

Fall Semester 2016
Iowa State University

ENGL 520 - Computational Analysis of English

Final Project Descriptions
Submission Deadline: 9 December 2016

Instructions: Here are some final projects you can consider working on. You can come up with alternate ideas if you want, but I should approve them before you start on them. The project carries 20 marks. I expect to see three deliverables related to this project. A 1 page report that should be submitted before 5th November, describing your approach, timeline etc. (2.5 Marks). Your actual program (10 marks). Your final report and Oral exam (2.5 + 5 Marks).

Idea 1

Implement a preposition tester. Have a collection of 10 paragraphs or so, in your back-end. When the user comes to the UI, you should "randomly" pick one of the texts, create a fill-in-the-blank test out of it for prepositions, and display it to the user. You should create a total of 5 blanks for the user to fill out. Once the user fills this up, and submits, you should show a message saying how many of those did he/she get right. For finding prepositions, you can use a POS tagger.

Idea 2

Implement a question-answer system. Have a collection of 5 texts (300-400 words long) and a set of 5 questions for each text (questions that require one word answers - like factual questions etc. Not grammatical ones). When the user comes to your application, you should randomly pick one text and show it to him. Once the user reads it and clicks next, you should display a list of questions to them, about the text they just read, with some text areas where the users can type their answers. Once they click submit, you should tell them how many questions did they get right.

Idea 3

Read about Eliza program online. Try to write such a program that converses with users, and converts a user sentence into a question. Your program does not need to be extra-ordinary, but should work for atleast some

cases. You can use whatever language processing steps you want to (tagging, parsing, or just regular expressions.. up to you!). It is okay if the questions you generate are ungrammatical, as long as you can justify the output according to the program.

Idea 4

Write a program that compares the n-grams used in two texts and shows the degree of similarity between them as output. If more than 75% of ngrams (n- can be something of your choice. Usually, 1–5 is the range used) overlap between the texts, then your program should print: "the texts seem to be very similar and perhaps one is a plagiarised version of the other.". Else, your program should print "These texts have XX% ngrams common between them"

Idea 5

Write a program that creates multiple choice questions from a paragraph of text. Use a thesaurus or some such resource to generate 4 distractor words for the word you choose to use as a fill in the blank question. You can use Wordnet, POS tagger, Parser - whatever you want.