Project Proposal

General-Purpose Sentence Representation Learning

Team members: Fanglin Chen, Mi Fang, Wenqing Zhu

1. Main Objective

Implement two different models for encoding sentences, SkipThought and DiscSent; Evaluate the sentence embeddings on different tasks such as classification and semantic analysis; Compare these two models' performance.

2. Dataset

For SkipThought, we train an encoder-decoder model that tries to reconstruct the surrounding sentences of a passage. Sentences that share semantic and syntactic properties are thus mapped to similar vector representations. We evaluate the capability of our encoder as a generic feature extractor after training on the BookCorpus dataset.

For DiscSent, the models are trained on a combination of data from BookCorpus, the Gutenberg project, and Wikipedia. After sentence and word tokenization and lower-casing, identify all paragraphs longer than 8 sentences and extract a NEXT example, pairs of sentences for ORDER and CONJUNCTION from each.

3. Model

SkipThought: We treat skip-thoughts in the framework of encoder-decoder models. An encoder maps words to a sentence vector and a decoder is used to generate the surrounding sentences. In our model, we use an RNN encoder with GRU activations and an RNN decoder with a conditional GRU. GRU has been shown to perform as well as LSTM on sequence modelling tasks while being conceptually simpler.

DiscSent: In this model, we propose three objective functions for use over paragraphs extracted from unlabeled text to train a single encoder in order to find the coherence relations. Three functions are: (a) Binary ordering of sentences (ORDER). (b) Next sentence (NEXT). (c)Conjunction prediction (CONJUNCTION)

Three encoding models: (a) A simple 1024D sum-of-Words (CBOW) encoding. (b) A 1024D GRU recurrent neural network. (c) A 512D bidirectional GRU RNN

4. Performance Evaluation

Use SentEval for evaluating the quality of sentence embeddings. Implement "batcher" and "prepare" functions in SentEval to preprocess and convert sentences into sentence embeddings, then SentEval can use these embeddings as features in different tasks to see the performance.

comment/feedback:

SB: This is a reasonable project topic, but I expect that you'll have speed issues with SkipThought. The standard version of the model, trained on Toronto Books, takes about two weeks to converge with one okay GPU. So, unless you have access to multiple good GPUs, you likely won't have time for any trial and error. I'd advise against just training both models for a shorter time: It's more important to know which one produces better representations than to know which one is faster to train. In real applications, you usually only need to train these once, so it's fine if it takes a long time.

To speed these up, consider either using much smaller representations for both models (100–300D) or switching to a smaller and more restricted source of text for training. Both of these will limit what we can learn, but you may have no better option. Feel free to stop by if you have any questions. You might also consider replacing SkipThought with a faster model, like FastSent from Hill et al.: https://github.com/fh295/SentenceRepresentation

KC: i agree with SB's concern on the computational overhead especially with skip-thought. some simplification should be sought after.