Preprocessing

Im Preprocessing habe ich ein gemeinsames BPE-Modell für die Ziel- und die Quellsprache verwendet, zusätzlich habe ich es von 50'000 auf 100'000 Symbole erweitert, nach dem Motto je mehr desto besser. Es hat für mich zumindest Sinn gemacht es zu verdoppeln, da ich auf der einen Seite ja jetzt 2 Sprachen hatte, was sicher schon zu mehr unterschiedlichen Symbolen führt, und ich auf der anderen Seite auch ein effektiv grösseres BPE-Modell haben wollte. Der Rest beim Preprocessing war ziemlich standardmässig, tokenisiert und truecaseing mit mosesdecoder, selbstverständlich vor dem BPE.

Training

Im Code für das Training habe ich zwei weitere Anpassungen vorgenommen. Zuerst habe ich die Source-Sequenz umgekehrt. Damit hatte ich kurz ein paar Probleme, da ich nicht wusste, dass die .reverse() Methode in place umkehrt, konnte es dann nach mehreren Anläufen erreichen. Danach wollte ich noch Dropout implementieren, da ich unbedingt Overfitting verhindern wollte. Nachdem ich ein bisschen nachgeschaut habe, habe ich mich für eine Dropout-Rate von 0.5 entschieden, wie sie auch im Dropout-Paper von Srivastava et al. 2014 verwendet wurde. An den restlichen Trainingsparametern habe ich nichts mehr verstellt, da ich mir sicher war, dass ich das Baseline-System so schlagen kann. Trainiert habe ich über 6 Epochen, wie vorgegeben.

Resultat

Mit dem Training hätte ich ab Dienstagmittag beginnen können, musste jedoch auf den Neustart warten. Am Mittwochabend musste ich dann das Training wegen dem 2. Neustart nochmal beginnen. Das Training hat überraschend lange gedauert (57h), und das Resultat war sehr ernüchternd. Ich habe auf dem devset einen BLEU-Score von nur 22.27 erreicht, und da ich glaube ich den neuen daikon-code verwendet habe, ist das eine ziemliche Verschlechterung gegenüber dem Baseline-System. Ich vermute, dass mein System bei diesen doch eher kleinen Trainingsdaten und 6 Epochen Training eine Dropout-Wahrscheinlichkeit von 50% wohl zu hoch war und das Modell underfitted ist. Ich hatte nach dem Training jedoch keine Zeit (und auch kein Platz auf dem Server) mehr, es ein zweites Mal zu trainieren mit einer kleineren Wahrscheinlichkeit. Ich bin mir auch nicht sicher, ob ein Dropout im Decoder-LSTM überhaupt Sinn macht, auch das hätte ich gerne noch ausprobiert.