# What is a Transformer?

<https://medium.com/inside-machine-learning/what-is-a-transformer-d07dd1fbec04>

# Transformers: Implementing NLP Models in 3 Lines of Code

<https://towardsdatascience.com/transformers-implementing-nlp-models-in-3-lines-of-code-475639c3611d>

<https://towardsdatascience.com/understanding-nlp-techniques-and-machine-learning-2fdbe78922f2>

<https://medium.com/analytics-vidhya/nlp-glossary-for-beginners-c3093529ee4>

<https://medium.com/analytics-vidhya/neural-networks-for-word-embeddings-4b49e0e9c955>

<https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1>

<https://towardsdatascience.com/giving-some-tips-for-data-science-interviews-after-interviewing-60-candidates-at-expedia-395fff7e073b>

<https://towardsdatascience.com/introduction-to-spark-nlp-foundations-and-basic-components-part-i-c83b7629ed59>

<https://medium.com/analytics-vidhya/nlp-glossary-for-beginners-c3093529ee4>

<https://medium.com/analytics-vidhya/neural-networks-for-word-embeddings-4b49e0e9c955>

<https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1>

**Introduction to Word Embedding and Word2Vec**

<https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa>

<https://github.com/soroosh-rz>

# 7 concepts of databases every data engineer should know

<https://oleg-agapov.medium.com/7-concepts-of-databases-every-data-engineer-should-know-5ea4a357c149>

<https://www.coursera.org/learn/language-processing/peer/982Gp/recognize-named-entities-on-twitter-with-lstms/review/ak74l3sSEeu8hgrPuuYC6w>

### **Help how to do and submit assignment1 - predict tags on stackoverflow**

1. Open [https://colab.research.google.com](https://colab.research.google.com/), click **Sign in** in the upper right corner, use your Google credentials to sign in.
2. Click **GITHUB** tab, paste <https://github.com/hse-aml/natural-language-processing> and press Enter
3. Choose the notebook you want to open, e.g. week1/week1-MultilabelClassification.ipynb

<https://aws.amazon.com/big-data/datalakes-and-analytics/what-is-a-data-lake/>

<https://aws.amazon.com/blogs/big-data/introducing-the-data-lake-solution-on-aws/>

# Introducing the Data Lake Solution on AWS

<https://aws.amazon.com/blogs/big-data/introducing-the-data-lake-solution-on-aws/>

<https://aws.amazon.com/big-data/datalakes-and-analytics/>

<https://aws.amazon.com/big-data/datalakes-and-analytics/what-is-a-data-lake/>

<https://colab.research.google.com/drive/1N9CWIXysFH7sSeZAlFAIBmabQyamkHw4>

Week 2

<https://colab.research.google.com/drive/1Xy3uXCddtK40z79TcARyNfP1UCFohvB2>

Word embeddings

<https://colab.research.google.com/drive/1nOUR17sQL3epdMySuIBuYLgjTZWWYc7G>

**RL**

<https://github.com/BoYanSTKO/Practical_RL-coursera>

**Below an excellent course**

<https://github.com/jiadaizhao/Advanced-Machine-Learning-Specialization>

**Book**

<https://ibm.ent.box.com/notes/552985044778?v=WelcomePrework>

<https://ibm.ent.box.com/notes/628111590035?v=PreworkAssignment>

<https://ibm.ent.box.com/notes/545105102419>

<https://cloud.google.com/learn/what-is-apache-spark>

4th Week

<https://colab.research.google.com/drive/16caRg-gC-d_UkB6E6jEgI6IFqRZ-5MhU>

@@@@@@@@@@@@@@@@@@@@@@@@

Rubric

There is a correct implementation of the function generate\_equations which passes basic tests.

0 pts

No

1 pt

Yes

There is a correct implementation of the function sentence\_to\_ids. It appends special end symbol at the end of the sequence and placeholders if the length of input is smaller than desirable length.

0 pts

No

1 pt

Yes

Placeholders self.ground\_truth, self.ground\_truth\_lengths and self.learning\_rate\_ph are initialized correctly in the function declare\_placeholders.

0 pts

No

1 pt

Yes

Placeholders self.embeddings and self.input\_batch\_embedded are initialized correctly in the function create\_embeddings.

0 pts

No

1 pt

Yes

The lines in the functions build\_encoder and build\_decoder are filled in correctly and the following variable are created with corresponding values:

* encoder\_cell and decoder\_cell as RNN cells; DropoutWrapper is used for both of them;
* dynamic\_rnn function is used for creating encoder;
* self.ground\_truth\_embedded is initialized as embeddings\_lookup using self.embeddings;
* infer\_helper as helper from tf.contrib.seq2seq (e.g. GreedyEmbeddingHelper);
* decoder as BasicDecoder from tf.contrib.seq2seq.

0 pts

Not all lines are filled-in.

1 pt

All lines are filled-in, but there are some mistakes, for example parameters in DropoutWrapper are not set, or initial state of the decoder is not final\_encoder\_state.

3 pts

All lines are filled-in correctly.

The lines in the functions compute\_loss and perform\_optimization are filled in correctly and the following variables are created:

* self.loss is defined as sequence\_loss which operates with logits (self.train\_outputs.rnn\_output), self.ground\_truth and the defined weights.
* self.train\_op is initialised as follows: self.loss is used as a loss function, value of the learning rate is the value of corresponding placeholder, optimizer is Adam, gradients are clipped.

0 pts

Not all lines are filled-in.

1 pt

All lines are filled-in, but there are some mistakes, for example, gradients are not clipped or other optimizer is used.

3 pts

All lines are filled-in correctly.

Functions predict\_for\_batch and predict\_for\_batch\_with\_loss are filled in correctly.

0 pts

No

1 pt

Yes

There are experiments of the network training present in the notebook.

0 pts

There are no experiments at all.

1 pt

There is at least one experiment with the smaller number of epochs.

3 pts

There is at least one experiment with the recommended hyperparameters.

Final evaluation scores are present in the notebook. MAE score for the best model is around 50 or lower on the test data and invalid\_number\_prediction\_count always equals zero.

0 pts

No

3 pts

Yes

5th week

<https://colab.research.google.com/drive/1DIvJ8Q20echfbsIjOxn3Kl35VSuVWLQC>

<https://colab.research.google.com/drive/1DIvJ8Q20echfbsIjOxn3Kl35VSuVWLQC#scrollTo=ECXLE3F6DO0Q>

Maria BakhanovaTeaching Staff · [6 months ago](https://www.coursera.org/learn/language-processing/peer/xbHJG/stackoverflow-assistant/discussions/threads/yi2-cuEgEeqr3AoY5SfHyQ/replies/s56cNOHoEeqr3AoY5SfHyQ/comments/18UVaueHEeqr3AoY5SfHyQ)

Hi,

I recommend you to do the following in Google Colab:

* start the work as usual (download resources with the first cell and etc.)
* open utils.py in your local text editor and fill in the gaps
* when you are ready with utils.py, remove utils.py in Colab with *!rm utils.py*
* upload your modified file: click at [>] to open the left pane, choose file tab, click [upload] and choose your [*utils.py*]
* restart runtime: *Runtime --> Restart runtime*
* import: *from utils import \**

Week4

<https://colab.research.google.com/drive/1nHH1hyV7GEkwvUFJDKNXPR-Wm6QBZXz1#scrollTo=vACs7YnMSO-e>

<https://colab.research.google.com/drive/1nHH1hyV7GEkwvUFJDKNXPR-Wm6QBZXz1#scrollTo=REBUflBiSO-2>

Text, letter

Description automatically generatedGraphical user interface, text, application

Description automatically generatedGraphical user interface, text

Description automatically generatedGraphical user interface, text, application

Description automatically generated

<https://www.coursera.org/learn/language-processing/peer/3wvhr/learn-to-calculate-with-seq2seq-model/review/jVYwcIQJEeundBJZhzCd-w>

<https://www.coursera.org/learn/language-processing/peer/3wvhr/learn-to-calculate-with-seq2seq-model/review/jVYwcIQJEeundBJZhzCd-w>