PRML Assignment 3 Report

Part 1

> $$\begin{aligned}\mathbf{z} &= [\mathbf{h}_{t-1}, \mathbf{x}_t] \\ \mathbf{f}_t &= \sigma(W_f * \mathbf{z} + b_f) \\ &= \sigma(W_{fh} * \mathbf{h}_{t-1} + W_{fx} * \mathbf{x}_t + b_f) = \sigma(\mathbf{net}_f) \\ \mathbf{i}_t &= \sigma(W_i * \mathbf{z} + b_i) \\ &= \sigma(W_{ih} * \mathbf{h}_{t-1} + W_{ix} * \mathbf{x}_t + b_f) = \sigma(\mathbf{net}_i) \\ \widetilde{\mathbf{c}_t} &= tanh(W_c * \mathbf{z} + b_c) \\ &= tanh(W_{ch} * \mathbf{h}_{t-1} + W_{cx} * \mathbf{x}_t + b_f) = \sigma(\mathbf{net}_{\widetilde{c}}) \\ \mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \widetilde{\mathbf{c}_t} \\ \mathbf{o}_t &= \sigma(W_o * \mathbf{z} + b_o) \\ &= \sigma(W_{oh} * \mathbf{h}_{t-1} + W_{ox} * \mathbf{x}_t + b_f) = \sigma(\mathbf{net}_o) \\ \mathbf{h}_t &= \mathbf{o}_t \odot tanh(\mathbf{c}_t)\end{aligned}$$

Differentiate one step of LSTM

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial \mathbf{o}_t} &= \frac{\partial(\mathbf{o}_t \odot tanh(\mathbf{c}_t))}{\partial \mathbf{o}_t} \\ &= diag[tanh(\mathbf{c}_t)]\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} &= \frac{\partial(\mathbf{o}_t \odot tanh(\mathbf{c}_t))}{\partial \mathbf{c}_t} \\ &= diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t))]\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial \mathbf{f}_t} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} * \frac{\partial \mathbf{c}_t}{\partial \mathbf{f}_t} \\ &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} * diag[\mathbf{c}_{t-1}] \\ &= diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1}]\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial \mathbf{i}_t} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} * \frac{\partial \mathbf{c}_t}{\partial \mathbf{i}_t} \\ &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} * diag[\widetilde{\mathbf{c}_t}] \\ &= diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}_t}]\end{aligned}$$

$$\begin{aligned}
\frac{\partial \mathbf{h}_t}{\partial \widetilde{\mathbf{c}_t}} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} * \frac{\partial \mathbf{c}_t}{\partial \widetilde{\mathbf{c}_t}} \\
&= \frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} * diag[\mathbf{i}_t] \\
&= diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t]
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_{t-1}} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} * \frac{\partial \mathbf{c}_t}{\partial \mathbf{c}_{t-1}} \\
&= \frac{\partial \mathbf{h}_t}{\partial \mathbf{c}_t} * diag[\mathbf{f}_t] \\
&= diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{f}_t]
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-1}} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_o} * \frac{\partial \mathbf{net}_o}{\partial \mathbf{h}_{t-1}} + \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_f} * \frac{\partial \mathbf{net}_f}{\partial \mathbf{h}_{t-1}} \\
&\qquad + \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_i} * \frac{\partial \mathbf{net}_i}{\partial \mathbf{h}_{t-1}} + \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_{\widetilde{c}}} * \frac{\partial \mathbf{net}_{\widetilde{c}}}{\partial \mathbf{h}_{t-1}} \\
&= diag[tanh(\mathbf{c}_t)] * \frac{\partial \mathbf{o}_t}{\partial \mathbf{net}_o} * \frac{\partial \mathbf{net}_o}{\partial \mathbf{h}_{t-1}} \\
&+ diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1}] * \frac{\partial \mathbf{f}_t}{\partial \mathbf{net}_f} * \frac{\partial \mathbf{net}_f}{\partial \mathbf{h}_{t-1}} \\
&+ diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}_t}] * \frac{\partial \mathbf{i}_t}{\partial \mathbf{net}_i} * \frac{\partial \mathbf{net}_i}{\partial \mathbf{h}_{t-1}} \\
&+ diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t] * \frac{\partial \widetilde{\mathbf{c}}_t}{\partial \mathbf{net}_{\widetilde{c}}} * \frac{\partial \mathbf{net}_{\widetilde{c}}}{\partial \mathbf{h}_{t-1}} \\
&= diag[tanh(\mathbf{c}_t) \odot \mathbf{o}_t \odot (1 - \mathbf{o}_t)] * \frac{\partial \mathbf{net}_o}{\partial \mathbf{h}_{t-1}} \\
&+ diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1} \odot \mathbf{f}_t \odot (1 - \mathbf{f}_t)] * \frac{\partial \mathbf{net}_f}{\partial \mathbf{h}_{t-1}} \\
&+ diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}_t} \odot \mathbf{i}_t \odot (1 - \mathbf{i}_t)] * \frac{\partial \mathbf{net}_i}{\partial \mathbf{h}_{t-1}} \\
&+ diag[\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t \odot (1 - \widetilde{\mathbf{c}}_t^2)] * \frac{\partial \mathbf{net}_{\widetilde{c}}}{\partial \mathbf{h}_{t-1}} \\
&= tanh(\mathbf{c}_t) \odot \mathbf{o}_t \odot (1 - \mathbf{o}_t) \odot W_{oh} \\
&+ \mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1} \odot \mathbf{f}_t \odot (1 - \mathbf{f}_t) \odot W_{fh} \\
&+ \mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}_t} \odot \mathbf{i}_t \odot (1 - \mathbf{i}_t) \odot W_{ih} \\
&+ \mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t \odot (1 - \widetilde{\mathbf{c}}_t^2) \odot W_{ch}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \mathbf{h}_t}{\partial \mathbf{x}_t} &= tanh(\mathbf{c}_t) \odot \mathbf{o}_t \odot (1 - \mathbf{o}_t) \odot W_{ox} \\
&+ \mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1} \odot \mathbf{f}_t \odot (1 - \mathbf{f}_t) \odot W_{fx} \\
&+ \mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}_t} \odot \mathbf{i}_t \odot (1 - \mathbf{i}_t) \odot W_{ix} \\
&+ \mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t \odot (1 - \widetilde{\mathbf{c}}_t^2) \odot W_{cx}
\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial W_o} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_o} * \frac{\partial \mathbf{net}_o}{\partial W_o}\\ &= (tanh(\mathbf{c}_t) \odot \mathbf{o}_t \odot (1-\mathbf{o}_t)) * \frac{\partial \mathbf{net}_o}{\partial W_o}\\ &= (tanh(\mathbf{c}_t) \odot \mathbf{o}_t \odot (1-\mathbf{o}_t)) * \mathbf{z}^T\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial W_f} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_f} * \frac{\partial \mathbf{net}_f}{\partial W_f}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1} \odot \mathbf{f}_t \odot (1-\mathbf{f}_t)) * \frac{\partial \mathbf{net}_f}{\partial W_f}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1} \odot \mathbf{f}_t \odot (1-\mathbf{f}_t)) * \mathbf{z}^T\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial W_i} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_i} * \frac{\partial \mathbf{net}_i}{\partial W_i}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}}_t \odot \mathbf{i}_t \odot (1-\mathbf{i}_t)) * \frac{\partial \mathbf{net}_i}{\partial W_i}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}}_t \odot \mathbf{i}_t \odot (1-\mathbf{i}_t)) * \mathbf{z}^T\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial W_c} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_c} * \frac{\partial \mathbf{net}_c}{\partial W_c}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t \odot (1-\widetilde{\mathbf{c}}_t^2)) * \frac{\partial \mathbf{net}_c}{\partial W_c}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t \odot (1-\widetilde{\mathbf{c}}_t^2)) * \mathbf{z}^T\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial \mathbf{b}_o} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_o} * \frac{\partial \mathbf{net}_o}{\partial \mathbf{b}_o}\\ &= (tanh(\mathbf{c}_t) \odot \mathbf{o}_t \odot (1-\mathbf{o}_t)) * \frac{\partial \mathbf{net}_o}{\partial \mathbf{b}_o}\\ &= (tanh(\mathbf{c}_t) \odot \mathbf{o}_t \odot (1-\mathbf{o}_t))\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial \mathbf{b}_f} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_f} * \frac{\partial \mathbf{net}_f}{\partial \mathbf{b}_f}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1} \odot \mathbf{f}_t \odot (1-\mathbf{f}_t)) * \frac{\partial \mathbf{net}_f}{\partial \mathbf{b}_f}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \mathbf{c}_{t-1} \odot \mathbf{f}_t \odot (1-\mathbf{f}_t))\end{aligned}$$

$$\begin{aligned}\frac{\partial \mathbf{h}_t}{\partial \mathbf{b}_i} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_i} * \frac{\partial \mathbf{net}_i}{\partial \mathbf{b}_i}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}}_t \odot \mathbf{i}_t \odot (1-\mathbf{i}_t)) * \frac{\partial \mathbf{net}_i}{\partial \mathbf{b}_i}\\ &= (\mathbf{o}_t \odot (1-tanh^2(\mathbf{c}_t)) \odot \widetilde{\mathbf{c}}_t \odot \mathbf{i}_t \odot (1-\mathbf{i}_t))\end{aligned}$$

$$\frac{\partial \mathbf{h}_t}{\partial \mathbf{b}_c} = \frac{\partial \mathbf{h}_t}{\partial \mathbf{net}_c} * \frac{\partial \mathbf{net}_c}{\partial \mathbf{b}_c}$$

$$= (\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t \odot (1 - \widetilde{\mathbf{c}}_t^2)) * \frac{\partial \mathbf{net}_c}{\partial \mathbf{b}_c}$$
$$= (\mathbf{o}_t \odot (1 - tanh^2(\mathbf{c}_t)) \odot \mathbf{i}_t \odot (1 - \widetilde{\mathbf{c}}_t^2))$$

Differentiate through time

> 对某个时刻 k 求导

$$\frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_k} = \frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-1}} * \frac{\partial \mathbf{h}_{t-1}}{\partial \mathbf{h}_{t-2}} * \frac{\partial \mathbf{h}_{t-2}}{\partial \mathbf{h}_{t-3}} \quad \dots$$
$$= \prod_{i=k}^{t-1} \frac{\partial \mathbf{h}_{i+1}}{\partial \mathbf{h}_i}$$

> 由此可得对任意时刻的任意变量的导

Part 2

Initialization

> Why not init to zero ?

If all weights are set to 0, the derivative with respect to loss function is the same for every W, then all the weights have the same values in the subsequent iteration.
This makes the hidden units symmetric and continues for all the `n` iterations you run.
Therefore, we can’t set the weights all to zero.

However, we can still set biases to zero, which will not cause any troubles as non zero weights take care of breaking the symmetry and the values vary in every neuron.

> How to init ?

There are many ways to initialize the weights.

- `random`
  Initialize weights randomly, following standard normal distribution.

  > convenient and fast
  > can lead to too large or too small gradients

- `RELU`
  Use RELU/ leaky RELU as the activation function.

  > avoid vanishing / exploding gradient issue

- `gradient clipping`
  Set a threshold value, and if a chosen function of a gradient is larger than this threshold, we set it to another value.

  > avoid exploding gradient issue

Training

> Dataset

At first I use 2048 poems to train, but I found it can easily lead to overfitting, and the perplexity will keep increasing after a short term of decreasing.
Finally, 16384 poems from `全唐诗` are picked as training set, and 2730 poems as development set.

> Hyperparameters

Vocabulary size: 5851
Batch size: 32
Sentence length: 48
Hidden size: 512
Input size: 256

> Softmax Problem

At first stage of my training, I found that at some point the Loss wouldn't decrease at all no matter how I choose the learning rate.
Finally, I figured out that I shouldn't have applied `Softmax` to the result before calculating the loss, for `torch.CrossEntropyLoss` had already done this for me.

> Perplexity

At first, I use $\prod 1/p(s_i \mid s_{i-1}, s_{i-2}, s_{i-3} \ldots)$.
However, soon I found the result can easily reach `INF` , as long as there is a single small possibility among them.
Therefore, I choose another form $\sum log(1/p(s_i \mid s_{i-1}, s_{i-2}, s_{i-3} \ldots))$.

`initial perplexity` : 416.69

`final perplexity` : 238.83

Generating

> 日

`generate('日', temperature=0.7, split_len=6)`
日照解晴雕，必對酒酣未。
女媧門前年，君問君恩深。
我本是教餘，酒闌綸谷疆。
道蘊誇則貴，有鳥衮桃李。

`generate('日', temperature=0.8, split_len=8)`
日高閣朝來官看，樓臺閣上國期吟。
我道是詠洗鉢舟，此是驥劒外戚戚。
片雲午藥州城秋，月中有鳥樹蘚擔。

> 红

`generate('紅', temperature=0.7, split_len=8)`
紅蓼陰陽頭榼酒，久雨風雪壓鴨櫳。
身慵蝶言帝里巷，不憎幽愁借問君。
有鳥道在因說客，不如之得地萬里。

`generate('紅', temperature=0.8, split_len=6)`
紅氣自顧余，出門前山屐。
勸君酒闌慚，仍愁著遶宇。
家國計衰羸，佞爲愛酒半。
有鳥去年回，南山抱雲褭。
是燈月圓令，云我來喚將。
被婢怯摘草，茲山城裏客。

> 山

`generate('山', temperature=0.7, split_len=6)`
山河光彩筆，歲暮去年少。
喜君王孫處，得頭白日日。
款山頭白日，得意氣氤界。
世人生計聞，女媧郎爲魚。

```
generate('山', temperature=0.8, split_len=8)
```

山館山橫寒無事，此去怨離來從來。
愛醉昏亭上將軍，握賊房前年來十。
我迴屹資舉壘屯，見說帝鄰里巷觥。
大陂桂兵苑岳陽，學北風雨露濕紗。

> 夜

```
generate('夜', temperature=0.7, split_len=8)
```

夜深院榷逆旅鳥，送客來時節世間。
野岸上國人生道，以地換宴帚作尉。
有鳥下視時候日，生賢者尋歸來訪。
我來荒涼風乾走，北渚閣遊人生宅。

```
generate('夜', temperature=0.8, split_len=6)
```

夜木落澗郊，寢久客路岐。
窮巷梧桐溪，此地上天台。
獲處士槍菊，備使君家住。
楚客過風莢，得在屐狒瑚。
莫至不知君，計寮賊死天。
我聞君交銷，宵裳削文翁。
木葉敝耳裏，朝朝士魂興。

> 湖

```
generate('湖', temperature=0.7, split_len=6)
```

湖亭亭塘愛，今春風雨過。
有酒酣去年，或本志業倚。
對酒酣羞人，云是非唯史。
在天地上鷹，有鳥啼和氣。

```
generate('湖', temperature=0.8, split_len=8)
```

湖上詩郡門依稀，或堪憑手把酒古。
犯吏岫曉鼓鼓中，將軍城外府雨雪。
惜青門前年年年，古來史役城下馬。
憤朝朝迴騰饒此，水國門前年後反。

> 海

```
generate('海', temperature=0.7, split_len=8)
```

海上國途年將軍，層冷火春暖處處。
我來山蜩絮結構，吾聞君貳適意愛。

我本得君心能遊，得道在戶梓龜識。
彼此地不得聞君，東南國神仙館酒。

`generate('海', temperature=0.8, split_len=6)`
海萍疑浦中，孔官道羸年。
羡君北河南，聲苦從我來。
我本邊頭搶，一榻取太豔。
有路侍弄栢，因醉寫快此。
盡日暖竹苧，茲林靜境歸。
將軍鎖樹俯，壯錢臺閣家。

> 月

`generate('月', temperature=0.7, split_len=8)`
月中雉木落第宅，我有客含宜酒座。
見說遺于君身慵，最亦瞻夜回首望。
道州城隅文榴菊，不有鳥啄實格殿。
煎博恩深就花嘗，之風雨露滴巷岸。
我今朝朝遷塵煩，君子於來往事日。
世情遙郊千齡量，覺來從俗違戮攻。

`generate('月', temperature=0.8, split_len=6)`
月鎖竹曙落，此地上國把。
有鳥鳴亭皎，有鳥染淚濕。
和蘆葉下第，甚滴齒怨坐。
凭鞋寒爐濤，道好風騷人。

Optimization

I tried `SGD with momentum` and `Adam`.

`SGD with momentum`

> Fast but kind of unstable
> You should tune the learning rate and momentum many times

`Adam`

> Very stable but kind of slow
> You don’t have to adjust the learning rate too often