase for this assignment.

After some time spent in the Internet to find more info about techniques and models that can be utilized to show any satisfactory results it was found out that the solution can have a form of:

1) Writing from scratch model architectures and such steps as: data preprocessing, training and evaluation.
2) Use already written model architectures(i.e. Torchvision.models) and still code from scratch the steps from previous case
3) Use already written and pretrained models and use built-in methods, functions, and structures to preprocess data, fine-tune and evaluate models.

After not so long analyzing the amounts of work in all cases it was decided to:

1) Use plan from the 3rd case
2) Read more about the topic(read some research papers, references, guides) and watch additional video resources.

Models:

After choosing the overall structure it was a time to choose models to be trained and since the task is Seq2Seq text generation models of “transformer” architecture were chosen: T5, Bart for conditional generation, Bert for conditional generation, ParaGeDi. Last two were classified as “read about” and hence were not fine-tuned in the final solution.

All the models can be found in Hugging face’s models section and hence the hugging face library called “transformers” was used to load models.

Since “transformers” library was used - all data preprocessing, fine-tuning & evaluation of model steps were performed with the use of appropriate modules such as: AutoTokenizer, DataCollatorForSeq2SeqLM, Seq2SeqTrainingArguments, Seq2SeqTrainer and some modules for Bart model and tokenizer configuration.

Results:

During the “preparations” phase it was decided:

1) Which structure the solution will have
2) Which resources to use as references, which ones to consider as “good to know”, and which ones to just skip
3) Which libraries to use
4) Which steps are necessary to take to create the solution