“fixed-length distributed sentence representations” – what is **distributed**?

“The decoder is auto-regressive” – what is **auto-regressive**?

Can cast current framework into seq2seq, see tensorflow seq2seq tutorials

Not a lot of hyperparameter search

Combine all data files/source files together

Dropout – can try other methods ALSO

Save optimizer state also

Note – tf.logging does not print time (do it yourself)

**Question – how to take sentence representation? Last layer? Average over layers?**

**Datasets:**

1. NLI:
   1. Collection of about 1 million sentence pairs from SNLI and MultiNLI corpora
   2. **@vineetm said: allnli** 1 million sentences (seems OK!)
2. Constituency parsing:
   1. Work with Noisy Parses
3. NMT on En-De:
   1. 4.5 million En-De sentence pairs from WMT15
      1. nmt.de-en.de, nmt.de-en.en, (see tokenized ones ALSO)
      2. **Question –** Some sentences WRONG – german file has English sentences?

**Tentative file structure:**

1. Preprocessing script(s) that write tokenized data to new file
   1. Study data, find datasets in downloaded data
      1. NLI
         1. Very simple – each line has two sentences, and a class at the end. Sentences separated by period (**check this**)
      2. Constituency parsing – **get a parser for this!**
      3. NMT on En-De
         1. Two files with same number of lines, corresponding lines are the translation
         2. **IMPORTANT –** files seem to be wron0067
   2. A
   3. A
2. File for pipeline/data reading methods
   1. Create iterable dataset objects, with batch and padding specifications, and return them
3. Models file
   1. NOTE – decoder, every step is conditioned on encoder hidden representation!
   2. NLI:
      1. Given **premise** and **hypothesis** sentence, classify relationship between them as **entailment, contradiction** or **neutral**.
      2. Encode **p** and **h** as vectors *u* and *v*.
      3. Concatenate them as [u;v|u-v|u\*v] (Hadamard product)
      4. Feed this into an MLP for 3-way classification – Single layer MLP with dropout of 0.3 (try removing droput, see what happens)
   3. Constituency parsing:
      1. Input is sentence, output is linearized parse tree
   4. NMT
   5. a
   6. a
   7. a
   8. a
   9. a
4. Main file
5. Evaluation file (load from saved model)
6. SentEval tasks script(s)

**Evaluation tasks:**

1. Done using *SentEval* repo
2. Tasks:
   1. Text classification:
      1. MR
      2. CR
      3. SST
      4. TREC
      5. SUBJ
      6. MPQA
      7. Evaluation metric – classification accuracy
   2. Paraphrase identification:
      1. MRPC
      2. Metric – accuracy, F1
   3. Entailment and semantic relatedness:
      1. SICK relatedness (SICK-R)
         1. Evaluation metric – Pearson correlation
      2. Entailment labels, SICK-E
         1. Evaluation metric – Classification accuracy
   4. Semantic textual similarity:
      1. Measure relatedness of sentences using only cosine similarity
      2. STS benchmark tasks (STS12, STS13, STS14, STS15, STS16, STSB)
      3. Evaluation – Pearson correlation
3. A
4. A
5. A
6. A