Student Name: **Qiaoyang ZHONG**

NetID: **24112456g**

Q1

Yes.

- Preliminaries:

  Preparing for the training data: We tag each token with `B, M, E, S`, which means `Begin of word, Mid of word, End of word, Single character word` respectively. For example,

  `仰天大笑出门去` -> `B M M E B E S`

  The sentence with annotation is supervised data.

- RNN Training:

  Input is sentence like `仰天大笑出门去` , and **output** is annotation like `B M M E B E S`, and we train one character in one recurrent step.

- BERT Training:

  Input is sentence like `仰天大笑出门去` , and **output** is annotation like `B M M E B E S`, and we can finetune the bert model applying annotated training data. Ignore `<cls>` token.

Q2

Yes

- CRF

$$P(y_i|X) = \frac{1}{Z} exp(\sum_{t=1}^{T}\sum_{k=1}^{K} \lambda_t f_{trans}(y_{t-1}, y_t) + \mu_t f_{state}(y_t, X))$$

  Input: y_t is the t-th labeling (that is `B M E S`), X is the seqeunce, f_trans is transition feature function, f_state is the state feature function (since applying `X` directly may cause sparsity, conservational method is using hand-craft features, like word

length, word entropy, prefix-suffix, etc), λ_t, μ_t is learnable parameters.

- SVM

Input: We split the sentence into a batch of words, and we choose the true word (ground truth) as *positive input*, and choose the random combination word as *negative word*.

The word features can be generated as hand-craft features above.

Process: We can apply SVM to classify whether the word is the real word. And we apply greedy method to segment the word sequence.

Q3

Yes

- First method

  Encoder input: the sequence like `仰天大笑出门去`

  Decoder output: the annotation like `B M M E B E S`

- Second method

  Encoder input: the sequence like `仰天大笑出门去`

  Decoder output: sequence like `<cls>仰天大笑<sep>出门<sep>去<eos>`

  And in this situation, the sequence length is changed.