ON NEURONS INVARIANT TO SENTENCE STRUCTURAL CHANGES IN NEURAL MACHINE TRANSLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

To gain insight into the role neurons play, we study the activation patterns corresponding to meaning-preserving paraphrases (e.g., active-passive). We compile a dataset of controlled syntactic paraphrases in English with their reference German translations and demonstrate our model-agnostic approach with the Transformer translation model. First, we identify neurons that correlate across paraphrases and dissect the observed correlation into possible confounds. Although lower-level components are found as the cause of similar activations, no sentence-level semantics or syntax are detected locally. Later, we manipulate neuron activations to influence translation towards a particular syntactic form. We find that a simple value shift is effective, and more so when many neurons are modified. These suggest that complex syntactic constructions are indeed encoded in the model. We conclude by discussing how to better manipulate it using the correlations we first obtained.[1]

1 INTRODUCTION

Understanding the roles neurons play is important for the interpretability of neural machine translation (NMT) models. Finding neurons that are either invariant or sensitive to sentence structures explains how NMT models encode such structures and what similarities they have learned to abstract away. Furthermore, it enables control of the output by direct manipulation of neurons.

Little previous work analyzed the interaction between input properties and individual neurons (see §7). We focus on how differences between structures of input sentences are represented and the resulting effect on the output translation. We propose a methodology to analyze the correlation patterns of the network activations between a source sentence and its structural paraphrase. We examine two test cases: the active/passive voice transformation and a paraphrase transforming a clause into a noun phrase (see Table 1). The work is motivated by the findings of Bau et al. (2019), who detected neurons that are highly correlated across LSTM translation models that differ in their initializations. They showed that these neurons are the most important for the model performance. Bau et al. (2019) further manipulated individual neurons to control semantic features at the word level (e.g., gender, tense). We extend their method to study the representation of syntactic structures, which influences the global organization of the sentence, rather than individual words.

Table 1: Examples produced by our paraphrasing engine

	Source	Paraphrased
Active Voice/ Passive Voice	*She **took** the book*	*The book **was taken** by her*
Adverbial Clause/ Noun Phrase	*The party died down before **she arrived***	*The party died down before **her arrival***

To carry out the analysis, we compile a dataset (section §2), consisting of English paraphrase pairs and matching German references. The sentence pairs have similar semantic meaning but a mini-

[1] The code and data will be released upon publication.

mal and controlled change, which allows us a controlled analysis of the activation patterns while attempting to minimize the effect of potential confounds.

We examine the correlation between the activation of neurons (see §4) (1) across models given identical input and (2) between a single model's neurons and themselves, given the source or the paraphrase. We detect strong correlation patterns, some appear in both measurements. This leads us to dissect the correlation to potential confounds, and indeed we discover that similar positional encoding and high overlap in lexical identity are the main contributors to the correlation between paraphrases. This suggests that the strongest correlations are incurred by similar input encoding and not by high-level abstractions learned by the model. Moreover, local correlation patterns do not distinguish between sentence paraphrases and lower level similarities, because the overlap of token embeddings and positional encoding is not a feature that is exclusive to meaning-preserving paraphrases.

We then experiment with controlling the translation output. It is done by simple addition to neuron values - the difference in mean activations over two sentence forms (§5). We show that this manipulation generates outputs that are more similar to the desired form. We find that the way we change activation values is important, but the effect is not localized: many neurons have to be modified to yield a noticeable effect. Lastly, we compare different methods for selecting subsets of neurons to be manipulated (§6). Counter-intuitively, we find that neurons most correlated across paraphrases are better in controlling sentence structure, as opposed to those with the least correlation (i.e. where activation was most changed between different structures). We attribute that to neurons generally important for performance and polysemy of neuron roles.

Overall, we find that strong correlations of neuron activation over paraphrases are explained by shallow features, the positional and token embeddings. Therefore, some neurons represent the input features, but high-level information is not localized. Moreover, we show how the syntactic forms generated during inference can be naïvely controlled, but require a large amount of neurons to modify. This suggests the distinction between different sentence structures is encoded in the model, probably in a widespread manner. Lastly, the neurons that have the most impact on such manipulations are the ones most important for translation in general and not those that differ most over sentence structure paraphrases.

2 Dataset: Minimal Paraphrase Pairs

We aim to isolate representations of specific distinctions in sentence phrasing. To achieve that, we curate a dataset of sentence pairs with controlled syntactic variations. Specifically, we require sentence pairs with the following attributes:

- **Similar Meaning**, to have invariant semantics.
- **Minimal Change**, to facilitate the experimental setup and the interpretation of the results.
- **Controlled Change**, where paraphrasing is consistent and well-defined. As opposed to lexical paraphrases that tend to be idiosyncratic, we require the same distinction to be applied to all instances.
- **Reference Translation**, since we examine translation models.

Existing paraphrasing tools and datasets fail to satisfy our criteria (see §7). Therefore, we develop our own paraphrasing method, which we use to compile two parallel sets: active voice to passive voice and an adverbial clause to a noun phrase. Sentence examples can be found in Table 1.

The proposed process is automatic, following predefined syntactic rules while utilizing several NLP models. First, we identify sentences that match some source patterns (active voice, adverbial clause) according to Dependency Parsing, POS tags (Honnibal et al., 2020) and Semantic Role Labeling (Gardner et al., 2018). Then, we rephrase the sentence to the desired structure. We complement missing prepositions by choosing the one with the highest probability as predicted by BERT (Devlin et al., 2019). For example, the sentence *"She felt accomplished when she met the investor"* requires the preposition "*with*" in the noun phrase form *"She felt accomplished during her meeting with the investor"*, and the temporal preposition *when* is replaced with *during*. In ambiguous instances, we choose whether or not to insert a preposition by opting for the sentence with the higher probability

Table 2: Minimal Paraphrase Pairs count, as derived from WMT19 English-German dev set, before (left) and after (right) filtering.

	Paraphrased	Valid
Adverbial Clause to Noun Phrase	376	114
Active Voice to Passive Voice	3107	1169

according to GPT2 Language Model (Radford et al., 2019). When replacing a verb with a noun (e.g. *arrival* is replaced with *arrive*), we look for the most suitable conversion in existing lexicons, including Nomlex (Macleod et al., 1998), AMR's[2] and Verb Forms[3]. The fine-grained details and step-by-step procedure can be found in Appendix A, along with examples.[4]

We apply our paraphrasing engine to the development set of WMT19 English-German (Barrault et al., 2019). Some paraphrases result in disfluent sentences. For example, the sentence *"He took his time"* is converted to *"His time was taken by him"*, which is syntactically well-formed, but also anomalous. Therefore, we manually filtered the data.[5] The number of examples is given in Table 2.

3 Technical Setup

Model. We demonstrate our model-agnostic methodology with the Transformer model for Machine Translation (Vaswani et al., 2017). We use the fairseq implementation (Ott et al., 2019), which was trained on the WMT19 English-German train set (Barrault et al., 2019). The embedding dimension is 1024 with learnt token embeddings and sinusoidal positional encoding.

Notations and Definitions. For our purposes, neurons are any of the 1024 values in the output embedding as produced by each of the 6 layer blocks. We refer to trained models with different random initialization as m_1, m_2. We denote the set of source sentences $S = \{s_1, s_2, ..., s_n\}$, and its corresponding paraphrased set with $P = \{p_1, p_2, ..., p_n\}$ (e.g., s_i is an active voice sentence and p_i is its passive counterpart). The activation of a neuron in model m, location (layer and index) l on sentence s_i is $x_S^{m,l}[i]$, while $x^{m,l}$ is a vector of size n. Following previous works (Liu et al., 2019; Wu et al., 2020) we consider only the last sub-word token activation for each word[6]. Since the number of words may differ between paraphrases, we average activation values over the words in a sentence, to allow for a uniform sample size.[7]

Dataset. For all experiments we use our minimal paraphrases dataset (see §2). Due to space considerations, we present results on the active/passive set in the main paper, while clause/noun phrase results can be found in Appendices B.2 and C.

4 Detecting Correlation Patterns

To detect activation patterns, we measure Pearson correlation[8] between neural activations. The correlation will allow us to examine how neurons activate under different conditions.

[2]`https://amr.isi.edu/download/lists/morph-verbalization-v1.01.txt`

[3]`https://github.com/monolithpl/verb.forms.dictionary`

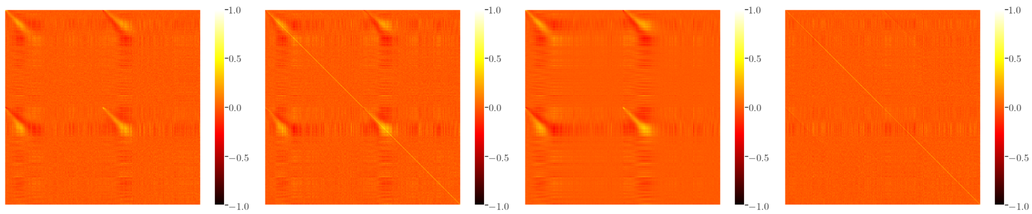
[4]The paraphrasing engine code and the dataset derived from WMT19 will be released upon publication.

[5]Two in-house annotators made binary predictions as to whether the generated paraphrases are fluent, with 75% observed agreement and 0.6 Cohen's kappa. We also tried using Direct Assessment (Graham et al., 2017) and eliciting fluency scores through crowdsourcing, as well as attempting to threshold the probability given by GPT2 or SLOR (Kann et al., 2018). Neither of these approaches worked in a satisfactory manner.

[6]We experimented with taking all sub-word tokens into the calculation, with similar results.

[7]Notably, experiments with taking minimum or maximum activation instead of average have shown similar results in the sense that the effects we present in §4 and §4.1 are included in the minimum and maximum maps. When applicable, we experimented with no pooling at all, with similar results.

[8]We experimented with Spearman correlation as well, but did not observe major differences.



(a) ModelCorr (b) ParaCorr (c) PosCorr (d) TokenCorr

Figure 1: Activation correlation of first layer neurons in the encoder, using the active-passive dataset.

First, we follow Bau et al. (2019) and define *ModelCorr* to be the correlation between any pair of neurons across models, when given the same input of source sentences:

$$ModelCorr(l, l') = \rho(x_S^{m_1,l}, x_S^{m_2,l'}) \tag{1}$$

We extend this definition to capture correlation across paraphrases, denoted with *ParaCorr*. Given the exact same model instance, we look at activations over a set of sentences and their correlation to the activations over the paraphrased set:

$$ParaCorr(l, l') = \rho(x_S^{m_1,l}, x_P^{m_1,l'}) \tag{2}$$

Figures 1a and 1b show ModelCorr and ParaCorr correlation maps.[9] Some of ParaCorr's observed effect also appears in ModelCorr, suggesting it might be unrelated to the examined variable, i.e. paraphrases. Moreover, ModelCorr indicates a strong correlation between neurons of the same location in different models, but the Transformer architecture in itself does not account for positions.

4.1 Controlling for Confounds

In this section, we show that strong activation correlations between paraphrases are a product of low-level cues. Namely, we inspect how the propagation of token identity and positional information greatly influences the correlation. This is a relevant confound to note for previous work adapting correlation analysis on neurons (Bau et al., 2019; Wu et al., 2020; Meftah et al., 2021). The positional encoding in our setting is sinusoidal, therefore the same positions are encoded exactly the same across models. Paraphrases have a minor change in sentence length, which incurs similar positional encoding. As for tokens, paraphrases have a large overlap of bag-of-words.

We define *PosCorr* as an activation correlation between sentences with identical positional encoding but different token embeddings. Formally ($\hat{S}$ is a set of random token sequences matching S in lengths):

$$PosCorr(l, l') = \rho(x_S^{m_1,l}, x_{\hat{S}}^{m_1,l'}) \tag{3}$$

Indeed, PosCorr isolates the strong correlation effect observed both in ModelCorr and ParaCorr (Fig. 1c). Repetition through the layers is probably due to the residual connections, which propagate the positional encoding. Indeed, when we looked at correlations of neurons inside the layer block - before the first residual connection - the effect seen in PosCorr was missing (see Appendix B.1). The implication is that input representation, and not higher-level learnt representation, is likely the cause of strong correlations.

As input representation is composed of tokens and their positions, the counterpart correlation to PosCorr is *TokenCorr*, to account for token embeddings. We strip an input set S from its positional encoding, denoted by $\tilde{S}$, and compare its activations to those of the intact S:

$$TokenCorr(l, l') = \rho(x_S^{m_1,l}, x_{\tilde{S}}^{m_1,l'}) \tag{4}$$

TokenCorr (Fig. 1d) captures the diagonals phenomenon of ParaCorr, explained by paraphrases having a large bag-of-words overlap (the effect is not present in ModelCorr since token embeddings are

[9]We feature only the first layer due to resolution constraints. Any effect shown is present in all layer block pairs but weakens when moving away from the main diagonal or when the layers are higher.

learnt). This implies that individual token identities, and not necessarily sentence-level semantics, contribute to strong correlations. This distinction is made apparent when we consider how word order may affect meaning. For example, *"Rose likes Josh"* has a widely different meaning than *"Josh likes Rose"*, although the sentences are comprised of the exact same bag of words.

We further dissect the observed correlation for possible causes. First, we compare activations on different sentences that only share the relevant syntactic structure (e.g., two random active voice sentences). No strong correlation is observed (between -0.17 to 0.20). This suggests that the effect observed in the TokenCorr experiment, where the same tokens are fed to the model (Fig. 1d, Eq. 4) is not explained by a similar sentence structure (i.e., active voice). In another experiment, we combine both PosCorr and TokenCorr: we strip the original sentence from its positional embedding and replace the tokens with random ones – i.e., no input is shared between the compared conditions. As little correlation is detected (between -0.27 to 0.31), we rule out the possibility that the correlation is caused by neurons of constant value.

Overall, our confound analysis implies the following: (1) strong activation correlation is greatly due to low-level components and not high-level learnt knowledge, (2) strong correlation detected across paraphrases may not be exclusive to sentences with similar meaning and different structure, and (3) sentence structure is not detected locally using correlation analysis.

5 Manipulating Activations

Being able to manipulate neurons allows us to control the translation output (without additional training), which in turn adds a causative dimension to our understanding of neurons. We look into changing the activation values to force the output translation to have a desired syntactic structural feature (e.g., active or passive voice). Although we did not detect individual neurons that have a strong positive or negative correlation across paraphrases, these distinctions could still be encoded in a decentralized manner in the model, and therefore susceptible to manipulation. We address three main questions:

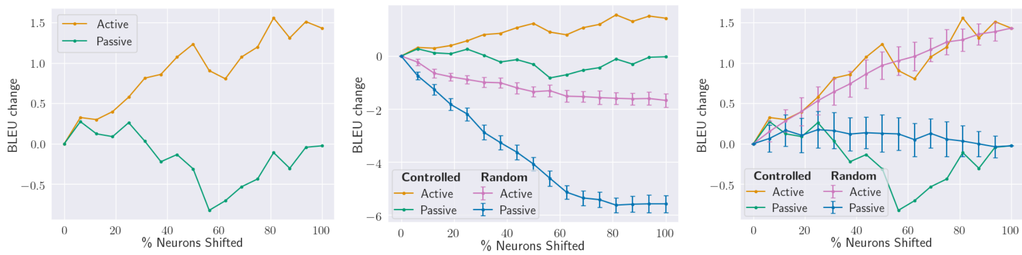
1. Can we effectively control structural properties of the output by changing neuron values?
2. Does the exact value matter or only the identity of the modified neurons?
3. How would we choose only a sub-set of neurons to manipulate?

5.1 Setup

Our technique is a simple translation of the activation values towards the average activation of a desired syntactic structure. In doing so, we extend the approach of Bau et al. (2019), who modified individual neurons according to their average activations.

We denote with y_c^l the average activation of the neuron in position l under a condition c. The formulation is general, but in this work we focus on the paraphrase's form, e.g., active voice where $y_c[l] \equiv \frac{1}{n}\sum_{i=1}^{n} x_S^{m_1,l}[i]$. The vector of average activations of all m neurons as recorded under condition c is denoted with $y_c \in \mathbb{R}^m$. Manipulation from c_1 to c_2 is defined as $\frac{1}{\|y_{c_1}-y_{c_2}\|}(y_{c_1} - y_{c_2}) \in \mathbb{R}^m$. This vector defines a subtraction for every neuron, but may applied to any subset of neurons. The normalization term is introduced to make manipulations comparable in size when all neurons are modified. We investigate two parameters: the direction of manipulation (from c_1 to c_2) and the set of neurons we apply it to. For an additional experiment on scaling the magnitude of the manipulation, see Appendix C.5.

We evaluate whether the manipulation increases the similarity of the output to a reference with the target form (c_2), relative to similarity with source form (c_1). We measure BLEU score between our model's translation and Google translations, which (in the absence of manual references) we consider as references to both source and target forms. This is a reasonable assumption given the performance gap between the models we train and Google Translate. Later, we discuss evaluation by additional methods to complement BLEU (see §5.3).



(a) Controlled baseline (b) Random direction comparison (c) Random selection comparison

Figure 2: Manipulating the outputted translation to be active voice when feeding passive voice as input. Lines present BLEU change with active and passive references, as a function of the amount of neurons manipulated (x-axis)

5.2 Experiments

We present experiments manipulating passive voice inputs towards active voice translations. The opposite manipulation (active input to passive translation) and the results on the clause/noun-phrase set can be found in Appendix C.

Baseline Manipulation. We modify an increasing amount of neurons, choosing first the neurons most correlated to themselves according to ParaCorr (i.e., we rank by $ParaCorr(l, l)$ with the higher values first). The motivation to use the correlation as a rank is based on Bau et al. (2019), who ranked neurons according to their correlation across models. We manipulate passive voice inputs towards active voice translations. Our outputs become more similar to active voice than passive voice (Fig. 2a), suggesting that sentence structure is indeed encoded in the model. Moreover, the information is used by the model when generating translations and it can be controlled.

Direction of Manipulation. We explore the importance of the manipulation direction by shifting towards a random vector $y_r \in \mathbb{R}^m$ (e.g., manipulate from average passive activation to a random value). We repeat the process 100 times and show the average results with standard deviation in Fig. 2b. We find it to be substantially worse, implying the success of the manipulation is tied to the direction we shift towards, and not an artifact of value modification.

Selection of Manipulation. We test whether there is a preferable subset of neurons to manipulate by randomly choosing neurons to manipulate[10] (while applying the value modification as the baseline). Results (Fig. 2c) do not indicate that a controlled selection of neurons (according to ParaCorr ranking) is better than random. Overall, it seems that a large subset of neurons has to be modified to obtain the desired outcome, which agrees with our correlation results, where the active/passive feature was not localized. The correlation between paraphrases can shed light on what sub-sets of neurons could still be better for manipulation, which we discuss later in section 6.

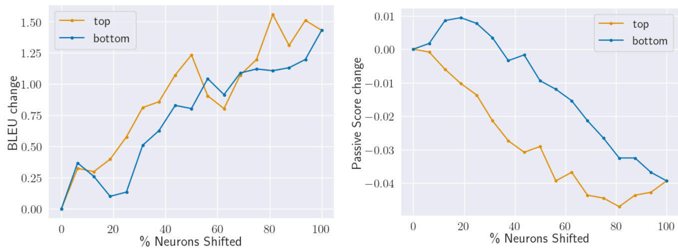
5.3 Beyond BLEU

BLEU score captures translation quality on the surface and not necessarily how good (or bad) it is at preserving meaning or capturing form (active vs. passive). Therefore, we employ additional evaluation measures.

Passive Score. Specifically for the active/passive dataset, we use a dependency parser and POS tagger to detect passive form[11]. The scorer shows a decrease of detected passive voice when we manipulate the passive input towards active translation (see figure 3b). The magnitude of the decrease

[10]We also experimented with choosing random neurons under the constraint they have the same distribution among the 6 encoder layers as the controlled case. The results were the same.

[11]Using Spacy (Honnibal et al., 2020), we consider a sentence to be in passive voice if the root lemmatization is *"werden"* and it has a child of dependency *"oc"* (i.e., clausal object) with a tag indicating a participle form.



(a) BLEU (b) Passive Score

Figure 3: Top ParaCorr neurons are better for manipulation. Manipulating the output translation to be active voice when feeding passive voice as input. Comparing the choice of neurons to manipulate when starting from the top or bottom according to the rank given by ParaCorr. (1) Measuring by BLEU against active voice references and (2) measuring passive score that automatically detects passive voice.

may seem small, but the scorer might have a limited recall: the baseline translation (no manipulation) of active voice sentences is detected as 0.94% passive, while passive voice input is translated into 37.38% passive.

Qualitative Analysis. A native German speaker examined a sample of output translations and found successful manipulations (see examples in appendix D). She discussed 'fail' cases – where the translation changed (i.e. unequal strings) but did not result in the desired form. This analysis has yielded that in some cases, the manipulation changed between a stative passive and dynamic passive, rather than between an active and a passive (the distinction between these passive types is more evident in German). In other cases, the manipulation was not applicable. For instance, some verbs could not be translated to an adverbial verb form and demand either to appear as a noun phrase or be replaced with a synonym verb (an example is in Appendix D). These suggest that the manipulation was successful even when not automatically detected as such, and is limited according to the target language and the model capabilities to generalize to synonyms while controlling the sentence structure.

6 Significance of Specific Neuron Sets

In our baseline manipulation in §5.2 we chose what neurons to modify according to the rank given by ParaCorr (i.e., sorting all neurons by $ParaCorr(l, l)$, high to low). Under an intuitive interpretation, neurons that still positively correlate when systematic changes are made to the input are those invariant to that change. Neurons with a negative correlation are specific to the change. Following these, we expect that applying our manipulation on a set of neurons with the lowest rank would yield better results than top ranked neurons. Contrary to intuition, we observe the opposite phenomenon, as seen in figure 3. We perform some tests, in an attempt to explain this.

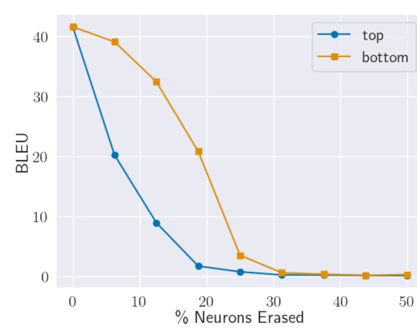


Figure 4: Top ranked neurons have a stronger impact on the translation quality of a test set, measured in BLEU. Erasure of neurons from the top or bottom of the rank given by the value of correlation between paraphrases.

Model Performance. Going back to the foundations of our methodology, Bau et al. (2019) identified important neurons (in an LSTM) by ranking the most correlated neurons across models. To verify the "importance" notion, they delete (i.e., set the activations to zero) neurons from the top versus the bottom of the rank and examined the impact on the model performance. We apply this experiment in our settings: we set to zero an increasing amount of neurons, according to ParaCorr. We measure BLEU on a held-out set of 552 active voice sentences extracted from the WMT19 test set. Results (figure 4) show that top ranked neurons have a stronger impact on the translation quality than lower ranked do, suggesting that ParaCorr partially ranks neurons by their general importance. This might explain the above counter-intuitive result; Top neurons have the most impact on translation, and hence have the most impact when manipulated.

Role Overlap. The top ParaCorr neurons are the same neurons that account for lexical identity and positional information. This fact explains why they have the most impact when manipulating sentence structure. Sentence structure is tied to word order, especially in active-passive, where the subject and direct object replace positions. Word tokens are the building blocks for the semantic meaning of the sentence (which should remain the same across paraphrases), even when bag-of-words is not exclusive to a specific meaning. The first evidence to support this claim is seen in §4, where most of the strong correlations in ParaCorr are explained by similarity in the tokens and the positional embeddings between the inputs (i.e., TokenCorr and PosCorr, respectively). In an additional test, we check how many of the top ParaCorr neurons are also top PosCorr and TokenCorr neurons. Figure 5 shows that for any count x, the set of top x ParaCorr neurons have an intersection with the sets of top x PosCorr or TokenCorr neurons.

7 Related Work

Understanding Neural Networks in NLP. Various approaches were previously proposed to explain neural networks (Belinkov & Glass, 2019), each with a methodology that differs from ours. Probing tasks investigate whether linguistic properties of the input text can be effectively predicted from model representations (Jawahar et al., 2019; Tenney et al., 2019; Slobodkin et al., 2021). They shed light on what information is kept within a model, but not necessarily on what is used, or how. Other works study causation, for example, Vig et al. (2020) employs mediation analysis theory to interpret what parts of a model generate a certain semantic behavior. Analysis of the interaction between input and output while exploiting internal knowledge was done by He et al. (2019), relying on gradients. Some works analyze attention heads (Voita et al., 2019) or follow attention flow in the network (Abnar & Zuidema, 2020). Visualization tools interpret activations and with some exceptions (e.g., Lenc & Vedaldi, 2015), they are mostly limited to qualitative examples. Other works interpreting individual neurons include Durrani et al. (2020), which use probing-like methods for a more fine-grained analysis. Challenge sets (Choshen & Abend, 2019; Warstadt et al., 2020), as well as adversarial examples (Alzantot et al., 2018), expose challenging cases by analyzing the NMT system's behavior, rather than representation. Elazar et al. (2021) analyze semantically equivalent input by clustering representation. They also improve prediction by continual training, while we manipulate translation post training.

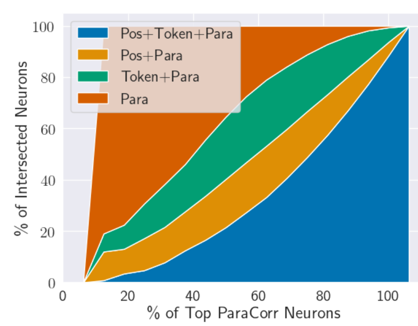


Figure 5: Top ParaCorr neurons intersect with neurons most related to token embeddings and positional encoding. The x-axis represents the amount of top ParaCorr neurons as a percentage of all the neurons in the encoder. The y-axis shows how many of these x neurons are also in the set of top x TokenCorr neurons or top x PosCorr neurons. The y-axis scale is a percentage out of x. Results are reported for the active-passive set.

Analysis in other domains. Some Computer Vision work resembles our approach. Lenc & Vedaldi (2015) delve into the interaction between input transformation and its representation along the layers, while Goodfellow et al. (2009) examine invariant neurons, those that are selective to high-level features but are robust given semantically identical transformation. Their methodologies do not fit the NLP domain since they rely on a mathematically well-defined input transformation. We propose an alternative with our paraphrases in section §2, and thus we analyze the relation between the input and individual representations. In the field of neuroscience, analysis on the encoding of linguistics in the human brain have been done in a similar methodology to ours: Friederici (2011) analyzed the correlation of neuroimaging where subjects are presented with sentences with subtle syntactic variations or violations, and well correlated regions are considered to process syntax.

Individual neurons analysis using correlation. Correlation between activations of individual neurons have been analyzed before in different settings. Bau et al. (2019) used it to detect neurons that are most correlated across LSTM models, while showing these are the most important for performance. They manipulated individual neurons to control single words in the output (e.g., gender, tense). The technique to identify neurons that activate similarly in different models was previously suggested by Dalvi et al. (2019), who found neurons in LSTM models to have role polysemy, aligning with our discussion in §6. Later, Wu et al. (2020) employed correlation to examine similarities of different Transformer architecture. Meftah et al. (2021) adapted correlations analysis to quantify the impact of fine-tuning by measuring activations of neurons before and after domain adaptation.

Controlling active-passive voice in translation. Manipulation of the sentence structure of a translated sentence, and specifically voice, was explored by Yamagishi et al. (2016). They controlled voice (active/passive) in Transformer translation from Japanese to English, when an indicator was given as input. Unlike ours, their method required additional model training.

Paraphrases Existing paraphrasing tools vary by how localized the edits they perform on the sentence are. Some alter the lexical level (Ribeiro et al., 2018), other alter whole phrases (Ganitkevitch, 2013; Bhagat et al., 2009), some are sentence-level paraphrases (Dolan et al., 2004), while some split source sentences into sub-sentences (e.g., Dornescu et al., 2014; Lee & Don, 2017). Other than paraphrasing tools, some existing datasets include the PPDB database (Pavlick et al., 2015) that contains sentence paraphrases that are lexical, phrasal, or syntactic. Zhang et al. (2019); Dolan & Brockett (2005) include paraphrase and non-paraphrase pairs, the former with high lexical overlap, while (Hu et al., 2019) contains multiple paraphrases of lexical diversity. None of these match our criteria for paraphrases (§2).

8 CONCLUSION

In this work, we propose a novel approach to understanding Neural Machine Translation models. With our curated dataset, we measured correlation to detect activation patterns across paraphrases. By a meticulous confound analysis, we found that similarity of activations across paraphrases is likely due to similarity of sequence length or word identity overlap, which are important components of paraphrases but are not exclusive to meaning-preserving variations. We emphasize how these confounds must be taken into account when attempting to detect local correlation under any experimental setup. Our results imply that the strongest correlations observed are due to the input representation, and to high-level abstraction.

We investigated activation manipulation to control translation to be of a specific sentence structure. Our experiments show that changing the activation value towards the averages activation under a target feature increases the similarity of translation to the target form. Our results thus imply that sentence structure is captured by the model, but in a decentralized way. This suggests that sentence structure is not a localized feature in representation. This aligns well with our correlation analysis, where we did not detect strong localized activation patterns after eliminating confounds.

Works in neuroscience also indicate that sentence structure might not be a local phenomenon but spread across the encoder latent representation. Blank et al. (2016); Reddy & Wehbe (2020) find that syntax processing is distributed across the language system in the human brain. Fedorenko et al. (2012) suggest that lexical information may play a more critical role than syntax in the representation of linguistic meaning, which relates to our findings in §6.

We hope our paraphrasing engine and the dataset derived from WMT19 will be beneficial to others for network analysis tasks or otherwise. Moreover, our correlation methodologies are model-agnostic and can be applied with any variation in input, which we hope will inspire future work in this field.

REFERENCES

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.385. URL https://www.aclweb.org/anthology/2020.acl-main.385.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2890–2896, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1316. URL https://www.aclweb.org/anthology/D18-1316.

Loïc Barrault, Ondřej Bojar, Marta R Costa-Jussa, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, et al. Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pp. 1–61, 2019.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. Identifying and controlling important neurons in neural machine translation. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1z-PsR5KX.

Yonatan Belinkov and James Glass. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72, March 2019. doi: 10.1162/tacl_a_00254. URL https://www.aclweb.org/anthology/Q19-1004.

Rahul Bhagat, Eduard Hovy, and Siddharth Patwardhan. Acquiring paraphrases from text corpora. In *Proceedings of the Fifth International Conference on Knowledge Capture*, K-CAP '09, pp. 161–168, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605586588. doi: 10.1145/1597735.1597764. URL https://doi.org/10.1145/1597735.1597764.

I. Blank, Zuzanna Balewski, Kyle Mahowald, and Evelina Fedorenko. Syntactic processing is distributed across the language system. *NeuroImage*, 127:307–323, 2016.

Leshem Choshen and Omri Abend. Automatically extracting challenge sets for non-local phenomena in neural machine translation. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pp. 291–303, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/K19-1028. URL https://www.aclweb.org/anthology/K19-1028.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, Anthony Bau, and James Glass. What is one grain of sand in the desert? analyzing individual neurons in deep nlp models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 6309–6317, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

Bill Dolan, Chris Quirk, and Chris Brockett. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*, pp. 350. Association for Computational Linguistics, 2004.

William B. Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, 2005. URL https://www.aclweb.org/anthology/I05-5002.

Iustin Dornescu, Richard Evans, and Constantin Orăsan. Relative clause extraction for syntactic simplification. In *Proceedings of the Workshop on Automatic Text Simplification - Methods and Applications in the Multilingual Society (ATS-MA 2014)*, pp. 1–10, Dublin, Ireland, August 2014. Association for Computational Linguistics and Dublin City University. doi: 10.3115/v1/W14-5601. URL `https://www.aclweb.org/anthology/W14-5601`.

Nadir Durrani, Hassan Sajjad, Fahim Dalvi, and Yonatan Belinkov. Analyzing individual neurons in pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4865–4880, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.395. URL `https://www.aclweb.org/anthology/2020.emnlp-main.395`.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard H. Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and improving consistency in pretrained language models. *CoRR*, abs/2102.01017, 2021. URL `https://arxiv.org/abs/2102.01017`.

Evelina Fedorenko, A. Nieto-Castañón, and N. Kanwisher. Lexical and syntactic representations in the brain: An fmri investigation with multi-voxel pattern analyses. *Neuropsychologia*, 50: 499–513, 2012.

A. Friederici. The brain basis of language processing: from structure to function. *Physiological reviews*, 91 4:1357–92, 2011.

Juri Ganitkevitch. Large-scale paraphrasing for natural language understanding. In *Proceedings of the 2013 NAACL HLT Student Research Workshop*, pp. 62–68, Atlanta, Georgia, June 2013. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/N13-2009`.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pp. 1–6, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-2501. URL `https://www.aclweb.org/anthology/W18-2501`.

Ian Goodfellow, Honglak Lee, Quoc Le, Andrew Saxe, and Andrew Ng. Measuring invariances in deep networks. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta (eds.), *Advances in Neural Information Processing Systems*, volume 22, pp. 646–654. Curran Associates, Inc., 2009. URL `https://proceedings.neurips.cc/paper/2009/file/428fca9bc1921c25c5121f9da7815cde-Paper.pdf`.

Yvette Graham, Timothy Baldwin, Alistair Moffat, and Justin Zobel. Can machine translation systems be evaluated by the crowd alone. *Natural Language Engineering*, 23(1):3–30, 2017.

Shilin He, Zhaopeng Tu, Xing Wang, Longyue Wang, Michael Lyu, and Shuming Shi. Towards understanding neural machine translation with word importance. 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)*, pp. 953–962, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1088. URL `https://www.aclweb.org/anthology/D19-1088`.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spacy: Industrial-strength natural language processing in python, 2020. URL `https://doi.org/10.5281/zenodo.1212303`.

J. Edward Hu, Rachel Rudinger, Matt Post, and Benjamin Van Durme. Parabank: Monolingual bitext generation and sentential paraphrasing via lexically-constrained neural machine translation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6521–6528, Jul. 2019. doi: 10.1609/aaai.v33i01.33016521. URL `https://ojs.aaai.org/index.php/AAAI/article/view/4618`.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3651–3657, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1356. URL `https://www.aclweb.org/anthology/P19-1356`.

Katharina Kann, Sascha Rothe, and Katja Filippova. Sentence-level fluency evaluation: References help, but can be spared! In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pp. 313–323, Brussels, Belgium, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/K18-1031. URL https://www.aclweb.org/anthology/K18-1031.

John Lee and J. Buddhika K. Pathirage Don. Splitting complex English sentences. In *Proceedings of the 15th International Conference on Parsing Technologies*, pp. 50–55, Pisa, Italy, September 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W17-6307.

K. Lenc and A. Vedaldi. Understandinspacyg image representations by measuring their equivariance and equivalence. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 991–999, 2015. doi: 10.1109/CVPR.2015.7298701.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. Linguistic knowledge and transferability of contextual representations. 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)*, pp. 1073–1094, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1112. URL https://www.aclweb.org/anthology/N19-1112.

Catherine Macleod, Ralph Grishman, Adam Meyers, Leslie Barrett, and Ruth Reeves. Nomlex: A lexicon of nominalizations. In *In Proceedings of Euralex98*, pp. 187–193, 1998.

Sara Meftah, N. Semmar, Y. Tamaazousti, H. Essafi, and F. Sadat. Neural supervised domain adaptation by augmenting pre-trained models with random units. *ArXiv*, abs/2106.04935, 2021.

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.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 425–430, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-2070. URL https://www.aclweb.org/anthology/P15-2070.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Aniketh Janardhan Reddy and Leila Wehbe. Syntactic representations in the human brain: beyond effort-based metrics. *bioRxiv*, 2020.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 856–865, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1079. URL https://www.aclweb.org/anthology/P18-1079.

Aviv Slobodkin, Leshem Choshen, and Omri Abend. Mediators in determining what processing bert performs first. *arXiv preprint arXiv:2104.06400*, 2021.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL https://www.aclweb.org/anthology/P19-1452.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart M. Shieber. Causal mediation analysis for interpreting neural NLP: the case of gender bias. *CoRR*, abs/2004.12265, 2020. URL https://arxiv.org/abs/2004.12265.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan 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*, pp. 5797–5808, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1580. URL https://www.aclweb.org/anthology/P19-1580.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392, 07 2020. ISSN 2307-387X. doi: 10.1162/tacl_a_00321. URL https://doi.org/10.1162/tacl_a_00321.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

John Wu, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. Similarity analysis of contextual word representation models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4638–4655, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.422. URL https://www.aclweb.org/anthology/2020.acl-main.422.

Hayahide Yamagishi, Shin Kanouchi, Takayuki Sato, and Mamoru Komachi. Controlling the voice of a sentence in japanese-to-english neural machine translation. In *WAT@COLING*, 2016.

Yuan Zhang, Jason Baldridge, and Luheng He. PAWS: paraphrase adversaries from word scrambling. *CoRR*, abs/1904.01130, 2019. URL http://arxiv.org/abs/1904.01130.

A Compilation of Minimal Paraphrase Pairs

A.1 Tools and Techniques

We explain, in greater detail, the main tools we use when we paraphrase, as briefly discussed in section §2.

Pattern Detection. Making sure we change form but not semantics, we rely on syntax patterns, not word-based. We use dependency parsing (including Part of Speech tagging) and Semantic Role Labeling combined (by Honnibal et al. (2020) and Gardner et al. (2018), respectively) to detect active form and adverbial clauses by type (see table 3).

Sentence Probability Used for choosing between two sentence options (with or without a certain preposition?). We use gpt2 model Radford et al. (2019) by huggingface Wolf et al. (2020) to get sentence probability for each option and opt for the higher.

Word Insertion.

Input: a sentence $X = x_1, x_2, ..., x_n$, a position *i*, and a set of possible words $W = \{w_1, ..., w_m\}$

1: Define $X' = x_1, \ldots, x_{i-1}, [MASK], x_i, \ldots, x_n$
2: Send X' into a trained BERT masked language model Devlin et al. (2019) by Wolf et al. (2020) and get $y \in \mathbb{R}^n$, a probability vector for each word in the vocabulary (d is the size of the vocabulary of the BERT model)
3: Define $w_k \in W$ s.t. $w_k = \max_{i=1,\ldots m} y[w_i]$ to be a new word at position *i* of sentence X (the word with the highest probability, according to BERT, out of the given set W).

Output: a sentence $(x_1, x_2, \ldots, x_{i-1}, w_k, x_i, \ldots, x_n)$

We can make this an **Optional Word Insertion** by returning either the input or output sentence, using Sentence Probability.

Noun Derivation.

Input: a verb (lemma form)
We prioritize choosing the noun form from AMR morph verbalization [12]. If we don't find it there, we choose between Nomlex Macleod et al. (1998) form and present participle form according to Verb Form Dictionaries[13] (if exists), deciding according to Word Insertion.
Output: either a noun or None

Preposition Sets. Using Word Insertion requires a set of options as input. In our paraphrasing process we use the following predefined sets to insert prepositions. **Temporal prepositions:** 'as', 'aboard','along', 'around', 'at', 'during', 'upon', 'with', 'without' **General prepositions:** 'as', 'aboard', 'about', 'above', 'across', 'after', 'against', 'along', 'around', 'at', 'before', 'behind', 'below', 'beneath', 'beside', 'between', 'beyond', 'but', 'by', 'down', 'during', 'except', 'following', 'for', 'from', 'in', 'inside', 'into', 'like', 'minus', 'minus', 'near', 'next', 'of', 'off', 'on', 'onto', 'onto', 'opposite', 'out', 'outside', 'over', 'past', 'plus', 'round', 'since', 'since', 'than', 'through', 'to', 'toward', 'under', 'underneath', 'unlike', 'until', 'up', 'upon', 'with', 'without'.

A.2 Active Voice to Passive Voice

The active to passive paraphrasing process is done on sentences that include a nominal subject and a direct object. We discard any sentence of question and coordination, possible passive form (root verb is in past participle), and when the root verb has a *"to"* auxiliary.

1: If the subject is a proper noun, convert it to object form
2: If the direct object is a proper noun, convert it to subject form
3: Switch the subtree spans of subject and object
4: Add "by" just before the span of the new object
5: If an auxiliary verb is one of "can", "may", "shall", convert it to "could", "might", "should" respectively.
6: If root verb is a gerund or present participle, replace it with "being". Otherwise, remove it altogether.
7: Add suitable auxiliary according to the new subject form of singular/plural, and the tense.
8: If the sentence includes a negation word, remove it and add "not" before the auxiliary.
9: Replace the root verb to its past participle form (using the Verb Forms Dictionary[14]).
10: If the sentence includes a particle, move it after the root verb.
11: If the sentence includes a dative, try to replace it using Optional Word Insertion.

We'll go over an example:

Active to Passive: example

Input: He can't take the book.

[12] https://amr.isi.edu/download.html
[13] https://github.com/monolithpl/verb.forms.dictionary
[14] https://github.com/monolithpl/verb.forms.dictionary

1: $''He'' \leftarrow ''Him''$
2: NA
3: Switch *"him"* with *"The book"*
4: $''him'' \leftarrow ''by\ him''$
5: $''can'' \leftarrow ''could''$
6: NA
7: Add *"be"*
8: $'''t'' \leftarrow ''not''$
9: $''take'' \leftarrow ''taken''$
10: NA
11: NA
Output: The book could not be taken by him.

1. Extract	**Extract Adverbial Clause** detect type by Semantic Role Labeling (Gardner et al., 2018)			
	Cause/Reason		**Temporal**	**Purpose**
	Possessive	Non-Possessive		
2. Match pattern	root "have" and marker "because"	root isn't: "have"/ "be"/"do"/"can"	marker "as"/"before"/ "after"/"until"/"while" or adverbial modifier "when"	participle "to"
3. aux	remove root's auxiliaries			
4. det/Noun	remove direct object's determinants	Noun Derivation A.1		
5. Possession	Nominal subject to possessive form			
6. Preposition	replace "because" with "because of"		If "as"/"while"/ "when", replace by Word Insertion A.1[a]	Replace "to" with "for"
7. Additions	If negation, add "lack of"	If there is a direct object[b] Optional Word Insertion A.1[c]		

[a] Using temporal prepositions set.
[b] If there is a direct object of the form "¡xxx¿self" in the non-possessive cause/reason case, we instead add "self" before the derived noun and remove this object.
[c] Using general prepositions set.

Table 3: The paraphrasing process from adverbial clause sentence to a noun phrase.

The complete process of paraphrasing a sentence with an adverbial clause to one with a noun phrase substituting it is detailed in table 3. We'll demonstrate a few examples [15].

Purpose clause

Input: She sat under the sun to enjoy the warmth.
1: Extract *"to enjoy the warmth"*
2: Found matching participle *"to"*
3: NA
4: $''enjoy'' \leftarrow ''enjoyment''$
5: NA
6: $''to'' \leftarrow ''for''$
7: $''thewarmth'' \leftarrow ''ofthewarmth''$
Output: She sat under the sun for enjoyment of the warmth.

[15]The flow of purpose clause conversion lacks optional determiner addition before the new noun phrase. It will be fixed upon publication.

Cause/Reason clause, possessive form

Input: She was at the library for a long time because she had an unresolved problem.

1: Extract *"because she had an unresolved problem"*
2: Found matching root *"had"* and a marker *"because"*
3: Remove *"had"*
4: Remove *"an"*
5: $"she" \leftarrow "her"$
6: $"because" \leftarrow "because\ of"$
7: NA

Output: She was at the library for a long time because of her unresolved problem.

Cause/Reason clause, non-possessive form

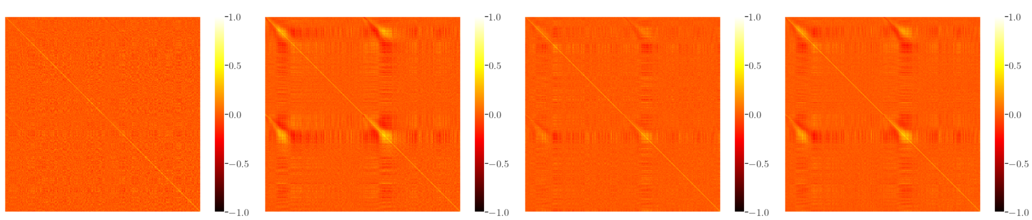
Input: This robot is very advanced because it flies itself.

1: Extract *"because it flies itself"*
2: Found matching root *"flies"* and a marker *"because"*
3: NA
4: $"flies" \leftarrow "flight"$
5: $"it" \leftarrow "its"$
6: $"because" \leftarrow "because\ of"$
7: $"flight" \leftarrow "self\ flight"$

Output: This robot is very advanced because of its self flight.

B Detecting Correlation Patterns

B.1 Inside the Layer Block



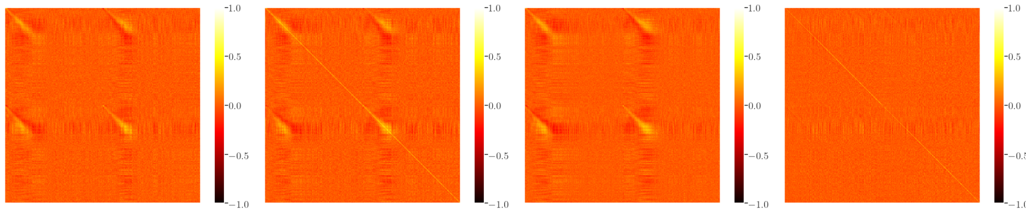
(a) Post Attention (b) Residual + Norm (c) Linear Layers (d) Residual + Norm

Figure 6: Activation correlation between paraphrases (ParaCorr), using the active-passive dataset. A view inside the first encoder layer block, step-by-step: (a) attention heads, (b) adding residual connections and applying normalization, (c) fully connected layer, followed by ReLU and another fully connected layer, (d) adding residual connections and applying normalization - the output of the layer block.

In section 4 we measure the correlation of activations only at the output of the encoder layer block, following previous work (Wu et al., 2020). We also take a look at intermediate activations, see figure 6. This strengthens our hypothesis that the strong correlation seen in PosCorr (figure 1c) is due to the sinusoidal positional encodings, as they are propagated through the network with residual connections. The PosCorr effect appears only after the first residual connection, weakens through the fully-connected layers, and strengthens again after additional residual connection.

B.2 Adverbial Clause versus Noun Phrase

Here we present the same correlations methods detailed in section 4but measured on the adverbial clause versus noun phrase sets. See figure 7.

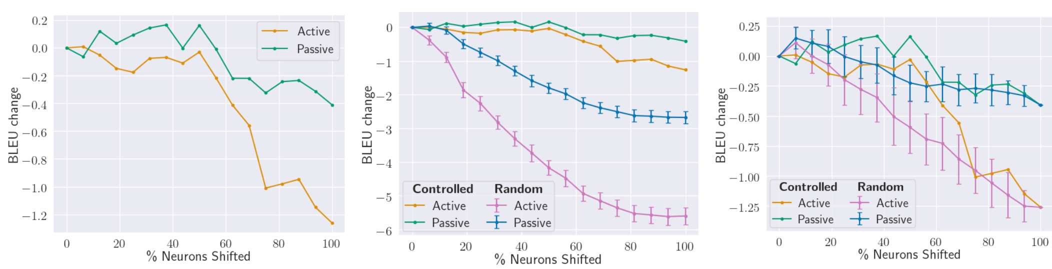


(a) ModelCorr (b) ParaCorr (c) PosCorr (d) TokenCorr

Figure 7: Activation correlation of first layer neurons in the Transformer encoder, using the clause/noun-phrase dataset.

C Manipulation of Neurons

C.1 Active to Passive



(a) Baseline (b) Random direction comparison (c) Random selection comparison

Figure 8: Manipulating output translation to be passive voice when feeding active voice as input. Lines present BLEU change with active and passive references according to the amount of neurons manipulated (x).

To complete all variations of the manipulation experiment, we first showcase the shift from active voice input to passive voice translation (the opposite direction than what we showed in the paper). We see that the translation is more similar to the target form (passive voice) than the input form (active voice). The positive change in BLEU is more subtle in this manipulating, and again getting maximal change requires many neurons to be modified (at least 50%), see Fig. 8a. With the random experiments of direction (Fig. 8b) and neurons selection (Fig. 8c), we get similar results - our controlled direction is better while choosing a subset of neurons is not easy.

C.2 Manipulation on a Test Set

We repeat the manipulation on a held-out test set: 552 sentences that we detect as active voice from the WMT19 test set. While our experiments on the dev set are valid, as we manipulate from one set (e.g. passive voice) by measuring on another (e.g. active voice), one might argue that we can't know the effect the shared semantic meaning (on the set level) has on the success rate. To cover all bases, we manipulate the test set according to average activations measured on the dev set. Here we do not have a passive voice counterpart, so we manipulate active voice inputs to passive voice translations. The passive voice detection score (see §5.3) shows a monotonous increase (up to 0.6% more) as we modify more neurons (see figure 9). The trend matches our expectations. Moreover, we see again that manipulating top ranked neurons (rank given by ParaCorr) has a greater effect than bottom ranked ones. This is again consistent with what we saw with the development set and BLEU score in section 6.

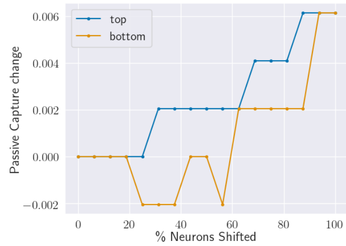
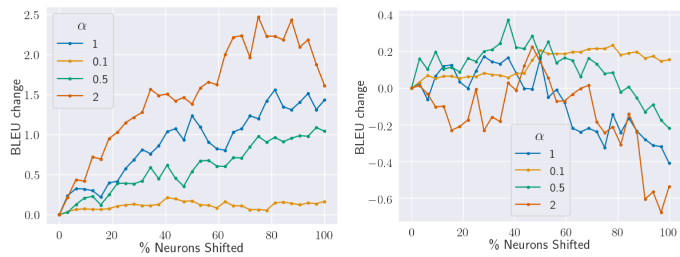


Figure 9: Manipulating neurons to get passive voice translation given an active voice input from the test set. Comparing the effect of manipulating first top versus bottom neurons, according to ParaCorr. We measure passive form detection



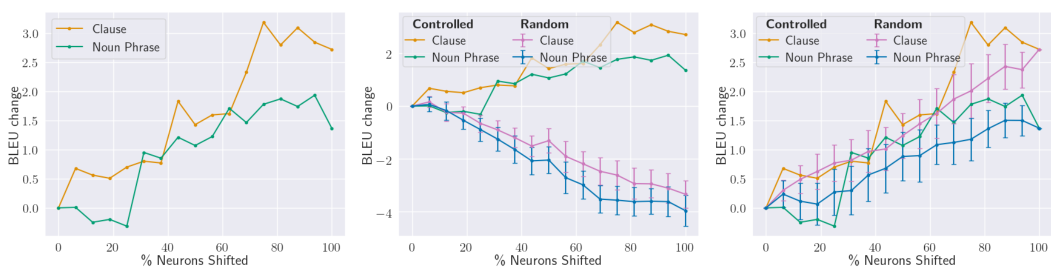
(a) Passive to Active

(b) Active to Passive

Figure 10: Comparing various magnitudes α for manipulation $\frac{\alpha}{\|y_{c_1}-y_{c_2}\|}(y_{c_1} - y_{c_2}) \in \mathbb{R}^m$. BLEU score measured against reference of target form, when manipulating increasingly more neurons according to top rank of ParaCorr.

C.3 Noun Phrase to Adverbial Clause

Manipulating from a noun phrase to an adverbial clause is consistent with the results we saw for passive to active manipulation, see Fig. 11 We repeat the same succession of experiments on the



(a) Baseline

(b) Random direction comparison

(c) Random selection comparison

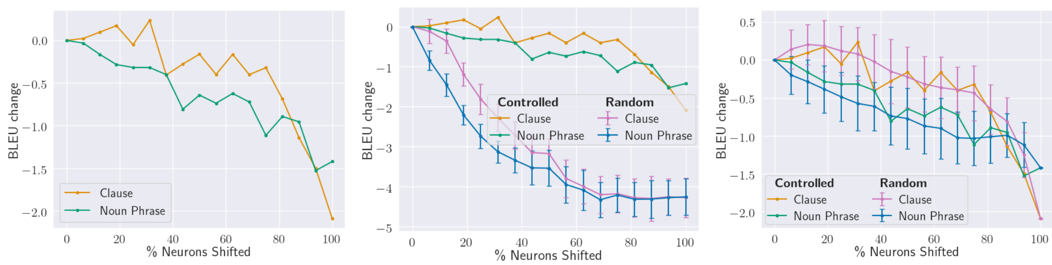
Figure 11: Manipulating output translation to be with an adverbial clause when feeding a sentence with a noun phrase as input. Lines present BLEU change with active and passive references according to the amount of neurons manipulated (x).

adverbial clause versus noun phrase dataset.

C.4 Adverbial Clause to Noun Phrase

Manipulating neurons to convert input with an adverbial clause to output translation with a noun phrase is not outright successful. In the controlled case (where we employ direction by our records of average activation of each paraphrase form and select an increasing set of neurons to manipulate according to top or bottom ParaCorr rank), we are still closer to the clause form than noun phrase. We propose several possible explanations:

1. The clause versus noun phrase dataset is substantially smaller than the active versus passive one (114 examples compared to 1,169 instances). A small dataset may include more noise or simply make the target syntactic form harder to capture.
2. Adverbial clause form may be more common in the train set so the model regularizes to the statistically more acceptable option. We see hints for that when we compare the manipulation towards active form as more successful than passive form (§5 and Fig. 8).
3. Noun phrase form may not be distinctive enough to be encoded in the model.
4. The target form may not be natural in the target language. As we discuss in our qualitative analysis in section 5.3, fail cases revealed instances where the target form was either not possible for a native German speaker, or required replacement of the verb to a synonym. This replacement demands another layer of manipulation from the model, one that it may not even know to generalize.



(a) Baseline (b) Random direction comparison (c) Random selection comparison

Figure 12: Manipulating output translation to be with a noun phrase when feeding a sentence with an adverbial clause as input. Lines present BLEU change with active and passive references according to the amount of neurons manipulated (x).

C.5 Manipulation Magnitude

The manipulation operation defined in 5 is normalized, then applied to chosen neurons. The reason is for different manipulations to be comparable in size. Another manipulation parameter to experiment with is a scalar α to re-scale with, i.e. manipulation from c_1 to c_2 is defined as subtraction of $\frac{\alpha}{\|y_{c_1}-y_{c_2}\|}(y_{c_1} - y_{c_2}) \in \mathbb{R}^m$. We experimented with a small grid search for alpha values without an apparent option being better than the baseline ($\alpha = 1$). See figure 10 for results[16]. There is no definitive conclusion of what magnitude would be consistently better in every manipulation. Similar trends were found in the clause dataset: $\alpha = 2$ was best when manipulating from paraphrased form noun phrase back to original form of adverbial clause, and worse in the other way around. This could be tied to the general effect we see in §C.4 that there is one direction of manipulation more effective, which is changing from paraphrased form to original form and should be further investigated in future work.

D Qualitative Analysis of Manipulation

Sentence examples of successful manipulation from passive voice input to active voice translation, as examined by a native German speaker, can be found at table 4.

As we discuss in §5.3, sometimes a manipulation is not applicable in the target language. For example, the adverbial clause sentence from our dataset *"In Lyman's case, she reported the alleged rape to military police less than an hour after it occurred."*, is translated into a noun phrase sentence regardless of input form (i.e. if we insert either this as input or its noun phrase paraphrase) or manipulation (i.e. with or without manipulation). "it occurred" is immediately translated into the

[16]We experimented with even greater values ($\alpha \in \{5, 10, 100, 1000\}$), each with a more drastic BLEU drop, therefore we discard their inclusion in the figure to allow the y-axis range to capture the subtle trends of the variables presented.

Table 4: Example of successfully manipulated sentences, from passive voice input to active voice translation. Manipulation is done by shifting the values of all the neurons in the encoder towards their average activation on active voice sentences. Correctness of sentence voice and fluency was verified by a native German speaker.

Input sentence: passive voice	Baseline translation: passive voice	Manipulated translation: active voice
The scene was described by police as very gruesome.	Der Tatort wurde von der Polizei als sehr grauenvoll beschrieben.	Die Polizei beschrieb den Tatort als sehr grauenvoll.
During the excavations, the remains of a total of five creatures were collected by them.	Bei den Ausgrabungen wurden von ihnen die Überreste von insgesamt fünf Lebewesen gefunden.	Bei den Ausgrabungen fanden sie die Überreste von insgesamt fünf Lebewesen.
From ”dream” to ”megalomania”: the Bit Galerie is discussed by TV readers	Vom ”Traum” zum ”Größenwahn”: Die Bit-Galerie wird von TV-Lesern diskutiert	Vom ”Traum” zum ”Größenwahn”: TV-Leser diskutieren über die Bit-Galerie

Table 5: Example of adverbial clause and noun phrase translations, showcasing the limitations of BLEU comparison to Google Translate references and the challenge of translating an output in adverbial clause form. Either manipulation here did not have any effect (e.g. manipulation from clausal input resulted in translation identical to the one without manipulation)

	Adverbial Clause	**Noun Phrase**
English	In Lyman’s case, she reported the alleged rape to military police less than an hour after it occurred.	In Lyman’s case, she reported the alleged rape to military police less than an hour after its occurrence.
Human Reference	In Lymans Fall meldete sie die mutmaŝsliche Vergewaltigung der Militärpolizei weniger als eine Stunde nach dem überfall.	
Google Translate	In Lymans Fall meldete sie die mutmaŝsliche Vergewaltigung weniger als eine Stunde nach ihrem Auftreten der Militärpolizei.	In Lymans Fall wurde die mutmaŝsliche Vergewaltigung von ihr weniger als eine Stunde nach ihrem Auftreten der Militärpolizei gemeldet.
Our Translation	In Lymans Fall meldete sie die angebliche Vergewaltigung weniger als eine Stunde nach dem Vorfall der Militärpolizei.	In Lymans Fall meldete sie die angebliche Vergewaltigung weniger als eine Stunde nach ihrem Auftreten der Militärpolizei.

German parallel of ”its occurrence” when translating the clause version, and it is translated into a wrong noun phrase when translating the noun phrase version (the German parallel of ”appearance” rather than ”occurrence” in this context , i.e. ”Auftreten” and ”Vorfall”, respectively). A native German speaker suggested we opt to replace ”occurred” with ”happened”, otherwise it could not be translated to a clause form. Even the human reference (of WMT) is with the ”its occurrence” noun phrase.

of of