

## Parsing mit neuronalen Netzen: Netzwerk

Nun sollen Sie das neuronale Netzwerk als Klasse **Parser** implementieren.

Aus Effizienzgründen werden die **Wortrepräsentationen** nicht aus der gesamten Buchstabenfolge berechnet sondern aus **Suffixen/Präfixen** fester Länge. Dies erleichtert die Parallelverarbeitung ohne Qualitätseinbußen.

Die Eingabe des Netzwerkes besteht daher aus zwei Tensoren: Der erste Tensor enthält alle **Wortsuffixe** fester Länge (z.B. 10). Kürzere Wörter werden am Anfang mit Paddingsymbolen aufgefüllt. Der zweite Tensor enthält alle **Wortpräfixe** derselben Länge in umgekehrter Reihenfolge, ebenfalls mit Padding.

Die Tensoren speichern Buchstaben-IDs und haben die Dimension  $N \times L$ , wobei  $N$  die Satzlänge und  $L$  die Präfix-/Suffix-Länge ist. Diese Vorverarbeitung müssen Sie nicht implementieren.

Das neuronale **Netzwerk** erledigt folgende Aufgaben:

- Berechnung der **Wortrepräsentationen**: Ein Forward-LSTM und ein Backward-LSTM berechnen beide eine Repräsentation für jedes Wort auf Basis (der Embeddings der Buchstaben) seines Suffixes bzw. Präfixes. Die Endzustände der beiden LSTMs werden konkateniert und ergeben die Wortrepräsentation.

*Anm.: Da die beiden LSTMs auf unterschiedlichen Eingaben operieren, können Sie hier kein bidirektionales LSTM verwenden.*

- Berechnung der **Konstituenten-Repräsentationen**: Ein (tiefes) bidirektionales LSTM verarbeitet die Wortrepräsentationen und liefert eine Repräsentation für jede Satzposition.

Durch Subtraktion der Repräsentationen von Satzpositionen werden dann die Konstituentenrepräsentationen berechnet.

Da Sie für die Spanberechnungen Repräsentationen für die Positionen 0 und Länge+1 brauchen, sollten Sie am Anfang und Ende des BiLSTM-Eingabetensors einen Nullvektor hinzufügen.

- Berechnung der **Bewertungen** der Kategorien: Dieses Feedforward-Netzwerk mit einer Hidden Layer transformiert jede Konstituentenrepräsentation in einen Vektor, der für jede Kategorie einen Score liefert. Die Größe der Hidden Layer sollte frei wählbar sein. Die Bewertung des Labels "keine Konstituente" (mit Label-ID 0) sollte auf 0 gesetzt werden.

In dieser Übungsaufgabe implementieren Sie noch nicht das Gesamtsystem, sondern nur das neuronale Netzwerk. Sie können also noch nicht auf realen Daten trainieren.

### Vorüberlegungen

- Wie **gliedern** Sie das Netzwerk sinnvoll in Teilnetzwerke?

- Welche Argumente sollten die **Konstrukteure** der Teilnetzwerke erhalten?
- Welche Argumente sollten die **Forward**-Funktionen der Teilnetzwerke erhalten und welche Rückgabewerte sollten sie liefern?
- Wo sollte **Dropout** angewendet werden?
- Wie berechnen Sie am besten die **Konstituenten-Repräsentationen**?
- Wie können Sie ihr Netzwerk separat **testen**?

**Anmerkung:** Aus Effizienzgründen sollten Sie möglichst viel parallel verarbeiten.  
(Eine parallele Verarbeitung mehrerer Sätze müssen Sie aber nicht versuchen.)