April 10 2015

Computer Science 51

TF: Allison Buchholtz-Au

Limor Gultchin, gultchin@college.harvard.edu
Meg Knister, mknister@college.harvard.edu
Zac Bathen, zbathen@college.harvard.edu
Gal Koplewitz, galkoplewitz@college.harvard.edu

Final Project Draft Specification – Autocorrect

Brief Overview and Features

We hope to implement an effective Autocorrect application, which operates on some text as it is being written, and automatically replaces misspelled words with more likely alternatives. This will require various "moving parts" specified in the 'Featured' section below -- but it basically boils down to:

(1) An effective data structure for the dictionary for storing words, that can be accessed quickly enough for real time comparisons.

(2) A word-similarity algorithm, that ranks different possibilities from the dictionary based on evaluation of # of same letters, ordering, etc. This is an 'upgraded spellcheck,' in the sense that it indicates not only if spelling is wrong, but also offers corrections.

For the above two, we will be using algorithms described in the following resources (online guides, as well as academic articles on language processing):

http://norvig.com/spell-correct.html
http://www.lleess.com/2013/03/auto-correct-algorithm-in-python.html
http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en/us/pubs/archive/36180.pdf
http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=52A3B869596656C9DA285DCE83A0339F?doi=10.1.1.146.4390&rep=rep1&type=pdf

(3) Running corrections in real time -- this will require some form of event handling. The more feasible version is after every word (so run after space is pressed), and more difficult version could evaluate and offer options after every letter is clicked. Should not be too difficult to find online resource for doing this in Python.

(4) User interface -- whether textbox, web-based, etc.

Cool Additions

(5) Basic Machine learning for expanding dictionary with new words used frequently by user. This could be done in various ways (physical addition by user, used over a certain number of times, etc).

(6) Using different, topical dictionaries that offer corrections in line with a theme (e.g. Pirate dictionary, that corrects "friend" to "matey", "car" to "ship", "man" to "sea rat", "bird" to "parrot", etc).

<u>Technical Specification</u>

Some initial ideas (will become more specific over the coming week, as we read through the technical articles and become more proficient in Python):

- **Dictionary module**

  Values of nodes and how they are connected to each other (in the form of a BK tree as data structure, possibly?)

  Functions -

  1. Lookup/Search – look up possible words and words matches in the dictionary, while making use of a helper Comparison function, that would sort and decide upon the closest word that does exist in the dictionary. According to a ranking heuristic that we will decide on, this latter will choose whether to replace the original word or not.
  2. Add function (for adding words to dictionary, if they are frequently used, etc).
  3. Delete function (?)

- **Event module**

  Initializer - calls the handler function every time a space bar is hit (stretch goal – every time a new letter is typed.

  Handler - call the search function from the dictionary module

Methods -

Correct - will replace the current string, if not found in the dictionary, to the most similar word that is found in the dictionary.

- **Display**

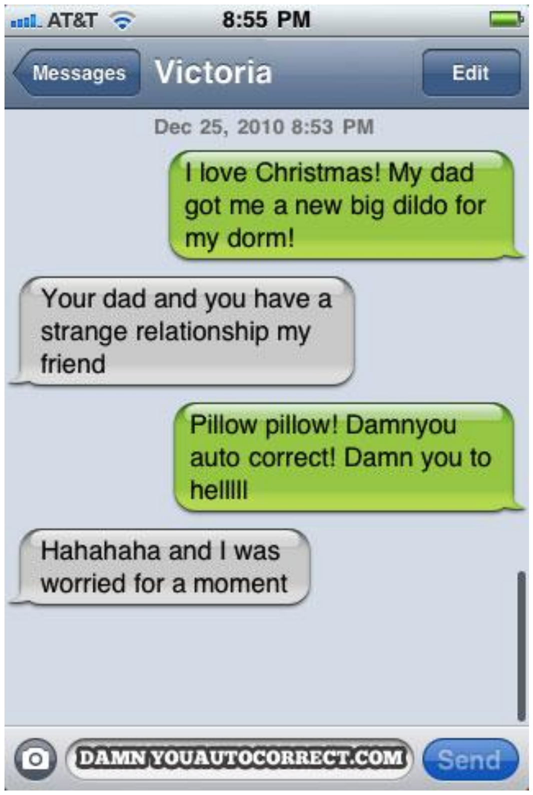
  Functions

  1. Open - Brings up the user interface
  2. Text - display user input in real time
  3. Replace - re-display word after searching dictionary
  4. Notify - Notify users of change

Next Steps

- Read about algorithms used for this in industry, academia, in the following links:
  http://norvig.com/spell-correct.html
  http://www.lleess.com/2013/03/auto-correct-algorithm-in-python.html
  http://blog.notdot.net/2007/4/Damn-Cool-Algorithms-Part-1-BK-Trees
  https://www.topcoder.com/community/data-science/data-science-tutorials/using-tries/
  Scientific articles:
  http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en/us/pubs/archive/36180.pdf
  http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=52A3B869596656C9DA285DCE83A0339F?doi=10.1.1.146.4390&rep=rep1&type=pdf
- Complete python workshop in Codecademy
- Download Python environment (either PyCharm or sublime).

Concluding Words



Dec 25, 2010 8:53 PM

I love Christmas! My dad got me a new big dildo for my dorm!

Your dad and you have a strange relationship my friend

Pillow pillow! Damnyou auto correct! Damn you to helllll

Hahahaha and I was worried for a moment