Learning to Generate Reviews and Discovering Sentiment

1 Idea

Given sufficient representation capacity, the neurons in a network can disentangle higher level features like sentiment from an otherwise incoherent latent representation. Authors identify a neuron that encapsulates almost all of the sentiment attribute of a body of text, the value of which can be tuned to modify the sentiment of generative functions conditioned on the representation.

2 Method

- Character level language modelling is used for benchmarking, trained on the Amazon product review dataset (82 million product reviews).

- Neural network architecture used comprised of a single layer multiplicative LSTM 4096 units wide. Training comprised of 1 epoch over 128-batch subsequences of length 256, with 1 million weight updates. Time taking across 4 GPUs was 1 month.

- Model states are initialized to zeros. Tanh is applied to bound values between -1 and 1.

- For each byte, the model updates its hidden state and predicts a probability distribution over the next possible byte.

- Logistic regression classifier trained on the latent representation for different NLP tasks (relatedness, classification, paraphrasing). L1 penalty for text classification used instead of L2. (Claim is that this performed better with lesser data)

3 Observations

- Outperforms state-of-the art sentiment analysis systems, doesn't perform as well on subjective-objective and opinion polarity tasks. Sets a new baseline for movie reviews in the Stanford Sentiment Treebank.
- L1 regularization is known to reduce sample complexity when there are many irrelevant features or outliers, since it is less sensitive to them.
- A single unit within the mLSTM directly corresponds to most of the sentiment classification. The sentiment unit achieves close to state of the art sentiment classification results and the improvement on adding the remaining 4095 units of the mLSTM representation is minor.
- Since the model is trained on Amazon reviews, it didn't generalize well to Yelp-style document reviews commenting about hospitality, location and ambience.
- The model performs reasonably well on paraphrase detection, but not on semantic relatedness.
- A couple things remain unclear
  - Why does the sentiment get captured in such a predictable manner? Could be some property of the mLSTM. Previous work by Karpathy shows that certain units are activated for different syntactic requirements in code.
  - Cross-domain training still seems unrealistic.