Comparison of Generative Text Strategies

Abstract: Text generation may someday help us finish unfinished works of literature. To that end, I am interested in comparing methods of text generation to see what the best strategy would be to undertake that task. In the course of this research, I will compare word- and character-based models using the literary works of Jane Austen as my input text. In comparing the two styles, I was able to find that under limited memory conditions, character-based performed better, and word-based would be better applied with a significantly larger corpus of input data.

---

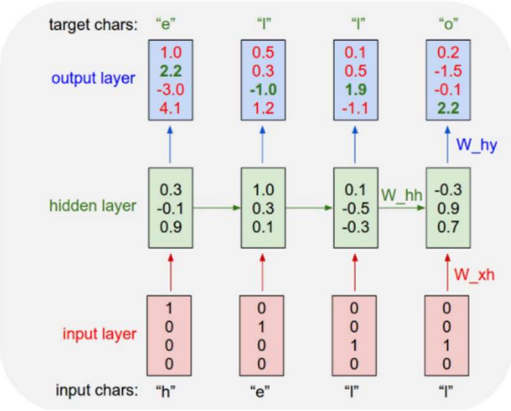
1. Introduction

In 1817, Jane Austen began work on a novel about the people of a seaside town called Sanditon only to abandon it when she was confined to her bed with the illness that would ultimately take her life six months later. The unfinished book has become popular with modern day writers interested in taking a stab at finishing the novel (Sutherland, 2019). One of the possible applications of text generation in the future could be to do just that—finish unfinished works in the style of the original author. This idea has already been tested in music, with the application of AI to finishing Beethoven's 10th symphony (Hall, 2021). As a first step to this task, I will be using NLG with the text of Austen's novels to compare word-based and character-based text generation.

2. Literature Review

2.1. RNNs & LSTMs

The models used in this research are both LSTM models, a form of RNNs. In his post "The Unreasonable Effectiveness of Recurrent Neural Networks" (2015), Karpathy explains why he used RNNs in his character-based model. As opposed to regular convolutional neural networks, which only will only accept a fixed vector input and output a fixed vector, RNNs are more flexible. They allow us to train on sequences of inputs and outputs. Even if the input isn't sequential (like an image), the model can still learn on it in a sequential manner, to the benefit of the overall model. RNNs also have the benefit of memory. Using the example of the word 'hello' the model calculates the probability distribution of the possible next letter.



Importantly, the model cannot just rely on the direct input to make its determination. As he points out: "Notice also that the first time the character 'l' is input, the target is 'l', but the second time the target is 'o'. The RNN therefore cannot rely on the input alone and must use its recurrent connection to keep track of the context to achieve this task" (Karpathy, 2015).

2.2. Word vs. Character

Since both models utilize LSTM one of the main ways they differ is in how the text is preprocessed before being fed into the model. As mentioned above, Karpathy employs a character-based model where the model determines the probability of what the next character in the word will be. Similarly, a word-based model will use probability distribution but for 'guessing' the next word in the text instead of character. This is the method employed by Nazarko in her tutorial.

Much has been made of models like GPT-2, an extremely good word-based NLG from OpenAI, but in smaller examples data scientists often use character-based models. This speaks to the drawbacks of word-based models.

In a character-based model, each character is a separate unit, that will give the researcher much more data for the same 1 page of words than the word-based model. That is why a model like GPT-2 is trained on 8 million webpages (Radford, 2019). It needs a lot more text to have the same amount of data as a character-based model.

Another issue is the sheer number of unique inputs word-based models must take to make their predictions accurately. Character-based models have only each letter of the alphabet plus a small variety of punctuation. Less than 100 unique inputs. For word-based models, each word is a unique input. In the English language there are estimated to be over 200,000 words. That can result in a model with a lot of parameters very quickly (Tauscher, 2020).

One interesting thing that both models do is leave in a variety of formatting that would normally be removed in natural language processing tasks. This is because generating realistic text requires the inclusion of things like punctuation and capitalized letters in the model.

3. Methods

In order to compare the two styles of text generation I followed the tutorial by TensorFlow based on Andrej Karpathy's blog post entitled "The Unreasonable Effectiveness of Recurrent Neural Networks". That post uses character-based text generation with an LSTM model.

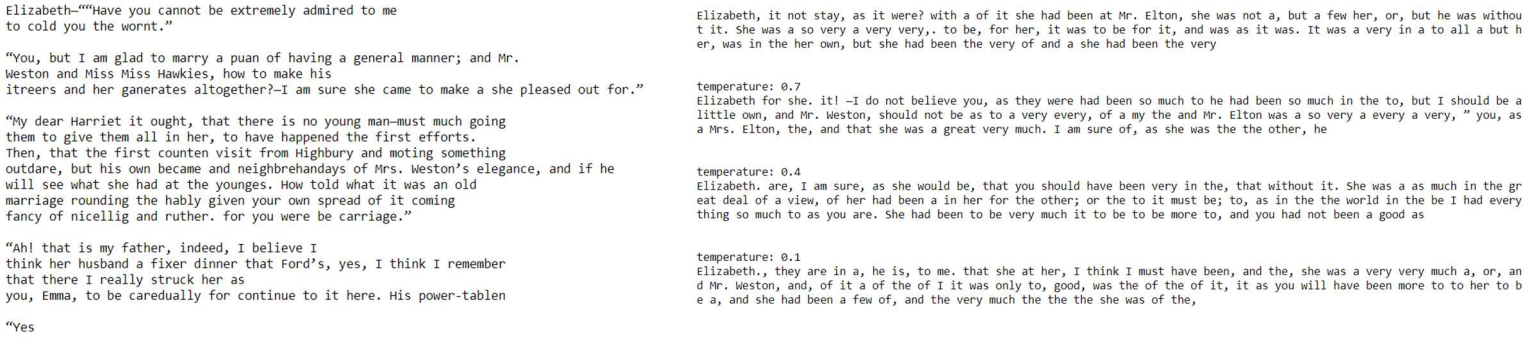
The word-based text generation code I got from the blog post "Practical text generation using GPT-2, LSTM and Markov Chain" (Nazarko, 2021). Specifically, I used the LSTM portion of the tutorial.

I used the text for Austen's novel 'Emma' to test the models, downloaded from the Project Gutenberg website. Then I replaced that with text from the unfinished novel Sandition and ran the character-based model again.

4. Results

I first ran both models with only 10 epochs and compared the quality of the text generated after the training of the model. The input word I used for both was 'Elizabeth'. What I found was that the word-based model at all 4 temperature options was of pretty low quality, while the character-level text was surprisingly decent.

I then ran the models with 50 epochs each and checked the results.



(left to right: character-based, word-based models)

Again, I thought the results from the character-based generator were much better. In the 10-epoch test it was able to generate a section of text as if a character is speaking. In the 50-epoch version it does something that resembles a conversation. The only drawback of the character-level generator is that it is making up some of its own words. "Forgituine", for example, is not a real word. In the word-based generator you can see that the text is starting to come together and make more sense but it is not quite on the same level. The sense of formatting is not natural, although I can see it starting to form, and although all the words are actual words, the way they are put together doesn't make sense yet. This suggests that the word-based model will get up to speed but it could take at least 100 or more epochs.

Finally, Since the character-based model performed the best in my tests, I decided to try replacing the input text with the text from *Sanditon*. I then changed the input sentence for text generation to “It was impossible not to feel him hardly used: to be obliged to stand back in his own house and see the best place by the fire constantly occupied by Sir Harry Denham.” Which is the final sentence from the unfinished draft of the book. The results are below:

```
It was impossible not to feel him hardly used: to be obliged to stand back in his own house and see the best place by the fire
constantly occupied by Sir Harry Denham. "And if this is the case we shall be out again on business relative but them to be the
following hamp an her hard and her prusetSin the constant effliving your fartule in the invertioned of a very indifferent accou
nt. Mary, I nature may be neiesbyed, a sweetly modest and yet may not have been able to present baronet taken place or we shoul
d have seem that Lady Denham was hear in a stately gentleman which, placed over the diese for I left her she wast of being usef
ul escaptions to invite one of the dirgh of points of mind a respectable impression which the sight the contrasuce of judgies;
and who is not much good news for Mr. and Mrs. Parker, whose manners deceasing her on purpose to be ill-useded. It was solichee
t in the cerear to fancy themselves ill if there is perveal? Willingden to his sister and toose chaises. The projuity of the st
oom, again to Mr. Heywood only one, you see you was and he recommended Mr. Hollis that a diffications, and without a rical, add
ry acquainted with exceptionation. Her first hurns for Mr. Hollis'ther Susan. The family, principal mover and actor. She had be
en to wonder a fewerar of Miss Diana possible at this relation and disinterested as they were warm.
```

As you can see, the letter jumble in the words is pretty significant, although the snippet appears to represent something of a conversation. Of course, the final input sentence is referencing the portrait of a dead man so I may have been a little unfair in using that as a test of the model—the context beyond the single sentence is important in this case.

5. Conclusions

Can AI finish unfinished works of literature? The answer I gleaned from my research is not yet, at least. Although NLG has the ability to learn the style and syntax of a particular writer’s voice, it is a long way off from the larger issues of story and structure required to realistically complete an author’s work in full. While I came to the conclusion that the character-based model produced an output that sounded more realistic as to the literary voice—despite the occasional fake word—when looking at the top end models of today such as GPT-2, I believe that the word-based model would have performed better when fed more input data. The problem with word-based models arises when an author has only a few small works that the researcher is trying to emulate. If it were possible to produce better results without such a heavy load of input data, the word-based model would be superior.

6. Bibliography

Hall, S. (2021, September 28). *Beethoven's unfinished Tenth Symphony completed by Artificial Intelligence*. Classic FM. Retrieved November 22, 2021, from https://www.classicfm.com/composers/beethoven/unfinished-tenth-symphony-completed-by-artificial-intelligence/.

Karpathy, A. (2015). The Unreasonable Effectiveness of Recurrent Neural Networks. The unreasonable effectiveness of recurrent neural networks. Retrieved November 22, 2021, from http://karpathy.github.io/2015/05/21/rnn-effectiveness/.

Nazarko, K. (2021, January 15). Practical text generation using GPT-2, LSTM and Markov chain. Medium. Retrieved November 22, 2021, from https://towardsdatascience.com/text-generation-gpt-2-lstm-markov-chain-9ea371820e1e.

Radford, A. (2019, February 14). Better language models and their implications. OpenAI. Retrieved November 22, 2021, from https://openai.com/blog/better-language-models/.

Sutherland , K. (2019, September 6). Continuing Jane Austen's unfinished novel Sanditon. OUPblog. Retrieved November 22, 2021, from https://blog.oup.com/2019/09/continuing-jane-austens-unfinished-novel-sanditon/.

Tauscher, J. (2020, September 13). Word vs. character text generation. Medium. Retrieved November 22, 2021, from https://medium.com/@john.l.tauscher/word-vs-character-text-generation-80a6dbba123a.