Towards Opening the Black Box of Neural Machine Translation: Source and Target Interpretations of the Transformer

Javier Ferrando[1], **Gerard I. Gállego**[1], **Belen Alastruey**[1],
Carlos Escolano[1], **Marta R. Costa-jussà**[2]
[1]TALP Research Center, Universitat Politècnica de Catalunya
[2]Meta AI
{javier.ferrando.monsonis,gerard.ion.gallego,
belen.alastruey,carlos.escolano}@upc.edu
costajussa@meta.com

Abstract

In Neural Machine Translation (NMT), each token prediction is conditioned on the source sentence and the target prefix (what has been previously translated at a decoding step). However, previous work on interpretability in NMT has mainly focused solely on source sentence tokens' attributions. Therefore, we lack a full understanding of the influences of every input token (source sentence and target prefix) in the model predictions. In this work, we propose an interpretability method that tracks input tokens' attributions for both contexts. Our method, which can be extended to any encoder-decoder Transformer-based model, allows us to better comprehend the inner workings of current NMT models. We apply the proposed method to both bilingual and multilingual Transformers and present insights into their behaviour.

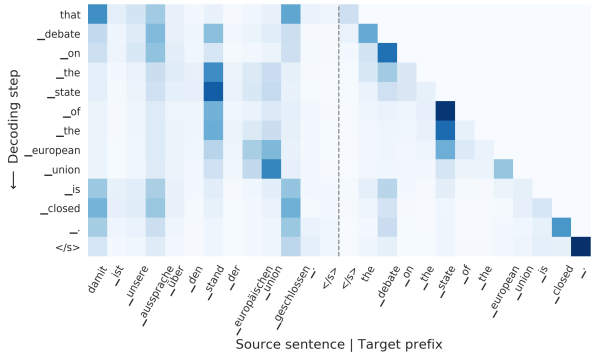


Figure 1: ALTI+ results for a De-En translation example. We obtain source sentence and target prefix (columns) interpretations for every predicted token (row).

1 Introduction

Transformers (Vaswani et al., 2017) have become the state-of-the-art architecture for natural language processing (NLP) tasks (Devlin et al., 2019; Raffel et al., 2020; Brown et al., 2020). With its success, the NLP community has experienced an urge to understand the decision process of the model predictions (Jain and Wallace, 2019; Serrano and Smith, 2019).

In Neural Machine Translation (NMT), attempts to interpret Transformer-based predictions have mainly focused on analyzing the attention mechanism (Raganato and Tiedemann, 2018; Voita et al., 2018). A large number of works in this line have investigated the capabilities of the cross-attention to perform source-target alignment (Kobayashi et al., 2020; Zenkel et al., 2019; Chen et al., 2020), compared with human annotations. Gradient-based (Ding et al., 2019) and occlusion-based methods (Li et al., 2019) have also been evaluated against human word alignments. The former computes gradients with respect to the input token embeddings to measure how much a change in the input changes the output, the latter generates input attributions by measuring the change in the predicted probability after deleting specific tokens. However, there is a tension between finding a faithful explanation and observing human-like alignments, since one does not imply the other (Ferrando and Costa-jussà, 2021).

The decoding process of NMT systems consists of generating tokens in the target vocabulary based on the information provided by the source sequence and the previously generated tokens (target prefix). However, most of the work on interpretability of NMT models only analyses source tokens. Recently, Voita et al. (2021a) proposed using Layer Relevance Propagation (LRP) (Bach et al., 2015) to analyze the source and target contributions to the model prediction, and later analyzed its behaviour during training (Voita et al., 2021b). Nonetheless, they apply their method to obtain global explanations, as an average over the entire dataset, not to get input attributions of a single prediction. Gradient-based methods have also been extended to the target prefix (Ferrando and Costa-jussà, 2021), although they do not quantify the relative contribution of source and target inputs.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 8756–8769
December 7-11, 2022

Concurrently, encoder-based Transformers, such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), have been analysed with attention rollout (Abnar and Zuidema, 2020), which models the information flow in the model with a Directed Acyclic Graph, where nodes are token representations and edges, attention weights. In the computer vision literature, Chefer et al. (2021b,a) combined this method with gradient information. Recently, Ferrando et al. (2022) have presented ALTI (Aggregation of Layer-wise Tokens Attributions), which applies the attention rollout method by substituting attention weights with refined token-to-token interactions. In this work, we present the first application of a rollout-based method to sequence to sequence Transformers. Our key contributions are[1]:

- We propose a method that measures the contributions of each input token (source and target prefix) to the encoder-decoder Transformer predictions;
- We show how contextual information is mixed across the encoder of NMT models, with the model keeping up to 47% of token identity;
- We evaluate the role of residual connections in the cross-attention, and show that attention to uninformative source tokens (EOS and final punctuation mark) is used to let information flow from the target prefix;
- We analyze the role of both input contexts in low and high-resource scenarios, and show the model behaviour under hallucinations.

2 Background

In this section, we provide the background to understand our proposed method by briefly explaining the encoder-decoder Transformer-based model in the context of NMT (Vaswani et al., 2017) and the Aggregation of Layer-wise Token-to-token Interactions (ALTI) method (Ferrando et al., 2022).

2.1 Encoder-Decoder Transformer

Given a source sequence of tokens $\mathbf{x} = (x_1, \ldots, x_J)$, and a target sequence $\mathbf{y} = (y_1, \ldots, y_T)$, an NMT system models the conditional probability:

$$P(\mathbf{y}|\mathbf{x}) = \prod_{t=1}^{T} P(y_t|\mathbf{y}_{<t}, \mathbf{x}) \quad (1)$$

where $\mathbf{y}_{<t} = (y_0, \ldots, y_{t-1})$ represents the prefix of y_t, with $x_J = y_0 =$ </s> used as a special token to mark the beginning and end of sentence. The Transformer is composed by a stack of encoder and decoder layers (Figure 2). The encoder generates a contextualized sequence of representations $\mathbf{e} = (\boldsymbol{e}_1, \ldots, \boldsymbol{e}_J)$ of the source sentence. The decoder, at each time step t, uses both the encoder outputs ($\mathbf{e}$) and the target prefix ($\mathbf{y}_{<t}$) to compute a probability distribution over the target vocabulary, from which a prediction is sampled.

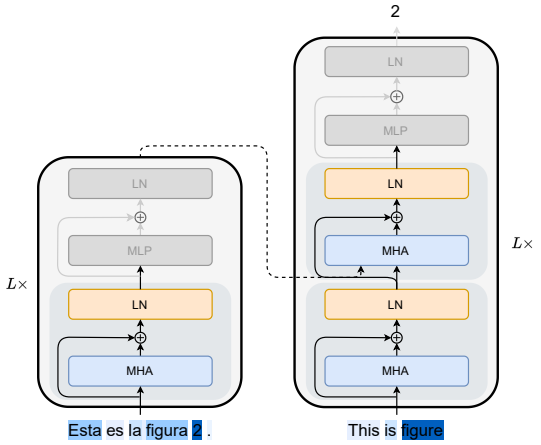


Figure 2: Encoder-Decoder Transformer.

Multi-head attention. The Transformer core building block, the multi-head attention mechanism (MHA) is in charge of combining contextual information in the hidden representations. Consider here $\mathbf{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_J)$ as the sequence of token representations[2] of dimension d entering layer l, and $\tilde{\mathbf{x}} = (\widetilde{\boldsymbol{x}}_1, \ldots, \widetilde{\boldsymbol{x}}_J)$ the output layer representations. Each of the H heads inside MHA computes vectors of dimension $d_h = d/H$:

$$\boldsymbol{z}_i^h = \sum_{j=1}^{J} \alpha_{i,j}^h \mathbf{W}_V^h \boldsymbol{x}_j \quad (2)$$

with $\alpha_{i,j}^h$ referring to the attention weight where token i attends token j, and $\mathbf{W}_V^h \in \mathbb{R}^{d_h \times d}$ to a learned weight matrix[3].

[1]Code available at https://github.com/mt-upc/transformer-contributions-nmt.

[2]We consider $\boldsymbol{x}_i$ as a column vector.

[3]The bias vector associated with $\mathbf{W}_V^h$ is omitted for the sake of simplicity.

The output of MHA for the i-th token (MHA_i) is calculated by concatenating each $\boldsymbol{z}_i^h$ and projecting the joint vector through $\mathbf{W}_O \in \mathbb{R}^{d\times d}$. This is equivalent to a sum over heads where each $\boldsymbol{z}_i^h$ is projected through the partitioned weight matrix $\mathbf{W}_O^h \in \mathbb{R}^{d\times d_h}$ and adding the bias $\boldsymbol{b}_O \in \mathbb{R}^d$:

$$\begin{aligned}\text{MHA}_i(\mathbf{x}) &= \mathbf{W}_O \text{ Concat}(\boldsymbol{z}_i^1, \ldots, \boldsymbol{z}_i^H) + \boldsymbol{b}_O \\ &= \sum_{h=1}^{H} \mathbf{W}_O^h \boldsymbol{z}_i^h + \boldsymbol{b}_O\end{aligned} \tag{3}$$

Layer normalization. Finally, a layer normalization (LN) is applied over the sum of the residual vector $\boldsymbol{x}_i$ and the output of the multi-head attention module, giving as output $\widetilde{\boldsymbol{x}}_i$:

$$\widetilde{\boldsymbol{x}}_i = \text{LN}(\text{MHA}_i(\mathbf{x}) + \boldsymbol{x}_i) \tag{4}$$

Merging Equations (2) to (4), we get:

$$\widetilde{\boldsymbol{x}}_i = \text{LN}\left(\sum_{j=1}^{J}\sum_{h=1}^{H} \mathbf{W}_O^h \alpha_{i,j}^h \mathbf{W}_V^h \boldsymbol{x}_j + \boldsymbol{b}_O + \boldsymbol{x}_i\right)$$

Considering $F_i(\boldsymbol{x}_j) = \sum_{h=1}^{H} \mathbf{W}_O^h \alpha_{i,j}^h \mathbf{W}_V^h \boldsymbol{x}_j$, we can formulate the previous equation as:

$$\widetilde{\boldsymbol{x}}_i = \text{LN}\left(\sum_{j=1}^{J} F_i(\boldsymbol{x}_j) + \boldsymbol{b}_O + \boldsymbol{x}_i\right) \tag{5}$$

2.2 Aggregation of Layer-wise Token-to-token Interactions (ALTI)

The layer normalization operation over a sum of vectors $\text{LN}(\sum_j \boldsymbol{u}_j)$, as in Equation (5), can be reformulated as $\sum_j L(\boldsymbol{u}_j) + \boldsymbol{\beta}$, where $L : \mathbb{R}^d \mapsto \mathbb{R}^d$ (see Appendix A.1). This allows us to express Equation (5) (Kobayashi et al., 2021) as an interpretable expression of the layer input representations (Figure 3):

$$\widetilde{\boldsymbol{x}}_i = \sum_{j=1}^{J} T_i(\boldsymbol{x}_j) + \boldsymbol{\epsilon} \tag{6}$$

where $\boldsymbol{\epsilon}$ contains bias terms (see Appendix A.2 for full derivation) and T_i transforms the layer input vectors:

$$T_i(\boldsymbol{x}_j) = \begin{cases} L(F_i(\boldsymbol{x}_j)) & \text{if } j \neq i \\ L(F_i(\boldsymbol{x}_j) + \boldsymbol{x}_i) & \text{if } j = i \end{cases} \tag{7}$$

with the residual connection $\boldsymbol{x}_i$ only considered in the transformed vector $T_i(\boldsymbol{x}_{j=i})$. Ferrando et al. (2022) propose to use the Manhattan distance between the output vector and the transformed vector as a measure of the impact of $\boldsymbol{x}_j$ on $\widetilde{\boldsymbol{x}}_i$:

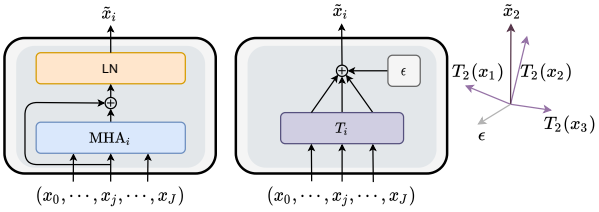


Figure 3: The self-attention block (left) at each position i can be decomposed as a summation of transformed input vectors (right). The closest vector ($T_2(x_2)$) *contributes* the most to $\widetilde{\boldsymbol{x}}_2$.

$$d_{i,j} = \|\widetilde{\boldsymbol{x}}_i - T_i(\boldsymbol{x}_j)\|_1 \tag{8}$$

By taking $-d_{i,j}$, larger distances reflect lower (more negative) influence. Then, distances are normalized $\in [0, 1]$ to obtain the *contribution of token representation j to token representation i*[4]:

$$c_{\widetilde{\boldsymbol{x}}_i \leftarrow \boldsymbol{x}_j} = \frac{\max(0, -d_{i,j} + \|\widetilde{\boldsymbol{x}}_i\|_1)}{\sum_{k=1}^{J} \max(0, -d_{i,k} + \|\widetilde{\boldsymbol{x}}_i\|_1)} \tag{9}$$

giving the matrix of layer-wise contributions $\mathbf{C}_{\widetilde{\boldsymbol{x}}_i \leftarrow \mathbf{x}} \in \mathbb{R}^{J\times J}$, where each row contains the contribution, or influence, of each $\boldsymbol{x}_j$ in $\widetilde{\boldsymbol{x}}_i$.

ALTI method (Ferrando et al., 2022) follows the Transformer's modeling approach proposed by Abnar and Zuidema (2020), where the information flow in the model is simplified as a Directed Acyclic Graph, where nodes are token representations, and edges represent the influence of each input layer token $\boldsymbol{x}_j$ in the output token $\widetilde{\boldsymbol{x}}_i$. ALTI proposes using token contributions $\mathbf{C}$ instead of raw attention weights α. The amount of information flowing from one node to another in different layers is computed by summing over the different paths connecting both nodes, where each path is the result of the multiplication of every edge in the path. This is computed by the matrix multiplication of the layer-wise contributions, giving the full encoder contribution matrix:

$$\mathbf{C}^{\text{enc}}_{\mathbf{e}\leftarrow\mathbf{x}} = \mathbf{C}^{L}_{\mathbf{e}\leftarrow\mathbf{x}} \cdot \mathbf{C}^{L-1}_{\widetilde{\mathbf{x}}\leftarrow\mathbf{x}} \cdot \cdots \cdot \mathbf{C}^{1}_{\widetilde{\mathbf{x}}\leftarrow\mathbf{x}} \tag{10}$$

We refer to $\mathbf{C}^{L}_{\mathbf{e}\leftarrow\mathbf{x}}$ as the contributions in the last layer of the encoder, where output vectors are $\mathbf{e}$.

[4]We use the term 'contribution' to refer to influences between token representations. 'Relevance' is used to allude to the influence of input tokens to model predictions.

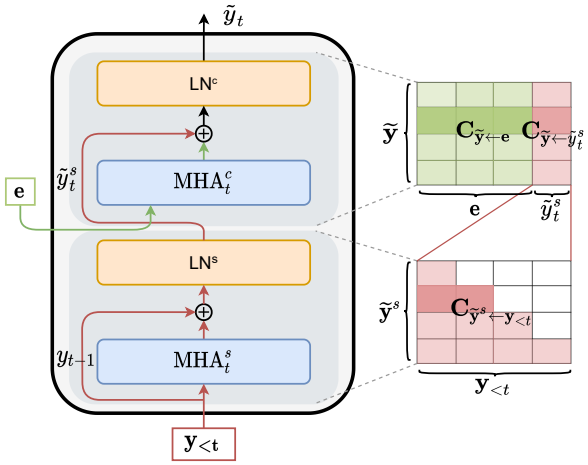


Figure 4: Self-attention and cross-attention modules in a decoder layer together with its contribution matrices.[5] In green, it's shown the information coming from the encoder (source), and in red, the information from the decoder (target prefix). Highlighted is shown contributions at a single time step t.

3 ALTI for the Encoder-Decoder Transformer (ALTI+)

The attention rollout and ALTI methods work for encoder-based Transformers. However, in the encoder-decoder Transformer, the cross-attention hinders its integration. In this section, we present ALTI+, which is the adaptation of ALTI method to the encoder-decoder Transformer.

3.1 Decoder Layer Decomposition

We decompose the self-attention and cross-attention of a decoder layer into interpretable expressions (Equation (6)), from which we can get the degree of interaction between input and output token representations (Equation (9)). Consider $\mathbf{y}_{<t} = (\boldsymbol{y}_0, \ldots, \boldsymbol{y}_j, \ldots, \boldsymbol{y}_{t-1})$ the set of vector representations of the target prefix tokens as input of a decoder layer, and $\widetilde{\boldsymbol{y}}_t$ the layer output (Figure 4).

Decoder self-attention. The layer normalization in the decoder self-attention (LN^s) is applied over the sum of the multi-head attention output and the residual $\boldsymbol{y}_{t-1}$. The self-attention block[6] can be written as:

$$\begin{aligned}\widetilde{\boldsymbol{y}}^s_t &= \text{LN}^s(\text{MHA}^s_t(\mathbf{y}_{<t}) + \boldsymbol{y}_{t-1}) \\ &= \text{LN}^s\left(\sum_{j=0}^{t-1} F^s_t(\boldsymbol{y}_j) + \boldsymbol{b}_O + \boldsymbol{y}_{t-1}\right)\end{aligned} \tag{11}$$

[5] We omit the MLP and its LN of the decoder layer.

[6] We refer as 'block' to the multi-head attention, residual, and layer normalization.

where F^s_t considers $\alpha, \mathbf{W}^h_V$ and $\mathbf{W}^h_O$ of the decoder self-attention. Analogous to Equation (7) we can obtain the transformed vectors of $\boldsymbol{y}_j$:

$$T^s_t(\boldsymbol{y}_j) = \begin{cases} L^s(F^s_i(\boldsymbol{y}_j)) & \text{if } j \neq t-1 \\ L^s(F^s_i(\boldsymbol{y}_j) + \boldsymbol{y}_{t-1}) & \text{if } j = t-1 \end{cases}$$

Following Equations (8) and (9) we get the decoder self-attention contributions $\mathbf{C}_{\tilde{\mathbf{y}}^s\leftarrow\mathbf{y}_{<t}} \in \mathbb{R}^{T\times T}$ reflecting the strength of the interaction between $\mathbf{y}_{<t}$ and $\widetilde{\boldsymbol{y}}^s_t$.

Decoder cross-attention. The output of the cross-attention block at time step t can be decomposed as:

$$\begin{aligned}\widetilde{\boldsymbol{y}}_t &= \text{LN}^c(\text{MHA}^c_t(\mathbf{e}) + \widetilde{\boldsymbol{y}}^s_t) \\ &= \text{LN}^c\left(\sum_{j=1}^{J} F^c_t(\boldsymbol{e}_j) + \boldsymbol{b}_O + \widetilde{\boldsymbol{y}}^s_t\right)\end{aligned} \tag{12}$$

where $\widetilde{\boldsymbol{y}}^s_t$, the residual connection, is the output of the self-attention block, and $\mathbf{e}$ the encoder outputs. We can obtain the transformed vectors of the encoder outputs $\boldsymbol{e}_j$ and the residual connection $\widetilde{\boldsymbol{y}}^s_t$:

$$\begin{aligned}T^c_t(\boldsymbol{e}_j) &= L^c(F^c_t(\boldsymbol{e}_j)) \\ T^c_t(\boldsymbol{y}^s_t) &= L^c(\widetilde{\boldsymbol{y}}^s_t)\end{aligned} \tag{13}$$

Following Equation (8), we can compute the Manhattan distance between the transformed vectors and $\widetilde{\boldsymbol{y}}_t$ and get the contributions $[\mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\mathbf{e}}; \mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\tilde{\boldsymbol{y}}^s_t}]$, with $\mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\mathbf{e}} \in \mathbb{R}^{T\times J}$ and $\mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\tilde{\boldsymbol{y}}^s_t} \in \mathbb{R}^{T\times 1}$.

The cross-attention residual $\widetilde{\boldsymbol{y}}^s_t$ contribution to $\widetilde{\boldsymbol{y}}_t$ reflects the total influence of the self-attention inputs $\mathbf{y}_{<t}$ to the decoder layer output $\widetilde{\boldsymbol{y}}_t$. Thus, we can get the full *decoder layer contribution matrix* $[\mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\mathbf{e}}; \mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\mathbf{y}_{<t}}]$ (Figure 5) by substituting the residual contributions ($\mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\tilde{\boldsymbol{y}}^s_t}$) with the self-attention contributions ($\mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\tilde{\boldsymbol{y}}^s_t}$), and weighting every row of $\mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\mathbf{y}_{<t}}$ by the corresponding value of the residual contribution of each time step.

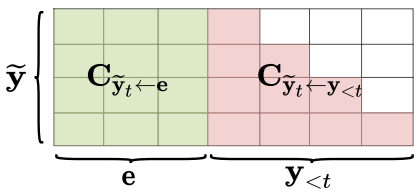


Figure 5: Full decoder layer contributions.

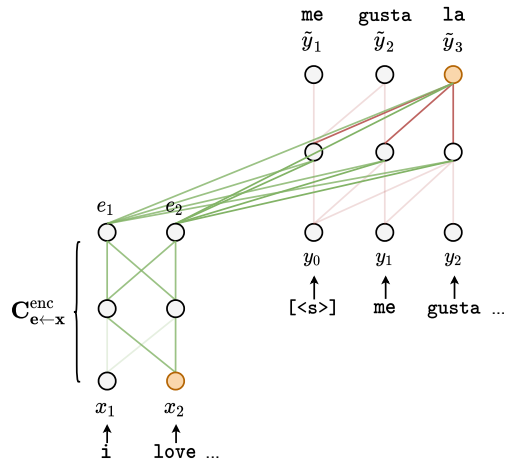


Figure 6: Source input attributions $\mathbf{R}^{\text{model}}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}}$.

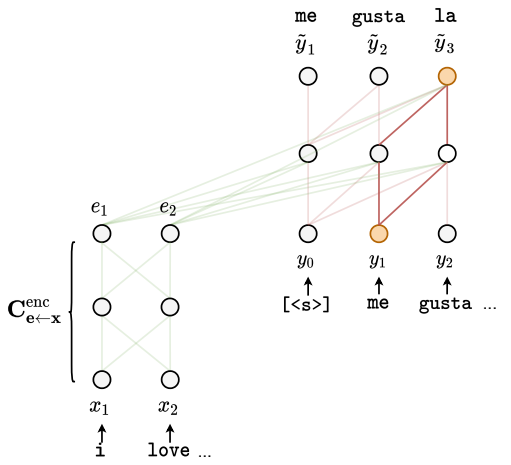


Figure 7: Target prefix input attributions $\mathbf{R}^{\text{model}}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{y}_{<t}}$.

3.2 Aggregating Contributions Through the Encoder-Decoder Transformer

In order to get *input token attributions*, we apply the same principle as attention rollout method. As described in §2.2, ALTI builds a graph where nodes are token representations and edges represent the contributions between tokens in each layer. The amount of information flowing from one node to another in different layers is computed by summing over the different paths connecting both nodes, where each path is the result of the multiplication of every edge in the path (Figures 6 and 7).

Algorithm 1: ALTI+ source relevance.

Input: $\mathbf{C}^{\text{enc}}_{\mathbf{e}\leftarrow\mathbf{x}}$ – encoder contributions
$\mathbf{C}^{l}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{e}}$ – contributions decoder layers
L – number of layers
Output: $\mathbf{R}^{\text{model}}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}}$ – source input relevancies
for $l \leftarrow$ *[1,2...L]* **do**
  $\mathbf{C}^{*l}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}} = \mathbf{C}^{l}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{e}} \cdot \mathbf{C}^{\text{enc}}_{\mathbf{e}\leftarrow\mathbf{x}}$
$\mathbf{R}^{1}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}} = \mathbf{C}^{*1}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}}$
for $l \leftarrow$ *[2,3...L]* **do**
  $\mathbf{R}^{l}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}} = \mathbf{C}^{l}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{y}_{<t}} \cdot \mathbf{R}^{l-1}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}} + \mathbf{C}^{*l}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}}$
$\mathbf{R}^{\text{model}}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}} = \mathbf{R}^{L}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}}$
return $\mathbf{R}^{\text{model}}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}}$

ALTI+ source tokens relevance. Algorithm 1 shows the process to obtain source sentence tokens relevance for the model prediction $R^{\text{model}}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}}$ (Figure 6). We first update the cross-attention contribution matrices (to $\mathbf{C}^{*l}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{x}}$) by multiplying each of them with the contributions of the entire encoder $\mathbf{C}^{\text{enc}}_{\mathbf{e}\leftarrow\mathbf{x}}$ to account for all the paths in the encoder and cross-attentions. We then iteratively aggregate edges from paths of the target prefix contributions $\mathbf{C}^{l}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{y}_{<t}}$.

ALTI+ target prefix tokens relevance. Target prefix input attributions (Figure 7) are computed by multiplying $\mathbf{C}_{\tilde{\mathbf{y}}\leftarrow\mathbf{y}_{<t}}$ in each layer:

$$\mathbf{R}^{\text{model}}_{\tilde{\mathbf{y}}_t\leftarrow\mathbf{y}_{<t}} = \mathbf{C}^{L}_{\tilde{\mathbf{y}}\leftarrow\mathbf{y}_{<t}} \cdot \mathbf{C}^{L-1}_{\tilde{\mathbf{y}}\leftarrow\mathbf{y}_{<t}} \cdot \cdots \cdot \mathbf{C}^{1}_{\tilde{\mathbf{y}}\leftarrow\mathbf{y}_{<t}} \quad (14)$$

4 Experimental Setup

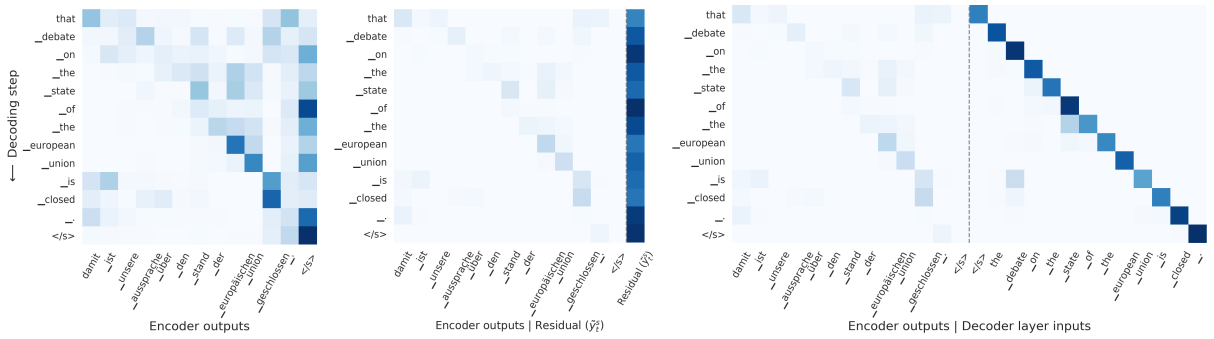
We analyze input token attributions in both bilingual and multilingual Machine Translation models. For the bilingual setting, we train a 6-layer Transformer model for the German-English (De-En) translation task. We use Europarl v7 corpus[7] and follow Zenkel et al. (2019) and Ding et al. (2019) data setup[8]. We use byte-pair encoding (BPE) (Sennrich et al., 2016) with 10k merge operations. For the multilingual model, we use M2M Transformer (Fan et al., 2021), a many-to-many multilingual translation model that can translate directly between any pair of 100 languages. We use FAIRSEQ (Ott et al., 2019) implementations, and the provided checkpoint for the M2M model (418M). We perform the quantitative analysis in 1000 sentences of the test set of IWSLT'14 German-English dataset. For the analysis in §5.5 we use FLORES-101 (Goyal et al., 2022) devtest split.

5 Analysis

In this section, we perform a set of experiments to measure the quality of the obtained contribu-

[7] http://www.statmt.org/europarl/v7

[8] https://github.com/lilt/alignment-scripts/tree/master/preprocess



(a) Attention weights (b) Contributions (c) Decoder layer contributions

Figure 8: (a) Cross-attention weights. (b) Cross-attention contributions $[\mathbf{C}_{\widetilde{\mathbf{y}}\leftarrow\mathbf{e}}; \mathbf{C}_{\widetilde{\mathbf{y}}\leftarrow\widetilde{\boldsymbol{y}}_t^s}]$ of the encoder outputs $\mathbf{e}$ and residual $\widetilde{\boldsymbol{y}}_t^s$ to the decoder layer output, as described in §3.1. (c) Total decoder layer contributions $[\mathbf{C}_{\widetilde{\mathbf{y}}\leftarrow\mathbf{e}}; \mathbf{C}_{\widetilde{\mathbf{y}}\leftarrow\mathbf{y}_{<t}}]$ with the self-attention contributions included.

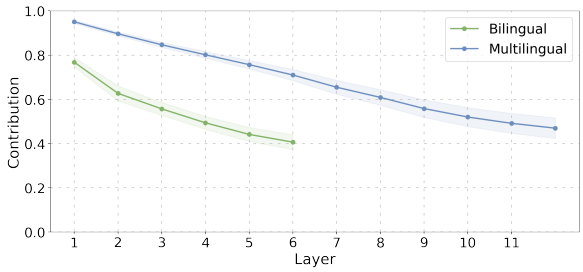


Figure 9: Contribution of the source input token to the encoder output representation at the same position. We show mean and SD for each layer of the bilingual and multilingual models.

Method	AER (↓)
Attention weights	47.7 ± 1.7
Vector-Norms	41.4 ± 1.4
Vector-Norms + LN + Res	42.5 ± 0.8
Our contributions $\mathbf{C}_{\widetilde{\mathbf{y}}\leftarrow\mathbf{e}}$	38.8 ± 1.3

Table 1: AER of the cross-attention contributions in the 5th layer of the bilingual model. We show mean and SD for models trained on five different seeds.

tions, and unveil different aspects of bilingual and multilingual NMT models.

5.1 Information Mix in the Encoder

Information from input source tokens gets mixed throughout the encoder. Intermediate layer representations acquire contextual information from other tokens in the sentence due to the self-attention mechanism. Brunner et al. (2020) analyze, for an encoder-based model, the contribution of input source tokens to its intermediate layer representations. They conclude that input source tokens contribute little (around 10% on average) to its corresponding last layer representation (encoder output). However, by training a linear classifier and, with nearest neighbor lookup based on the cosine distance, they are able to recover input token identity 93% of the times. We apply ALTI method (Equation (10)) across the Transformer encoder and analyze the input relevance of source tokens to intermediate encoder representations (Figure 9). Our results in the bilingual and multilingual models show that, indeed, input tokens highly contribute to their associated layer representations. In the last layer, 41% of the input contribution comes from the input token at the same position. The multilingual model is able to retain above 47% despite its 12 layers. The curves of both models in Figure 9 closely match the results obtained by Voita et al. (2019) relying on the mutual information between the input tokens and tokens representations across layers.

5.2 Alignment in Cross-attention

In order to evaluate the quality of the proposed cross-attention contributions (§3.1), we measure Alignment Error Rate (AER) against human-annotated alignments. As found out by Garg et al. (2019), the penultimate layer of Transformers tends to focus on learning the source-target alignment of words. Therefore, we analyze the cross-attention contributions $\mathbf{C}_{\widetilde{\mathbf{y}}\leftarrow\mathbf{e}}$ extracted from the 5th layer from the bilingual 6-layer model. We use gold alignments from Vilar et al. (2006), containing 508 sentence pairs. For comparison, we compute the AER of the raw attention weights and previous methods based on vector norms. *Vector-Norms* (Kobayashi et al., 2020) compute $\|F\|_2$ from Equa-

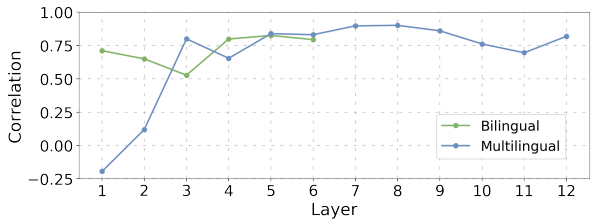


Figure 10: Pearson's r correlation between attention weight values given to EOS token (</s>) and the contribution of the residual in the cross attention.

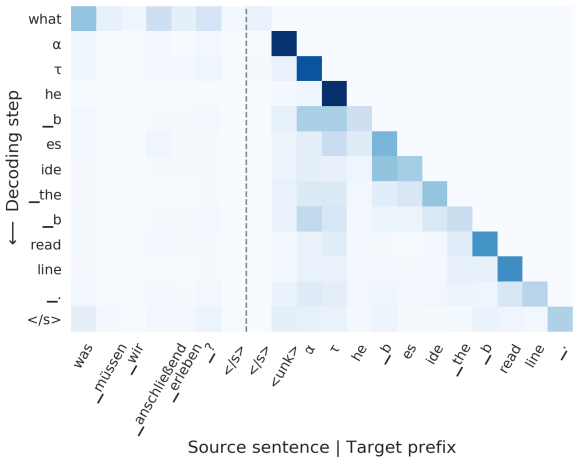


Figure 11: ALTI+ results for a hallucination after induced perturbation in the bilingual model.

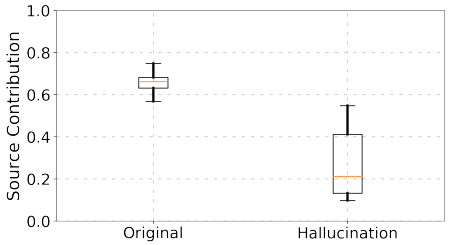


Figure 12: Source contribution in sentences without and with induced perturbation in the bilingual model.

tion (5), and *Vector-Norms + LN + Res* (Kobayashi et al., 2021) $\|T\|_2$ from Equation (6). As shown in Table 1, our method for estimating layer-wise contributions obtain the lowest AER, outperforming similar previous methods by at least 2.6 points on average. As can be observed in Figure 8, attention weights fail at showing alignments, with the </s> token concentrating large attention weights. Our method is able to filter this noise, showing almost no contribution from </s>. In §5.3, we analyze this phenomenon and try to find an explanation for it.

5.3 The Role of the End-of-Sentence Token

It has been hypothesized that attention given to special tokens is used by the model as a 'no-op' (Clark et al., 2019). Ferrando and Costa-jussà (2021) analyze attention weights of the cross-attention to source finalizing tokens (final punctuation mark and </s>), and find the value vectors (see Appendix B) associated with these tokens to be almost zero norm. Additionally, they find that attention weights to source finalizing tokens tend to increase when predicting tokens that heavily rely on the target prefix, such as postpositions, particles, or closing subwords. The proposed cross-attention decomposition in §3.1 allows us to analyze both the contributions of source tokens, and the residual connection (Figure 8 (b)). We measure the Pearson correlation between attention weights to </s> token and the contribution of the residual connection in the cross-attention. We can see in Figure 10 that there is a high correlation in almost every layer, especially in the last layers. This demonstrates that finalizing tokens are used to *skip source attention*, since the higher their attention score, the more information is flowing from the decoder (in the residual) coming from the target prefix.

5.4 Analyzing Hallucinations

A common issue of NMT models is hallucination, which are translations that are disconnected from the source text, despite being fluent in the target language (Müller et al., 2020). Hallucinations should be reflected in our method as a drop in the contribution of the source sentence. Thus, in this section, we induce hallucination and measure the source sentence contribution with ALTI+.

To induce hallucination, we perturb the target prefix sequence of the bilingual model by adding the <unk> token. Then, we follow the algorithm proposed by Lee et al. (2018) to detect which perturbed translations are hallucinations. They measure BLEU score of the generated translation with and without perturbation. They fix a minimum threshold BLEU score for the original translations (20 BLEU in our experiments), and a maximum score for the perturbed translations (3 BLEU in our experiments). The model is considered to hallucinate when both translations satisfy the thresholds.

Analyzing ALTI+ contributions, we can confirm that the bilingual model largely ignores source to-

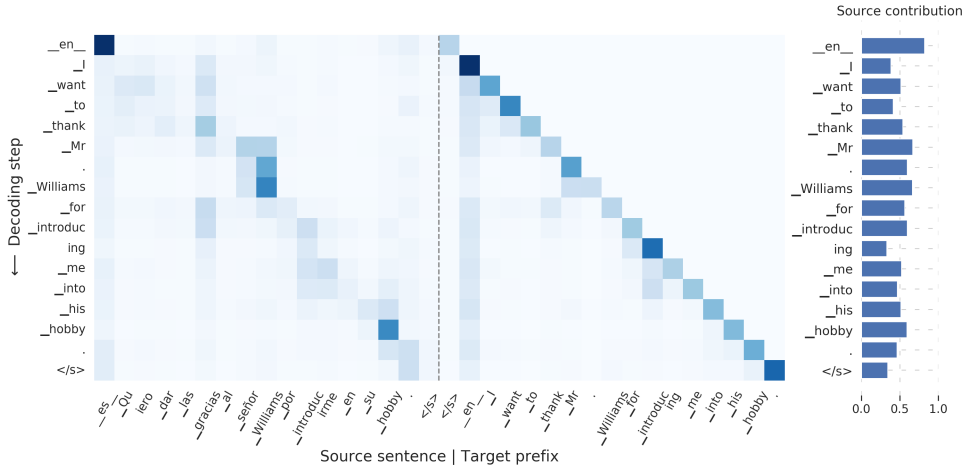


Figure 13: ALTI+ for a Es-En example in the multilingual model.

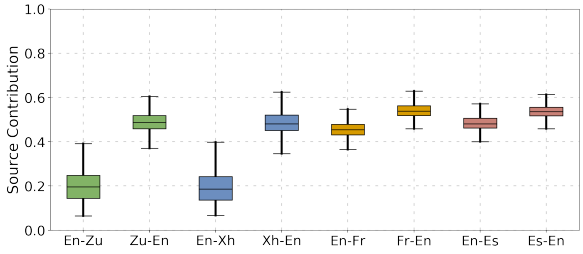


Figure 14: Source sentence contribution in different language directions from the FLORES-101 devtest split.

kens during hallucinations (Figures 11 and 12).

5.5 Multilingual Model Analysis

We analyze the behaviour of the multilingual model in different language pairs of FLORES-101 dataset. We include in the analysis high-resource languages, English (En), Spanish (Es), and French (Fr) and low-resource languages, Zulu (Zu) and Xhosa (Xh). High-resource languages have been defined in (Goyal et al., 2022) as languages with available bitext data beyond 100M samples, and low-resource languages are those with less than 1M.

Figure 13 shows an Es-En example in the multilingual model. We observe an almost uniform contribution of the language tags across different outputs. The only drop in its contribution seems to happen when translating proper nouns (e.g., "Mr. Williams") or anglicisms (e.g., "hobby"), which is observed for other language pairs too (Appendix C), and repeated across the dataset. We hypothesize that the model doesn't need to rely on the language tag since these words appear across different languages. Dependencies between generated tokens are also observed, the prediction "for" relies on "thanks", "Williams" on "Mr." and "into" on "introducing". The same example can be found in Appendix C for En-Zu and Zu-En pairs.

Figure 14 shows results of the source sentence contribution for En-Zu, En-Xh, En-Fr and En-Es pairs. We observe similar source contribution patterns between the high-resource pairs, and between those pairs involving a low-resource language. However, in the low-resource scenario, the source contribution is remarkably lower when translating from English. We hypothesize that, when the low-resource language is in the target prefix, the model tends to behave similarly to when it hallucinates (Figure 12), ignoring the source. But, when a high-resource language (En) is in the target prefix, it is less likely to lose track of the source and, thus, less prone to enter hallucination mode. Low-resource language sentences in the target side may be seen by the model as target prefix perturbations (§5.4), although further research is required.

6 Conclusions

We propose ALTI+, an interpretability method for the encoder-decoder Transformer that provides token influences to the model predictions for the two input contexts: source sentence and target prefix. By applying ALTI+ to a bilingual and a multilingual NMT model we are able to discover insights into the behaviour of these black-box models. Unlike previous methods, we can now observe dependencies between tokens in the predicted sentence, and quantify the total contribution of each of the contexts. This allows a deeper exploration of current NMT models. Our findings include: the role of the source EOS (</s>) token as a mean to avoid

incorporating source information, the absence of source contribution when producing hallucinations, and the lack of source contributions when translating from English to a low-resource language. ALTI+ overcomes the limitations of previous interpretability methods in NMT, and we believe it can help researchers and practitioners to better understand any encoder-decoder Transformer model.

Limitations

ALTI+ is able to measure the amount of contextual information in each layer representation of the Transformer. We use the influences of each input token to the last layer representation for evaluating input attributions for the model prediction. However, our method does not consider the softmax layer on top of the Transformer. Therefore, ALTI+ doesn't provide explanations for each of the output classes (target vocabulary), as opposed to gradient-based methods.

Ethical Considerations

ALTI+ provides explanations about input attributions in the Encoder-Decoder Transformer. By itself, we are not aware of any ethical implications of the methodology, which does not take into account any subjective priors. We perform experiments in Machine Translation. While we do not study biases in this application, we know they exist (Costa-jussà et al., 2022). In the future, we plan to further explore and mitigate them by using the information of source input attributions that ALTI+ provides. Also, understanding hallucinations by means of ALTI+ can help to avoid catastrophic and unsafe translations.

7 Acknowledgements

We would like to thank the anonymous reviewers for their useful comments. Javier Ferrando and Gerard I. Gállego are supported by the Spanish Ministerio de Ciencia e Innovación through the project PID2019-107579RB-I00 / AEI / 10.13039/501100011033.

References

Samira Abnar and Willem Zuidema. 2020. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4197, Online. Association for Computational Linguistics.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLOS ONE*, 10(7):1–46.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Gino Brunner, Yang Liu, Damian Pascual, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. 2020. On identifiability in transformers. In *International Conference on Learning Representations*.

Hila Chefer, Shir Gur, and Lior Wolf. 2021a. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 397–406.

Hila Chefer, Shir Gur, and Lior Wolf. 2021b. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 782–791.

Yun Chen, Yang Liu, Guanhua Chen, Xin Jiang, and Qun Liu. 2020. Accurate word alignment induction from neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 566–576, Online. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. 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*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Marta R. Costa-jussà, Carlos Escolano, Christine Basta, Javier Ferrando, Roser Batlle, and Ksenia Kharitonova. 2022. Interpreting gender bias in neural machine translation: Multilingual architecture matters. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(11):11855–11863.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Shuoyang Ding, Hainan Xu, and Philipp Koehn. 2019. Saliency-driven word alignment interpretation for neural machine translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 1–12, Florence, Italy. Association for Computational Linguistics.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Michael Auli, and Armand Joulin. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.

Javier Ferrando and Marta R. Costa-jussà. 2021. Attention weights in transformer NMT fail aligning words between sequences but largely explain model predictions. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 434–443, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Javier Ferrando, Gerard I. Gállego, and Marta R. Costa-jussà. 2022. Measuring the mixing of contextual information in the transformer.

Sarthak Garg, Stephan Peitz, Udhyakumar Nallasamy, and Matthias Paulik. 2019. Jointly learning to align and translate with transformer models. 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)*, pages 4453–4462, Hong Kong, China. Association for Computational Linguistics.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Sarthak Jain and Byron C. Wallace. 2019. 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)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7057–7075, Online. Association for Computational Linguistics.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2021. Incorporating Residual and Normalization Layers into Analysis of Masked Language Models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4547–4568, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2018. Hallucinations in neural machine translation. *NIPS 2018 Interpretability and Robustness for Audio, Speech and Language Workshop*.

Xintong Li, Guanlin Li, Lemao Liu, Max Meng, and Shuming Shi. 2019. On the word alignment from neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1293–1303, Florence, Italy. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Mathias Müller, Annette Rios, and Rico Sennrich. 2020. Domain robustness in neural machine translation. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 151–164, Virtual. Association for Machine Translation in the Americas.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Alessandro Raganato and Jörg Tiedemann. 2018. An analysis of encoder representations in transformer-based machine translation. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 287–297, Brussels, Belgium. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual*

Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Sofia Serrano and Noah A. Smith. 2019. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

David Vilar, Maja Popovic, and Hermann Ney. 2006. AER: do we need to "improve" our alignments? In *Proceedings of the Third International Workshop on Spoken Language Translation: Papers*, Kyoto, Japan.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. The bottom-up evolution of representations in the transformer: A study with machine translation and language modeling objectives. 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)*, pages 4396–4406, Hong Kong, China. Association for Computational Linguistics.

Elena Voita, Rico Sennrich, and Ivan Titov. 2021a. Analyzing the source and target contributions to predictions in neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1126–1140, Online. Association for Computational Linguistics.

Elena Voita, Rico Sennrich, and Ivan Titov. 2021b. Language modeling, lexical translation, reordering: The training process of NMT through the lens of classical SMT. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8478–8491, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Elena Voita, Pavel Serdyukov, Rico Sennrich, and Ivan Titov. 2018. Context-aware neural machine translation learns anaphora resolution. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1264–1274, Melbourne, Australia. Association for Computational Linguistics.

Thomas Zenkel, Joern Wuebker, and John DeNero. 2019. Adding interpretable attention to neural translation models improves word alignment. *CoRR*, abs/1901.11359.

A ALTI

A.1 Layer Normalization

The Layer normalization operation over input $\boldsymbol{x}$ can be defined as: $\text{LN}(\boldsymbol{x}) = \frac{\boldsymbol{x}-\mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})} \odot \gamma + \beta$, where μ computes the mean, σ the standard deviation, and $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ refer to an element-wise transformation and bias respectively. $\text{LN}(\boldsymbol{x})$ can be decomposed into $\frac{1}{\sigma(\boldsymbol{x})}\mathbf{L}\boldsymbol{x} + \boldsymbol{\beta}$, where $\mathbf{L}$ is a linear transformation including the mean and element wise multiplication.

Given a sum of vectors $\sum_j \boldsymbol{x}_j$ as input to LN we can rewrite the expression as:

$$\begin{aligned}\text{LN}(\sum_j \boldsymbol{x}_j) &= \frac{1}{\sigma(\sum_j \boldsymbol{x}_j)}\mathbf{L}\sum_j \boldsymbol{x}_j + \boldsymbol{\beta} \\ &= \sum_j \frac{1}{\sigma(\sum_j \boldsymbol{x}_j)}\mathbf{L}\boldsymbol{x}_j + \boldsymbol{\beta} \\ &= \sum_j L(\boldsymbol{x}_j) + \boldsymbol{\beta}\end{aligned}$$

A.2 Full derivation

$$\begin{aligned}\widetilde{\boldsymbol{x}}_i &= \text{LN}\left(\sum_{j=1}^{J}\sum_{h=1}^{H}\mathbf{W}_O^h \alpha_{i,j}^h \mathbf{W}_V^h \boldsymbol{x}_j + \boldsymbol{b}_O + \boldsymbol{x}_i\right) \\ &= \text{LN}\left(\sum_{j=1}^{J} F_i(\boldsymbol{x}_j) + \boldsymbol{b}_O + \boldsymbol{x}_i\right) \\ &= \sum_{j=1}^{J} L(F_i(\boldsymbol{x}_j)) + L(\boldsymbol{b}_O) + L(\boldsymbol{x}_i) + \boldsymbol{\beta}\end{aligned}$$

Defining $\boldsymbol{\epsilon} = L(\boldsymbol{b}_O) + \boldsymbol{\beta}$ we get to the expression in Equation (6):

$$\widetilde{\boldsymbol{x}}_i = \sum_{j=1}^{J} T_i(\boldsymbol{x}_j) + \boldsymbol{\epsilon} \tag{15}$$

B Values Norms

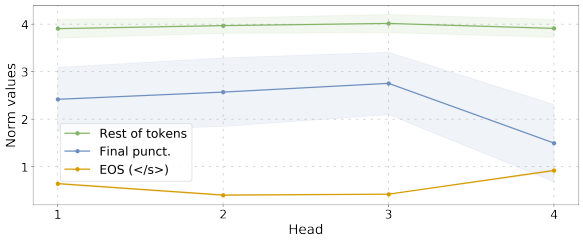


Figure 15: Norm of the value vectors (from encoder outputs) in the cross-attention of the alignment layer. We provide mean and SD for each head in the bilingual model. Similar patterns are observed across layers, and in the multilingual model.

C Examples

We include examples for the En-Zu language pair in the multilingual model in Figure 16 and 17, as well as for Es-En in Figure 18 and Fr-En in Figure 19.

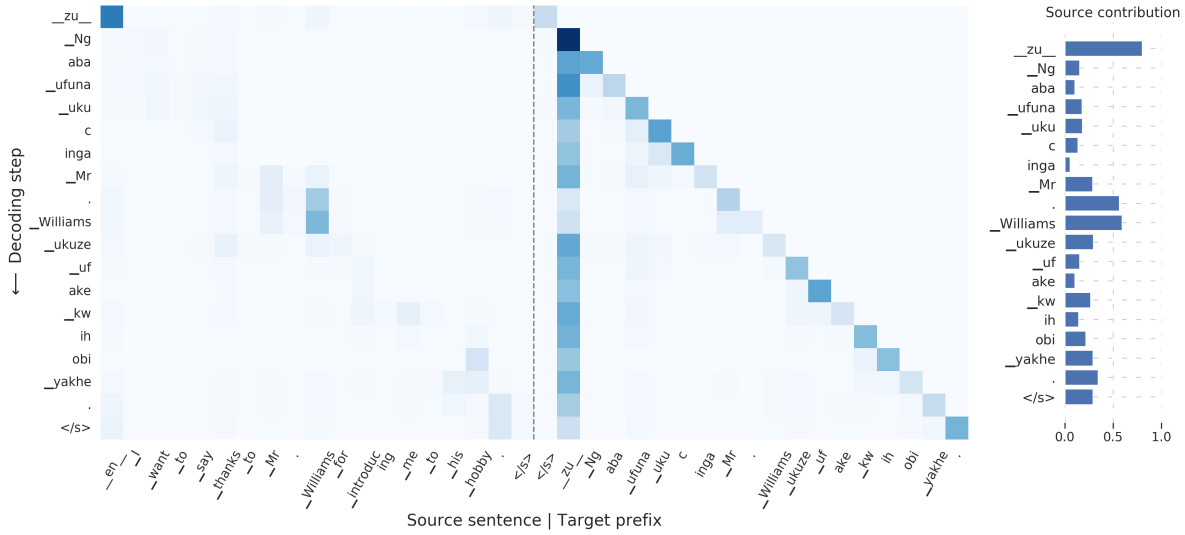


Figure 16: ALTI+ for a En-Zu example in the multilingual model.

Figure 17: ALTI+ for a Zu-En example in the multilingual model.

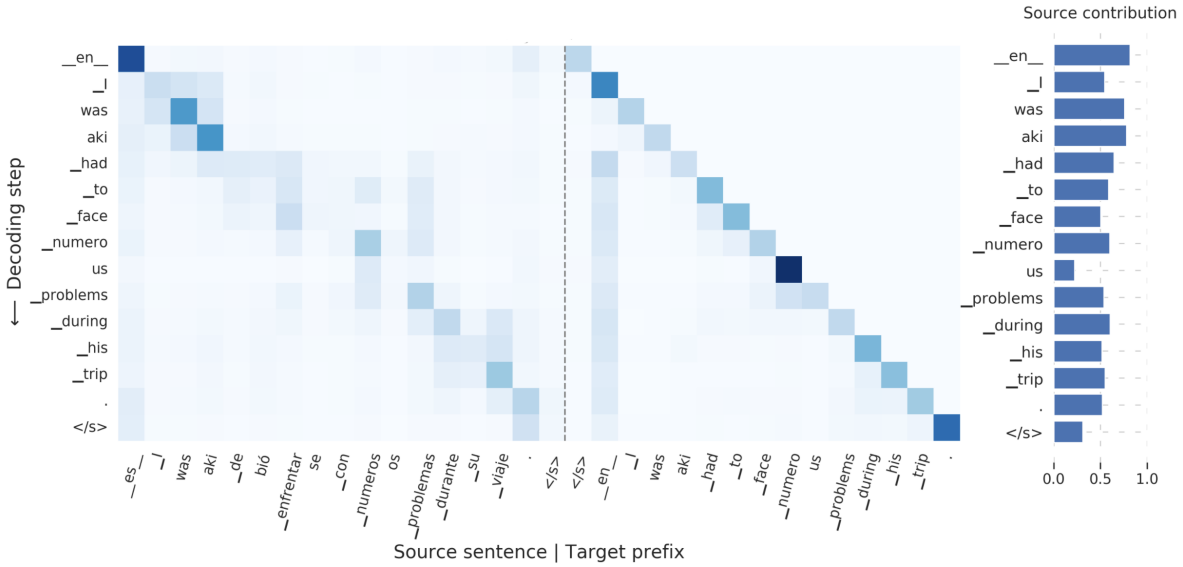


Figure 18: ALTI+ for a Es-En example in the multilingual model.

Figure 19: ALTI+ for a Fr-En example in the multilingual model.