Movie Reviews Under Review

Part 3: Knowledge Graphs and LSTM

Aubrey Molitor

MSDS 453: Natural Language Processing

Introduction

As a crucial part of identifying the themes and similarities in documents, entity and relation extraction identify major features of a document and how they relate to others. Part of speech identification, word and document vectorization, and recurrent neural networks are all useful tools to perform successful entity and relation extraction. By performing these techniques, natural language processing can take a step further to natural language understanding, which is key to creating generative, interactive speech. For classification, summarization, and text understanding, the next phase of movie review evaluation will be entity and relation extraction.

The Data

The data set for this project is the class corpus data set, compiled of ten reviews for each student in the class. The "trained data" was a 80% split of the corpus data frame, excluding the Genre of Movie (target variable). These texts were processed with a clean method to normalize, tokenize, lemmatize, stem documents and remove stopwords, then recombine the tokens to create a string per document. If train and test data were not needed, the data was processed as mentioned above directly from the corpus data frame, without dropping the target variable. Some of the experiments, however, will test across raw data and processed data as described above. The uses for these differing data will be described below.

Research Design and Methods

Part 1: Knowledge Graphs: SpaCy

For the knowledge graphs, two experiments were performed - one with The Toxic Avenger documents, and one with the entire corpus. The documents examined in the knowledge graph will not be processed, but will instead be included as raw sentences at the sentence level. The entity extraction will be extracted at each sentence via the spaCy dependency tree tokens, and the relations between those entities will be identified for each sentence via the spaCy Matcher. Then, these relations will be used to create a circular graph that demonstrates the connections between each of these entities in a visual representation.

Part 2: Knowledge Graphs: SpaCy + LLMs

The BERT Named Entity Recognition pre-trained model will be used in addition to the spaCy features for another strategy at entity and relation extraction. The NER pipeline will be used to define the dictionary to process each document and identify entities within each of them.

A score will be included for each word, and the type of entity (person, type, misc.) will also be defined. The relations will be collected and graphed as described in part 1.

Part 3: LSTM RNN

The LSTM models to classify documents will each use the processed train/test data as described in the data section. These models will all use the following features: a TensorFlow Sequential model; an input Embedding layer; RELU activation functions for LSTM hidden layers; softmax for the output layer; the Adam optimizer; and the SparseCrossEntropy loss function.

The inputs for these models will be doc2vec vectors of the entire corpus. The models will all attempt to predict the document's genre, which will be formatted with a Label Encoder. The default features of each of these models will be as follows: doc2vec embedding size of 100, 300 epochs, and 200 node units. Some of these features will be changed per model; if it is not mentioned, the values for that model are these defaults. After the model is created and fit to the training data, F1 scores for genre predictions will be calculated for the training and testing sets.

Model 1 will be a one-layer RELU model with each of the default features; this is to serve as a baseline. Model 2 will be similar, but instead of one LSTM layer with the RELU specification, it will be coded as a Bidirectional LSTM layer. Model 3 will be a two-layer LSTM RELU model with an epoch size of 400. Lastly, Model 4 will be a two-layer LSTM RELU model with a node size of 100.

The Results

Knowledge Graphs: SpaCy and SpaCy + LLMs

The spaCy + LLM model had more focused entities than the spaCy model with the Toxic Avenger documents. For example, those with the spaCy + LLM were names, movies, or organizations, without many filler words or attachments from document context: Daniel Craig, License to Kill, Casino Royale. These are all short, direct entities that accurately label an idea, person, or film. This performance was also similarly successful in the entity extraction for The Toxic Avenger documents. Those most frequent and relevant were Blair, Troma Entertainment, and The Toxic Avenger. There were a few slip-ups, such as two separate entities – "I don" and "t Feel At Home Anymore" – for the movie title that should be combined. There was also just the number 80. However, these were few exceptions to the generally productive entity extraction.

In contrast, the lone spaCy model with the Toxic Avenger movies had poorly defined entities: “originally Lloyd Kaufman,” “mutated poultry blair,” and “nerdy ferd junko who” are just some examples. Nearly every entity extracted has this kind of unclear mix of document-specific descriptors attached to what would be the entity. In this case, it seems the pre-trained model is much more effective at picking out clear entities. This model also failed to capture similar entities in the same terms. For example, “troma entertainment,” “troma film,” and “1984 troma entertainment” were all considered different entities. This separation made it so that each entity only connected to one other in the knowledge graph, forming a graph with little use for extractable usable information.

As for the relations, the spaCy + LLM model also proves superior. There are several repetitions of entities that have various connections in the knowledge graph. The Toxic Avenger, Peele, Evelyn, and The Conversation are just some of the entities that have several connections, building an informative web of connected ideas and the structure of those relationships. In contrast, the poor entity identification of the spaCy model made it difficult to form informative relations. There were several entities that should have had several connections, such as Troma, but because of the separation in words described above, each only had one relation. This made the graph useless in looking for connected lines and concepts coming from a single idea.

LSTM RNN

The LSTM models had varied performance across multiple iterations. For each model, the F1 score was never above .3 for the testing data, and usually around .15-.25. For the training data, the F1 was usually around .3-.5. However, across several runs, all of the models performed relatively inconsistently. Each model had a run at both the highest and lowest F1 scores. Still, across multiple iterations, the first and last models seemed to perform the best overall. The second and third models also tended to perform relatively similarly to each other, with model 3 tending to have a slight edge over model 2 when they tended to differ. Based on several iterations, the Bidirectional LSTM model (Model 2) was overall the worst, and the single-layer LSTM RELU model (Model 1) was generally the best. It was unexpected that the simplest model had a somewhat consistent rank above the others. For future iterations of this experiment, it would be useful to examine differences in the activation functions, loss functions, and doc2vec embedding size to see whether these changes could result in a stronger performance.

Conclusions

Despite the power of Deep Learning as it could be applied to NLP, the use of TensorFlow and LSTM RNN models seemed to have questionable benefits in this experiment. The best-performing classification model was, in fact, not a DL model at all, but one that only had one hidden layer. Because of this, further experimentation is needed to see what could pull a higher performance out of the TF models using LSTM RNNs.

However, the first section of the experiment still did bring a clear insight. The use of the spaCy + LLM model was much better for entity extraction and relation than the spaCy model alone. Both the entity results and the relation graphs showed how much more fruitful and productive the spaCy + LLM model was to identify clear, consistent entities and draw multiple relationships. SpaCy alone was strongly inconsistent in its ability to identify like entities, even for one movie – this hinted at how disastrous its performance proved to be across the entire corpus.

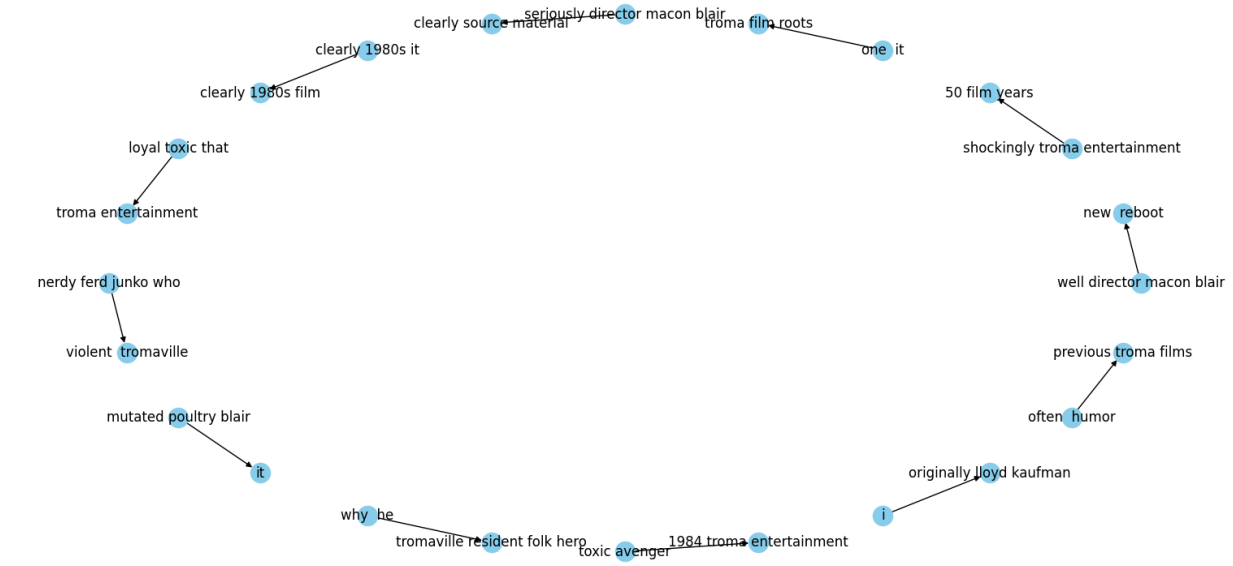
APPENDIX

1. Entities for The Toxic Avenger, spaCy + LLM
Entity: Fantastic Fest, Type: MISC, Score: 0.9965514540672302, Start: 12, End: 26
Entity: The Toxic Avenger, Type: MISC, Score: 0.9946606755256653, Start: 97, End: 114
Entity: Macon Blair, Type: PER, Score: 0.9995699524879456, Start: 0, End: 11
Entity: I Don, Type: MISC, Score: 0.8310539126396179, Start: 13, End: 18
Entity: t Feel At Home, Type: MISC, Score: 0.680655837059021, Start: 19, End: 33
Entity: This World Any, Type: MISC, Score: 0.8533236384391785, Start: 37, End: 51
Entity: Hold the Dark, Type: MISC, Score: 0.8280026912689209, Start: 57, End: 70
Entity: 80, Type: MISC, Score: 0.704387366771698, Start: 144, End: 146
Entity: Lloyd Kaufman, Type: PER, Score: 0.9994139671325684, Start: 178, End: 191
Entity: Joe Ritter, Type: PER, Score: 0.9993868470191956, Start: 196, End: 206
Entity: Blair, Type: PER, Score: 0.9994809031486511, Start: 3, End: 8
Entity: The Toxic Avenger, Type: MISC, Score: 0.9940410256385803, Start: 91, End: 108
Entity: Blair, Type: PER, Score: 0.9993820190429688, Start: 168, End: 173
Entity: Blair, Type: PER, Score: 0.9985255599021912, Start: 46, End: 51
Entity: The Toxic Avenger, Type: MISC, Score: 0.9832758903503418, Start: 54, End: 71
Entity: The Toxic Avenger, Type: MISC, Score: 0.9784103035926819, Start: 11, End: 28
Entity: Troma Entertainment, Type: ORG, Score: 0.9358675479888916, Start: 43, End: 62
Entity: Troma, Type: ORG, Score: 0.8976012468338013, Start: 24, End: 29
Entity: The Toxic Avenger, Type: MISC, Score: 0.9860642552375793, Start: 26, End: 43
Entity: Blair, Type: PER, Score: 0.9994927644729614, Start: 164, End: 169
Entity: The Toxic Avenger, Type: MISC, Score: 0.9779045581817627, Start: 18, End: 35

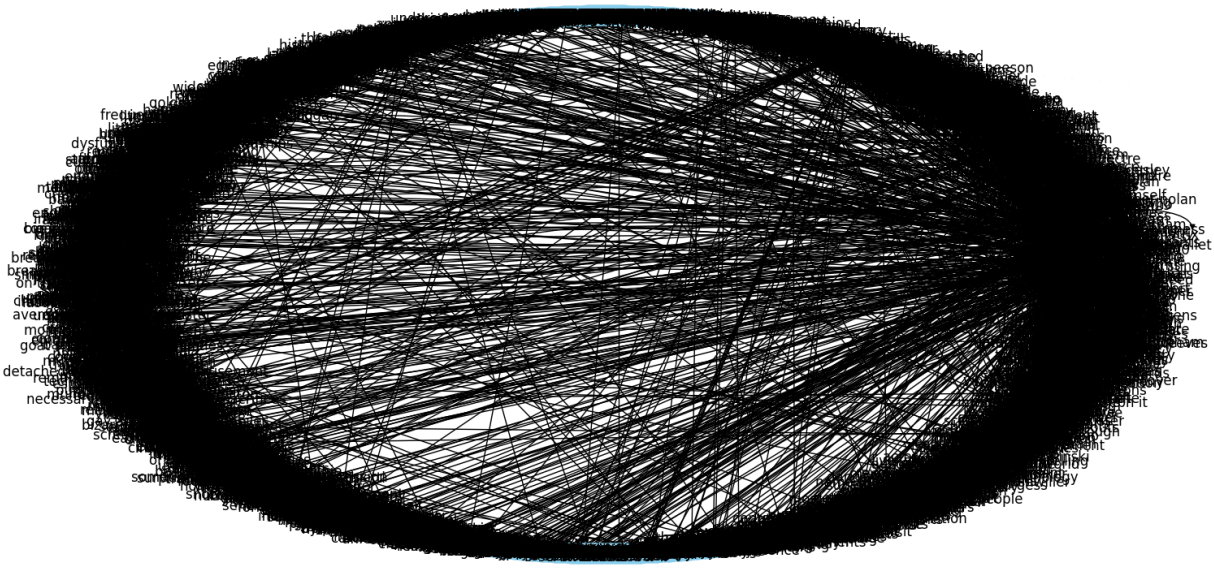
2. Four different runs of the LSTM Models, with F1 Score outputs:

F1 SCORES:	Run 1	Run 2	Run 3	Run 4
Mdl 1	0.25	0.25	0.15625	0.15625
Mdl 2	0.25	0.09375	0.25	0.09375
Mdl 3	0.25	0.125	0.15625	0.15625
Mdl 4	0.15625	0.1875	0.28125	0.15625

3. Knowledge Graph – The Toxic Avenger (Lone SpaCy Model)



4. Knowledge Graph - Entire Corpus (Spacy + LLM)



(The graph is included not for legibility of the words, but rather for the demonstration of multiple connections being drawn across the different entities. This is in contrast to the circle-like graph in Appendix 3, where no lines crossed the middle.)