

## 7. Übungsblatt

### Aufgabe 35      RNN

Betrachten Sie ein RNN mit einer versteckten Schicht, die wie folgt aktiviert wird :

$$h_{t+1} = Wx_t + Uh_t + b$$

$$W, U \in \mathbb{R}^{4 \times 4}, b \in \mathbb{R}^{4 \times 1}$$

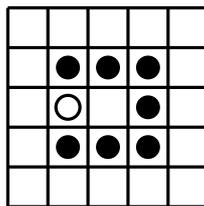
In den folgenden Skizzen betrachten wir den Agenten  $\circ$ , der sich entlang des Pfades von  $\bullet$  bewegen soll. Die Eingabe  $x_t$  in das RNN ist die Umgebung des Agenten, wobei sich der Vektor wie folgt zusammensetzt:

$$x_t = \begin{bmatrix} \uparrow \\ \rightarrow \\ \leftarrow \\ \downarrow \end{bmatrix} \in \{0, 1\}^4$$



Hierbei soll ein Eintrag 1 sein, wenn sich in der jeweiligen Richtung ein  $\bullet$  befindet. Die Ausgabe  $h_t$  soll dabei analog zu  $x_t$  die nächste Richtung des Agenten angeben. In der folgenden Anwendung wird nach dieser Ausgabe der Agent in die Richtung bewegt und beeinflusst die Eingabe  $x_{t+1}$

Bestimmen sie für das Labyrinth unten mit  $h_0 = (1, 0, 0, 0)^T$  die Gewichtsmatrizen  $W, U, b$ , sodass der Agent dem Pfad aus  $\bullet$  folgt.



Könnte ein normales MLP dieses Problem lösen? Was können wir über die Berechnungsfähigkeit der RNNs ableiten?

### Aufgabe 36      Gradienten in RNNs

Eine besondere Herausforderung bei RNNs ist das bestimmen eines geeigneten Gradientens. Dabei ist ein Aspekt das Lernen von Abhängigkeiten zwischen Elementen am Anfang und am Ende der Sequenz z.B. Subjekt und Verb in einem Nebensatz.

Wir betrachten nun als Vereinfachung nur die Verbindungen zwischen versteckten Neuronen mit linearer Aktivierungsfunktion.

$$h_t = U^t \cdot h$$

Weiterhin sei  $f(h_k)$  der Fehler, den die versteckte Schicht macht. Bestimmen Sie den Gradienten der Fehlerfunktion für  $U$  in Abhängigkeit von  $k > 1$ . Nehmen Sie dazu an, dass  $U$  eine Diagonalmatrix ist.

- Wie verhält sich der Gradient, wenn  $U = U_1$  bzw.  $U_2$  ist, für sehr große  $t$ ?

$$U_1 = \begin{bmatrix} 1.2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1.1 \end{bmatrix}, U_2 = \begin{bmatrix} 0.8 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.9 \end{bmatrix}$$

- Warum sind diese Werte problematisch für Gradienten-basiertes Training?
- Was sind mögliche Strategien, die bei den Problemen helfen können ?

### Aufgabe 37      Architekturen

Schlagen Sie für die folgenden Anwendungsgebiete jeweils eine Architektur vor und erklären Sie warum die jeweilige Architektur geeignet ist.

- Sie haben eine Sammlung von den Werken von Shakespear und wollen Texte im gleichen Stil generieren. Dabei sollen Sie auf der Ebene von Buchstaben arbeiten.
- Sie haben eine Sammlung von Katzen- und Hundebilder und wollen bei neuen Bildern entscheiden, ob es sich um einen Hund oder eine Katze handelt.
- Sie sollen ein Model entwickeln, das markiert, ob ein Buchstabe in einem Text zur wörtlichen Rede gehört oder nicht.
- Sie sollen ein Model entwickeln, das Englisch in Russisch und Deutsch übersetzt.
- Sie sollen ein Model entwickeln, das in der Lage ist einen Code zu zu reduzieren und den ursprünglichen Code zu rekonstruieren.

### Aufgabe 38      Bonus: Automatisches Differenzierungsframework (5)

Mit dieser Aufgabe wollen wir uns Klassifikation zuwenden. Dafür implementieren wir die Kreuzentropie, die auf eine Softmaxaktivierung folgt als eine differenzierbare Funktion. Dafür sollten Sie wie folgt vorgehen:

- Implementieren Sie für die Tensorklasse die exp Funktion
- Implementieren Sie für die Tensorklasse den natürlichen Logarithmus
- Erweitern Sie `sum(dim=None)` um den Parameter `dim` der es erlaubt entlang einer bestimmten Dimension die Elemente aufzuaddieren. Wenn `dim=None` soll die Funktion sich

wie bisher verhalten.

Bsp.:

$$\text{sum}\left(\begin{pmatrix} 1 & 2 & 3 \\ 3 & 4 & 5 \end{pmatrix}, \text{dim} = 1\right) = \begin{pmatrix} 6 \\ 12 \end{pmatrix}$$

$$\text{sum}\left(\begin{pmatrix} 1 & 2 & 3 \\ 3 & 4 & 5 \end{pmatrix}, \text{dim} = 0\right) = (4 \ 6 \ 8)$$

$$\text{sum}\left(\begin{pmatrix} 1 & 2 & 3 \\ 3 & 4 & 5 \end{pmatrix}, \text{dim} = \text{None}\right) = (18)$$

- Implementieren Sie `crossentropy`. Erinnern Sie sich dafür an den besprochenen Logarithmus der Softmaxfunktion.

Als nächstes wollen wir das erste Mal ein Klassifikationsproblem mit unserem Framework lösen. Dafür gibt es wieder eine Vorlage. Diesmal arbeiten wir mit dem Iris Datensatz <http://archive.ics.uci.edu/ml/datasets/iris>. Das Laden des Datensatzes im gleichen Ordner, so die Vorbereitung der Daten zu `batches` befindet sich in der Vorlage. Implementieren Sie nun folgende Schritte:

- Implementieren Sie ein Model.
- Trainieren Sie das Model.
- Erreichen Sie ein `Testaccuracy` von mindestens 90%.