

The CS106B Random Writer Contest

Due date: Friday, January 30

On Assignment #2, the second problem has you create a random-writer application that uses a Markov model to generate English text according to a pattern. Figure 1, for example, shows the output generated by two runs of the program (slightly shortened and reformatted so that the figure fits on one page), one that uses *Middlemarch* as its text and one that uses *Hamlet*. The styles are instantly recognizable as that of George Eliot and William Shakespeare. Moreover, the names of the characters make it clear what story served as the model. As impressive as the similarities are, however, there is still one major problem: the stories don't make any sense at all.

Your job for this contest is to submit a randomly generated text of not more than 2000 characters, along with the program and data files used to produce it. What we're looking for is the best story, poem, play, or other "literary" work. To produce it, however, you'll have to do some serious thinking about the design of the program. There are many ways you can extend the program to make it more likely to produce something better than the simple Markov model approach used in the assignment, including the following:

- *Change the program so that the basic unit is a word rather than a character.* Making this change will certainly produce more sensible stories, but does have pitfalls of its own. For one thing, the corpus of text you supply probably has to be much larger if you hope to have your output differ substantially from the input you supply. The complete works of Shakespeare, for example, would be a better starting point for a play-writing program than *Hamlet* alone. It will probably also be necessary to reduce the order of the model or to use a more complicated approach for choosing the next word than the simple Markov approach.
- *Add additional code to make increase syntactic and semantic consistency.* You are not required to stick with the basic Markov model used in the assignment and are free to extend your program in any way you want (subject to the rules described on the last page of this handout). It is, for example, a bit unnerving to have open quotation marks that are never closed. You could fix that by insisting on generating matching quotation marks. Semantic consistency is harder to obtain, but you could, for example, add code to ensure that the lines delivered by characters in the Shakespeare example are in fact delivered by characters who are on the stage. This change would eliminate, for example, the disconcerting situation of having Hamlet address Horatio after the stage direction "[Exeunt all but Hamlet]".
- *Add any features you want to enforce a particular form.* If you want your programs to write sonnets in the Shakespearean mode, for example, you are allowed to ensure that each sonnet has fourteen lines and to try (although doing so is subtle) to count syllables to approximate iambic pentameter. Similarly, if your program is writing a play, you can include speaker tags as part of your general formula for generating text. And you should not feel constrained to have your program produce its output with precisely the right spacing. You are free to modify the formatting (as I did for the examples in Figure 1) to change the line breaks or regularize the spacing.

Figure 1. Random writer examples using *Middlemarch* and *Hamlet* as sources

Non si pub dicer, ne tener a mente,
And builds a heaven in hell's despair.

Thus he had consisted at that time if Mary Garth, who appeared the more keenly he felt that he would sometimes it upset her gravity, "if you think of something for her flightiness, and did mischief when they're sorry," said Caleb, turning from useless words.

Neither of them knew how to use the preparations for me," said Dagley. "I can think of that childish kisses which were most in need of praise. He spoke to Miss Vincy.

Celia, in her cooing way, keeping here."

"Oh ay, they stick, do they? and the rest, without seeming nearer. There are so many things may happen," said Fred, who felt himself plodding along the edges, caused the pick of them; and if there had been urged. His sense of connections. That is a point I wanted up-stairs, poor dear old soul. A man whose learning. But my children in Middlemarch politics generally done something important step without capital practice did him in counteracting me secretly in future--not in future, you know. Garth has invented a new pattern of mittens?"

After the fashion.

"No, I only sketch a little grumbling remonstrated with his own account. Her friends. Take your brother with a still deeper undertone in which dominated them--the painter is not known of Raffles had sustained her point of view, as well as on all other subjects a kind of thing. He only neglects his work only: that was left. I thought you only cares about Mrs. Goby. You have not the sample of virtue or accomplishments in the House had been tinkering long enough to bear up at Fair time, if I didn't mean beating, you know."

Enter young Osric. Cousin Hamlet?

Ham. Not this, by no means!

Ham. I pray you.

Guil. We will, we will!

Laer. I thank your lordship were at leisure, I should impart
a thing to you from his mouth whose voice will draw both friend and foe,
Winner and loser?

Laer. Must there no offence i' th' world.

King. Thanks, dear my lord.

[Exeunt all but Hamlet.]

'Tis too much o'erleavens
To wash it white as snow, thou noble youth,
Have heaven?
But let it be. Horatio, I am dead;
Thou canst not then be false to any man.
That's an ill phrase, a vile phrase; 'beautified' is a vile
phrase.

But you'll be secret?

Both. Arm'd, my lord, put your discourse most eloquent music. Look you lay home to him.

[Leaps in after Laertes.

Laer. O, fear me not!

[Polonius].

Guil. My honour'd lord, 'tis true. A foolish figure!

But now, my cousin Hamlet?

Ham. Nay, but to live

Without the which he has impon'd, as you say,
my mother. Exit.

King. We doubt it is no matter in his heart,

Absent thee from felicity awhile,

And reason panders will.

My stronger than a mason, a shipwright, or a
carpenter?

Prizes

As announced in the first lecture, winning the grand prize in this contest entitles you to substitute a 100% for whatever individual score most negatively affects your grade at the end of the term. Thus, if you win this contest and end up bombing an assignment, the midterm, or even the final, we will overlook that misstep and count it as a 100%. We will probably also award various runner-up prizes and designate some entries as worthy of honorable mention, all of which increases your chances for winning the random contest drawing in March.

Official rules:

1. Only students registered in CS106B are eligible to submit entries.
2. Only one entry per person will be accepted.
3. All entries must be submitted electronically by 5:00P.M. on Friday, January 30, using the standard submission procedure. Late entries will not be accepted.
4. Each submission must consist of a C++ program that generates your contest entry, along with any data files it uses. You may also submit a text file containing your entry, but that is not necessary unless you want to reformat the output. The legal reformatting actions are (1) changing the spacing and line breaks and (2) shortening the entry by deleting characters from the end.
5. To ensure that your program generates exactly the entry you want, your program should begin with a call to `setRandomSeed(n)`, which sets the internal state of the random number generator so that it returns the same sequence each time for that particular value of n . You may play with different values of n until you get one that produces the output you like. To be fair, however, that output must change considerably for different starting points of the random number generator; changing the value of n in the `setRandomSeed` call must generate a story in which no more than five percent of the words match those in the story you submitted, at least on average. (It is also less than sporting to check the value of the random number generator and to deliver up your favorite canned masterpiece only if you happen to call `setRandomSeed` with a particular value.)
6. Your program must not simply reproduce some large, randomly chosen segment of your data files. The story you submit may not include more than ten consecutive words that also appear together in your data files.
7. Contest submissions should be sensitive to Stanford's individual and cultural diversity. Programs or narratives that have the effect of perpetuating negative stereotypes will be disqualified.
8. Contest entries will be evaluated initially by Eric Roberts and Kevin Miller. The best entries will then be evaluated by the entire course staff, which will choose the winner and runners-up.