Lemmatisierung mit einem Encoder-Decoder-System: Training und Verarbeitung

In dieser Woche werden Sie den Lemmatisierer fertigstellen und ein Programm für das Training und ein Programm für die Lemmatisierung schreiben.

Training

Schreiben Sie ein Programm `lemmatizer-train.py`, welches den Lemmatisierer trainiert. Als Loss-Funktion verwenden Sie `CrossEntropyLoss`, als Optimizer `AdamW`.

Verwenden Sie `argparse.ArgumentParser`, um folgende Kommandzeilen-Argumente einzulesen: `trainfile`, `devfile`, `paramfile`, `embeddings_size=100`, `lstm_size=400`, `num_epochs=50`, `batch_size=1000`, `dropout_rate=0.5`. Definieren Sie bei den optionalen Argumenten – das sind alle außer den ersten drei Argumenten – die oben angegebenen Defaultwerte. Versuchen Sie auf einer Grafikkarte zu arbeiten und verringern Sie die Batchgröße, wenn der Speicherplatz nicht reicht.

Erzeugen Sie ein `Data`-Modul. Speichern Sie es mit der Methode `save` in der Datei `args.paramfile + '.io'`, wobei `args.paramfile` ein Kommandozeilen-Argument ist.

Trainieren Sie den Lemmatisierer auf den Trainingsdaten und evaluieren Sie ihn nach jeder Epoche auf den Development-Daten. Verwenden Sie bei der Evaluierung die Methode `lemmatize`. Geben Sie die Genauigkeit (Anteil der korrekt lemmatisierten Wörter) aus und speichern Sie das aktuelle Modell in der Datei `args.paramfile + '.pth'`, falls die Genauigkeit höher als alle bisherigen Genauigkeiten ist.

Anwendung

Schreiben Sie ein Programm `lemmatizer.py`, welches den Lemmatisierer auf Eingabedaten anwendet. Das Programm erhält zwei obligatorische Kommandozeilen-Argumente `paramfile` und `inputfile` und das optionale Argument `batch_size`. Der Inhalt der Datei `inputfile` ähnelt `train.txt`, enthält aber keine Lemmas. Die Ausgabe des Programmes hat dasselbe Format wie die Trainingsdaten und geht auf den Bildschirm.

Geben Sie Folgendes ab:

- die Liste der Genauigkeiten nach den einzelnen Epochen
- die Ausgabe Ihres Lemmatisierers für die ersten 100 Wörter der Developmentdaten (Die Eingabedatei des Lemmatisierers können Sie mit folgendem Befehl erstellen: `head -100 dev.txt | cut -f1 > input.txt`)
- Ihren vollständigen Code inklusive dem (eventuell verbesserten) Code aus den letzten beiden Aufgaben