New Books

Sean Foley

I used Markhov Analysis to create a new book in the style of David Copperfield. I wanted to create books in the style of Dickens in general, but I only succeeded in downloading one book. My goal was to create a book that had a much different "plot" and meaning from the original book, so few long strings of original text.

There were several stages to my implementation. First, *get_books.py* downloaded my selected books and added them to the list in *dickens_texts.pickle*. Next, *markhov.py* loaded that list, created a prefix-suffix dictionary, and stored that in its own pickle. Finally, new_book.py loaded the Markhov dictionary and created a book of length *depth.*

The first step was fairly simple, and my only design consideration was storing a list so that I didn't have to redownload the books over and over again or make a bunch of separate pickle files. The next bit of code was more complicated. In the interest of correct pronunciation, I split the input book into a list of individual words and punctuation marks, which were treated the same.

To map potential prefixes to potential suffixes, I constructed a dictionary of lists, where the keys were prefixes and the lists contained every suffix for that prefix, including repeats. I chose a dictionary for the highest level data structure because it was very fast and easy to check if I'd already looked at a particular prefix, and it was simple to map my prefixes to suffixes. My original thought was to create a dictionary of dictionaries, where each prefix would map to a dictionary that mapped suffixes to their probabilities. I realized it would be much simpler to just pick a random suffix out of a list, and use the number of repeats of that suffix to represent its prevalence. Another design choice I made was the order of the Markhov analysis; because I wanted very unique books without long strings of original text, I went for small order (specifically 2).

Finally, *new_book.py* took the Markhov dictionary, started with a random seed prefix, and built a book by selecting random suffixes. If it got to the very end of the original book, which has no suffix, it would reseed itself with another random prefix.

My first short story and first full novel are in my GitHub repo under the names BLANK and BLANK. Because the order was so small and I did no adjusting for grammar, they're pretty nonsensical, but in an endearing way. My favorite paragraph so far is:

> *I walked to and fro upon the Norwood Road, while I wrote a note to Peggotty about crocodiles. However, I went to it again, Emly, you know what I had had my doubts and apprehensions on that subject, is always before me in my corner; scared by the sea, observed my aunt.*

Because of the way I built the code, the punctuation is, for the most part, correct. There's something funky with newlines that I haven't figured out, though, so the paragraphs are kind of jagged. I left in the Project Gutenberg text at the beginning and end; it would’ve been very easy to strip out, but I actually kind of like it, because you get sentences like:

> *There was a beggar himself, and the balance in sherry, my dear Sir, though she terminated, as if she should stay away from, the plain truth; though, now, if it had been made extremely snug and comfortable there and then went on. That will be linked to the Project Gutenberg Web pages for current donation methods and addresses. WILKINS MICAWBER,*
>
> *L. Beggared Outcast,*
>
> *II. Another Retrospect*
>
> *Most secret and confidential.*
>
> *MY DEAR SIR,*

And:

> *I dont watch his eye in idleness, but is in attendance as my child-wife to a work with the phrase Project Gutenberg), you see. Will you write and tell Mr. Dick, mildly.*

In combination with another project I was working on, I practiced incremental coding. I built small bits of code, tested and fixed them thoroughly until they did what I wanted, and then built more code around those working bits. The errors I got were much more intelligible than I got in the second project, because I had only made small changes since my last test. If I were to do this again, I would spend more time on it. I would sample a lot more books, and work out a way to make my sentences more intelligible while still unique.