Title: A conclusive visual and interactive assistant tool to ease exploration of rhymes and synonyms for poets (Rhyme-Viz)

This proposal is organized based on Heilmeier questions followed by a literature survey.

1. A common challenge in writing a poem is finding rhyming words, which shapes the stanza and makes it different from daily conversations [1]. This project creates a one-stop visualization of rhymes and synonyms allowing interactive exploration of word relationships in an organic manner.
2. Currently writers either rely on memory and experience or use thesauri or rhyming dictionaries. However, dictionaries lack interactive visualization. Reference [2] introduces a visual and interactive dictionary; however, it is not helpful for finding rhymes.
3. This project visualizes synonyms and rhymes with an interactive continuous tree structure which includes relevance of a rhyme and clustering by color and position. Natural language processing (NLP) is used to prioritize better rhymes in similar concepts [3-8].
4. This is useful for writers, poets, rappers and even scrabble players.
5. This project makes writing a poem much easier and faster, which can be measured through user studies.
6. Risks are relying on NLP to suggest relevant rhymes. The payoff is that the NLP gives useful rhymes
7. There are no monetary costs. D3, Python, and SQLite will be used.
8. The project will take 8 weeks. Three weeks focuses on language processing and the next three weeks focuses on visualization. The final two weeks is for testing. Each member has done literature review. Robert and Jag focus on back-end coding and language processing, and Majid and Patrick focus on interactive visualization.
9. Midterm check for success will be testing natural language processing features. Final check for success will be on visualization testing.

Survey:

* + - 1. Reference [1] explains what makes a poem sound different from daily conversation. A big part of this difference are rhymes. This reference discusses the importance and role of rhyme in a poem, in addition to different types and aspects of rhyming words. This reference helps us to gain a better insight on rhyming words, and represent this concept visually.
      2. A survey on different types of graphs for visualization, beside navigation, interaction and clustering in graphs, is presented in [9]. Rhyme-Viz has a graph based visualization, and we need to find a layout which would be suitable for interaction and clustering on rhyming words. We select the graph layout from this paper and modify it for our specific use.
      3. Reference [10] suggests a novel approach to find rhyming words in an on-line dictionary. The paper goes further and suggests a method to find the pronunciation of unknown words. This paper is useful for us since we plan to use language processing methods to find rhymes, and we will improve it by adding other rhyme finding algorithms, interactive visualization and synonyms.
      4. Python is a great language for processing linguistic data. There is a built-in split function that separates data into words. The Natural Language Tool Kit in Python contains a lot of functionality for language processing. Python has an advantage with searching text for a word, count number of words, and other functions is built in with the Natural Language Tool Kit [11].
      5. The d3 language is powerful for loading, binding, transforming, and transitioning data. The data drives the visualizations in D3. The downside is D3 can be time consuming with custom visualizations. Easier options are available, but not as customizable and can cost money. Graphs must be visually appealing to the readers’ eye and summarize the data quickly [12].
      6. Visualizations can display data better than the statistics that represent them, making the important data stick out instead of blending in with the rest. Infographics contain minimal data; data visualizations focus on showing the trends via graphs and other tools, making sure to not take away from visualizations with wordy infographics [13].
      7. Determining the “focus” and “topic” of a query is an important language processing goal. The paper [3] details a system of tags and templates to determine topic and focuses of queries including participants, circumstances, and activities. The authors created a set of queries, annotated it, and collected statistics. The framework they created can be used to build and train an effective model determining the focus of a query.
      8. Measuring semantic similarity is an important problem in linguistics using two types of models: knowledge resources such as thesauruses and distributional properties of words. By combining these models together, a more accurate model can be created. Using a combination of page-rank and distributional analysis, an accurate hybrid model was created [4].
      9. Human knowledge can be crowd-sourced to classify words and language, a task once the domain of trained experts. There are two basic categories of crowd-sourcing, model driven and user driven. Model driven crowd-sourcing uses humans to collect data around well-defined tasks. User driven crowdsourcing explores how humans behave as they perform tasks. The article provides use cases and guidelines for interface design for data collection [5].
      10. Article [6] gives an overview of a statistical method for tagging parts of speech for large quantities of text. It operates based on Markov chains which predict the part of speech of a word given its conditional part of speech probability distribution (where it depends on the parts of speech of previous words in the sentence). The disambiguation routine is relatively straightforward, and can also differentiate between homographs. We will likely need to employ a very similar algorithm in our application, so this is a good starting point with a simple implementation.
      11. Article [7] describes a sort of machine learning approach to language processing, specifically part-of-speech tagging, chunking, named entity recognition, and semantic role labeling. The authors assert that avoiding task-specific programs, and instead using an all-inclusive supervised learning approach based on unlabeled training data can produce the most efficient output.
      12. Article [8] gives a broad overview of NLP, specifically the Bayesian/Markov approach to determining the lexical semantics of specific words and their computational costs. The implementation of the statistical models is also discussed, including the implementation of Document Retrieval through Linguistic Knowledge (DR-LINK) interfaces.

References:

[1] Strachan, John, and Terry, Richard. Poetry. Edinburgh: Edinburgh University Press, 2011, Chapter 3, pp. 47-71.

[2] Scheiner, Robert, “Wordflex touch dictionary.” Wordflex.com, http://www.wordflex.com, (Accessed October 1, 2017).

[3] Andersson, Marta, Adnan Ozturel, and Silvia Pareti. "Annotating Topic Development in Information Seeking Queries." In LREC. 2016.

[4] Agirre, Eneko, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. "A study on similarity and relatedness using distributional and wordnet-based approaches." In Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 19-27. Association for Computational Linguistics, 2009.

[5] Chang, Nancy, Russell Lee-Goldman, and Michael Tseng. "Linguistic Wisdom from the Crowd." In Third AAAI Conference on Human Computation and Crowdsourcing. 2016.

[6] Church, Kenneth Ward. "A stochastic parts program and noun phrase parser for unrestricted text." In Proceedings of the second conference on Applied natural language processing, pp. 136-143. Association for Computational Linguistics, 1988.

[7] Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. "Natural language processing (almost) from scratch." Journal of Machine Learning Research 12, no. Aug (2011): 2493-2537.

[8] Chowdhury, Gobinda G. "Natural language processing." Annual review of information science and technology 37, no. 1 (2003): 51-89.

[9] Herman, Ivan, Guy Melançon, and M. Scott Marshall. "Graph visualization and navigation in information visualization: A survey." IEEE Transactions on visualization and computer graphics 6, no. 1 (2000): 24-43.

[10] Byrd, Roy J., and Martin S. Chodorow. "Using an on-line dictionary to find rhyming words and pronunciations for unknown words." In Proceedings of the 23rd annual meeting on Association for Computational Linguistics, pp. 277-283. Association for Computational Linguistics, 1985.

[11] Bird, Steven, Ewan Klein, and Edward Loper. Natural language processing with Python: analyzing text with the natural language toolkit. " O'Reilly Media, Inc.", 2009. pp.1-21

[12] Murray, Scott. Interactive Data Visualization for the Web: An Introduction to Designing with. " O'Reilly Media, Inc.", 2017.

[13] Borup, Rick. “Data Visualization for the Database Developer.” Paper presented at the Southwest Fox conference, Gilbert, AZ, October 2015.