Explainable AI
Visualization is all you need

Maria Ribalta
maria.ribalta@estudiantat.upc.edu

Pere-Pau Vázquez
pere.pau.vazquez@upc.edu

Abstract

The Transformer is one of the most famous architectures in NLP tasks since the publication of Attention is all you need in 2017. Explainable AI supplies the need to understand algorithms and artificial intelligence concepts that tend to work in a black-box way.

In Visualization Is All You Need, this rising discipline and the NLP top architecture have been merged to explain to users with basic notions on Neural Networks, everything that it is necessary to know on what is a Transformer and how it works.

The result is an interactive self-contained document that has been meticulously designed and created to approach the state-of-the-art architecture to the readers, with real examples and data extracted from a pre-trained Transformer in the task of Machine Translation from English to French.

1. Introduction

Ever since the publication of *Attention is all you need* [16] in 2017, the Natural Language Processing field (NLP) has been presenting state-of-the-art results with the use of the architecture introduced in the paper, the Transformer. It is currently one of the best models to solve many natural language tasks as Machine Translation, leaving behind far more complex models such as Recurrent Neural Networks (RNNs) or Convolutional Neural Networks (CNNs).

On the other hand, with the ever-growing availability of data and models that perform tasks in any field one can imagine, even outperforming the human knowledge and abilities; understanding the machines and algorithms used becomes a fundamental part of the field. Explainable Artificial Intelligence (XAI) tries to approach these complex and black-box-alike concepts in a visual and basic, yet informative, manner via the visualization of some features or results of Artificial Intelligence models.

In this project, an interactive and self-contained document that merges these two concepts has been created with the following objectives:

- Explain the architecture introduced in *Attention is all you need* [16] in a basic yet precise way via an interactive document.
- Use XAI to follow the sections covered in the paper, making them understandable for an audience that has basic notions on Deep Learning and/or Neural Networks.
- Clarifying and explaining clearly the main parts, features and characteristics of it through charts, plots and diagrams that allow to visualize them in a approachable way rather than mathematical complex expressions.
- Provide more detail on those sections which are not as specified in the paper but are helpful when trying to understand the nature behind the network.

The interactive document is addressed to a target reader that is not expected to have expertise on the subject but some elementary knowledge on Neural Networks. As for the Transformer essentials, that is covered in the document itself.

In contrast, this paper intends to summarize the the process and work behind the final web page under the title *Visualization is all you need* [2][1].

2. Related work

XAI is the discipline that tries to ease the task of understanding powerful and complex AI tools and models. Every day, a new paper can be found about a new architecture or algorithm that makes a task easier, faster or even better than a human would do. However, as Samek et al. [13] mention in their paper, said models are usually applied in a black box manner, where no information is provided on how they arrive to their predictions. This may cause a lack of transparency and/or comprehension that can eventually result in bad solutions, useless networks or even wrong diagnosis. Hence it is fundamental to acquire the necessary explanations of such intriguing methods.

[1] The link to the interactive document is available on the References

This field is starting to obtain more importance every time, however there are still few sources about Explainable AI or few libraries that permit it. A good example on interactive XAI is the *Beginner's Guide to Dimensionality reduction* [5] in which the user interacts with the page to get the insights of Dimensionality Reduction algorithms. Or the page *Explainable AI + VIS* [18] that contains several examples focused on different models and concepts.

Specifically for the Transformer, one can find *The illustrated Transformer* [3], a document that explains thoroughly and in great detail each of the parts of the model. This document contains basic illustrations and some animations to show the complex methods and concepts the original paper has. Another worth-mentioning example is *The Annotated Transformer* from Harvard's NLP Group [6], less extensive and detailed than the previous one, with more technical vocabulary and shortened visual approach, but very recommendable for those who are interested in the implementation.

However, one might want to have a level in between of both articles: visual but not as detailed as the first one, and technical and concise enough so that the main features and the formal concepts are internalized. In addition to this, none of them is interactive and both structures are rigid. This project fulfills the characteristics lacking in the aforementioned articles.

3. Methodology

In this section the methodology steps will be covered. They can be summarized as Understanding, Design, Extraction, Creation and Display.

3.1. Understanding the Transformer

The first task was to understand the algorithm, the architecture and the parts involved in all the process a Transformer performs, aiming special attention on the task of translation. In this step, it was also defined which model would be used.

The used Transformer implementation was from `fairseq` [12]. It can be found under the name `transformer.wmt14.en-fr` and it is already pretrained with the *ACL 2014 ninth workshop on statistical machine translation dataset* [1].

It also counts with an own tokenization algorithm and BPE to reduce the size of the data. However the modifications this last compression method may cause are not visible or reflected in any way in the final document as it was considered unnecessary.

In addition, no fine-tuning was applied to the architecture since the task of translating from English to French was already suitable and sufficient for the objectives of the project.

3.2. Design and drafts of the interactive plots

The second step was to investigate and analyze which parts were less clear and needed to be further detailed. Moreover, which of them could be shown with an interactive visualization.

The visualizations were divided into two subgroups: interactive plots and diagrams.

For the first one, plots, it was enhanced simplicity over detail, ensuring that the user understood the basis and introducing the detail if it was needed and possible. Some of this tips were extracted from *Visualization Best Practices for Explainable AI* [15] and some from Tufte's Fundamental Principles found in *The Visual Display of Quantitative Information* [14] on the need to focus on the content and data importance.

For the second one, diagrams, the colors used on them were the same as the ones displayed in the paper (except for the Attention diagrams due to the color palette used in the plots). In all, the color schemes are constant according to the part they represent. That is:

- Green for the Embeddings.
- Red for Positional Encoding.
- Blue for the Feed Forward layers.
- Yellow for the Normalization layers.
- Purple for the Attention.

3.3. Extraction of the data

The extraction of the data and the creation of the plots were highly codependent with each other as, in more than one occasion, the data extracted was not what it was expected to be, so it was required either to adapt the plots to the data or repeat the data extraction modifying the output.

Even though the library of the model offers an option to extract the data automatically, it was quite limited since it returned very simplified vectors or incomplete parts of what was desired for the interactive charts. Not to mention that not all the data that was interesting for the project was available via this option. Hence, the code was modified directly and manually to get it.

In addition to this, not only it was necessary to edit the source code of the Transformer itself, but also the client it uses to load the model and display the results, since they were apart and some key elements (like the token transformation into word) were only performed in the client.

For each sentence used in the document it was extracted the following data:

- The embeddings of the output of the Encoder. They were used to plot the Embeddings and Tokens subsections in the Basics section of the document.

- The Attention matrices from the Self-Attention Block from the Encoder and the Encoder-Decoder Attention from the Decoder, found in the Attention section of the document.

Aside from this, the weights applied for the Positional Encoding were extracted directly from the network rather than plotted from the mathematical expression.

Some more data was extracted but eventually not used.

3.4. Creation of the plots, charts and diagrams

Even though the ideas and structure of the charts and interactive visualizations were already defined, they were later adapted to the data format (and eventually the web page format too). The library used for the interactive charts was `Altair` and an open-source software called `draw.io` for the diagrams and animations. This step in the methodology until the last one, not included, have represented most of the time of this project.

3.4.1 Plots and charts

To create the plots as simplified and understandable as possible, it was decided to set some of the Transformer's parameters fixed for all designs, that is:

- The **batch** dimension was set to 1, so that each time only one sentence was used as input rather than a document with all of them.
- The **beam** dimension was set to 1. To avoid the extraction of unnecessary data.
- The **heads** dimension for the Multi-Head Attention was aggregated using the mean, as the code did with the last layer of it.
- The **end-of-sentence token** was kept when saving the data but omitted in the plots.

As a matter of fact, the original idea was to set the same font of the plots as in the web page, however it was discarded due to internal issues with the library when rendering this.

3.4.2 Diagrams and animations

The explanatory diagrams found in the Architecture section were specially designed for this project. It was done this way for several reasons:

First, this way it was ensured that the parts that were not as clear or simplified enough in the paper appeared as such in the document.

Second, so that the format, size and shapes could be adapted to follow the design and structure the document had.

Diagrams' colors, distribution, shape and orientation were chosen as similar to the original paper as possible, to ease the task of comparing both. Otherwise, designing a complete different structure, even explaining the same concepts, would difficult the reader to relate the parts or link ideas.

The only format that was not respected or displayed similarly were the residual connections, which appeared merged in *Attention is all you need* [16] along with the normalization layer. It was found proper to enable this difference as it does not modify excessively the diagram or wards off from the original appearance. In addition to this, the design implemented in the interactive document is visually more clear and it is widely used in the literature.

The comparison of the architecture diagrams is visible in Figure 1.

3.5. Display within an interactive document

In the last step, a web page was created in `HTML/CSS/JS` code using a template as a baseline, adapting the number of sections, the distribution of them, the design they had, etc.

3.5.1 Language

The language used needed to be concrete but basic enough so that the reader could understand all the characteristics of the Transformers with the proper concepts rooted. To do so, it was all written with technical and concise vocabulary but also an informal and rather colloquial tone. This way, it could be ensured that all the technicalities were covered but also comprehended.

4. Results

The front-end created for the results can be found in the web page *Visualization Is All You Need* [2]. The final plots will be introduced in this section, stating clearly how and why they were designed, what interactions the user has and the evolution of the charts until the final representation.

4.1. Embeddings and tokens

The embeddings included in the plots were extracted from the output of the Encoder, allowing the correspondence of each embedding with its token in English. The original dimensonality of the data was initially $Ba \times T_{input} \times E$, being Ba the batch size, T_{input} the dimension of the input (English tokens) and E the size of the embeddings (1024). The palette applied was `darkgreen` and the green color represents the embeddings during all the document.

In the eventual dimensions, the Ba was eroded as it was already set to 1 and could be dismissed, T_{input} was con-

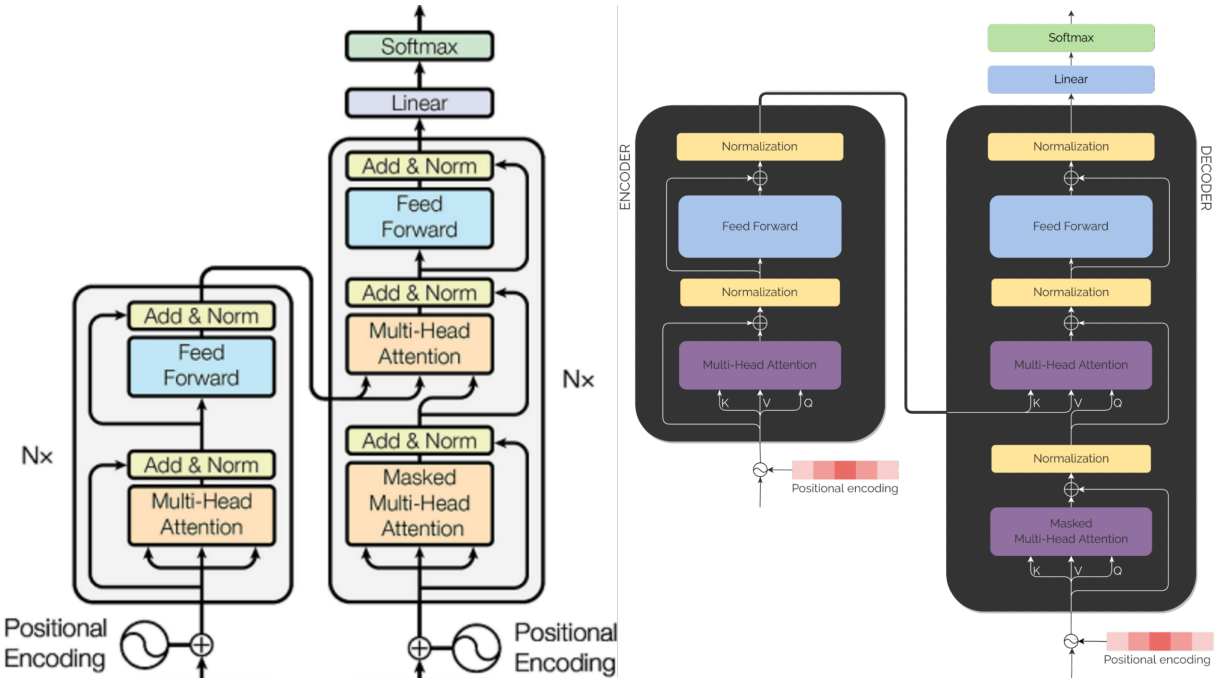

Figure 1. Original paper and current architecture diagrams

served and E was limited to 150 by truncation, as it already showed the concepts that needed to be explained.

The interaction enabled counts with a display of the value when hovering over the boxes and a dropdown with a sentence to display in the chart. In this case, two of the dropdown options were the subtraction of two 1-different-word pair of sentences to see the similarity between equal and different embeddings.

The same design of this chart is used in the Basics section for the Embeddings and Tokens subsections. However, the first one has the information of the tokens of a same word aggregated using the mean. The second one displays the tokens as found directly in the data returned by the architecture.

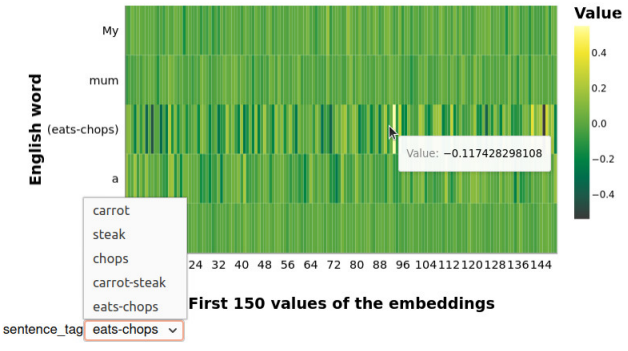

Figure 2. Interaction in Embeddings and Tokens plots

4.2. Attention

As can be found in many articles, such as in *Attention and its different forms* [9], there are different types of attention and three of them are used in the original paper. Furthermore, understanding single-headed attention is crucial to comprehend what Multi-Head Attention stands for. All in all, it was considered easier to explain the basics first as is done in *Attention and Augmented Recurrent Neural Networks* [11] or in *Single Headed Attention RNN: Stop Thinking With Your Head* [10].

To ease the task to the user, it was thought to introduce first the basic notions of Attention (*e.g.* the main formula, the concepts of K, Q and V, etc), the classes and finally a brief explanation about Heads and Multi-Head Attention.

The design and creation of the Attention plot was the most time-consuming of all the interactive charts displayed in the document. The data used in it was the output of the last Attention layer, located on the Self-Attention block found in the Encoder and on the Encoder-Decoder Attention block found in the Decoder.

The dimensions of the original output tensor were $T_{output} \times H \times Ba \times Be \times T_{input}$ where T_{input} and T_{output} were the number of tokens of the input and output respectively, H the number of heads, Ba the batch size and Be the beam dimension. To get the $3d$-array desired for the heatmap, the H dimension was aggregated with the mean operation and the Ba and Be dimension were deleted as they were settled to 1, as it has already been mentioned.

At first, it was included the last token dimension (the token indicating end-of-sentence) in both T_{output} and T_{input}, however this condensed the highest values of Attention and the plot appeared excessively highlighted for those values. This did not allow the user to focus on the interesting tokens, the ones that represented words, so the last token information was deleted.

After it, the color-scheme was selected so that the difference between high and low values of the legend were clearly distinguishable, palettes of a single color with a darkened (or brightened) gradient were not different enough. The eventual color scheme was `magma`, with black colors for low values, purple for middle-ranged values and yellow for the highest ones. One can easily observe in Figure 3 how the values of Attention of the last tokens of the sentence are clearly more different in the magma colors than in the previous palette.

It was in this case in which the colors were varied from those displayed in the original paper, since they appeared orange there but were settled purple in here. The whole process of design can be observed in Figure 3.

Finally, the interactions found are similar to the ones in Embeddings and/or Tokens. A display of the Attention value when hovering, along with the corresponding English and French token (only English in the case of Self-Attention). A dropdown with different sentences the user can choose to observe the behaviour followed by this block when changing the type of sentences, structures, etc.

4.2.1 Heads

Heads might be the one concept in the paper that appears too few for the importance and impact it has. In this project, it was considered necessary to explain deeply, but plainly, what they were and the role they played within the architecture and training, as it is one of the key points to achieve the speed and results it yields.

Unlike the visualization and code introduced by J. Vig [17], the possibility of plotting heads was discarded, as conceptually they did not provide much insight nor helped the understanding of the idea itself. One has to take into account that plotting heads is the visualization of an abstract process of an already complex and black-box-like concept.

In other words, heads are the number of partitions that an embedding (an already non-intuitive unit) suffers at a certain point after being forwarded along the layers (some non-linear mappings). Eventually, what is shown is something at such levels of complexity and transformations that the human intuition is not able to understand or relate. Not to mention that, by showing them, one might have more guidance on how the embeddings distribute or summarize the information of the word rather than clarifying the meaning of the head concept itself.

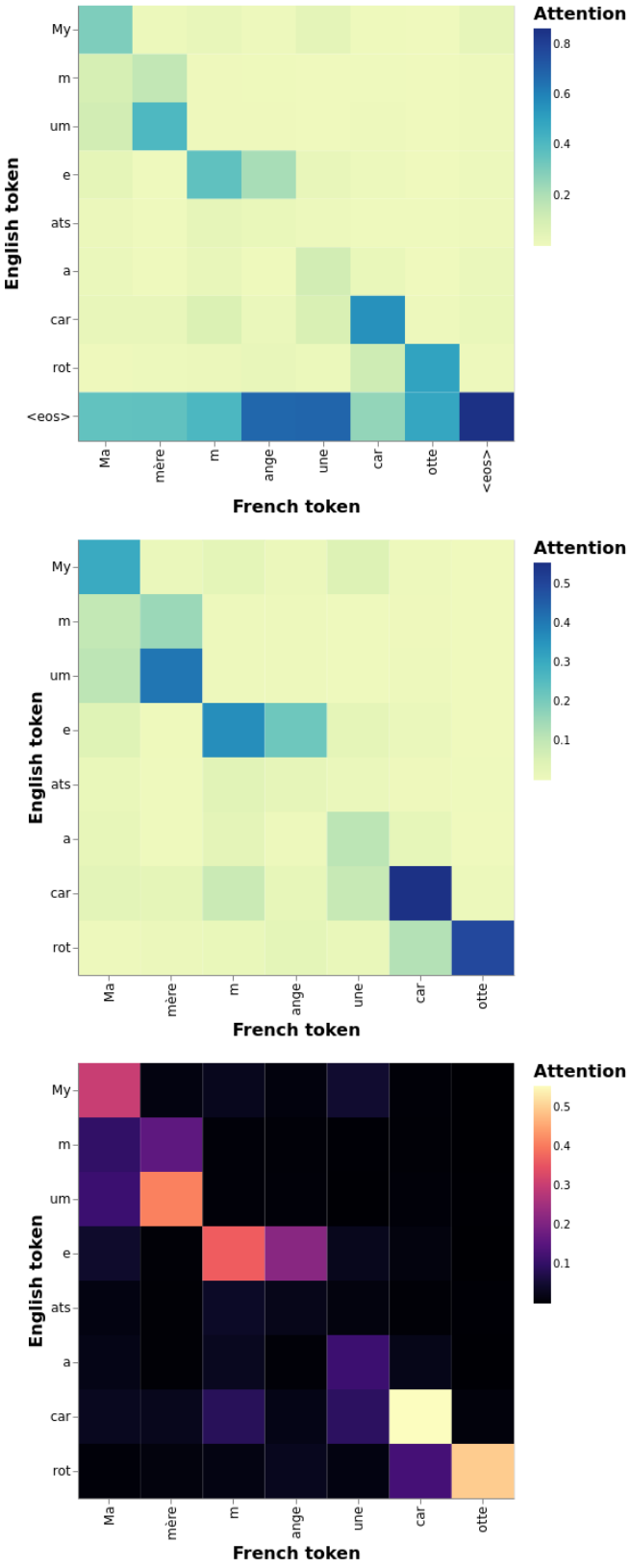

Figure 3. Attention Plot Evolution

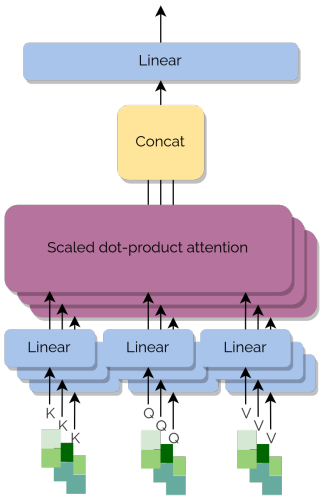

Figure 4. Last step of the Multi-Head Attention animation

Hence, to visualize how heads interacted with the architecture and its role within, it was proposed an animation, similarly as Jay Alammar [4] does for the Transformer forward process, but with the Multi-Head Attention process instead. The animation covers the process any vector performs once entering the Multi-Head Attention Block, providing special detail in the division of the embedding and distribution among the layers. At the same time, the design of the diagrams is identical to the shown in the previous section so that the reader is already familiar with them and is able to focus on the process rather than the architecture.

4.3. Positional Encoding

When talking about Positional Encoding, one finds itself in the opposite case of the Head concept. Its meaning and task is wide clear and easily understood in the article, nonetheless its usage is not the most known or characteristic of the architecture.

The idea is simple, adding weights to the word representations so that the machine has an idea of the order of the sequence, either by training said weights or aggregating a sinusoidal-like function, since according to the authors the results are pretty similar.

However, the reason why a sinusoidal behaviour is related in any way with a sentence's order is not found anywhere. To understand this, it was considered to expand the explanations of the original paper as does A. Kazemnejad [8] with the example of the bits or Jafar Ali Habshee [7] by displaying the mathematical expression in different dimensions. Hence, the final result combined the plots of the function (less detailed than the article mentioned) and a simplified example with bits.

In this case, it was chosen a color scheme based upon a red color with darkened gradient. Unlike the cases of the Embeddings, Tokens and Attention there was no need to highlight those boxes with clearly different values; a smooth change was suitable enough. The palette `reds` was perfect for this.

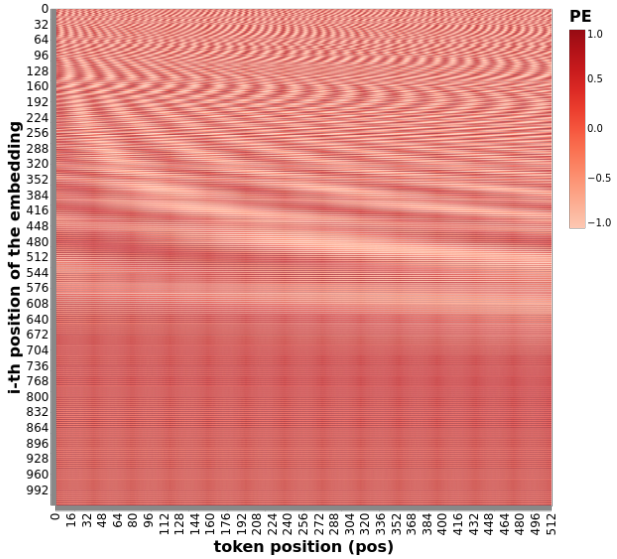

Figure 5. Positional Encoding Plot

5. Conclusions

The raising availability of new Machine Learning and Deep Learning models needs, at the same time, the availability of methods and documents that explain them.

Visualization Is All You Need provides an interactive document for anyone who wants to understand the logic and mechanics behind the Transformer models.

The charts and diagrams have been specially designed for this project and allow any user with basic knowledge on Neural Networks to understand in an easy, visual and interactive manner the architecture and functioning of it.

The keys of the project have been the initial deep understanding of the network and the specific and detailed design of each of the visualizations, either plots or diagrams. Enhancing, overall, a point of view of simplicity and homogeneity to easily link concepts in a technical but simple approach that allow the user to understand the basics of the state-of-the-art.

References

[1] Acl 2014 ninth workshop on statistical machine translation. `http://statmt.org/wmt14/translation-task.html`, 2014.

[2] Visualization is all you need. `https://www.cs.upc.edu/~ppau/XAI/XAI_transformer/`, 2021.

[3] Jay Alammar. The illustrated transformer. 2018.

[4] Jay Alammar. Visualizing a neural machine translation model (mechanics of seq2seq models with attention), 2018.

[5] Matthew Conlen and Fred Hohman. Beginner's guide to dimensionality reduction, 2018.

[6] Harvard NLP Group. The annotated transformer. 2018.

[7] Jafar Ali Habshee. On positional encodings in the attention mechanism. 2020.

[8] Amirhossein Kazemnejad. Transformer architecture: The positional encoding. 2019.

[9] Anusha Lihala. Attention and its different forms. 2019.

[10] Stephen Merity. Single headed attention rnn: Stop thinking with your head, 2019.

[11] Chris Olah and Shan Carter. Attention and augmented recurrent neural networks. *Distill*, 2016.

[12] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

[13] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models, 2017.

[14] Edward R. Tufte. *The Visual Display of Quantitative Information*. 1983.

[15] Jen Underwood. Visualization best practices for explainable ai, 2019.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[17] Jesse Vig. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–42, Florence, Italy, July 2019. Association for Computational Linguistics.

[18] HKUST VisLab. Explainable ai + vis, 2018.