Deep Learning - Project Proposal

Brendon Boldt

Sep/25/2017

1 Proposal

For my semester project, I plan to take the Voxelurn natural language system and making more robust using vector representation of words [2] [1]. Voxelurn, at its core, starts with a "core language" (which is itself a formal language) which is able to describe basic block-building actions (e.g., add, remove, select) and allows users to extend the language as they use it. This extension consists of allowing users to define new commands in terms of already known commands; the newly defined commands are added to the grammar of the formal language. The purpose of this extension is that language becomes more natural as people use the system.

The words in formal language are simply represented as individual tokens. In this way, a command such as "make red block" might be recognized while something like "build red block" would not be recognized at all. Using vector representation of words would allow the language system to be more robust since words with sufficiently similar vectors could be used in place of each other. Thus, words like "make" and "build" could be used in place of each other without each command needing to be explicitly learned. Normally the right-hand-side of formal language production rules consists of strings of symbolic tokens; this approach, instead, would turn the right-had-side of production rules into sequences of vectors in order to improve the flexibility of the language as a whole.

2 Tools

A demo and explanation of Voxelurn (as well as a link to the paper) can be found at `http://www.voxelurn.com/#/about`. The formal language and parsing aspect of the project is based on the SEMPRE project from Stanford University. Much of the Voxelurn project will be left as-is; though the parser will have to be modified in order to support vector representations of words. The word vectorization will be based on the TensorFlow machine learning library and be derived from the tutorial found at `https://www.tensorflow.org/tutorials/word2vec`.

3 Datasets

While I have not completely decided on particular datasets, I hope to largely use datasets that do not require extensive preprocessing. For the word vectorization, it would be ideal if a standard English corpus (such as the Penn Treebank) could provide the necessary vectorization of words. The Voxelurn project also contains sample datasets of user-entered commands; these could possibly be augmented by replacing words with dictionary synonyms in order to test the word vectorization's effectiveness.

References

[1] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[2] Sida I. Wang, Samuel Ginn, Percy Liang, and Christopher D. Manning. Naturalizing a programming language via interactive learning. *CoRR*, abs/1704.06956, 2017.