AUTOMATIC SCRIPT GENERATION USING DEEP LEARNING

Milestone-3 Presentation

Komal Atul Sorte
Student
ks7229@rit.edu

Dr. Hans Peter Bischof
Faculty Advisor
hpb@cs.rit.edu

Prof. Matthew Fluet
Course Advisor
mtf@cs.rit.edu
Target
Build and enable an AI agent such that it learns a piece of script and automatically generates text based on appropriate context.

Further Accuracy Improvement
By considering,
- Contextual meaning of words
- Synonyms (e.g. check or validate)
- Word embeddings
Data Pre-processing
1. Import the data and relevant libraries for data manipulation
2. Perform high level data inspection followed by detailed exploratory data analysis
3. Conduct data cleaning and treatment to get data in consumable format
4. Perform text analysis

Web Scraping
1. Search and identify apt data sources and elements
2. Understand how the data is organized and fetched on/from the source
3. Develop a crawler to extract data elements from source
4. Import data into the storage system in a form which could be further processed

Building AI Agent
- Divide data into train, validation and test
- 60%
- 20%
- 20%

Training Data Scripts
Validation Data Scripts
Testing Data Scripts

Final Data
Fine Tuning
Evaluate the model
LSTM

Prediction
Input Line
Must predict and write a script
Output Text/Dialogue

Two approaches – word Level and character level
Goal
WEEK 12
Prediction & Evaluation

- Fine-tuning
- Script Generator (Predictor)
- Evaluation & Testing
- Report
- Final Model
Fine-Tuning
Hyper-parameters

- number of training epochs: [0, 100]
- number of memory units to the layers: {128, 256, 512, 1024, 2048}
- dropout percentage: {10%, 20%}
- number of hidden layers: {1, 2, 3}
- batch size: {128, all instances}
Fine-Tuning

Constant Hyper-parameters

![Graph 1: Fine-tuning: Units vs Loss for Word Level Language Model](image1)

![Graph 2: Fine-tuning: Units vs Loss for Character Level Language Model](image2)
Script Generator (Predictor)
Script Generator

Load Model
- Character level language model
- Word level language model

Load Mappings
- Token to integer
- Integer to token
- Pickle API

Predict
- Encode tokens to int
- Predict next token & append
- Repeat until limit

Convert To Human Readable Format
- Encode integers to tokens in output

Sample Script
- Output script of specified len
Inputs considered for evaluation
- Lines already known to the model (*scripts used for training*)
- Lines from test data

Judgement based on following metrics
- Grammar
- Novelty
- Linkage between sentences
- Repetition

*Models fine-tuned again as per prediction outputs*
Underfit character level model

Monica: Theres nothing to tell Hes just some guy I work with Joey: Cmon youre going out with the guy Theres|

Model has not learned anything

Overfit character level model

scene: central perk chandler joey phoebe and monica are there monica: there t tt he t on a aou yo o o o t th h w a to o ou ah o oo ou oo h wo t a he h ou o o wo ta ah a o t th w o h h h o a a an w

Model has learned wrong aspects than anticipated although accuracies and loss obtained suggest otherwise
**Sample Input/Output**

**Overfit word level model**

Monica: There's nothing to tell. He's just some guy out down the city.

Monica: Who wasn't invited to the wedding.

Rachel: Ooh, I was kinda hoping that wouldn't be an issue.

Monica: Now I'm guessing that he bought her the big pipe organ and she's really not happy about it.

*Model has mostly remembered exact lines from training*

**Best word level model**

Rachel: Alright, don't tell me, don't tell me. I'm getting pretty good at this.

Chandler: Heyheyheyyy I cannot feel my legs.

Ross: You guys.

Monica: He's such a mess.

Joey: Ross, let me ask you a question.

*Outputs closer to human perspective – novelty to some extend*
Evaluation & Testing
Measure of Success & Results

Loss

- Cross-entropy loss for training and validation data for both approaches
- Training & validation loss decreases significantly but validation loss increases after convergence

Qualitative judgement and human survey

- Word level language model outputs more “realistic” human-like samples.
- Character level tends to produce output with words not in vocabulary.
Results – Cross-entropy Loss

Character Level Language Model

Training and validation loss vs epochs

Word Level Language Model

Training and validation loss vs epochs

Validation accuracy never really took off. Loss might not be the best way to look at the generalized model.
Conclusion
Conclusion & Future Work

Character level language model has problems with vocab in spite of validation accuracy and loss being better than word level.

Since world-level language model-based models are trained on world level inputs and anticipated to be produce word level outputs, they tend to generate more “realistic” human-like samples.

More details post Human survey - email will be sent today.

Future work
Train model over all the data
Use more metrics to evaluate the model


Thank You