Non-NLP Transformers

Atticus P. Johnston
CS 342: Neural Networks
Department of Computer Science
The University of Texas at Austin
Austin, TX 78712
atticusjohnston@utexas.edu

1 Background

Today, the deep learning landscape has adopted the transformer as the undisputed gold standard. And for an extremely valid reason; it revolutionised the entire field. Amidst all of the worship, this project seeks to evaluate the effectiveness of the distinguished transformer model on a basic classification task. The ultimate focus will be the comparison of the transformer's performance against that of standard feed-forward and convolution neural networks, with further experiments into the generalisability of the constructed model and optimal model architecture for the task.

The classification task will be predicting the quality of a wine based on its physicochemical properties, a rudimentary task typically handled by a standard feed-forward neural network. This project, despite understanding that the transformer architecture was constructed explicitly to deal with sequential data processing, of which this is not, hopes to unveil the facade of the transformer being the one solution for all deep learning problems.

During this process, a model with an understanding of the predictive power that a wine's physicochemical properties holds on the quality of the wine will be developed, thereby producing a powerful tool in wine quality evaluation.

Influential Papers This project is rather distinct in its principle direction, but inspirations have been derived from two papers. First, the foundation of Vaswani and colleagues' paper, *Attention Is All You Need* [3], on this project is unambiguous, having proposed the transformer architecture. Additionally, Cortez et al. [1], which is the original paper where the wine quality dataset was constructed, gave crucial inspiration for the performance metric employed for this project.

2 Dataset

The wine quality dataset used in this project is from Cortez et al. [2]. It includes 6497 instances of red (1599) and white (4898) *vihno verde* samples from the Minho region of Portugal, outlining eleven common physicochemical properties and a wine grading in a scale that ranges from zero (very poor) to ten (excellent). These final gradings of the wines are the median score of blind sensory evaluations conducted by at least three expert assessors. The data on grape type, wine price and brand is not included.

2.1 Features

Table 1 displays the eleven features for each wine instances and their corresponding statistics. These features, alongside the wine quality score, comprise the dataset, with rows being the different wines, and columns being the features.

Table 1: Physicochemical data statistics per wine type.

	Red			White		
Feature	Min	Max	Mean	Min	Max	Mean
fixed acidity (g (tartaric acid)$/dm^3$)	4.6	15.9	8.3	3.8	14.2	6.9
volatile acidity (g (acetic acid)$/dm^3$)	0.1	1.6	0.5	0.1	1.1	0.3
citric acid (g/dm^3)	0.0	1.0	0.3	0.0	1.7	0.3
residual sugar (g/dm^3)	0.9	15.5	2.5	0.6	65.8	6.4
chlorides (g (sodium chloride)$/dm^3$)	0.01	0.61	0.08	0.01	0.35	0.05
free sulfur dioxide (mg/dm^3)	1	72	14	2	289	35
total sulfur dioxide (mg/dm^3)	6	289	46	9	440	138
density (g/cm^3)	0.990	1.004	0.996	0.987	1.039	0.994
pH	2.7	4.0	3.3	2.7	3.8	3.1
sulphates (g (potassium sulphate)$/dm^3$)	0.3	2.0	0.7	0.2	1.1	0.5
alcohol (% vol.)	8.4	14.9	10.4	8.0	14.2	10.4

2.2 Preprocessing

To make the data suitable for the the model it was preprocessed in several ways. Each feature column (all except wine quality score) was normalised with min-max scaling[1], and the quality ratings were separated from the features into a new table. Each rating was then converted into a one-hot encoding with 10 classes. These two tables comprised the input and target tables for the model, and the white wine set was split into training, validation and test sets with ratio of 70/20/10 and batch size of 64.

3 Model description

The model was constructed with the PyTorch library. Specifically, the transformer architecture was constructed with using PyTorch's `TransformerEncoderLayer` and `TransformerEncoder` classes. Six encoder layers, with sixteen attention heads per layer and a hidden size of 128, were used for the transformer. Linear layers were placed before and after the transformer, the first as an embedding layer, and the latter as an output layer to output the ten classes. A rectified linear unit (ReLU) activation function was applied to the output of the final linear layer. Figure 1 provides a comprehensive summary of the model's architecture.

A significant amount of testing was conducted in pursuit of the optimal model architecture which heavily influenced the final model's attributes. The relationships between the features in the data were complex enough to warrant the use of six encoder layers, and a hidden size of 128 aided the model in capturing those relationships. The number of attention heads did not seem to influence the model's performance to a great extent, assumed to be because the data was not sequential and the multi-head attention mechanism did not play a large role in the model's learning.

The learning rate of the model was set to 1×10^{-4} as it allowed the model to learn properly and converge in a reasonable number of epochs, approximately 300. The model's performance between different learning rates seemed to have little variation, only the time taken to converge.

[1] $x_{\text{sc}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$

```
==========================================================================================================================
Layer (type:depth-idx)                        Input Shape               Output Shape              Param #
==========================================================================================================================
TransformerModel                              [64, 11]                  [64, 10]                  --
├─Linear: 1-1                                 [64, 11]                  [64, 128]                 1,536
├─TransformerEncoder: 1-2                     [64, 1, 128]              [64, 1, 128]              --
│    └─ModuleList: 2-1                        --                        --                        --
│    │    └─TransformerEncoderLayer: 3-1      [64, 1, 128]              [64, 1, 128]              99,584
│    │    └─TransformerEncoderLayer: 3-2      [64, 1, 128]              [64, 1, 128]              99,584
│    │    └─TransformerEncoderLayer: 3-3      [64, 1, 128]              [64, 1, 128]              99,584
│    │    └─TransformerEncoderLayer: 3-4      [64, 1, 128]              [64, 1, 128]              99,584
│    │    └─TransformerEncoderLayer: 3-5      [64, 1, 128]              [64, 1, 128]              99,584
│    │    └─TransformerEncoderLayer: 3-6      [64, 1, 128]              [64, 1, 128]              99,584
├─Linear: 1-3                                 [64, 128]                 [64, 10]                  1,290
├─ReLU: 1-4                                   [64, 10]                  [64, 10]                  --
==========================================================================================================================
Total params: 600,330
Trainable params: 600,330
Non-trainable params: 0
Total mult-adds (Units.MEGABYTES): 0.18
==========================================================================================================================
Input size (MB): 0.00
Forward/backward pass size (MB): 0.07
Params size (MB): 0.01
Estimated Total Size (MB): 0.08
==========================================================================================================================
```

Figure 1: Base transformer architecture summary.

Three regularisation techniques were employed to aid the model's generalisability and prevent it from over-fitting. Dropout was used at 50% since it is a powerful way to regulate learning and allowed the model to generalise to the unseen data the best. Weight decay was set to 1×10^{-5} as a means of ensuring that the model does not become too complex and also increase the stability of the model. A learning rate scheduler was used with a step size of 100 and a multiplier of 0.1. This allowed the model to converge faster and assist in maintaining the model's stability.

4 Performance

Performance metric Top-2 categorical accuracy was the adopted performance metric for the model and the further computational experiments. It measures a success when the target class is within the top two predicted classes provided by the model. The final value is the number of successes divided by the total number of outputs. It was used due to the subjective nature of the target output of the model, which permits the leniency in performance evaluation. Inspiration for this was drawn from Cortez et al. [1] who used a very similar performance metric.

The model's performance was evaluated on the withheld white wine testing set (10%) over 300 epochs. It was trained and evaluated in four separate tests, with all variables staying consistent. Figure 2 depicts a training and validation loss over all epochs similar to that of all tests. The mean top-2 accuracy for the transformer was 86.3%.

This result displays a high effectiveness of the transformer when applied to this task. However, as Figure 2 shows, the learning rates above 1 indicate that the model can be improved greatly.

5 Computational Experiments

Each of the following experiments was conducted with four trials, where all variables remained consistent, including the training, validation and testing split sets. The final top-2 accuracy representing each model is the mean of the top-2 accuracy from all tests.

5.1 Generalisation performance: red wine data

To evaluate the model's generalisability, it was tested on all of the red wine data which was completely unseen to the model. This experiment provides a great indicator of both the model's generalisability and the variation in physicochemical properties that contribute to a wine's quality between red and white wines. Prior to testing, the feature importances of red wines was unknown, and assumed to be similar to those of white wines, so it was expected for the model to possess somewhat reasonable accuracy on the red wine data.

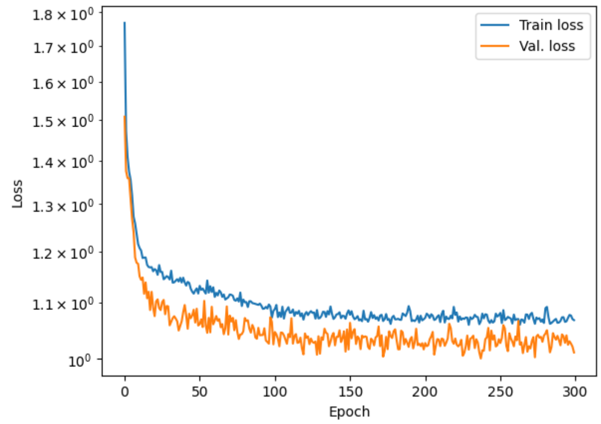

Figure 2: Base (6 encoder layers, 50% dropout) transformer model training and validation loss over 300 epochs.

The testing resulted in the model displaying a 76.28% top-2 accuracy on the red wine data, demonstrating that the model generalises to red wine data to a much lesser degree than that of white wine.

In Cortez et al. [1], they provide insight into the feature importance for each wine type of their support vector machine (SVM) model. Although not a neural network model, their SVM model displayed a high proficiency for predicting the wine quality, so it can be assumed that these feature importances hold in this project's case. Their findings show, aside from sulphates being the most important for each type, drastic feature importances between red and white wines, for instance, pH is ninth for white wines, but second for red wines.

With this understanding, it can be assumed that the model cannot, and did not, learn the relationship between the features of red wines, solely on white wine data, since it will weight the features as if they are attributed to white wines. Hence, this experiment shows that model generalisation on red wine data is extremely difficult for a model strictly trained on white wine data.

5.2 Model comparison: transformer vs. feed-forward and CNN

For this experiment, two more models were constructed: a feed-forward network and a CNN. It was intended for the complexities of the new models to be similar to that of the transformer. The feed-forward model comprised of an input and output layers, and one hidden layer in between, whereas the CNN had two convolutional layers with kernel sizes of three, and also one max pooling layer with a kernel size of two. Both models possessed hidden sizes of 128, 50% dropout and ReLU activation functions.

Table 2: Performances of various architectures.

Model	Mean top-2 accuracy (%)
feed-forward	83.57
CNN	83.34
transformer	**86.30**

The results of this experiment, shown in Table 2, indicate that the various models did not differ in performance to a high degree. Although the transformer boasted the greatest accuracy, it must be noted that significantly more effort was put into tuning the transformer model's hyperparameters and architecture, aiming for an optimal model. This was not the case with the feed-forward and CNN models since they were constructed to be "equal" in complexity to the transformer. A concept that is very difficult to define between different types of models. Since the transformer, even with this additional effort, did not perform levels higher than the other models, the difference can be interpreted as negligible, and all the models are as effective as each other on this data.

Moreover, the transformer model was notably slower in training than the other two models, so it can be considered worse for this task. This analysis provides a final judgment on how effective the transformer architecture is when applied to non-sequential data such as this. The transformer is a capable model for use in tabular classification tasks, certainly not a bad one, but the advantages of using it does not outweigh the negatives, in the added training time and unnecessary complexity of the model. This outcome can be expected, however, considering that the transformer was designed with sequential data in mind with its multi-head attention mechanism. Nevertheless, this experiment provides insight into the limits of the revolutionary transformer architecture, and satisfies the curiosity of those wondering about where those limits lie and asking the question "What is a transformer not the best at?"

5.3 Optimising model architecture: dropout and hidden layer analysis

In the process of constructing the transformer model, several experiments were conducted evaluating the effects of varying architectures on the performance of the model. In particular, specific experiments on the dropout value and the number of encoder layers were led.

Dropout was selected for experimentation as it is a prominent regularisation technique used to prevent over-fitting, and optimising the dropout rate encourages the model to be more robust and increase generalisability. The number of layers was experimented with due to it playing a crucial role in the model's capacity to capture complex relationships in the data, and in knowing how various complexities of the model perform on the data, it aids in understanding how complex the relationships actually are within the data features. Ultimately, the experimentation of hyperparameters is critical in the pursuit of an optimal model.

Other common hyperparameters were not experimented with due to individual reasons. The number of attention heads was assumed to not have a major influence on the model's performance as the data was not sequential. The learning rate was selected through "informal" testing and it was believed that the effect that varying learning rates has on the number of epochs that the model converges in, would lead to a biased experiment.

Table 3: Encoder layer number experiment results (50% dropout rate).

No. of encoder layers	Mean top-2 accuracy (%)
3	**86.43**
4	86.30
5	86.20
6	86.30
7	86.22
8	85.77

Table 4: Dropout value experiment results (6 encoder layers).

Dropout value (%)	Mean top-2 accuracy (%)
50	86.30
35	86.46
20	**86.89**

The two experiments (Tables 3 & 4) displayed that three hidden layers and 20% dropout performed the best. However, the increase is not very significant, with most variations in each test showing performances only slightly worse, indicating that for this model and dataset, tweaking these hyperparameters on a small scale is inconsequential. Further, it may have been noticed by the reader that the proposed "base" model in this project used neither of the optimal values discerned by this experiment. It was observed later to this experiment that the adoption of 20% dropout and three encoder layers resulted in a model that generalised terribly to the red wine data. As expected, however, with dropout being a predominate strategy to counter over-fitting.

6 Supplementary Material

6.1 Model Card

Model Card - Transformer to Predict Wine Quality

Model Details
- Developed by Atticus Johnston at The University of Texas at Austin in 2024.
- Transformer
- Trained on physicochemical properties of white wine data.

Intended Use
- Intended to be used for predicting a wine's quality rating based on certain physicochemical properties of that wine.
- Intended to be used by individuals with the ability to test various properties of wine and want an indication of the quality without requiring an expert assessor.

Factors
- The testing instruments and techniques used for gathering the data on the physicochemical properties of the wine.
- The expert assessor's ability in rating wine quality.
- Subjectivity in the wine quality ratings.

Metrics
- Evaluation metric is top-2 accuracy, which measures how many times the target wine quality is within the top two classes predicted by the model. This allows leniency due to the subjective nature of the wine quality targets.

Evaluation Data
- Wine quality dataset [2], white wine test data split and red wine data.
- Chosen for its completeness and large number of instances.
- All data normalised with min-max scaling, targets converted to one-hot encoding, white wine data split into testing set.

Training Data
- Wine quality dataset [2], white wine training and validation data split.
- Chosen for its completeness and large number of instances.
- All data normalised with min-max scaling, targets converted to one-hot encoding, data split into training and validation sets.

Ethical Considerations
- Could disrupt traditional wine quality prediction methods and potentially impact those in the industry.

Caveats and Recommendations
- Only trained on white wine data so does not generalise to red wines exceptionally.

Quantitative Analyses

White wine	86.30%
Red wine	76.28%

Table 5: Model performances on white and red wines.

6.2 Datasheet

MOTIVATION

For what purpose was the dataset created? Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.
The dataset was created with the purpose of predicting human wine taste preferences based on physicochemical properties of wines. The data was to fill the gap in the field of wine quality.

Who created this dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?
P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Associated with the University of Mihno and the Viticulture Commission of the Vinho Verde region.

COMPOSITION

What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.
They represent eleven measured physicochemical properties of wines, alonside a wine quality rating on a scale of 0 to 10.

How many instances are there in total (of each type, if appropriate)?
4898 instances of white wine. 1599 instances of red wine.

Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set? If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).
It is a sample containing only *vihno verde* wines from the Mihno region of Portugal. The larger set is all wines globally, and it is not representative since region plays a large factor in the attributes of wines and their qualities. It is unfeasible to gather the data on every wine in the world.

What data does each instance consist of? "Raw" data (e.g., unprocessed text or images) or features? In either case, please provide a description.
Raw data representing measurement results. Fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol, and wine quality rating.

Is there a label or target associated with each instance? If so, please provide a description.
The target is the wine quality rating. It is an integer.

Is any information missing from individual instances? If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.
No.

Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)? If so, please describe how these relationships are made explicit.
No.

Are there recommended data splits (e.g., training, development/validation, testing)? If so, please provide a description of these splits, explaining the rationale behind them.
No.

Are there any errors, sources of noise, or redundancies in the dataset? If so, please provide a description.
No.

Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)? If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a future user? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.
It is self-contained.

Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor-patient confidentiality, data that includes the content of individuals' non-public communications)? If so, please provide a description.
No.

Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety? If so, please describe why.
No.

Does the dataset relate to people? If not, you may skip the remaining questions in this section.
No.

COLLECTION

How was the data associated with each instance acquired? Was the data directly observable (e.g., raw text, movie ratings), reported by subjects (e.g., survey responses), or indirectly inferred/derived from other data (e.g., part-of-speech tags, model-based guesses for age or language)? If data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.
Directly observable. The data was recorded by a computer system that managed the process of wine sample testing and the final database was exported.

Over what timeframe was the data collected? Does this timeframe match the creation timeframe of the data associated with the instances (e.g., recent crawl of old news articles)? If not, please describe the timeframe in which the data associated with the instances was created. Finally, list when the dataset was first published.
May 2004 to February 2007.

What mechanisms or procedures were used to collect the data (e.g., hardware apparatus or sensor, manual human curation, software program, software API)? How were these mechanisms or procedures validated?
Unspecified for wine attributes. For the wine quality, each sample was evaluated by a minimum of three sensory assessors which graded the wine in a scale that ranges from 0 (very bad) to 10 (excellent). The final score was given by the median of these evaluations.

What was the resource cost of collecting the data? (e.g. what were the required computational resources, and the associated financial costs, and energy consumption - estimate the carbon footprint. See Strubell *et al.*[**?**] for approaches in this area.)
Unspecified.

If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)?
Not from larger set.

Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?
Unspecified.

Were any ethical review processes conducted (e.g., by an institutional review board)? If so, please provide a description of these review processes, including the outcomes, as well as a link or other access point to any supporting documentation.
No.

Does the dataset relate to people? If not, you may skip the remainder of the questions in this section.
No.

PREPROCESSING / CLEANING / LABELING

Was any preprocessing/cleaning/labeling of the data done(e.g.,discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)? If so, please provide a description. If not, you may skip the remainder of the questions in this section.
The database was transformed to include a distinct wine sample per row, and only the most common physicochemical tests were included to avoid discarding examples.

Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)? If so, please provide a link or other access point to the "raw" data.
No.

Is the software used to preprocess/clean/label the instances available? If so, please provide a link or other access point.
No.

Any other comments?
The data is separated in the two sets, one for each type of wine.

USES

Has the dataset been used for any tasks already? If so, please provide a description.
Yes, the creators of the dataset used it for prediction. There are many instances of individuals using it for own projects.

Is there a repository that links to any or all papers or systems that use the dataset? If so, please provide a link or other access point.
`https://www.kaggle.com/datasets/yasserh/wine-quality-dataset/code`. This provides 340 instances of people using the dataset.

What (other) tasks could the dataset be used for?
Predicting whether a wine was red or white based on its physicochemical properties.

Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses? For example, is there anything that a future user might need to know to avoid uses that could result in unfair treatment of individuals or groups (e.g., stereotyping, quality of service issues) or other undesirable harms (e.g., financial harms, legal risks) If so, please provide a description. Is there anything a future user could do to mitigate these undesirable harms?
No.

Are there tasks for which the dataset should not be used? If so, please provide a description.
No.

DISTRIBUTION

Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created? If so, please provide a description.
Yes, it has been published online.

How will the dataset will be distributed (e.g., tarball on website, API, GitHub)? Does the dataset have a digital object identifier (DOI)?
The dataset was published on the UCI Machine Learning Repository. DOI: 10.24432/C56S3T

When will the dataset be distributed?
Published on 10/6/2009.

Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)? If so, please describe this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.
The dataset is licensed under a Creative Commons Attribution 4.0 International license. This allows for the sharing and adaptation of the datasets for any purpose, provided that the appropriate credit is given.

Have any third parties imposed IP-based or other restrictions on the data associated with the instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms, as well as any fees associated with these restrictions.
No.

Do any export controls or other regulatory restrictions apply to the dataset or to individual instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any supporting documentation.
No.

MAINTENANCE

Who is supporting/hosting/maintaining the dataset?
UCI Machine Learning Repository,

How can the owner/curator/manager of the dataset be contacted (e.g., email address)?
Unspecified.

Is there an erratum? If so, please provide a link or other access point.
No.

Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)? If so, please describe how often, by whom, and how updates will be communicated to users (e.g., mailing list, GitHub)?
It has not been since 2009, so unlikely.

If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were individuals in question told that their data would be retained for a fixed period of time and then deleted)? If so, please describe these limits and explain how they will be enforced.
Does not relate to people.

Will older versions of the dataset continue to be supported/hosted/maintained? If so, please describe how. If not, please describe how its obsolescence will be communicated to users.
No.

If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so? If so, please provide a description. Will these contributions be validated/verified? If so, please describe how. If not, why not? Is there a process for communicating/distributing these contributions to other users? If so, please provide a description.
No.

References

[1] P. Cortez, A. L. Cerdeira, F. Almeida, T. Matos, and J. Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47:547–553, 2009. URL https://api.semanticscholar.org/CorpusID:2996254.

[2] P. Cortez, A. L. Cerdeira, F. Almeida, T. Matos, and J. Reis. Wine Quality. UCI Machine Learning Repository, 2009. URL https://doi.org/10.24432/C56S3T.

[3] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Neural Information Processing Systems*, 2017. URL https://api.semanticscholar.org/CorpusID:13756489.