Machine Learning applied in the context of Question Generation and Question Answering

Author: Chloe Thompson | cthompson68@qub.ac.uk
Supervisors: Dr Barry Devereux, Dr Joana Cavadas

Introduction

Question Answering (QA) and Question Generation (QG) are prominent research problems and a focus area in the space of Natural Language Processing (NLP). QG is the process of generating questions based on a given context paragraph, this context and generated question can then be passed for answering by the QA model.

Leveraging NLP techniques, it is aimed to develop a QG and QA system that can consume (un)structured text and, by capturing relevant information, return a set of comprehensive questions and answers.

This project aims to understand and demonstrate the use of NLP models, specifically BERT (Bi-Directional Encoder Representation from Transformers) [1] and GPT-2 (Generative Pre-Trained Transformer 2) [2] for QA and QG tasks. We also analyse how the training data used and hypertuning of models affects the generated questions and answers, from the QG and QA models respectively. The models are trained on SQuAD (Stanford Question and Answering Dataset) [3] and QuAC (Question Answering in Context Dataset) [4] and are evaluated using Bleu (Bi-lingual Evaluation Understudy Score) [5].

From this we develop a QA and QG pipeline that enables the evaluation of generated questions via an answering model and metrics such as F1, Exact Match and Bleu.

Literature Review

Language Modelling (LM) [6] is the use of mathematical processes to determine the probability of a given sequence of words, occurring in a sentence. This is the basis for NLP models and language analysis. With NLP becoming a prominent problem area, LM techniques were employed to develop and create models, such as BERT and GPT-2. These models use a form of Masked Language Modelling (MLM), either through token masking or weighted attention.

BERT [1] was an innovative development from Google, that used MLM to tackle the issue of past and future contexts of a target word in a language model. Google released two versions of BERT, base with 12 Transformer layers and large with 24 Transformer layers. BERT also requires specific tokens to denote the beginning, separation and end of sentences being passed as inputs and mask token is used to hide mask words for prediction.

GPT-2 [2] was created by OpenAI to translate, generate and summarise text, and comes in 4 size variations starting at 117 Million parameters, going up to 1542 Million parameters in size. Rather than using tokens to perform MLM, GPT-2 uses self-attention weights [7] to prioritise tokens to the right of the position being predicted.

This project utilises two datasets, SQuAD 2.0 and QuAC. These datasets contain context paragraphs, questions for the context, answers and the spans in the paragraph where the answers occur. The models are trained on both datasets separately to evaluate the variations in model outcomes, this being the answers or generated questions for QA and QG respectively.

Results

Both BERT and GPT-2 models were hyper-tuned using a Grid Search approach, varying parameters such as batch size, epochs and learning rate. The Table 1 below captures the results for the SQuAD BERT model, to give a representation of the hypertuning performed.

Parameters {Learning Rate, Epochs, Batch Size}	$\{5e^{-05}, 2, 4\}$	$\{3e^{-05}, 3, 4\}$	$\{3e^{-05}, 2, 2\}$	$\{2e^{-05}, 3, 2\}$	$\{3e^{-05}, 4, 4\}$	$\{5e^{-05}, 3, 2\}$	$\{5e^{-05}, 4, 2\}$	$\{2e^{-05}, 3, 4\}$	$\{3e^{-05}, 4, 2\}$
Exact Match	65.029	65.366	68.264	67.531	65.316	67.228	65.880	63.572	67.076
F1 Score	68.829	69.266	71.601	71.371	69.145	71.040	69.717	67.591	70.931

Parameters {Learning Rate, Epochs, Batch Size}	$\{3e^{-05}, 2, 4\}$	$\{2e^{-05}, 4, 2\}$	$\{2e^{-05}, 4, 4\}$	$\{2e^{-05}, 2, 2\}$	$\{2e^{-05}, 2, 4\}$	$\{5e^{-05}, 3, 4\}$	$\mathbf{\{3e^{-05}, 3, 2\}}$	$\{5e^{-05}, 2, 2\}$	$\{5e^{-05}, 4, 4\}$
Exact Match	62.882	66.520	64.313	67.177	60.692	67.775	**68.171**	68.415	66.428
F1 Score	66.799	70.479	68.285	70.779	64.845	71.320	**71.844**	71.798	70.405

Table 1: Hypertuning results BERT on SQuAD 2.0 for QA

It can be seen that parameters of $\{3\mathrm{e}^{-05}, \ 3, \ 2\}$ for Learning Rate, Epochs and Batch Size respectively, got the highest result for QA, both in F1 and Exact match. F1 is the balance between recalling from training examples and precision, while Exact Match is the measure of how close it was to producing the exact replica of the ground truth answer. This result is slightly below that of 76.07 F1 and 72.80 Exact Match, in the comparative paper for BERT QA with SQuAD 2.0 [8].

QG models are evaluated by generating multiple questions for each paragraph of context and running these questions through the QA model. In doing such the F1, Exact Match and Bleu scores can be evaluated, where the results are based on the outcomes of the answers produced.

Future Work

In conclusion, while the results are lower than that of comparative work, the project demonstrated the capability to create and develop a QA and QG pipeline. This pipeline had metric outcomes acceptable for the task and that prevented overfitting. Based on the current project outcomes, there are a number of areas that could be explored for further development:

1. The impact training with dataset combinations has both for QA and QG, such that models are trained on both datasets in turn.
2. Understanding the affect varying hypertuning techniques will have on model outcomes and evaluation, such as Bayesian Optimisation, Loss Functions and early stop mechanisms.
3. Analysing the question types via clustering to identify subsets of questions that could be improved upon, which in turn improves model performance
4. Deployment of a QA and QG pipeline with frontend interface.

Models

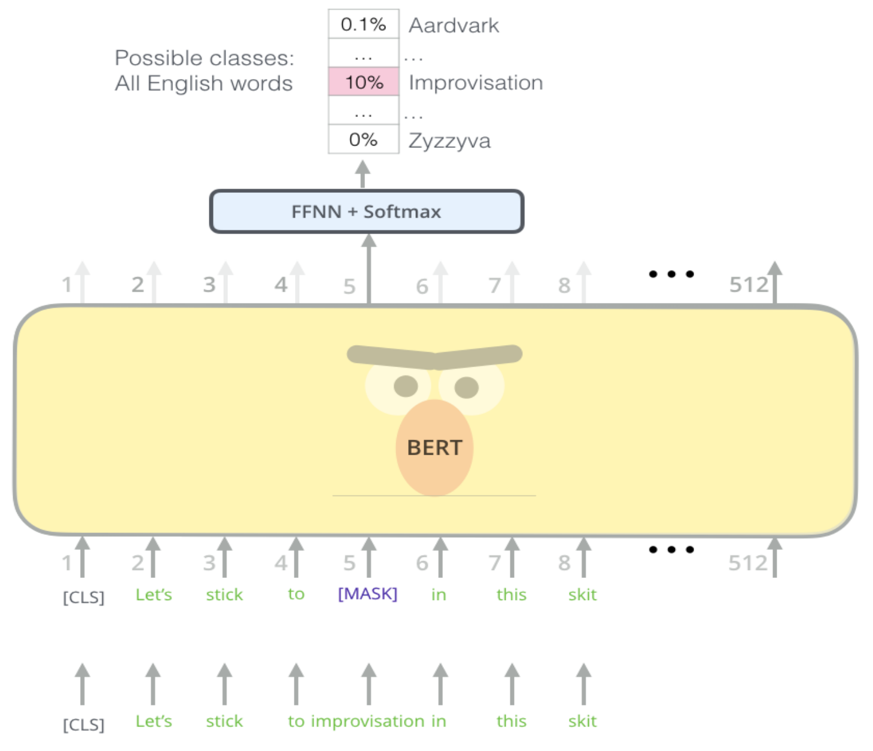


Figure 1: BERT Model

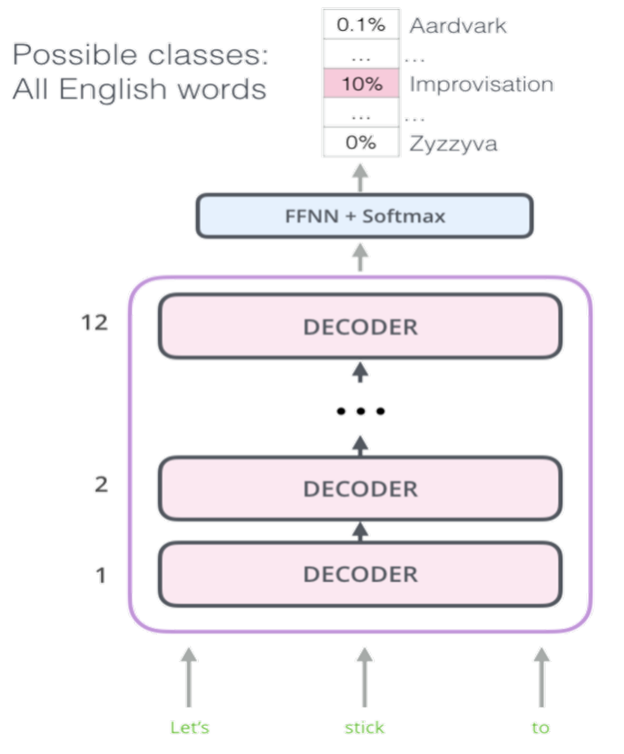


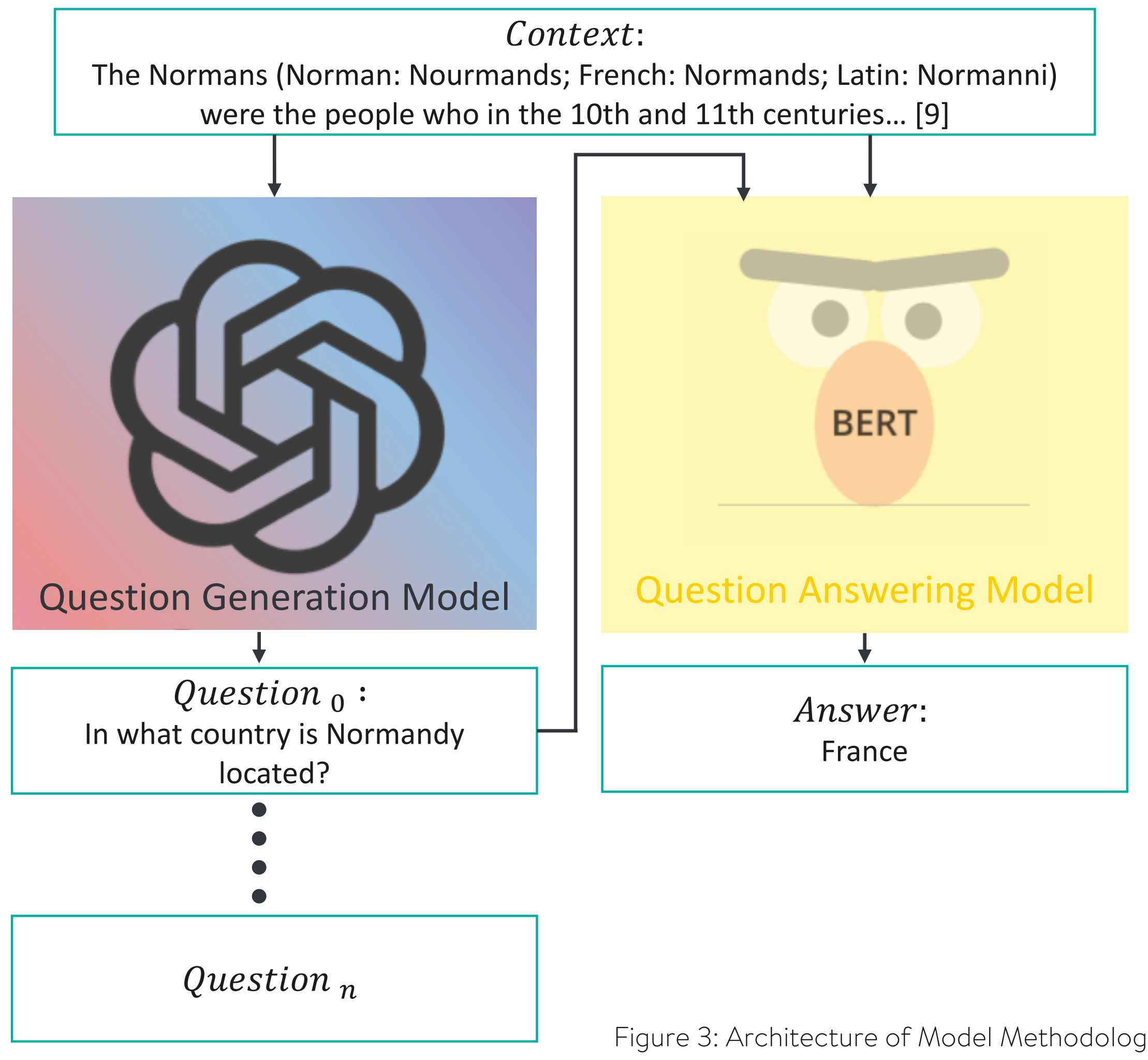
Figure 2: GPT-2 Model

A Transformer takes input sentences, which have been formatted into embeddings. These input embedding are created by taking the input sentence, tokenizing it with a word piece tokenizer, adding segment embeddings which distinguish the sentence or between the two sentences and finally adding position embeddings, which encode word order. Both BERT and GPT-2 have specific tokens that are also included in the embeddings.

A BERT Transformer architecture contains a number encoder layers, this depends on the model size, and these contain self-attention heads and a Feed-Forward Neural Network (FFNN). Combined these new representations of the embeddings are produced repeatedly until a vector output of a predicted word(s) or binary decision is produced, depending if the model is performing a MLM task or Next Sentence Prediction (NSP).

The GPT-2 Transformer architecture is similar to BERT, however, it is a series of decoder layers containing masked self-attention and a FFNN. The Transformer block produces an output matrix of token predictions with the highest probability word being chosen. This process can be repeated a number of times to produce words based on those previous in the sentence, which is how GPT-2 performs strongly for text generation.

Methodology



Context:
The Normans (Norman: Nourmands; French: Normands; Latin: Normanni) were the people who in the 10th and 11th centuries… [9]

$Question_0$:
In what country is Normandy located?

$Question_n$

Answer:
France

Figure 3: Architecture of Model Methodology

Acknowledgements

I would like to firstly acknowledge the support and guidance of my QUB supervisor Dr Barry Devereux, secondly my Aflac Northern Ireland supervisor Dr Joana Cavadas, and lastly Aflac Northern Ireland for access to compute resources.

References

Graphics: BERT and GPT-2 from Jay Alammar

[1] Jacob Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

[2] Alec Radford et al. Language Models are Unsupervised Multitask Learners

[3] Pranav Rajpurkar et al. Know What You Don't Know: Unanswerable Questions for SQuAD

[4] Eunsol Choi et al. QuAC : Question Answering in Context

[5] Kishore Papineni et al. BLEU: a Method for Automatic Evaluation of Machine Translation

[6] Ben Lutkevich (Search Enterprise AI). Language Modelling

[7] Raimi Karim Illustrated Self-Attention

[8] Yuwen Zhang et al. BERT for Question Answering on SQuAD 2.0

[9] Normals Wikipedia