On Neurons Invariant to Sentence Structural Changes in Neural Machine Translation

Gal Patel   **Leshem Choshen**   **Omri Abend**
School of Computer Science and Engineering
The Hebrew University of Jerusalem
`first.last@mail.huji.ac.il`

Abstract

We present a methodology that explores how sentence structure is reflected in neural representations of machine translation systems. We demonstrate our model-agnostic approach with the Transformer English-German translation model. We analyze neuron-level correlation of activations between paraphrases while discussing the methodology challenges and the need for confound analysis to isolate the effects of shallow cues. We find that similarity between activation patterns can be mostly accounted for by similarity in word choice and sentence length. Following that, we manipulate neuron activations to control the syntactic form of the output. We show this intervention to be somewhat successful, indicating that deep models capture sentence-structure distinctions, despite finding no such indication at the neuron level. To conduct our experiments, we develop a semi-automatic method to generate meaning-preserving minimal pair paraphrases (active-passive voice and adverbial clause-noun phrase) and compile a corpus of such pairs.[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 particular structural distinctions may explain how such structures are encoded, and validates the robustness of translation systems, which is a challenging but important problem (He et al., 2020; Freitag et al., 2020). Furthermore, understating how these encodings are used by the network may potentially enable controlling the output by direct manipulation of neurons.

Previous work analyzing what aspects of sentence structure are encoded in network representations mostly took a probing approach or focused on syntactic agreements. Works that compared activations did so either across models with identical input (Dalvi et al., 2019; Bau et al., 2019; Wu et al., 2020) or by representation words, not sentences (Antverg and Belinkov, 2021). Our novelty lies in using the same model with paraphrased input pairs, to analyze sentence structure encoding, without probing (c.f. §7). We derive inspiration from Computer Vision works that analyze model behavior under non-semantic changes to the input (Lenc and Vedaldi, 2015; Goodfellow et al., 2009).

For our proposed approach, we provide a dataset of minimal paraphrases (along with a code to extend it). We take two phenomena as test cases: active to passive voice and an adverbial clause to a noun phrase (see Table 1 and section §2).

To compare the activation patterns of sentences that may be comprised of a different number of tokens, we aggregate tokens representations. We then measure the correlation of neuron activations between paraphrases and provide a confound analysis. We find that the main contributors to strong correlation are similar positional encodings and bag-of-words overlap, suggesting strong correlation is derived from similar input encoding and not high-level abstractions learned by the model. The identification of these confounds may be beneficial to future work on network analysis (see §4).

Our findings suggest that these paraphrase distinctions are nonetheless encoded and used, as evident by our experiments of manipulating neurons (§5). We control the structure of the translation output (e.g., active/passive) by translating neuron activations in a fixed direction. We show that this manipulation generates outputs that are more similar to the desired form with an in-depth evaluation, using BLEU, dependency parsing, and manual analysis. We provide ablation studies that show that the results are not simply random artifacts of manipulation, with a non-local effect. Lastly, we compare different methods for selecting subsets of neurons

---

[1] The dataset is provided with paraphrase generation code: https://github.com/galpatel/minimal-paraphrases

	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***

Table 1: Examples produced by our paraphrasing engine

to be manipulated (§6) with counter-intuitive results, attributed to neurons of general importance and multiple roles per neuron.

Overall, the similarity between neuron activation over paraphrases is mostly explained by shallow input features: the positional and token embeddings. Therefore, some neurons represent input features, but sentence-level information is not localized, even in higher layers. Moreover, we show that sentence phrasing can be naïvely controlled, with a manipulation of a large number of neurons. This suggests that the distinction between different sentence structures is encoded in the model, probably in a distributed manner. Lastly, the neurons most effective for such manipulations are the ones most correlated across paraphrases, not necessarily those that vary the most.

2 Dataset: Minimal Paraphrase Pairs

We curate datasets that isolate specific sentence structure distinctions. To achieve that, 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, the same distinction is applied to all instances.
- **Reference Translation**, since we examine translation models.

Existing paraphrasing tools and datasets fail to satisfy these criteria (see §7). Therefore, we develop our own paraphrasing method, with which we 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 a Dependency Parsing and POS tags model (Honnibal et al., 2020) and a Semantic Role Labeling model (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 adverbial clause 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 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 and Verb Forms. See Appendix A for details and examples.[2]

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

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

We apply our paraphrasing engine to the WMT19 English-German development set (Barrault et al., 2019). Some results are disfluent. For example, the sentence *"He took his time"* is converted to *"His time was taken by him"*, which is syntactically well-formed, but anomalous. Therefore, we manually filter the data. For more details and other failed filtering approaches, see Appendix A.3. The number of pairs 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

---

[2]The dataset is provided with paraphrase generation code: https://github.com/galpatel/minimal-paraphrases

was trained on the WMT19 English-German train set (Barrault et al., 2019). The embedding dimension is 1024 with sinusoidal positional encoding.

Dataset. Minimal paraphrases (see §2). Due to space considerations, we present results on the active/passive set in the main paper. Clause/noun phrase results are in appendices §B.3 and §C.

Notations. We refer to trained models with a different random seed as m_1, m_2. We denote the set of source sentences $S = \{s_1, ..., s_n\}$, and its corresponding paraphrased set with $P = \{p_1, ..., p_n\}$ (e.g., s_i is an active voice sentence and p_i is its passive counterpart). We follow Liu et al. (2019); Wu et al. (2020) and take into account only the last sub-word token for each word (results with all sub-word tokens were similar). We consider as neurons the 1024 activation values in the output embedding of the 6 encoder layer blocks (following Wu et al., 2020)[3]. This leads to the following definition: the activation of some neuron l in model m, on sentence s_i is $x_S^{m,l}[i]$, while $x_S^{m,l}$ is a vector of size n (with n being the number of sentences in set S, i.e. one activation value per sentence, see §4.1).

4 Detecting Correlation Patterns

To detect activation patterns under experimental conditions, we measure Pearson correlation[4] between neural activations. While correlation analysis has been previously used to analyze neuron-level behavior (Bau et al., 2019; Dalvi et al., 2019; Wu et al., 2020; Meftah et al., 2021), our novelty lies with the independent variable being a property of the input (i.e., paraphrases) and not the model (e.g., architecture, initialization).

4.1 Sample Alignment Challenge

For every neuron, we measure its activation values while feeding a model with either the set of source sentences as input (e.g., active voice) or the paraphrased set (e.g., passive voice). Since the number of words may differ between paired sentences, so will the total number of tokens in these sentence sets. Consequently, the neural activation sample size will vary, which poses a challenge for testing correlation. Previous works did not face this difficulty, as they compared activation values given the same input corpus.

[3]Experiments on activations internal to each layer block have similar but weaker effects (Appendix B.2).

[4]Results with Spearman correlation were similar.

We overcome this with intra-sentence aggregation. Instead of having an activation value per word in the input corpus, we consider a single value per sentence, by pooling activations of words within a sentence. Ideally, this will allow for a sentence-level analysis of semantics and structure. Mean pooling was previously considered in several instances. Ethayarajh (2019) compared sentences by averaging their word vectors and Antverg and Belinkov (2021) aggregated words with specified attributes by averaging their representation, element-wise. The main results of this paper use mean pooling. In Appendix B.1 we also report results with min/max pooling, that show similar trends.

It is possible to consider other straightforward approaches, which we find less suitable. One option is a position-wise alignment of words (discarding the last words of the longer sentence). The difficulty here is that words of different semantics and syntactic roles are compared. For example, the third words of the active/passive sentence in Table 1 would be *the* and *was*, the former is a determiner of a direct object while the latter is an auxiliary of the root verb. Another option is functional correspondence alignment, where we measure correlation only between the functional tokens that indicate the structure change (e.g. *took* vs. *taken* and *arrived* vs. *arrival* from the examples in Table 1). That would result in an analysis based on similar single words with the same context words but in different syntactic forms. This could be problematic as it would capture a local syntactic change but not necessarily a sentence-level phrasing.

4.2 Baseline Experiments

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

$$ParaCorr(l, l') = \rho(x_S^{m_1,l}, x_P^{m_1,l'}) \quad (1)$$

ParaCorr should examine how neural networks represent differences in sentence structure (or similarities in semantics). We follow by comparing it to *ModelCorr* - the correlation between any pair of neurons across models, when given the same input:

$$ModelCorr(l, l') = \rho(x_S^{m_1,l}, x_S^{m_2,l'}) \quad (2)$$

ModelCorr is based on Bau et al. (2019) who detected generally important neurons in this way.

Figures 1a and 1b show ParaCorr and ModelCorr

correlation maps.[5] Some of ParaCorr's observed effect also appears in ModelCorr, suggesting the observed correlations 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 we had expected a different pattern as highly correlated neurons should be distributed differently for randomly initialized models.

4.3 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 present a minor change in sentence length: 2.0 ± 0.4 or 0.8 ± 0.8 token difference when paraphrasing active to passive or clause to noun phrase, respectively. The positional encodings are therefore similar. As for tokens, paraphrases have a large unigram overlap.

We define *PosCorr* as activation correlation between sentences with identical positional encoding but different token embeddings. Formally:

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

Where $\hat{S}$ is a set of random token sequences, uniformly sampled from the dictionary, with the same sequence lengths as S. PosCorr isolates the strong correlation effect observed both in ModelCorr and ParaCorr (Figure 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.2). The implication is that input representation, and not higher-level learned representation, is likely the cause of strong correlations.

TokenCorr accounts for token embeddings. We strip an input set S from its positional encoding,

denoted by $\tilde{S}$, and measure correlation:

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

TokenCorr (Figure 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 different across models). 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 different meaning than *"Josh likes Rose"*, although the sentences have the same bag of words. Even if word meaning is sufficient for a lot of cases, grammatical cues are still essential (Mahowald et al., 2022).

We further dissect the observed correlation for possible confounds. First, we compare activations of sentence pairs that share only 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 (Figure 1d, Eq. 4) is not explained by a similar structure cue (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., nothing is shared between the compared conditions. As little correlation is detected (between -0.27 to 0.31), we rule out the possibility of neurons with constant values.

Overall, our confound analysis implies the following: (1) strong activation correlation is greatly due to low-level components and not high-level learned knowledge, (2) strong correlation detected across paraphrases may not be exclusive to sentences with similar meaning but different structures, and (3) sentence structure is not localized to a specific set of neurons in our analysis.

5 Manipulation of Neurons

Manipulation of neurons allows us to control the output translation (without additional training) and adds a causative explanation to the role neurons play. We look into changing the activation values to force the output to have a desired syntactic structural feature (e.g., active or passive voice). Although we did not observe individual neurons that have a strong positive/negative correlation across paraphrases in §4, these sentence-structure distinc-

---

[5]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 (i.e. correlation across layers) or when the layers are higher.

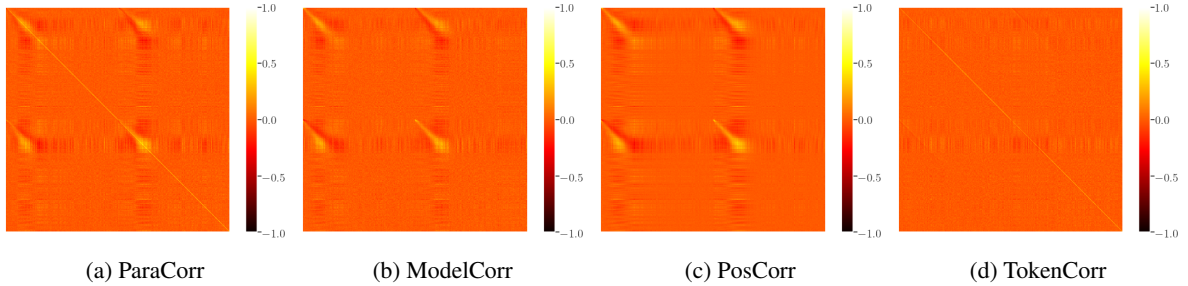| (a) ParaCorr | (b) ModelCorr | (c) PosCorr | (d) TokenCorr |

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

tions 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 the output structural properties by changing neuron values?
2. Does the exact activation value matter or only the identity of the modified neurons?
3. How to choose a set of neurons to manipulate?

5.1 Setup

We denote with $\bar{x}_c[i]$ the average activation of neuron i given a set of input sentences with property c, e.g., the property of passive voice. For a model with a total of n neurons (in the encoder), we have a vector of average behavior $\bar{x}_c \in R^n$. An intervention on neuron i with a current value of $x[i]$ from property c_1 towards property c_2 is a linear translation between their averages:

$$\hat{x}[i] = x[i] - \beta(\bar{x_{c_1}} - \bar{x_{c_2}})[i] \qquad (5)$$

So far, the formulation is based on previous work (Bau et al., 2019). Our preliminary experiments showed it is essential that the scaling factor β includes a normalization term because comparing the effects of different manipulations (i.e. different target properties c_2) can be confounded by activation magnitude. Therefore, $\beta = \frac{\alpha}{\|\bar{x_{c_1}} - \bar{x_{c_2}}\|}$. This leaves us with various parameters to experiment with: the properties c_1, c_2 we wish to manipulate, the subset of neurons we intervene with, and the scaling factor α. We explore the former two in §5.2 and the latter in Appendix C.5, having $\alpha = 1$ as default.

We evaluate whether manipulation increases the similarity of the output to a reference with the target form (c_2), relative to the similarity with the source form (c_1). We measure BLEU scores between our model's translation and Google translations, which (in the absence of manual references) we consider as references to both the source and target forms. This is a reasonable assumption given the performance gap between the models we use

and Google Translate. Later we discuss evaluation by additional methods to complement BLEU (see §5.3).

5.2 Experiments

We present experiments on manipulating passive voice inputs toward active voice translations. The reverse 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, first selecting the neurons most correlated across paraphrases (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 used it as an indicator of important neurons. We manipulate passive voice inputs toward active voice translations. Outputs become more similar to active voice than passive voice (Figure 2a), suggesting that sentence structure is indeed encoded in the model, even if we did not detect the distinction at the neuron-level in §4. 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, i.e. the value we shift towards. A random manipulation, with a random vector $y_r \in \mathbb{R}^n$, is defined by:

$$\hat{x}[i] = x[i] - \beta(\bar{x_{c_1}} - y_r)[i] \qquad (6)$$

Repeating 100 different random vectors (Figure 2b), we find it to be substantially worse. This implies that success is tied to the specific values we manipulate, not an artifact of any modification.

Selection of Manipulation. We test whether there is a preferable subset of neurons to manipu-

(a) Controlled baseline (b) Random direction comparison (c) Random selection comparison
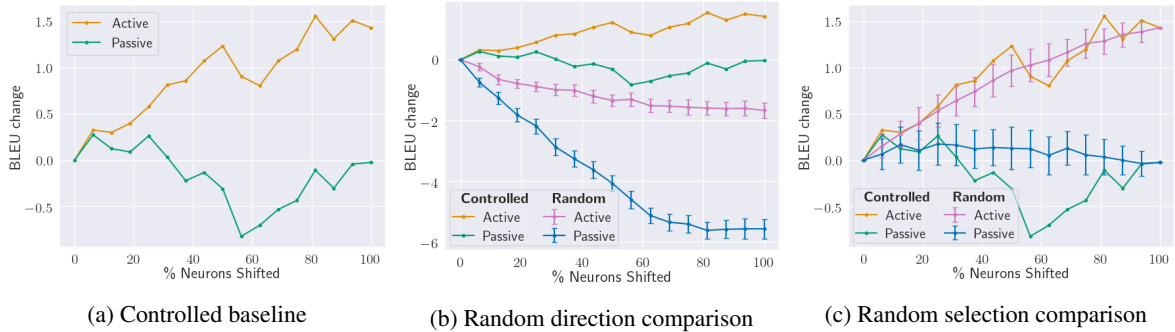
Figure 2: Manipulating the outputted translation to be of active voice when feeding passive voice input. Lines present BLEU change with active and passive references, as a function of the amount of neurons manipulated (x-axis). For random experiments (b) and (c) we report the average of the measured BLEU and its standard deviation.

late by randomly selecting a subset of neurons.[6] Results (Figure 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. Notwithstanding, we study subset choice even further in §6.

5.3 Beyond BLEU

BLEU score captures translation quality on the surface and not necessarily how good (or bad) it is at 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 a POS tagger to detect passive form[7]. The scorer is not intended to be perfect in capturing all passive instances[8], but it could serve as a complementary measure to indicate trends. We observe a decrease of detected passive voice when we manipulate the passive input towards active translation (see Figure 3b), solidifying the BLEU results.

Qualitative Analysis. A native German speaker examined a sample of output translations and found successful manipulations (see Appendix D). She discussed failed outputs – where the translation changed (i.e. unequal strings) but did not result in

the desired form. They did not degrade the translation. In some cases, sentences changed between stative passive and dynamic passive, rather than between passive and active (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 in an adverbial form in German, which demands them either to appear as a noun phrase or to be replaced with a synonym verb (an example is in Appendix D). These suggest that the manipulation has successfully modified the desired attributes in the sentence, even when not automatically detected as such. Moreover, it may be limited by the nature of the target language and the model's capabilities to generalize to synonyms while controlling the sentence structure.

Held-out Test Set. We repeated the manipulation experiment on a held-out test set: 552 active voice sentences from the WMT19 test set. This allows us to examine if the successful manipulation effect extends to a setting where the manipulated sentences do not contribute to the measure of average activation of the source form. As can be seen in C.2, the manipulation still results in the desired change in passive form detection.

Linearity Caveat. As Ravfogel et al. (2021) noted, positive results can indicate a causal effect, while negative results should be interpreted carefully since we have a linear manipulation in a non-linear setting. We leave the exploration of non-linear techniques for future work.

6 Specific Neuron Set Selection

Although finding a subset of neurons that carries a specific functionality is a difficult problem (Sajjad

---

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

[7]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.

[8]Limited recall: baseline translation of passive sentences (without manipulation) gets a score of 37.38%
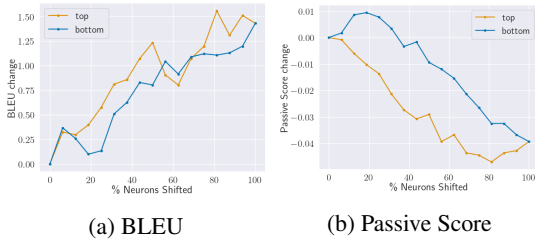
(a) BLEU   (b) Passive Score

Figure 3: Top ParaCorr neurons are better for manipulation. Manipulating the output translation to be in 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.
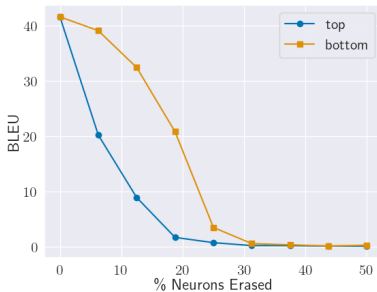

Figure 4: Top ranked neurons have a stronger impact on the translation quality of a test set, measured in BLEU. Erasure of neurons from the top or bottom of the rank given by the value of correlation between paraphrases.

et al., 2021), we look for subsets that are *relatively* better for manipulation. In our baseline (§5.2) we chose which neurons to manipulate according to the rank given by ParaCorr (i.e., sorting neurons by $ParaCorr(l, l)$, high to low). Under an intuitive interpretation, neurons that positively correlate when a systematic change is made to the input are those invariant to that change. Neurons with a negative correlation are specific to the change in sentence structure. Following these, we expect bottom-ranked neurons to be better for manipulation than top-ranked neurons. Contrarily, we observe the opposite phenomenon (Figure 3). The following tests may explain it.

Model Performance. Top-ranked neurons are important for overall performance. We follow Bau et al. (2019) who identified important neurons by deleting them (i.e., setting activations to zero) and examining the impact on the model performance. We delete an increasing amount of neurons, according to ParaCorr rank. We measure BLEU on a

held-out set of 552 active voice sentences, and their references, extracted from the WMT19 test set. Results (Figure 4) show that top-ranked neurons have a stronger impact on the translation quality than bottom-ranked do, suggesting that ParaCorr partially ranks neurons by their general importance.

Role Overlap. Some of the top ParaCorr neurons account for lexical identity and positional information. This fact explains why they have the most impact when manipulating sentence structure. Word order is essential for active-passive paraphrasing, where the subject and direct object exchange places. Notably, when we tested non-paired active-passive sentences the phenomena did not repeat itself, see Appendix C.6. Word tokens are the building blocks for the semantic meaning of a sentence (which is the same across paraphrases), even when a bag-of-word is not exclusive to a single 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 in the 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.
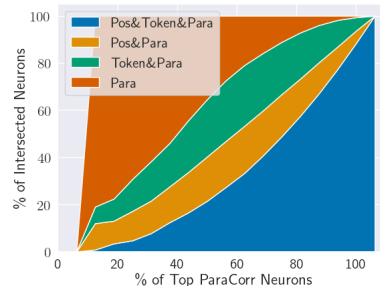

Figure 5: Top ParaCorr neurons intersect with neurons most related to token embeddings and positional encoding. The x-axis represents the amount of top ParaCorr neurons as a percentage of all the neurons in the encoder. The y-axis shows how many of these x top ParaCorr neurons are also in the set of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

7 Related Work

Understanding NLP Neural Networks. Various approaches were previously proposed (Belinkov

and 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 what is used, or how (Antverg and Belinkov, 2021). Others employ mediation analysis theory: Vig et al. (2020); Finlayson et al. (2021) study semantic behavior and syntactic agreement, respectively. Some works analyze attention heads (Voita et al., 2019) or follow attention flow (Abnar and Zuidema, 2020). Visualization tools interpret activations, but with some exceptions (e.g., Lenc and Vedaldi, 2015), they are mostly limited to qualitative examples. Durrani et al. (2020) interpreted individual neurons with probing-like methods for fine-grained analysis. Challenge sets (Choshen and Abend, 2019; Warstadt et al., 2020) and adversarial examples (Alzantot et al., 2018), expose challenging cases by analyzing the NMT system's behavior, rather than representation. Elazar et al. (2021) analyzes semantically equivalent inputs by clustering their embeddings. They improve prediction by continual training, while we manipulate translation post-training.

Interpretation in other domains. Some Computer Vision work inspired our approach. Lenc and Vedaldi (2015) study the interaction between input transformation and its representation through the layers, while Goodfellow et al. (2009) examine invariant neurons, those that are selective to high-level features but robust given semantically identical transformations. Their methodologies do not fit the NLP domain since they rely on a mathematically well-defined input transformation (e.g., rotation). We propose an alternative with our paraphrases in §2. Linguistic encoding in the human brain has been studied in neuroscience works. Friederici (2011) analyzed the correlation of neuroimaging where subjects are presented with sentences with subtle syntactic variations or violations, and found that well-correlated regions are considered to process syntax. Fedorenko et al. (2016) presented human subjects with various inputs, which are analogous to our correlation experiments: word lists (TokenCorr), meaningless grammatical sentences (PosCorr), non-words lists (combination of TokenCorr and PosCorr), and regular sentences (ParaCorr).

Individual neurons analysis with correlation. Bau et al. (2019) detected neurons that correlate 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), with their linguistic role identified by probing with a GMM classifier. The technique to identify neurons that behave 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 architectures. Meftah et al. (2021) adapted correlations to quantify the impact of fine-tuning by measuring activations of neurons before and after domain adaptation.

Controlling active-passive voice in translation. Yamagishi et al. (2016) controlled voice (active/passive) in RNN-based machine translation, from Japanese to English, when an indicator was given as input. Their method required additional model training, unlike ours.

Paraphrases. Existing paraphrasing tools vary by how localized their edits 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 subsentences (e.g., Dornescu et al., 2014; Lee and Don, 2017). Other than paraphrasing tools, 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 and 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

With our curated dataset, we introduced a model-agnostic methodology to detect activation patterns across paraphrases. By a meticulous confound analysis, we found that activation similarity is likely due to shallow features of sequence length or word identity, which 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 ex-

perimental setup. We controlled syntactic structures of generated output, which provides evidence of the ability of models to capture them. While we found the modification technique to be important for manipulation success, the selection of a subset of neurons was more challenging. Future work should test additional architectures and language pairs or examine the representation significance of our paraphrase pairs in other NLP tasks.

Limitations

Our work has some limitations. First, we compile the minimal paraphrases dataset to analyze representations of an isolated difference. Even so, human language is complex and any input transformation could not be mathematically well-defined. Our paraphrases may have other differences than the ones we indicate, which may introduce noise to our analysis. For example, we don't know the effect specific verbs (e.g., more common/rare ones) have when they appear in noun form, or what possible bias 'by' introduces when we add it for passive voice.

Secondly, we demonstrate our model-agnostic methodology in a specific setting with a transformer model for en-de translation. Our insights of what is captured (or isn't) may change when experimenting on other architectures or language pairs.

We aggregate token representations to sentence representation, discussing our choice and other possible approaches in §4.1. We do this to overcome potential differences in the number of tokens the paraphrases contain. However, the aggregation may lose encoded information along the way. We find some evidence of that when examining other pooling techniques (min/max) in Appendix B.1.

Our manipulation shows that many neurons have to be modified for a successful outcome (at least 50%). Still, when manipulating more and more neurons, the effect is not always monotone in every setting (see Figure 9, Figure 12 and Figure 13). We conducted our qualitative analysis on the output with all of the encoder neurons modified and see positive results, with a discussion of how to choose neurons relatively better (§6).

In the following Appendix sections we also address: the filtering required for the automatically generated paraphrases (§A.3), analysis on neurons internal to the layer block (§B.2), possible alternative explanation for the observed results for top vs.

bottom ranked neurons (§C.6), unsuccessful manipulation cases (§C.4) and manipulation magnitude (§C.5).

Acknowledgements

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.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2890–2896, Brussels, Belgium. Association for Computational Linguistics.

Omer Antverg and Yonatan Belinkov. 2021. On the pitfalls of analyzing individual neurons in language models. *CoRR*, abs/2110.07483.

Loïc Barrault, Ondřej Bojar, Marta R Costa-Jussa, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, et al. 2019. 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)*, pages 1–61.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2019. Identifying and controlling important neurons in neural machine translation. In *International Conference on Learning Representations*.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Rahul Bhagat, Eduard Hovy, and Siddharth Patwardhan. 2009. Acquiring paraphrases from text corpora. In *Proceedings of the Fifth International Conference on Knowledge Capture*, K-CAP '09, page 161–168, New York, NY, USA. Association for Computing Machinery.

Leshem Choshen and Omri Abend. 2019. Automatically extracting challenge sets for non-local phenomena in neural machine translation. In *Proceedings*

of the 23rd Conference on Computational Natural Language Learning (CoNLL), pages 291–303, Hong Kong, China. Association for Computational Linguistics.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, Anthony Bau, and James Glass. 2019. 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, pages 6309–6317.

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.

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

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In Proceedings of the Third International Workshop on Paraphrasing (IWP2005).

Iustin Dornescu, Richard Evans, and Constantin Orăsan. 2014. 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), pages 1–10, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Nadir Durrani, Hassan Sajjad, Fahim Dalvi, and Yonatan Belinkov. 2020. Analyzing individual neurons in pre-trained language models. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4865–4880, Online. Association for Computational Linguistics.

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

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. 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 55–65, Hong Kong, China. Association for Computational Linguistics.

Evelina Fedorenko, Terri L. Scott, Peter Brunner, William G. Coon, Brianna Pritchett, Gerwin Schalk, and Nancy Kanwisher. 2016. Neural correlate of the construction of sentence meaning. Proceedings of the National Academy of Sciences, 113(41):E6256–E6262.

Matthew Finlayson, Aaron Mueller, Sebastian Gehrmann, Stuart Shieber, Tal Linzen, and Yonatan Belinkov. 2021. Causal analysis of syntactic agreement mechanisms in neural language models. 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 1828–1843, Online. Association for Computational Linguistics.

Markus Freitag, George Foster, David Grangier, and Colin Cherry. 2020. Human-paraphrased references improve neural machine translation. In Proceedings of the Fifth Conference on Machine Translation, pages 1183–1192, Online. Association for Computational Linguistics.

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

Juri Ganitkevitch. 2013. Large-scale paraphrasing for natural language understanding. In Proceedings of the 2013 NAACL HLT Student Research Workshop, pages 62–68, Atlanta, Georgia. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In Proceedings of Workshop for NLP Open Source Software (NLP-OSS), pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

Ian Goodfellow, Honglak Lee, Quoc Le, Andrew Saxe, and Andrew Ng. 2009. Measuring invariances in deep networks. In Advances in Neural Information Processing Systems, volume 22, pages 646–654. Curran Associates, Inc.

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

Pinjia He, Clara Meister, and Zhendong Su. 2020. Structure-invariant testing for machine translation. In Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering, ICSE '20, page 961–973, New York, NY, USA. Association for Computing Machinery.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spacy: Industrial-strength natural language processing in python.

J. Edward Hu, Rachel Rudinger, Matt Post, and Benjamin Van Durme. 2019. 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.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Katharina Kann, Sascha Rothe, and Katja Filippova. 2018. Sentence-level fluency evaluation: References help, but can be spared! In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 313–323, Brussels, Belgium. Association for Computational Linguistics.

John Lee and J. Buddhika K. Pathirage Don. 2017. Splitting complex English sentences. In *Proceedings of the 15th International Conference on Parsing Technologies*, pages 50–55, Pisa, Italy. Association for Computational Linguistics.

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

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. 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)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Kyle Mahowald, Evgeniia Diachek, Edward Gibson, Evelina Fedorenko, and Richard Futrell. 2022. Grammatical cues are largely, but not completely, redundant with word meanings in natural language. *CoRR*, abs/2201.12911.

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

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 NAACL-HLT 2019: Demonstrations*.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. 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)*, pages 425–430, Beijing, China. Association for Computational Linguistics.

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

Shauli Ravfogel, Grusha Prasad, Tal Linzen, and Yoav Goldberg. 2021. Counterfactual interventions reveal the causal effect of relative clause representations on agreement prediction. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 194–209, Online. Association for Computational Linguistics.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. 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)*, pages 856–865, Melbourne, Australia. Association for Computational Linguistics.

Hassan Sajjad, Nadir Durrani, and Fahim Dalvi. 2021. Neuron-level interpretation of deep nlp models: A survey. *ArXiv*, abs/2108.13138.

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

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, 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.

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

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. 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*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

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. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

John Wu, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2020. Similarity analysis of contextual word representation models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4638–4655, Online. Association for Computational Linguistics.

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

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: paraphrase adversaries from word scrambling. *CoRR*, 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-based 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 $y \in \mathbb{R}^d$, 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. $w_k = \max_{i=1,...m} y[w_i]$ 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 [9]. If we don't find it there, we choose between Nomlex form (Macleod et al., 1998) and present participle form according to Verb Form Dictionaries[10] (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'.

---

[9] https://amr.isi.edu/download.html
[10] https://github.com/monolithpl/verb.forms.dictionary

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 those where 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[11]).
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.
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.

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. [12]

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.

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.

A.3 Filtering Results

As mentioned in §2, some sentences generated by our paraphrasing process are disfluent. Therefore,

---

[12]The flow of purpose clause conversion could arguably lack optional determiner addition before the new noun phrase. It could be easily added to the generation code for any future use.

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 an adverbial clause sentence to a noun phrase.

we manually filtered the data. 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.

B   Detecting Correlation Patterns

B.1   Pooling Techniques

In section 4 we measure the correlation of activations over paraphrases. Since paraphrases vary in their sequence length, the subsequent random variables representing the activations for each sentence structure (i.e. activations over active voice versus activations over passive voice), vary as well. While previous works (Bau et al., 2019; Dalvi et al., 2019; Wu et al., 2020; Meftah et al., 2021) compared activations over all input words, our settings necessitate pooling. In §4 we presented the results where we used mean activation per sentence. In Figure 6 we compare the heatmaps of different poolings. As is evident, the major confounds (diagonals and concentrated neuron groups of strong correlation) are present across the techniques.

B.2   Inside 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 7. 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.
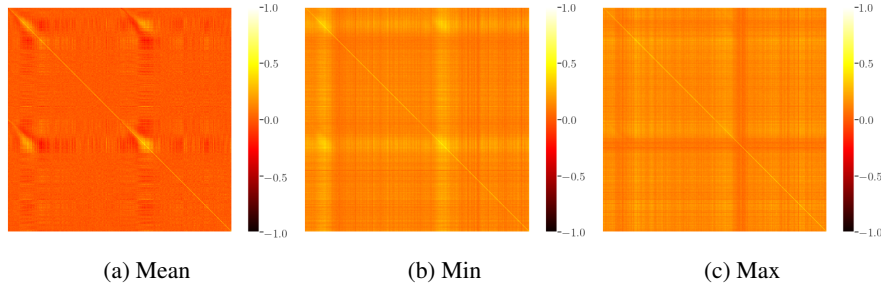
| (a) Mean | (b) Min | (c) Max |

Figure 6: Activation correlation between paraphrases (ParaCorr), using the active-passive dataset. The correlation is done sentence-wise, while the pooling technique of token activation varies. The major confounds are present across all techniques.
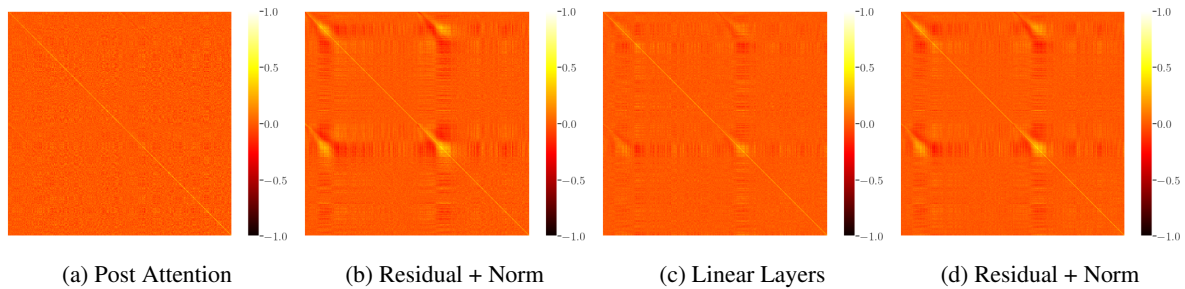

| (a) Post Attention | (b) Residual + Norm | (c) Linear Layers | (d) Residual + Norm |

Figure 7: 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.

B.3 Adverbial Clause versus Noun Phrase

Here we present the same correlation methods detailed in section 4 but measured on the adverbial clause versus noun phrase sets. See Figure 8.

C Manipulation of Neurons

C.1 Active to Passive

To complete all variations of the manipulation experiment, we first showcase the shift from active voice input to passive voice translation (the opposite direction of what we showed in the main 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 manipulation, and again getting maximal change requires many neurons to be modified (at least 50%), see Figure 9a. With the random experiments of direction (Figure 9b) and neurons selection (Figure 9c), we get similar results - our controlled direction is better while choosing an optimal 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 automatically detect as active voice sentences 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 another (e.g. active voice), one might argue that we can't know the effect of 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 10). 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.
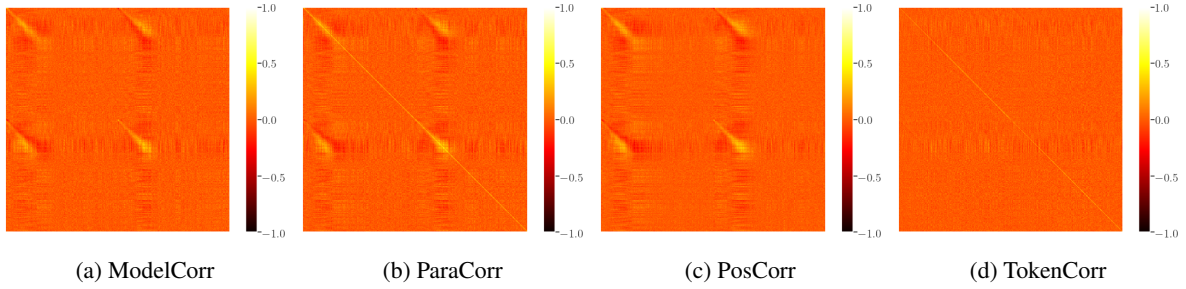
(a) ModelCorr   (b) ParaCorr   (c) PosCorr   (d) TokenCorr

Figure 8: Activation correlation of first layer neurons in the Transformer encoder, using the clause/noun-phrase dataset.

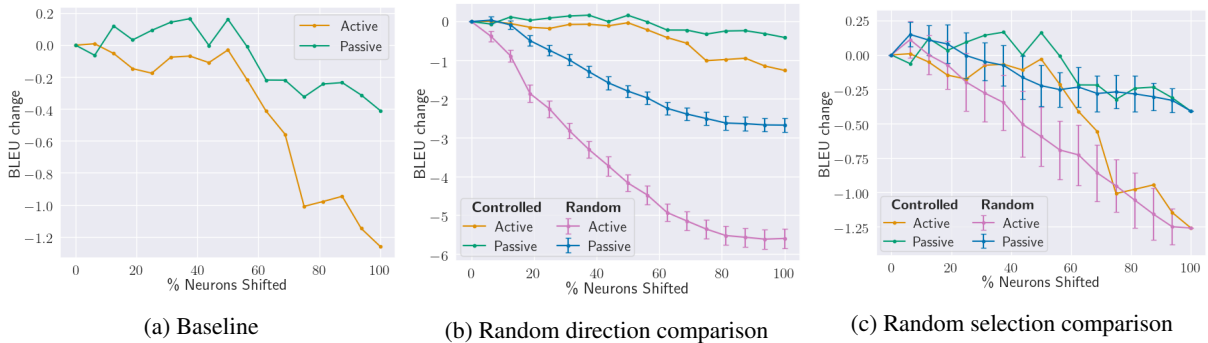(a) Baseline   (b) Random direction comparison   (c) Random selection comparison

Figure 9: Manipulating output translation to be in 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).
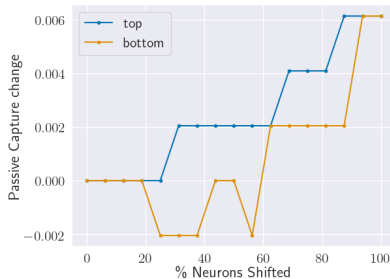

Figure 10: Manipulating neurons to get passive voice translation given an active voice input from the test set. Comparing the effect of manipulating first top versus bottom neurons, according to ParaCorr. We measure passive form detection

(a) Passive to Active   (b) Active to Passive

Figure 11: Comparing various magnitudes α for manipulation step $\frac{\alpha}{\|x_{c_1}^- - x_{c_2}^-\|}(x_{c_1}^- - x_{c_2}^-)$. 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 Figure 12 We repeat the same succession of experiments on the 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 (see Figure 13).
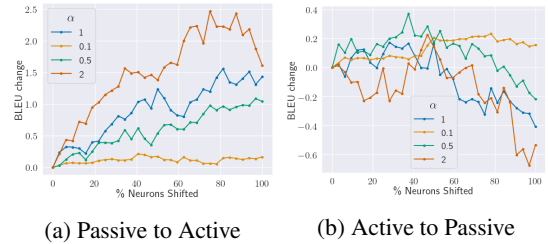
In the controlled case (where we employ direction by our records of average activation of each paraphrase form and select an increasing set of neurons to manipulate according to the top or bottom ParaCorr rank), we are still closer to the clause form than the noun phrase. We propose several possible 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

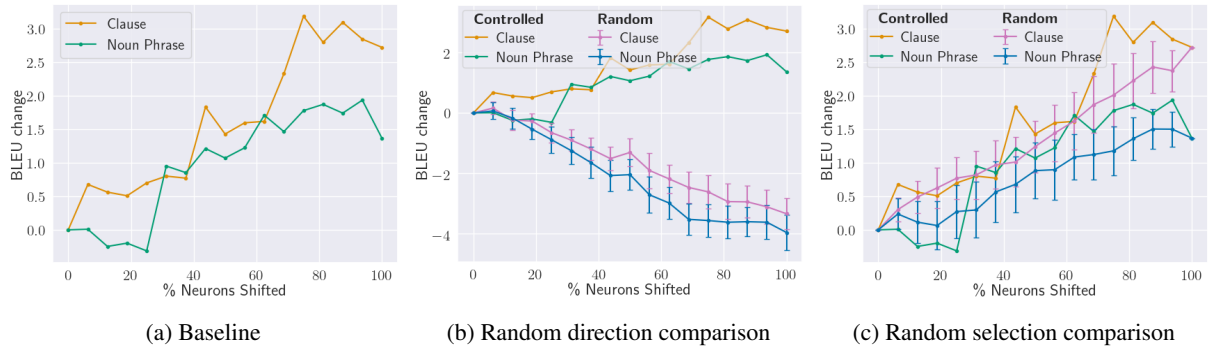| (a) Baseline | (b) Random direction comparison | (c) Random selection comparison |

Figure 12: 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).

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 Figure 9).

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 a replacement of the verb to a synonym. This replacement demands another level of manipulation from the model, one that it may not even know to generalize.

C.5 Manipulation Magnitude

As defined in section 5, manipulation from sentence feature c_1 to c_2 is a subtraction of the term $\frac{\alpha}{\|\bar{x_{c_1}} - \bar{x_{c_2}}\|}(\bar{x_{c_1}} - \bar{x_{c_2}})$ (applied to chosen neurons). We experimented with a small grid search for possible values for the scaling factor α, without an apparent option being better than the baseline ($\alpha = 1$). See Figure 11 for results[13]. 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

---

[13]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.

when manipulating from paraphrased form noun phrase back to the original form of the adverbial clause, and worse the other way around. This could be tied to the general effect we discuss in §C.4 where one direction of manipulation is more effective: changing from paraphrased form to original form. This should be further investigated in future work.

C.6 Unparalleled Sentences Manipulation

As seen in §6, top ParaCorr neurons were better for manipulation than those from the bottom of the rank. One possible explanation we introduced was the fact that many of those top ParaCorr neurons are also top PosCorr and TokenCorr neurons. Therefore, the effectiveness might be derived from the role polysemy of these neurons, especially when the paraphrasing calls for a change of word order (e.g. active-passive requires subject and object swap) or token identity (e.g. clause to noun phrase requires a transformation between verb and noun). This is true for cases where the paraphrases are parallel pairs, therefore they share those shallow features (tokens and word order).

Here we present an experiment of manipulation where there are no parallel pairs of paraphrases. We randomly split the sentence pairs into two sets. From one we take only the active sentences, and from the other we take only the passive sentences, resulting in an active set and passive set of unrelated sentences. We repeat the manipulation experiment as detailed in §5.2 but use those unparalleled sets for measuring the averaging activation of neurons under the active voice feature and under the passive voice feature (i.e., for measuring $\bar{x_{c_1}}$ and $\bar{x_{c_2}}$). We do so for 100 different splits of the data. Measuring the mean and standard deviation of the

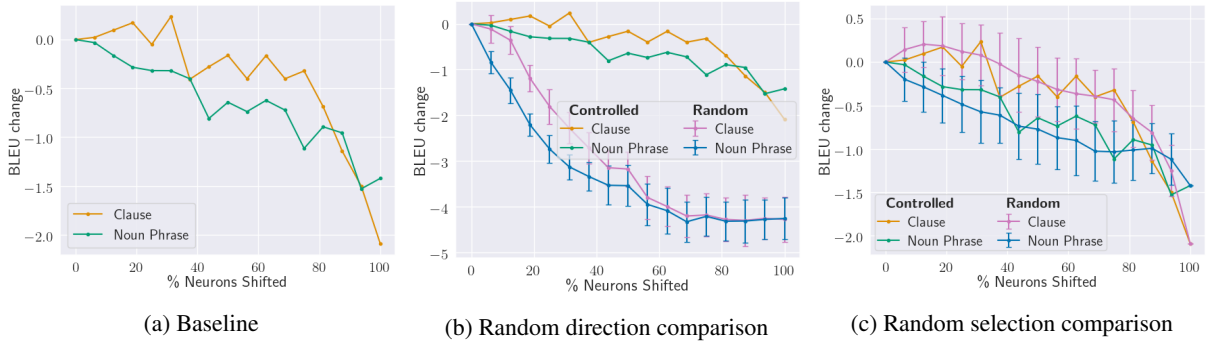(a) Baseline (b) Random direction comparison (c) Random selection comparison

Figure 13: 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).

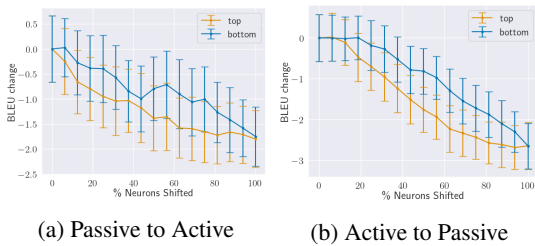

(a) Passive to Active (b) Active to Passive

Figure 14: Comparing the impact of manipulating top ParaCorr neurons versus bottom ones, where the modification value of neurons is determined by average activation under unparalleled sets of sentences, i.e. $\bar{x}_{c_1}$ and $\bar{x}_{c_2}$ are measured by active and passive non-pairs. The lines represent the mean and standard deviation over 100 different unparalleled sets.

BLEU against the objective reference, the results are presented in Figure 14. Notably, the standard deviation of the experiment is reported, not the standard error of the mean. The results may match the intuition where the least correlated neurons between paraphrases are those most sensitive to the active-passive feature, but since nothing is shared across those sentences, the expected noise level is high, and any measure is hard to explain.

When we measured the correlation of such unparalleled sentences, we got an average correlation (per neuron over 100 different splits of the dataset into unparalleled sets) ranging from -0.04 to 0.04, with a standard deviation between 0.03 to 0.06.

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 in 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 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 into a clause form. Even the human reference (of WMT) is with the "its occurrence" noun phrase. See Table 5.

211

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 toward 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 an adverbial clause and a 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 mutmaßliche Vergewaltigung der Militärpolizei weniger als eine Stunde nach dem überfall.	
Google Translate	In Lymans Fall meldete sie die mutmaßliche Vergewaltigung weniger als eine Stunde nach ihrem Auftreten der Militärpolizei.	In Lymans Fall wurde die mutmaßliche 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.