Project 2: Sentence VAE

May 3, 2019

This project will help you learn and implement a deep generative language model. In summary, your task is to:

- Pre-process the training, validation, and test data;
- Implement the models as described below;
- Optionally implement one or more of the `extra`'s;
- Evaluate your model on the test data;
- Perform a qualitative analysis;
- Write a report on the entire process.

1 Deterministic Language Model

Implement a deterministic language model, using for example the following architecture:

- At each step, an RNNLM parameterises a categorical distribution over the vocabulary of English words V_x, i.e.
  $X_i|x_{<i} \sim \mathrm{Cat}\left(f\left(x_{<i};\theta\right)\right)$ and

$$\begin{aligned} f\left(x_{<i};\theta\right) &= \mathrm{softmax}\left(\mathbf{s}_i\right) \\ \mathbf{e}_i &= \mathrm{emb}\left(x_i;\theta_{\mathrm{emb}}\right) \\ \mathbf{h}_i &= \mathrm{RNN}\left(\mathbf{h}_{i-1},\mathbf{e}_{i-1};\theta_{\mathrm{RNN}}\right) \\ \mathbf{s}_i &= \mathrm{affine}\left(\mathbf{h}_i;\theta_{\mathrm{out}}\right) \end{aligned} \tag{1}$$

- Embedding layer (emb), for the RNN one (or more) GRU or LSTM cell(s), and an affine layer to map from the dimensionality of the RNN to the vocabulary size.

This will be our Baseline.

2 Sentence VAE

Implement a deep generative language model, using :

- Generation of one word at a time without Markov assumptions, but $f(\cdot)$ conditions on z in addition to the observed prefix/history x.
- Gaussian Sen VAE parametrises a categorical distribution over the vocabulary for each given prefix, and, it conditions on a latent embedding based on the following generative story:
  $Z \sim \mathcal{N}(0, I)$,
  $X_i | z, x_{<i} \sim \operatorname{Cat}\left(f\left(z, x_{<i}; \theta\right)\right)$

  and architecture

$$\begin{aligned} f\left(z, x_{<i}; \theta\right) &= \operatorname{softmax}\left(\mathbf{s}_i\right) \\ \mathbf{e}_i &= \operatorname{emb}\left(x_i; \theta_{\text{emb}}\right) \\ \mathbf{h}_0 &= \tanh\left(\operatorname{affine}\left(z; \theta_{\text{ init }}\right)\right) \\ \mathbf{h}_i &= \operatorname{RNN}\left(\mathbf{h}_{i-1}, \mathbf{e}_{i-1}; \theta_{\text{RNN}}\right) \\ \mathbf{s}_i &= \operatorname{affine}\left(\mathbf{h}_i; \theta_{\text{out}}\right) \end{aligned} \tag{2}$$

An example of inference Network:

$$\begin{aligned} \mathbf{e}_i &= \operatorname{emb}\left(x_i; \phi_{\text{emb}}\right) \\ \mathbf{f}_i &= \operatorname{RNN}\left(\mathbf{f}_{i-1}, \mathbf{e}_i; \phi_{\text{fwd}}\right) \\ \mathbf{b}_i &= \operatorname{RNN}\left(\mathbf{b}_{i+1}, \mathbf{e}_i; \phi_{\text{bwd}}\right) \\ h &= \operatorname{dense}([f_n; b_1]; \phi_{hid}) \\ \mu &= \operatorname{dense}(h; \phi_{loc}) \\ \sigma &= \operatorname{softplus}(\operatorname{dense}(h; \phi_{scale})) \\ z &= \mu + \sigma \odot \epsilon \end{aligned} \tag{3}$$

- with $\epsilon \sim \mathcal{N}(0, I)$

In addition, implement KL annealing and Free-bits. Use word dropout, and as an extra use variational dropout in the RNN.

3 Metrics

Evaluate your model using the following metrics:

- Negative log-likelihood (NLL) is denoted by the assumption that the dataset consists of N sequences of length M:
  $-\log p(x) = -\frac{1}{N} \sum_i^N \sum_j^{M_i} \log p\left(x_{ij} | x_{i,<j}\right)$

- In variational models the NLL is not available available and can be approximated with importance sampling:
$-\log p(x) \approx -\frac{1}{N}\sum_{i=1}^{N} \log\left(\frac{1}{S}\sum_{k}^{S}\left[\frac{p(z_{ik},x_i)}{q(z_{ik}|x_i)}\right]\right) \quad z_{ik} \sim q\left(z_i|x_i\right)$

- For latent variable models we report the negative ELBO:
$-\mathrm{L}(\theta,\phi,x) = \frac{1}{N}\sum_{i=1}^{N}\left(\mathrm{KL}(q\left(z_i|x_i\right)\|p\left(z_i\right)) - \mathbb{E}_{q(z_i|x_i)}\left[\sum_{j=1}^{M_i}\log p\left(x_{ij}|z_i\right)\right]\right)$

- Perplexity (PPL) is defined as the inverse of the geometric mean per-word likelihood. It can be computed as the exponent of average per-word entropy, given N sequences:
$\mathrm{PPL} = \exp\left(\frac{\sum_{i=1}^{N}\log p(x_i)}{\sum_{i=1}^{N}|x_i|}\right)$
where $|x_i|$ is the length of the i-th sentence in the dataset, including the end of sentence token but excluding the start of sentence token. Note that in the variational models an approximation of the perplexity based on the importance sampled negative log-likelihood.

- Accuracy is defined as the fraction of correctly predicted words with greedy decoding from the models. For variational models, this quantity is computed by decoding from the approximate posterior mean mu.

4 Qualitative Analysis

- Sample 10 sentences from the Sentence VAE models are produced by greedy decoding from a sample z from the prior distribution.

- Test the reconstruction of sentences from the test dataset using the approximate posterior mean and 10 samples.

- Test homotopy by interpolating two sentences in the test set and different α parameters.
decode from $z_\alpha = \alpha * z_1 + (1-\alpha) * z_2$, where z_1 is the encoding of the first sentence and z_2 is the encoding of the second sentence.

5 Data

All relevant data (including details about file formats) are available from https://uva-slpl.github.io/nlp2/projects.html.

In this project, you will work with the Penn Treebank (PTB) dataset, that consists of English sentences. The training data is *02-21.10way.clean*, the validation data is *22.auto.clean*, and the test data is *23.auto.clean*. Note that the sentences in the PTB dataset are annotated with syntactic trees.

We are making available *training* data (which you can use to perform parameter estimation), *validation* data (which you can use to debug your implementation as well as to perform model selection), and finally in due time *test* data (which you will use to conduct your final empirical comparison).

6 Report

You should use LaTeX for your report, and you should use the ACL template available from http://acl2017.org/downloads/acl17-latex.zip.

We expect reports (5 pages plus references). The typical submission is organised as follows:

- **Abstract**: conveys scope and contributions;
- **Introduction**: present the problem and relevant background;
- **Model**: technical description of models;
- **Experiments**: details about the data, experimental setup, findings, and qualitative analysis;
- **Conclusion**: a critical take on contributions and limitations.

7 Submission

You should submit a `.tgz` file containing a folder (folder name `lastname1.lastname2.lastname3`) with the report as a single `pdf` file. Your report may contain a link to an open-source repository (such as github), but please do not attach code or additional data to your submission. You can complete your project submission on Canvas.

8 Assessment

Your report will be assessed according to the following evaluation criteria:

Implementation

- Deterministic LM
- Sentence VAE
- KL annealing and Free-bits
- Importance Sampling for NLL
- Word dropout
- **Extra**: Variational dropout for RNN

Report

- **Scope** (max 2 points): Is the problem well presented? Do students understand the challenges/contributions?
- **Theoretical description** (max 3 points): Are the models presented clearly and correctly?
- **Empirical evaluation** (max 3 points): Is the experimental setup sound/convincing? Are experimental findings presented in an organised and effective manner?
- **Writing style** (max 2 points): use of LaTeX, structure of report, use of tables/figures/plots, command of English.

9 Resources

A non-exaustive list of resources that might help you along the way:

1.