Final Report Section Write-Up

**Abstract**

in this paper, we exstracted the LiSCU which is a dataset of literary pieces and their summaries paired with descriptions of characters that appear in them (presented by "Let Your Characters Tell Their Story": A Dataset for Character-Centric Narrative Understanding). It was created from various online study guides such as shmoop, SparkNotes, CliffsNotes, and LitCharts. We also created another example dataset using coreference resolution including context with Harry Potter and the Philosopher's Stone. We present our findings concerning the results from training the seq2seq hugging face model using the LED longformer from allenai base 16384 on the LISCU dataset and additionally evaluating on the Harry Potter coref dataset. We also present our findings using reinforcement learning and generating character graphs.

Edit:

In this paper, we present a method of approach to create summaries of characters’ stories in literary pieces, based on coreference resolution, training a Longformer model (with Seq2Seq), reinforcement learning, and character relationship graphs. We created an example dataset using coreference resolution including context with Harry Potter and the Philosopher's Stone. We extracted data from the LiSCU dataset, which is a dataset of literary pieces and their summaries paired with descriptions of characters that appear in them (presented by "Let Your Characters Tell Their Story": A Dataset for Character-Centric Narrative Understanding). We present our findings with the results from training the seq2seq hugging face model using the LongformerEncoderDecoder (LED) from Allen et. al. on the LISCU dataset and additionally evaluating on the Harry Potter coreference dataset. We also present our findings and results using reinforcement learning and generating character relationship graphs.

**Introduction**

Hi, this is for our final research project in cS 4650. the 3 of us wanted charater summaries. motivation is abstractive summarization for longer texts like literary novels, ancient texts, or even scripts of movies.

Edit:

The ability to automatically generate summaries of texts has been a long-standing goal that is actively researched in the field of natural language processing. One of the key challenges in this task of summarization is identifying the salient characters and their relationships to the plot, and more specifically their relationships to each other. This is abstractive summarization.

This paper presents our approach to creating summaries of characters’ stories in literary pieces using natural language processing techniques. Specifically, we utilize coreference resolution, training a LongformerEncoderDecoder (LED) model (with Seq2Seq), and reinforcement learning to generate character-centric summaries. We also create character relationship graphs and present those findings.

We trained the LongformerEncoderDecoder (LED) model to generate summaries of the characters’ stories. The Seq2Seq approach is used to map the input sequence of our data to the output sequence of a generated summary, with the LED model being trained to perform this task.

Additionally, we incorporate reinforcement learning to fine-tune the generated summaries and improve their coherence and readability. We also create character relationship graphs to visualize the relationships between the characters in the story and how they affect the plot.

We extracted data from the LiSCU dataset which consists of literary pieces and their summaries paired with descriptions of characters that appear in them. To evaluate the effectiveness of our approach, we also created a dataset using coreference resolution with Harry Potter and the Philosopher's Stone.

This project is for our team’s final research project in the Natural Language (C.S. 4650) class at Georgia Institute of Technology. It is a culmination of an implementation based on what we learned in class and our own research. The rest of this paper is organized as follows: we discuss related Engines in section 2, followed by an explanation of the Data and Model in section 3. This is followed by a detailed explanation of our Methodology in Section 4. We present our experimental Results in Section 5, and conclude with sections 6 through 8 which are titled Limitations, Ethics, and Acknowledgement respectively.

Engines

To generate our dataset, we used an M1 Mac with whatever stuff ajay has. to train our models along with the reinforcement learning and character graph generation we used a google colab pro subscription which offers access to Nvidia Tesla K80, T4, P4, and P100 GPUs, even so we had to reduce our batch size and input sequence length and output token length more than we would like because of computational and time constraints by Google Colab pro's device gpus and storage availability.

Edit:

To generate our dataset, we used an M1 Mac and shell scripts to extract data from the LiSCU dataset and perform coreference resolution with each Harry Potter novel. We used the spaCy library to perform this coreference resolution. We also utilized Python to preprocess the data and prepare it for training the model.

To train our models, along with the reinforcement learning and character graph generation, we used a Google Colab Pro subscription, which offers access to Nvidia Tesla K80, T4, P4, and P100 GPUs. These GPUs provide a significant performance boost compared to using only a CPU, allowing us to train our models much faster. For our models, we used the Hugging Face transformers library to implement our Seq2Seq models and the reinforcement learning algorithm.

The data and model are further discussed in the next section, and limitations and challenges that we ran into with respect to these engines and libraries are discussed in section 6.

Data and Model

**Limitations**

We used a pretrained allen ai model on the english dataset and also used english core language from spacy for ner. results may vary on what allen ai or spacy have trained for their language bases.

we also have issues with scalability to long texts as we had originally proposed feeding in the entire book to output character summaries given a character but no modern day model would be able to handle that many tokens. the models we could find could take in 16384 tokens for their input sequence length like the xl longformer or the led longformer.

even if the xl longformer or the led longformer could handle the 16384 token length for the input sequence length, running that with a batch size of 8 would cause memory issues and running with a batch size that wouldnt throw a memory issue would cause the runtime to take upwards of 10 hours on colab pro as most of the time we were given devices with 16 gb gpu ram. we chose to use 4096 tokens for the input sequence length and 512 for the output length with a batch size of 4 and the training took 3 hours and 30 minutes with the 16 gb gpu ram. given better devices with more advanced gpus and more storage, the model could output better summaries.

Edit:

We ran into a few challenges and limitations when working on this project. The main one being scalability. We worked with the LED model because of the Longformer’s ability to take in up to 16k tokens using a sliding window technique. However, the first Harry Potter novel (one of the books we worked very closely with) alone is 76,944 tokens long. As a result, we went chapter by chapter to complete our methodology.

Even with access to powerful GPUs, we still faced computational and time constraints. We had to reduce our batch size and input sequence length and output token length more than we would have preferred, to fit the available GPU memory and storage constraints. We also had to carefully manage the training process to ensure that we made efficient use of the available resources, for example, by using early stopping to prevent overfitting and reduce training time.

Despite these challenges, we were able to train our models and generate character-centric summaries for the literary pieces in our dataset. Had we been given more time, we would have done more training, hyperparameter tuning. Coreference resolution would also be improved, and the reinforcement learning aspect of our project would have been more fleshed out.