INTRODUCTION

- We worked on the task of extractive question answering using SQuAD 2.0 which adds unanswerable questions to about half of the dataset.
- QANet is an efficient architecture built for SQuAD 1.1 that uses an Encoder Block module to replace RNNs.
- The Encoder Block relies on a Transformer-like component which uses Positional Encodings, Self Attention, and Position-wise Feedforward Networks.
- We experimented with Data Augmentation through back-translation, local self-attention, and augmented loss function, and adding tag features.

PROBLEM

- Given a context paragraph C, and a question Q, provide the token span C[[i,j]] which answers the question or state it’s unanswerable.
- There are two metrics reported for this task:
  - Exact Match (EM) measures the percentage of questions that received the exact correct answer span prediction.
  - F1 gives a averaged measure of overlap between the predicted and ground truth answer spans. To calculate this both are treated as a bag of tokens.

DATASET

- Our current SQuAD 2.0 dataset contains 129,941 samples.
- The dev and test set contain 6078 and 5915 samples, respectively with roughly equal % of unanswerable/answerable questions.
- The train set has roughly twice as many answerable questions as unanswerable.
- Majority of questions begin with “What.”

Figure 1. SQuAD 2.0 Question Breakdown

For many years, Saddam had two rights under the sovereignty of House of Saddam. The National Islamic Front first gained influence when General George Akinlai led military action against Saddam. Saddam was the leader of the government of Iraq in 1980. Saddam built a strong economic base with money from foreign Islamist banking systems, especially those linked with South Africa. He also used his oil wealth and took control of the strategic Central Bank of Iraq. Saddam used the steel and many local government workers in the university and military academy while serving as minister of education.

QUESTION ANSWERING

QANet on SQuAD 2.0

MODEL ARCHITECTURE

- Model Details:
  - A layer dropout was added between each sublayer
  - TAG Features: POS, dependency, entity
  - Batch size: 32
  - Hidden size: 128
  - MAX context length(train): 400
  - Learning rate: 1e-3
  - Dropout prob: 0.1
  - Optimizer: Adam
  - \( \beta_1 = 0.8 \)
  - \( \beta_2 = 0.999 \)

- We observe that char embeddings significantly impact performance and that the tag features provide marginal gains.
- It was not immediately clear to us how much the some of the other components helped or hurt

Figure 2. Model Diagram

RESULTS

- Our best model achieved a F1 score of 0.68 on dev and 0.64 on test set. It achieved an EM score of 0.62 and 0.60 on dev and test set respectively.

Figure 3. Dev F1 and NLL plots on the well-tuned baseline with char-CNN embeddings using different hyperparameters

- The efficiency of a Rankine cycle is usually limited by the working fluid. Without the pressure limit of stainless steel) and condenser temperatures are around 30 °C. This gives a theoretical Carnot efficiency of about 63% compared with an actual efficiency of 42% for a modern coal-fired power station. This low turbine entry temperature (compared with a gas turbine) is reaching supercritical levels for the working fluid, the temperature range the cycle can operate over is quite small; in steam turbines, turbine entry temperatures are typically bounded by steam temperature and condenser temperature.

Figure 4. Dev F1 and Loss plots of improvements over baseline

- We want to further explore local attention on a dataset with larger context length.
- We would like to dig into why there was a large gap between our dev and test results.
- It would be interesting to incorporate techniques such as a span reader and pointer networks and try adding an RNN before or after the encoder blocks as this might help with transformers.

FUTURE WORK

- We observe that char embeddings significantly impact performance and that the tag features provide marginal gains.
- It was not immediately clear to us how much the some of the other components helped or hurt

Figure 5. Bidirectional attention visualization using similarity matrix between question and context.

CONCLUSION

- Our QANet implementation reached an F1 of 63.841 and EM of 60.389 on the test leaderboard and improved over a well-tuned baseline.
- Model requires more reasoning capabilities to understand unanswerable questions and perform better on “why” questions.
- Larger context length is not detrimental in this context(max 400 tokens), thought extreme lengths might be
- In practice we did not find this model more efficient than the BiDAF baseline

REFERENCES

- Anuprit Kale, Anuprit@stanford.edu
- Edgar Velasco, edgarv1@stanford.edu

Figure 6. Model metrics(dev) on different question types.

Figure 7. Model metrics(test) by context length groups.