HW0: Sample Assignment

Alexander Rush
srush@seas.harvard.edu

Sam Wiseman
swiseman@seas.harvard.edu

January 17, 2016

1 Introduction

This note lays out our expectation for a homework submission in *CS287: Statistical Natural Language Processing*. While you do not have to follow this template to the letter, we do expect that write-ups have a very clear structure and to cover all the elements described in this note. With this in mind, the burden is on the presenter to demonstrate why deviations from the standard are necessary.

All write-ups should include a short introduction. In this section you should summarize the underlying problem in high-level language and describe the extensions that you have decided to propose in your implementation. When you describe these extensions you should carefully cite the papers of interest. For instance, it will often be useful to cite the work seen in class (Murphy, 2012). Alternatively, you can also cite papers inline, for instance the work of Berger et al. (1996).

2 Problem Description

In general homeworks will be given using precise but informal language. As part of the assignment, we expect you to write-out a definition of the problem and your model in formal language. For this class, we will use the following notation:

- $\boldsymbol{b}, \boldsymbol{m}$; bold letters for vectors.
- $\boldsymbol{B}, \boldsymbol{M}$; bold capital letters for matrices.
- $\mathcal{B}, \mathcal{M}$; script-case for sets.
- B, M; capital letters for constants and random variables.
- b_i, x_i; lower case for scalars or indexing into vectors.

In natural language processing, it is also very common to use discrete sets like $\mathcal{V}$ the vocabulary of the language, or $\mathcal{T}$ a tag set of the language. We might also want one-hot vectors representing words. These will be of the type $\boldsymbol{v} \in \{0,1\}^{|\mathcal{V}|}$. In a note, it is crucial to define the types of all variables that are introduced. The background is the right place to do this. NLP is also full of sequences. For instance sentences, $w_1, \dots, w_N$, where here N is a constant length and $w_i \in \mathcal{V}$ for

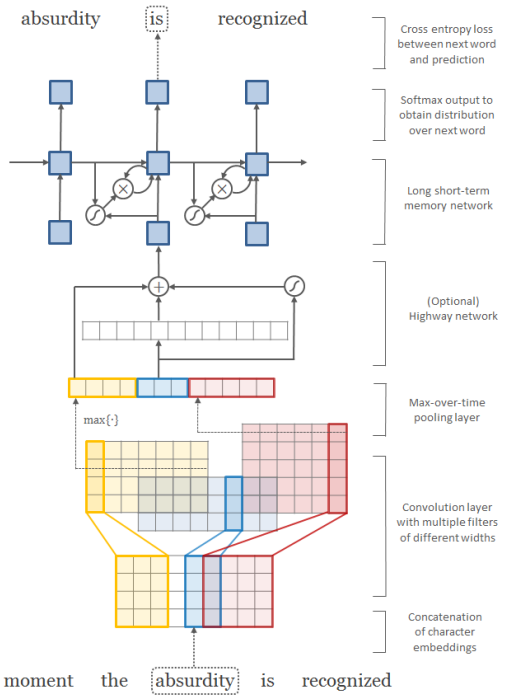

Figure 1: Sample network description.

all $i \in \{1, \ldots N\}$. If we pretend sentences are all the same length, we can have scoring function over sentences, $s : \mathcal{V}^N \mapsto \mathbb{R}$. One might be defined as:

$$s(w_1, \ldots, w_N) = \sum_{i=1}^{N} p(w_i | w_{i-2}, w_{i-1}),$$

where p is the bigram probability, which we will cover later in the class.

3 Model and Algorithms

Finally we get to the model itself. This section should formally describe the model used to solve the task proposed in the previous section. This section should try to avoid introducing new vocabulary or notation, when possible use the notation from the previous section.

This section is also a great place to include other material that describes the underlying structure and choices of your model, for instance here are some example tables and algorithms from full research papers:

- diagrams of your model, see Figure 1.
- feature tables, see Table 1.
- pseudo-code, see Algorithm 1.

Mention Features	
Feature	Value Set
Mention Head	$\mathcal{V}$
Mention First Word	$\mathcal{V}$
Mention Last Word	$\mathcal{V}$
Word Preceding Mention	$\mathcal{V}$
Word Following Mention	$\mathcal{V}$
# Words in Mention	$\{1, 2, \ldots\}$
Mention Type	$\mathcal{T}$

Table 1: Sample feature table.

Algorithm 1 Sample pseudo-code algorithm.

procedure LINEARIZE($x_1 \ldots x_N$, K, g)
  $B_0 \leftarrow \langle(\langle\rangle, \{1, \ldots, N\}, 0, \boldsymbol{h}_0, \mathbf{0})\rangle$
  for $m = 0, \ldots, M-1$ **do**
    for $k = 1, \ldots, |B_m|$ **do**
      for $i \in \mathcal{R}$ **do**
        $(y, \mathcal{R}, s, \boldsymbol{h}) \leftarrow \text{copy}(B_m^{(k)})$
        for word w in phrase x_i **do**
          $y \leftarrow y$ append w
          $s \leftarrow s + \log q(w, \boldsymbol{h})$
          $\boldsymbol{h} \leftarrow \delta(w, \boldsymbol{h})$
        $B_{m+|w_i|} \leftarrow B_{m+|w_i|} + (y, \mathcal{R} - i, s, \boldsymbol{h})$
        keep top-K of $B_{m+|w_i|}$ by $f(x, y) + g(\mathcal{R})$
  return $B_M^{(k)}$

4 Experiments

Finally we end with the experimental section. Each assignment will make clear the main experiments and baselines that you should run. For these experiments you should present a main results table. Here we give a sample Table 2. In addition to these results you should describe in words what the table shows and the relative performance of the models.

Besides the main results we will also ask you to present other results comparing particular aspects of the models. For instance, for word embedding experiments, we may ask you to show a chart of the project word vectors. This will lead to something like Figure 2. This should also be described within the body of the text itself.

5 Conclusion

End the write-up with a very short recap of the main experiments and the main results. Describe any challenges you may have faced, and what would have been improved in the model.

Model	Acc.
BASELINE 1	0.45
BASELINE 2	2.59
MODEL 1	10.59
MODEL 2	13.42
MODEL 3	7.49

Table 2: Table with the main results.

Figure 2: Sample qualitative chart.

References

Berger, A. L., Pietra, V. J. D., and Pietra, S. A. D. (1996). A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1):39–71.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.