Deep Learning Software

School of Information Studies
Syracuse University

How can we use Deep Learning in NLP?

Code available for sentiment and relation analysis: www.socher.org

Tools for word representations

- word2vec, available from Google in C
- gensim (python), open source by Radim Rehurek
- Deeplearning4j (java)

Use word vectors as features in current classifiers

- doc2vec

Google trained word model – trained on 100B words news data resulting in 3M phrases with layer size 300

Program Neural Nets for your own tasks

- TensorFlow, available from Google

gensim packages Word2vec and doc2vec

From RaRe technologies, Radim Rehurek

- https://rare-technologies.com/word2vec-tutorial/

Word2vec: Create a generator to get sentences as lists of tokens

- Do whatever preprocessing and tokenizing you need
- Create a model with Word2vec
- Evaluate analogies with Google's file: questions-words.txt
- Other functions show similar, and dis-similar, words

Doc2vec:

- In order to use word dense vectors as features in a current classifier, use RNN to combine word vectors for entire paragraph, sentence or (short) document

TensorFlow

Framework from Google for scalable ML

- Framework of reusable components

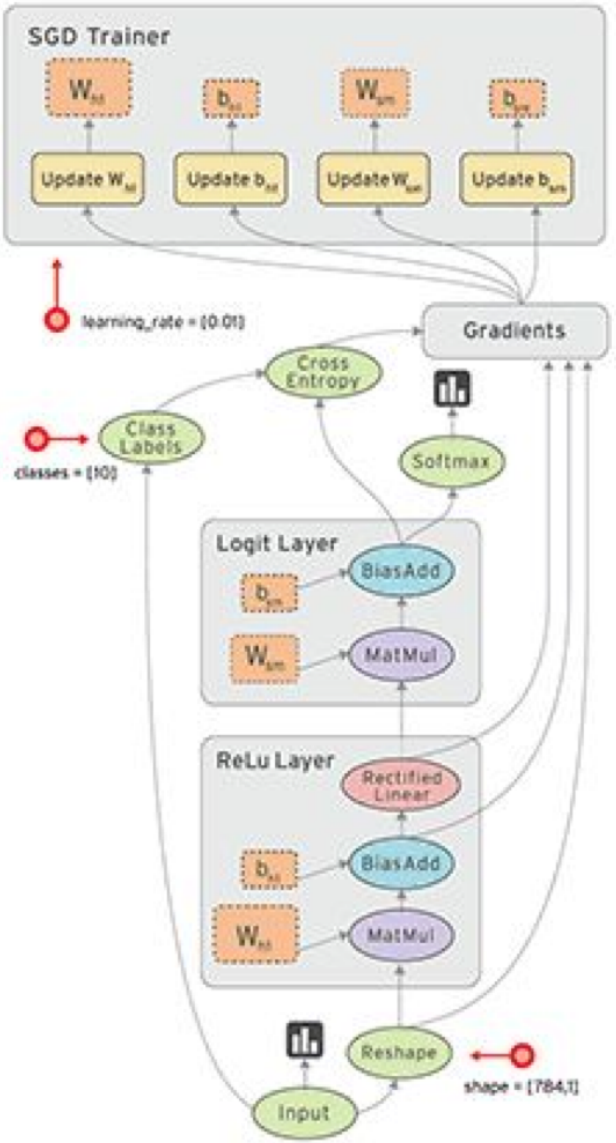
Define computation as a graph

- Graph is defined with python fns
- Compiled, optimized, executed
- Nodes represent computations
- Data (tensors) flow along edges

Manage distributed, heterogeneous systems

But must program details of neural net classifiers

Higher level packages in progress



TensorFlow slides from Martin Wicke, ACM webinar 2016

What's available in TensorFlow now

Tutorials on tensorflow.org:

- Image recognition (convolutional NN):
  - https://www.tensorflow.org/tutorials/image_recognition
- Word embeddings:
  - https://www.tensorflow.org/versions/word2vec
- Language Modeling:
  - https://www.tensorflow.org/tutorials/recurrent
- Translation:
  - https://www.tensorflow.org/versions/seq2seq

Lynda tutorial on deep learning

- All SU ids can access a Lynda account

Conclusion

Deep learning is promising area for NLP

- Learning features of classification from unlabeled data
- But may not totally free us from designing manual features and using labeled data to make NLP representations

Lots of research on what are the right NN algorithms and text representations

Future:

- More software packages to make it easier to apply DL to your own NLP task
- Possibilities for improvements in some of the harder tasks of NLP