Identifiability

*Note: Sub-titles are not captured in Xplore and should not be used

1st Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

2nd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

***Abstract*—Here the abstract**
***Index Terms*—**

I. Introduction

Recent advances in the field of natural language processing (NLP) are partly due to the use of the Transformer architecture, first introduced in [1]. Underlying the operation of Transformers, which are capable of producing not only highly human-like language, but also of performing a plurality of tasks ([2], [3], [4]), is the attention mechanism [5]. Initially, this mechanism was used in receptive neural networks (RNNs), but the transformer architecture has taken over. The basic idea is to allow the model to weight the importance of each word within a sequence according to its context. In this way, the model can capture long-range relationships between words without relying on rigid sequential structures. Put simply, the attention mechanism observes a sequence of inputs and decides, at each step, which other parts of the sequence are important. In mathematical terms ([6]), given $m \in \mathbb{N}$ pairs of key and value vectors (k_i, v_i), $k_i, v_i \in \mathbb{R}^d$ where d is the dimension of the embedding, for a given query vector $q_j \in \mathbb{R}^d$ the attention mechanism computes an output vector o_j as

$$o_j = \sum_{i=1}^{m} \alpha(q_j, k_i) g(v_i).$$

The function $g(\cdot)$ is a linear application from $\mathbb{R}^d$ to $\mathbb{R}^d$, while function $\alpha(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$ computes the attention weights. For instance, in the case of transformers,

$$\alpha(q_j, k_i) = \frac{exp(\langle q_j, k_i \rangle)}{\sum_{l=1}^{m} exp(\langle q_j, k_l \rangle)}, \qquad g(v_i) = v_i.$$

Identify applicable funding agency here. If none, delete this.

In literature, the function α is generally referred to as *attention pooling* and it can be thought as a similarity measure. In general we can ask α to have various properties, but in general, for a fixed query vector q_j, we require the weights $\alpha(q_j, k_i)$ to form a convex combination, i.e.,

$$\sum_{i=1}^{m} \alpha(q_j, k_i) = 1, \quad \alpha(q_j, k_i) \geq 0 \qquad j = 1, \ldots, m.$$

With this notation, we can see the intuition behind the attention: for simplicity assume the function α to be the normalized dot product,

$$o_j = \frac{1}{C}[\langle q_j, k_1 \rangle g(v_1) + \ldots + \langle q_j, k_m \rangle g(v_m)],$$
$$C = \sum_{i=1}^{m} \langle q_j, k_i \rangle.$$

Now if, for example, q_j attends mostly k_1, then

$$o_j \approx \frac{1}{C}[\langle q_j, k_1 \rangle g(v_1)] \approx g(v_1).$$

Therefore, we can think of attention weights as a "guide map" to values vectors $g(v_i)$.

The Transformer architecture employs the so-called *self-attention* to transform an embedded input sequence $z = \{z_1, \ldots, z_n\}$ to a contextualized output sequence $y = \{y_1, \ldots, y_n\}$ by using α to capture how much each token contributes to the contextualization of each other token. One of the most debated aspects in analyzing Transformers is the actual role of attention weights ([7]). While they are often interpreted as a key indicator of how the model distributes importance among input tokens, recent studies have questioned their centrality. Some works suggest that AWs (Attention weights) might be artifacts of the model's parameterization rather than reliable representations of its reasoning process ([8]). Furthermore, the issue of identifiability in Transformers

adds another layer of complexity. Specifically, the existence of multiple attention weight configurations that yield the same output, as shown in [9], raises questions about the extent to which these weights can be uniquely attributed to specific decision-making processes. This ambiguity highlights the need for a more nuanced understanding of how attention mechanisms function and whether they provide meaningful interpretability or simply reflect mathematical symmetries inherent in the architecture.

II. Transformers and Identifiability

The transformer architecture is made of several pieces: the Multi-Head attention, the Add&Norm layer and the Feed-Forward layer. Here we combine the transformed input with the original input via residual connections. For a more complete discussion, we refer to [1] and [10]. Following the notation and argumentation of [9], [11] and [12] we focus on the identifiability AWs, specifically in the study of the following linear application

$$\begin{aligned} f\colon \mathbb{R}^{d_s\times d_s} &\to \mathbb{R}^{d_s\times d} \\ A &\mapsto AT \end{aligned} \tag{1}$$

where the attention matrix A is defined as:

$$A = \textit{softmax}\Big(\frac{QK'}{\sqrt{d_q}}\Big) \in \mathbb{R}^{d_s\times d_s}$$

and $Q \in \mathbb{R}^{d_s\times d_v}$ is the query matrix, $K \in \mathbb{R}^{d_s\times d_v}$ is the key matrix, the symbol $'$ denotes the transposed matrix, $V \in \mathbb{R}^{d_s\times d_v}$ is the value matrix. The term d_s represents the input sequence length, while $d_v = d/h$ is the head dimension which is simply the ration of the embedding dimension over the number of heads. Finally, matrix $T = VH \in \mathbb{R}^{d_s\times d}$, where $H \in \mathbb{R}^{d_v\times d}$ is the concatenation of the various heads.

We recall that identifiability refers to the property whereby a model's parameters can be uniquely determined given the observed outputs. Formally, a model is identifiable if, for any two parameter sets θ_1 and θ_2, $f(x,\theta_1) = f(x,\theta_2)$ implies $\theta_1 = \theta_2$. If we relate the previous concept in computational terms, following the paper [11], we have that given two matrices A_1, A_2 such that $||A_1 - A_2|| < \lambda$, the predictions should be more or less the same for a given task. This concept is clearly related with the injectivity of the map f, for which we have the following proposition:

Proposition II.1. *f is injective iff $Ker(T') = \{0_{d_s}\}$.*

Moreover, taking into account

Remark II.1. *Let $f\colon X \to Y$ be a linear transformation between vector spaces and let $dim(X) = n$ and $dim(Y) = m$, with $n, m \in \mathbb{N}$.*

- *If the function f is injective, then $n \leq m$;*
- *if the function f is surjective, then $m \leq n$.*

[9] shows that a sufficient condition for f to fail to be 1-1 and for AW to be nonidentifiable is that: $d_s > d_v$. Both papers [12] and [11] tries to restore the identifiability of AW. First of all they note that it is not sufficient to prove that there exists $\tilde{A}$ such that

$$(A + \tilde{A})T = AT$$

because it is not guaranteed that the AW matrix $A + \tilde{A}$ comes from the attention computation. Indeed the softmax generates a probability distribution over the token, hence

$$\begin{aligned} (A + \tilde{A}) &\geq 0, \\ \tilde{A}T &= 0, \\ \tilde{A}\mathbf{1} &= 0. \end{aligned} \tag{2}$$

Moreover, in [12] they prove that

$$\text{rank}(A + \tilde{A}) \leq d_k, \tag{3}$$

where d_k is the size of key vectors and typically $d_k = d_v$.

A. Proposed solutions to restore identifiability

Paper [12] proposes two possible solutions: Contrary to the regular Transformer setting where $d_k = d_v$, a possibility is to decrease the value of d_k as it will reduce the possible solutions $\tilde{A}$ of (3) beucase we are putting harder constraints on the rank of attention logits. Another possibility would be to keep the value vector size and token embedding dimension to be more than (or equal to) the maximum allowed input tokens, i.e., $d_v \geq d_{s-max}$. On the other hand, in [11] they tackle this issue on another perspective. Their approach can be presented as it follows: given f such that $f(A) = AT$, we want to determine a second function g such that when we compute $g(A)$ then $g(A)$ is identifiable w.r.t to f. In the above g is the projection of A onto $Ker([T,1]')^{\perp}$. They show that even if $A_1T = A_2T$ with $A_1 \neq A_2$, $proj_{Ker([T,1]')^{\perp}}(A_1) = proj_{Ker([T,1]')^{\perp}}(A_2)$ and so restoring the identifiability in this space.

B. Limitations and criticality of the solutions

The solution of Naim and Asher in [11] can be achieved in many ways. One very simple possibility is to

define the map ϕ as the projection of A onto the equivalence class $[A]$ induced by the following equivalence relation

$$A_1 \sim A_2 \iff f(A_1) = f(A_2).$$

Clearly if $A_1T = A_2T$, then $\phi(A_1) = \phi(A_2)$ and therefore we restore the identifiability of AW matrix. However, It is noticeable that there are infinitely many solution as any equality generates a equivalence relationship which induces a projection map similar to ϕ. Another example, would be

$$A_1 \sim A_2 \iff f(A_1) = f(A_2) \text{ and } A_1\mathbf{1} = A_2\mathbf{1} = 1$$

or again,

$$\begin{aligned} A_1 \sim A_2 \iff & f(A_1) = f(A_2), \\ & A_1\mathbf{1} = A_2\mathbf{1} = 1, \\ & A_1 \geq 0, A_2 \geq 0. \end{aligned}$$

Among the limitations present on [11] some are addressed by [12], by acknowledging how the process for generating attention relates to the key (K) and query vectors (Q) obtaining the constraint (3). However neither of the studies notices a potential circularity problem implicit in the previously described equivalence relation. The map f is derived from $T = VH = (EW^v)H$, where E is the resulting token embedding $E \in \mathbb{R}^{d_s \times d}$. It is essential to underline that maintaining the same T may be detrimental as we keep E fixed while A_1, A_2 are highly dependent on the input:

$$\textit{input} \to E \to A \to \textit{output}.$$

This means that the way each token is embedded, portrayed in E, has a potential influence in $Ker(T')$. Under the analysis of [11] and [12] this issue is ignored. Meaning that an equivalence relationship $A_1 \sim A_2$ actually is dependant on the specific embedding $A_1 \sim_E A_2$, as W^vH are fixed during evaluation. If we were to use this equivalence relationship in identifying the relevant components of A, we could be discarding components of the attention attributing this behaviour to the model without considering the semantic influence of the specific tokens. For example, we could have two similar entries corresponding to E_1, A_1 and E_2, A_2 such that $A_1 \sim_{E_1} A_2$ but $A_1 \not\sim_{E_2} A_2$.

III. Exploiting the structure of AW matrix

The definition of identifiability used in the literature often relies on perturbation analysis of the A matrix. According to this definition, whenever $||A_1 - A_2|| \geq \lambda$, distinct predictions should be expected. However, this assumption is fragile as it fails to account for the information structure embedded in the attention matrices.

First, it is essential to underline that any analysis performed on AW matrices is inherently task-dependent, making it difficult to discuss them in purely general terms. Furthermore, some studies reveal that AW matrices often exhibit clear structures arising from the training process. For example, in [13], the authors observe that in translation tasks, the most important and confident heads play consistent and often linguistically interpretable roles, which are reflected in specific structures of the AW matrices. In paper [14] the authors show that BERT's attention heads exhibit patterns such as attending to delimiter tokens, specific positional offsets, or broadly attending over the whole sentence, with heads in the same layer often exhibiting similar behaviors.

Consider the following extreme task: given an input sequence, the output is simply the sequence in reverse. For instance, given the sequence "The sun is shining," the output is "shining is sun The." In this case, the attention weight matrix is antidiagonal, roughly of the form:

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

In tasks where AW matrices exhibit such distinct structures, simple perturbations of A cannot be assumed to yield the same output, even under identifiability constraints. We observe two key issues:

- The choice of the norm $||\cdot||$ is crucial in defining when two matrices are close,
- It is necessary to consider whether a perturbation disrupts the inherent structure of the matrix.

To continue with the above example, consider the norms

$$||A||_1 = \max_{1 \leq j \leq d_s} \sum_{i=1}^{d_s} |a_{ij}|$$

(the maximum absolute column sum of the matrix) and

$$||A||_\infty = \max_{1 \leq i \leq d_s} \sum_{j=1}^{d_s} |a_{ij}|$$

(the maximum absolute row sum of the matrix). Using these norms, the matrices

$$A_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad A_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

have the same norm but encode entirely different information for the given task. In light of the above, it is critical to reconsider the definition of identifiability by incorporating the structural properties of attention matrices specific to a task. This perspective allows us to rigorously define perturbations and 'similar' matrices in a way that aligns with the task's requirements.

A. Definition of structure-aware norm

To encode the structure S of AW matrices for a given task in a norm, we define the following norm

$$||A||_S = \frac{1}{d_s^2} \sum_{i,j} w_{ij}|a_{ij}| \quad w_{i,j} > 0 \tag{5}$$

where w_{ij} is the weight for that entry. First of all let us prove that (5) defines a norm on $\mathcal{M}_{d_s}(\mathbb{R}_{\geq 0})$, i.e., the set of square matrices of dimension d_s with non-negative real entries.

Proposition III.1. (5) *defines a norm on* $\mathcal{M}_{d_s}(\mathbb{R}_{\geq 0})$.

Proof. First of all let us prove the positive definiteness property: if $||A||_S = 0$ then $A = 0$. This is straightforward as each $w_{ij}|a_{ij}|$=0 and since $w_{ij} > 0$, this implies that $|a_{ij}| = 0$ for any i, j. Now $\forall \lambda \in \mathbb{R}$,

$$||\lambda A||_S = \frac{|\lambda|}{d_s^2} \sum_{i,j} w_{ij}|a_{ij}| = |\lambda| ||A||_S$$

which proves the absolute homogeneity. Lastly, to prove the triangle inequality, it is sufficient to observe that given two matrices A, B,

$$w_{ij}|a_{ij} + b_{ij}| \leq w_{ij}|a_{ij}| + w_{ij}|b_{ij}|$$

as a consequence of the triangle inequality on the absolute value. □

Coming back to the example of A_1 and A_2 in (4) if we let $w_{ii} = 1$ and $w_{ij} = \epsilon$ for $i \neq j$, with $\epsilon > 0$ relatively small, then noticing that $d_s = 4$

$$||A_1||_S = \frac{1}{4}, \quad ||A_2||_S = \frac{\epsilon}{4}$$

and so $||A_1 - A_2||_s \not\approx 0$ as it takes into account the structure of the matrices. On the other hand,

$$A_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad A_2 = \begin{bmatrix} 0.1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0.1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

as they encode the same information they share similar norm $|| \cdot ||_S$

B. Extracting the structure

In order to obtain the relevant structures present in attention we can take varied approaches. A possibility is to take inspiration from [11]. As explained in II-B, the equivalence relation $A_1 \sim A_2$ implicitly assumes a common embedding, i.e., $A_1 \sim_E A_2$ which can lead to an inconsistency such as $A_1 \sim_{E_1} A_2$ but $A_1 \not\sim_{E_2} A_2$. A simple way to avoid this is to isolate the impact E has on the $ker(T')$ for a specific instance. We propose the following strategy: Define the subspace R

$$R = \Big(\bigcap_i^N ker(T_i')\Big)^{\perp}, \tag{6}$$

where N is the training set size, and T_i are the T obtained from each sample i in the training set. Then we can build an orthonormal basis of R and the elements of this basis $B = \{S_1, ..., S_n\}$ with $n = dim(R)$ would be the relevant structures present in the attention.

Note that in this way we can isolate the effect E has on each instance by comparing $ker(T')^{\perp}$ to R. For example, assuming $dim(ker(T')^{\perp} \setminus R) \neq 0$, the structures contained in $proj_{ker(T')^{\perp} \setminus R}(A)$ are specifically sent to zero, i.e., ignored in that head due to the meaning encoded in the embedding of the tokens.

$$A_1 \sim A_2 \iff proj_R(A_1) = proj_R(A_2)$$

1) A brute force approach: In this section we present a practical approach to extract the general structure of AW matrices. In order to apply formula (5) and hence compute the structure-aware norm, we must be able to know the entries of the following matrix:

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1d_s} \\ \vdots & \vdots & \cdots & \vdots \\ w_{d_s1} & w_{d_s2} & \cdots & w_{d_sd_s} \end{bmatrix}. \tag{7}$$

Let us assume to have $x_1, \ldots, x_n$, n points from dataset X that are *diverse*, *representative* and *sufficient*. In other words, we can approximate the dataset X with n

data points. We then proceed to compute the respective attention matrices $A_1, \dots A_n$ and calculate

$$\tilde{A} = \sum_{i=1}^{n} A_i.$$

As next step, we apply the Cut_l operator for $l \geq 0$, such that

$$Cut_l(x) = \begin{cases} x, & x \geq l \\ h(x), & \text{otherwise} \end{cases} \tag{8}$$

where $0 \leq h(x) < x$ penalize low values (for instance, $h(x) = 0$). We apply the operator (8) to matrix $\tilde{A}$. To conclude, as AW matrices are normalized by rows, we approximate (5) with

$$Norm(Cut_l(\tilde{A})).$$

The general idea of this procedure is to extract the dominant structural patterns from the set of attention matrices by aggregating information across multiple representative data points. By applying the Cut_l operator, we filter out low-significance entries while preserving the most stable attention weights. Finally, normalizing by rows ensures that the resulting structure-aware norm captures the relative importance of attention patterns across different positions. This brute-force approach provides a straightforward way to estimate the weight matrix W required for computing the structure-aware norm. However, its effectiveness depends on the quality of the selected $x_1, \dots, x_n$.

To construct a subset of points that is both diverse and representative, we propose applying k-means clustering and selecting $x_1, \dots, x_n$ as the cluster centroids in the embedded space E. This approach minimizes the within-cluster sum of squares (WCSS), ensuring that the chosen points effectively represent the underlying distribution of the dataset, while simultaneously maximizing the between-cluster sum of squares (BCSS), promoting diversity. The number of clusters, which depends on the size of X plays a crucial role in guaranteeing sufficiency. However, selecting the optimal number of clusters remains a critical challenge. A small number of clusters may fail to capture the dataset's variability, while an excessive number may introduce noise rather than meaningful structure. By leveraging this clustering-based selection process, we ensure that the extracted attention weight matrix structure is both representative of the dataset and robust to variations in individual samples. This allows us to compute a structure-aware norm that more accurately reflects the intrinsic patterns in the attention mechanism.

IV. Related Work

To the best of our knowledge, there is no other research that has addressed the problem of identifiability in terms of the structure of attention matrices. In paper [15] they introduce a notion that is similar to our concept of structural equivalent matrices. Given a set of two parameters for the model, θ' and θ^*, they are said to be $\sim_L$ identifiable if and only if

$$f'(x) = Af^*(x),$$
$$g'(x) = Bg^*(x)$$

where A, B are invertible matrices, $f(x)$ is some representation of the input and $g(y)$ is a context representation function. Under some appropriate condition, the authors prove the (linear) identifiability for a variety of models such as BERT, GPT-2 and GPT-3. Their work is aligned with ours in that they admit that a certain amount of *flexibility* is needed to be able to achieve the identifiability of a model and thus the connection between parameters and output. In their case, their equivalence relation require linear transformations that preserve information ($det(A), det(B) \neq 0$).

Paper [16] focuses on how to formalise in mathematical terms the concept of explainability. It is worth emphasising that although the concepts of identifiability and explainability have distinct definitions, they are extremely connected. Indeed, a more identifiable model tends also to be more explicable, because it provides unambiguous and interpretable parameters. In specific, they introduce the notion of *consistent representation* and *explainable representation*. We take inspiration from their point of view of how information should flow in a model, to ground the idea of using equivalence classes. Indeed, as we mention in Section II-B, using a fixed T (which is dependent on the embedding E) for different attention matrices A_1, A_2 represents circular reasoning. The main problem is that we are not sure that *similar* inputs x_1, x_2 produces similar attention matrices A_1, A_2 and that close inputs have close embedding representations, which would allow us to fix the map T. In other terms, we should be aware of how information flows in the process

$$\textit{input} \to E \to A \to \textit{output}.$$

Not considering the injectivity of application T and considering equivalence relations instead of the concept of similarity in terms of distance allows us to overcome these problems.

V. Experimental Setup

To demonstrate our concerns about the identifiability concept within the Attention Mechanism, we conducted experiments on several models incorporating attention. First, we examine the argument presented in Section III regarding diagonal attention weight matrices. We trained an encoder–decoder architecture comprising two Long Short-Term Memory (LSTM) layers and a Dense layer, with the objective of reversing the order of a sequence of tokens represented via one-hot encoding. The model was trained for `epochs` = 150, achieving 0.99 accuracy on the test set. We then computed the cross attention weights—i.e., between the decoder outputs and the encoder outputs—and visualized their distribution for a given input, as shown in Figure 1. We next

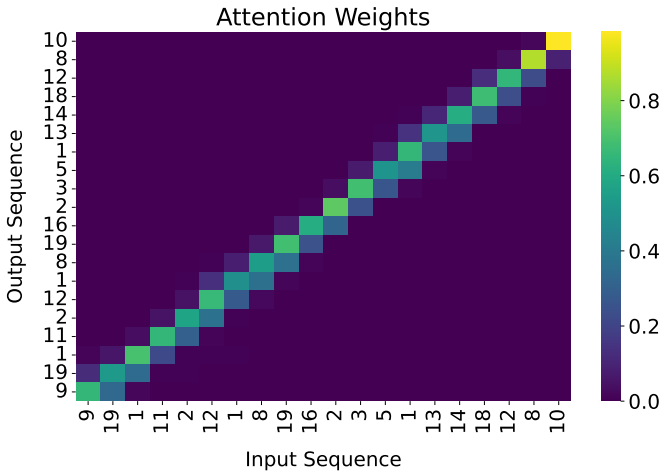

Fig. 1: The distribution of attention weights for a given input of length 20. The structure is reflecting the task of reversing the sentence.

extracted the underlying structure from the attention weight matrices. Given that the dataset X was generated randomly, we selected 100 input samples and followed the procedure described in Section III-B1. Figure 2 illustrates the critical role of the threshold l in equation (8) (with $h(x) = 0$) for structure extraction. It can be observed that a threshold of $l = 0.4$ serves as a good approximation of the optimal value. These experiments highlight the importance of considering both the task and the inherent structure of attention weight matrices when addressing identifiability. As theoretically discussed in Section III-A, it is necessary to consider structure-aware perturbations in order to properly assess whether the attention mechanism is identifiable.

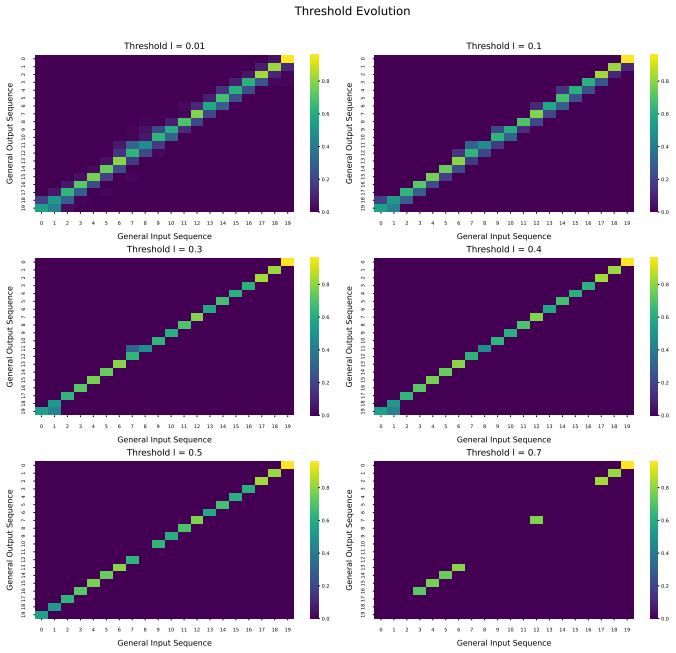

Fig. 2: The impact of threshold l in determining the W matrix in (7).

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[2] S. Islam, H. Elmekki, A. Elsebai, J. Bentahar, N. Drawel, G. Rjoub, and W. Pedrycz, "A comprehensive survey on applications of transformers for deep learning tasks," 2023.

[3] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in vision: A survey," *Association for Computing Machinery*, 2022.

[4] W. Guan, I. Smetannikov, and M. Tianxing, "Survey on automatic text summarization and transformer models applicability," in *Proceedings of the 2020 1st International Conference on Control, Robotics and Intelligent System*. Association for Computing Machinery, 2021.

[5] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:11212020

[6] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into deep learning," 2023.

[7] A. Bibal, R. Cardon, D. Alfter, R. Wilkens, X. Wang, T. François, and P. Watrin, "Is attention explanation? an introduction to the debate," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2022.

[8] S. Jain and B. C. Wallace, "Attention is not Explanation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019.

[9] G. Brunner, Y. Liu, D. Pascual, O. Richter, M. Ciaramita, and R. Wattenhofer, "On identifiability in transformers," *arXiv: Computation and Language*, 2019.

[10] J. Thickstun, "The transformer model in equations." [Online]. Available: https://johnthickstun.com/docs/transformers.pdf

[11] O. Naim and N. Asher, "On explaining with attention matrices," in *European Conference on Artificial Intelligence*, 2024.

[12] R. Bhardwaj, N. Majumder, S. Poria, and E. H. Hovy, "More identifiable yet equally performant transformers for text classification," in *Annual Meeting of the Association for Computational Linguistics*, 2021.

[13] E. Voita, D. Talbot, F. Moiseev, R. Sennrich, and I. Titov, "Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2019.

[14] K. Clark, U. Khandelwal, O. Levy, and C. D. Manning, "What does BERT look at? an analysis of BERT's attention," in *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics, 2019.

[15] G. Roeder, L. Metz, and D. P. Kingma, "On linear identifiability of learned representations," *ArXiv*, vol. abs/2007.00810, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:220302489

[16] L. Wolf, T. Galanti, and T. Hazan, "A formal approach to explainability," in *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. Association for Computing Machinery, 2019.

[17] C. Molnar, *Interpretable Machine Learning*, 2nd ed. Independently published, 2022. [Online]. Available: https://christophm.github.io/interpretable-ml-book

[18] S. Wiegreffe and Y. Pinter, "Attention is not not explanation," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 2019.

[19] A. Jacovi and Y. Goldberg, "Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness?" in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020.