**Leitfaden für nachvollziehbare Schritte**

1. Kurze Darstellung des Problembereichs / Aufriss des Themas

1.1 Inhaltlich

Es soll ein Neuronales Netz aufgebaut werden, das Text als input nimmt, um einen dem Eingang ähnlichen Text auszugeben.

1.2 Begründung desThemas

**Darstellung der Relevanz des Themas?**

Diese Form eines Neuronalen Netzes ist als „proof of concept“ interessant, um die Fähigkeit von NNs Text zu erzeugen zu pürfen.

***Darstellung eines persönlichen Erkenntnisinteresses.***

Texterzeugung aus Neuronalen Netzen ist ein sehr interessantes Feld. Während hier eine sehr grundlegende Form der Spracherzeugung trainiert werden soll, ist die Vorgehensweise sehr interessant.

2. Nachvollziehbare Schritte

2.1 Der Stand der Forschung / Auswertung der vorhandenen Literatur / Tutorials ...

*Wurde das Problem früher bereits untersucht?*

Das Problem ist in vielerlei Form bereits geprüft worden.

*Welche Aspekte wurden untersucht und welche nicht?*

2.2 Fragestellung

Kann ein Neuronales Netzwerk mit einfachen Mitteln einen echt klingenden Text aus einem gegebenen Sample erstellen?

2.3 Stand der Forschung

unbekannt

2.4 Wissenslücke

Die Erstellung eines NNs zur Textanalyse ist in ihrer Grundform nicht neu, allerdings in dieser speziellen Anwendung bisher nicht.

2.5 Methode

2.5.1 Datenaufbereitung

Die Daten, welche aus github geladen wurden, müssen bereinigt werden. Diverse Sonderzeichen können von Python nicht importiert werden und müssen im Voraus durch lesbare Zeichen ersetzt werden.

Dies wurde in Notepad++ durchgeführt, wo Zeichen wie schräge Anführungszeichen und ähnliches ersetzt wurden.

2.5.2 Import der nötigen Module



2.5.3 Import der Daten



2.5.4 Daten werden auf lowercase gesetzt

Dieser Schritt ist sinnvoll, um die Anzahl der characters nahezu zu halbieren, da keine Großbuchstaben nötig sind.



2.5.5 Erzeugen der Character-list

Nun müssen die einzelnen Zeichen, die unique im Datensatz vorkommen in eine liste übersetzt werden, da das Modell nicht mit strings arbeiten kann.



Heirzu wird über sorted(), list() und set() ein set unique charakter erstellt, zur Info ausgegeben und dann in zwei dictionaries übergeben. Mid diesen Dictionaries können nun die Zeichen in zugehörige Zahlen übersetzt und zurückübersetzt werden.

2.5.6 Aufteilen des Texts in batches

Der Text muss nun in batches aufgeteilt werden, die am Stück gelesen werden. Als Länge sind hier 40 Zeichen gewählt. Diese Länge ist frei wählbar, wurde aber auf 40 belassen, da dies in der Tensorflow Dokumentation als guter Wert genannt wird.

Es werden 40 character lange sets erstellt, die jeweils um 3 Zeichen versetzt über den gesamten Text laufen:



2.5.7 Dummy Variablen / Vektorisierung:

Für jedes Zeichn pro Textschnipsel wird der Wert für X am entsprechenden Index auf TRUE gesetzt. Für Y wird immer der erwartete Folgebuchstabe auf TRUE gesetzt. Hierdurch wird erreicht, dass das NN den erwarteten 41. Buchstaben nach einer Folge von 40 Buchstaben erlernt



2.5.8 Model erzeugen:

Entsprechend der tensorflow Dokumentation wird das Model als LSTM-model aufgebaut. Dieses eignet sich besonders gut für die vorhersage von Text. Der Shape des Models ist durch die Anzahl der unique-Characters im Text bestimmt:



2.5.9 Sample Funktion, um Text zu erzeugen:

In der Sample-Funktion wird mittels einer softmax Berechnung die Wahrscheinlichkeit für den folgenden Buchstaben errechnet. Nach dem winner-takes-all Prinzip wird nur der höchste Output gewertet.

preds ist der Output des Models und temperature ist ein Maß dafür, wie konservativ der Text erzeugt werden soll. (kleiner= näher am original, größer = mehr Fehler aber interessanterer Output)



2.5.10 output am Ende jeder Epoche:

Um den Fortschritt des Modells verfolgen zu können wird am Ende jeder Epoche ein Sample an verschiedenen Temperatures erstellt. Hierbei werden 400 Zeichen aus einem zufälligen Anfangsseed von 40 Zeichen generiert. Jedes generierte Zeichen verschiebt den betrachteten Bereich um eine Stelle, sodass ab dem 41. Zeichen der komplette betrachtete Text außerhalb des Seeds ist und sich komplett aus sich selbst fortschreibt:



**Beispiel:**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*----- Generating text after Epoch: 1*

*----- temperature: 0.2*

*----- Generating with seed: ""yes, i quite agree." he didn't say another word o"*

*"yes, i quite agree." he didn't say another word of the troush the stream and start of the streen to the door and the streared of the streathed of the streathed of the fore the streen of the door the streathed the door the door the leaky to the train of the class with a famely and said. "harry said had got to the fleen of the floor the lead on the streaned of the stands of the fore the streathed the class the door the streathed at the streathed o*

*----- temperature: 0.5*

*----- Generating with seed: ""yes, i quite agree." he didn't say another word o"*

*"yes, i quite agree." he didn't say another word of she leaves the start and head to hermione snapped the marris, the frounds of the look as pooked of the lade the stinds and surred the stands of the clear pooked at the haght me the sarded and said. "you're manut of the fursh to a larsly on the mors and was the must from the stresped of books snepped as he looked of the dark making to the froth of seen his should the mist on the streally of the e*

*----- temperature: 1.0*

*----- Generating with seed: ""yes, i quite agree." he didn't say another word o"*

*"yes, i quite agree." he didn't say another word of the ohlagiff in frofu. "heakly of the hand, parknightzan recky bark the 's sade. an, thair grount, slapled. he goned of the wailsuly tire end mright enged in the ?" "... his best doon, arounded his amasling lad, went that even strdnon, and the pack beew softtwent last ittring with. we all velp's domepering ont by of acrus shepprion. slapt baday. harry sapcing to pimped on the strad. no got ence*

*----- temperature: 1.2*

*----- Generating with seed: ""yes, i quite agree." he didn't say another word o"*

*"yes, i quite agree." he didn't say another word of that waslve deep tignt off oundays, drevin'rrted, the de'd fore the hailsling hoy in the stroct malforc to pamplinn. "grays, but looked chairs, anyns, hagrid but out doined at plaksed cars must less unuporly on the hounath." "yer the flut mamine on harry cronefos fblaop chrbomered infics to him, less in the fron as im, whenking eljouloefuny. "yerking." "..." saroush ie dummledousess he was, pro*

2.5.10 Fitting des Models:

Das eigentliche Fitting des models ist nun nur noch ein kleiner codeblock, in dem die Anzahl der Epochen und die batch-size festgelegt werden:



**2.6 Ergebnisse**

Die Ergebnisse dieses Versuchs sind interessant, wenn auch in der Praxis vermutlich nicht sehr nützlich:

Wir haben herausgefunden, dass das NN in der Lage ist, selbständig Text zu erzeugen, der mit ein Wenig Anpassung der Parameter aus größtenteils echten Worten besteht, die in scheinbar sinnvollem Zusammenhang stehen.

Allerdings ist das Modell weder auf Worte, noch auf Sinn Trainiert und fügt nur Buchstabe für Buchstabe zusammen, wodurch Sinn nur absolut zufällig entstehen kann.

*n the same condemnation? 23:41 and we is the morning of the commandments of the lord the lord, and the lord the lord thy god of hosts and the sons of the lord the god of god, and shall be shall be the lord the lord will not be the lord the lord, and shall be the son of the priest, and the men of the lord the god of god of the people, and shall be the lord of hosts.21:2 and the lord shall be the lord the god of the lord the lord the god*

**2.7 Ausblick**

Das Modell in der vorliegenden Form wird wohl nie in der Lage sein, logisch kohärenten Text zu generieren. In weiteren Projekten muss ein NN erzeugt werden, das nicht im ganz Kleinen auf jeden Buchstaben achtet, sondern vielleicht Worte in Zusammenhang bringt, um einen sinnhafteren Zusammenhang im Text zu erzeugen.