# Errata: Bachelor thesis

s184399 - Mikkel Godsk Jørgensen

January 2022

Citations and figure references are renumbered in this document.

## Section 2.1: The transformer model - Figure 1 (p. 4)

**(Error in illustration)**. The brown output matrix was too tall possibly making the illustration confusing. The reader should be able to assume that the heights and widths of the boxes can be interpreted as the dimensions of the matrices. The corrected version is seen in figure 1.

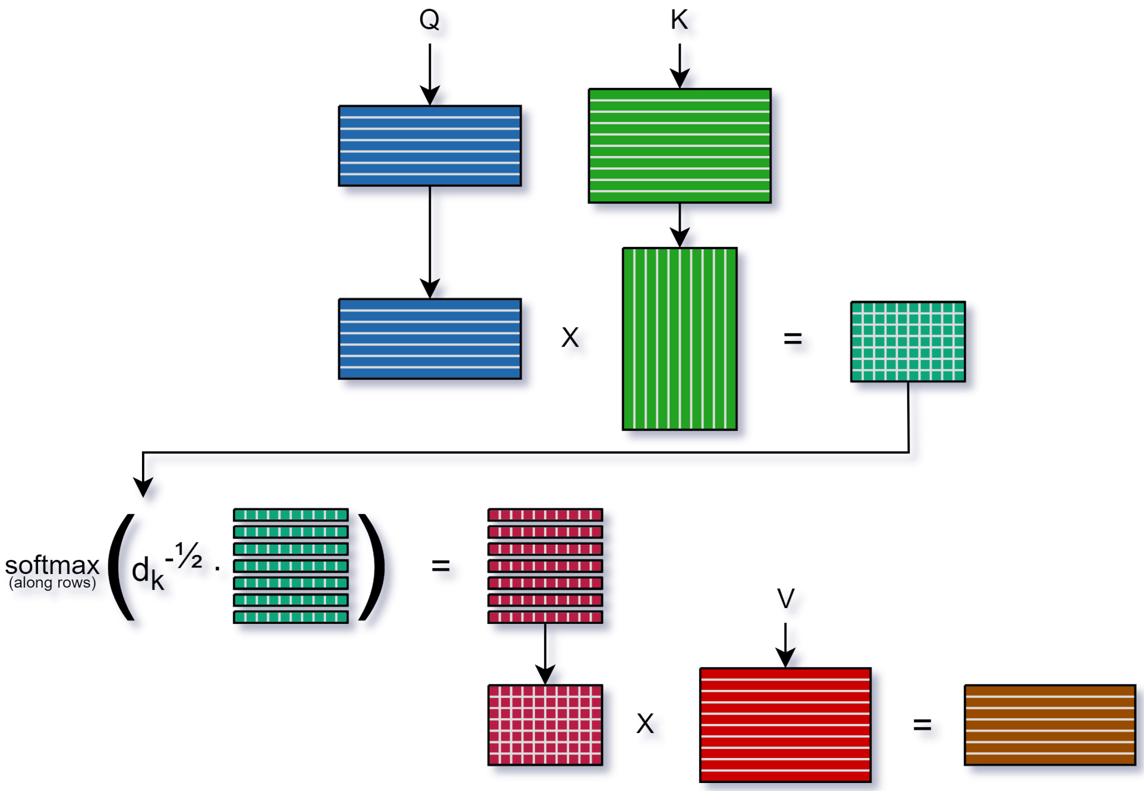

Figure 1: The scaled dot-product attention illustrated. As in the definition from [Vaswani et al. 1, 2017], the queries, keys and values are stored in rows. This illustration is heavily inspired by [Ji et al. 2, 2019].

## Section 2.1: The transformer model - Input sequence (pp. 5-6)

**(Notation error)**. First of all, the sequence $(x_1, x_2, ..., x_n)$ does not have $x_i \in \mathbb{R}^{d_{\text{model}}}$. The constant $d_{\text{model}}$ denotes the size of the embeddings, hence $\text{InputEmbedding}(X) \in \mathbb{R}^{n \times d_{\text{model}}}$. In [Vaswani et al. 1, 2017] (*"attention is all you need"*), the input sequence is specified as a sequence of embeddings within $\mathbb{R}^{d_{\text{model}}}$. However they do not specify the space of the tokens.

Furthermore, I believe the notation and description can easily be cause for confusion since it is slightly illogical. To clarify, we have a *padded* sequence $(x_1, x_2, ..., x_n)$ at page 5. In the example at the top of page 6, the sequence length should more logically be denoted $n = 384$, which would also correspond to having a query matrix $Q \in \mathbb{R}^{n \times d_{\text{model}}}$. However, since we are using self-attention we necessarily have that $m = n$.

In order to make the sentence "Dette er mit bachelor projekt" into a sequence $(x_1, x_2, ..., x_n)$, it is first tokenized using e.g. WordPiece and padded to have a length $n = 384$. If longer, it is cropped.

## Section 2.1.1: BERT - The empty string as input (p. 7)

**(Ambiguous description)**. With "the BERT-encoding is actually not the zero-vector", I actually mean the vector $C$. However, this is also true for the output sequence, i.e. none of the $T_i$ vectors are the zero-vector.

## Section 3.7: Similarity measures - Leacock-Chodorow similarity (p. 14)

**(Notation error)**. Correction: $\mathrm{sim_{lch}(a, a) = \max\left[-\log(1/2D)\right]}$.

## Section 4.3: Figure 5.a (p. 19)

**(Title and labels were not very informative)**. The corrected version is seen in figure 2.

## Section 4.2: Preliminary experiment: Building a classifier using BERT (p. 16)

**(Missing table reference)**. The last sentence on the page should read:

> The results are summed up in tables 4 and 5.

## Section 4.2: Preliminary experiment: Building a classifier using BERT (p. 17)

**(Missing word)**. Lines 9-11 should read:

> Hence it seems that the pooled output sentence embedding may not be of much use as an embedding as also pointed out in section 2.1.1.

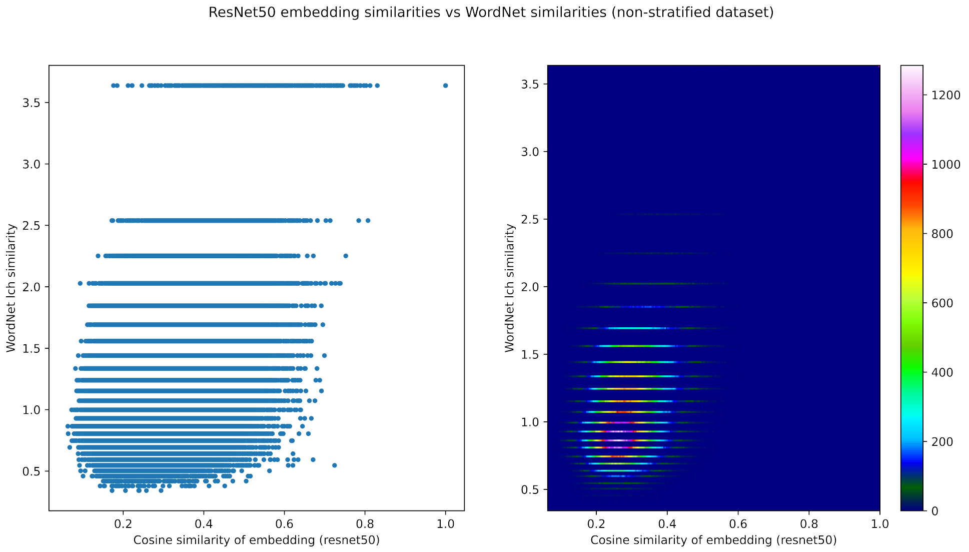

Figure 2: This dataset was not stratified. The empirical correlation was found to be approximately 0.39.

## Section 4.3: Preliminary experiment: The correlation of pairwise similarities between WordNet synsets and ResNet50-embeddings

**(Missing word)**. Lines 3-5 should read:

> Here I use a similar definition of semantic meaningfulness being that there is a relationship between the semantic similarity of the synsets and the embeddings of appurtenant random images.

## Section 4.4: Interpretation of results (p. 30)

**(Wrong terminology)**. The *test set* should be the *validation set*. This was written before I changed the names of the partitions to be less convoluted. Hence the paragraphs should read:

**Interpretation of results:** Comparing to the baseline accuracies from table 9, we see that the addition-based model seems to consistently outperform the baseline on the validation set. Given WordNet hints, we see an improvement of roughly 7 percentage points measured on the top-1 accuracy score. Furthermore, given Wikipedia-hints the improvement is even greater at roughly 23 percentage points higher accuracy. There is however the caveat that the hints on the validation set did not have dynamic-dropping applied.

Furthermore, we see that the concatenation-based model seems to yield an improvement over the baseline when given Wikipedia-hints. Here the improvement is measured to be roughly 20 percentage points. Lastly, training the addition-based model again from scratch yields very similar results with the largest difference in accuracy on the validation set being in the third decimal place. This suggests that this result is reliable.

# 1 Section 4.4: Experiment - Constraining $\lambda$ to the unit interval (p. 30)

**(Lack of clarity)**. As for the entire section, I am using the average word embedding. Although not stated here, it was stated on p. 23 that only the average word embedding would be used.

# 2 Section 4.4.6: Experiment - Varying $p_{\text{kept}}$ between 0 and 1 (p. 33)

**(Lack of clarity)**. With "the model" I am referring to the addition-based model. This is also apparent from the figure titles.

# References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[2] S. Ji, Y. Xie, and H. Gao, "A mathematical view of attention models in deep learning," *Texas A&M University*, 2019.