IDENTIFYING AND CONTROLLING IMPORTANT NEURONS IN NEURAL MACHINE TRANSLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural machine translation (NMT) models learn representations containing substantial linguistic information. However, it is not clear if such information is fully distributed or if some of it can be attributed to individual neurons. We develop unsupervised methods for discovering important neurons in NMT models. Our methods rely on the intuition that different models learn similar properties, and do not require any costly external supervision. We show experimentally that translation quality depends on the discovered neurons, and find that many of them capture common linguistic phenomena. Finally, we show how to control NMT translations in predictable ways, by modifying activations of individual neurons.

1 INTRODUCTION

Neural machine translation (NMT) systems achieve state-of-the-art results by learning from large amounts of example translations, typically without additional linguistic information. Recent studies have shown that representations learned by NMT models contain a non-trivial amount of linguistic information on multiple levels: morphological (Belinkov et al., 2017), syntactic (Shi et al., 2016b), and semantic (Hill et al., 2017). These studies use trained NMT models to generate feature representations for words, and use these representations to predict certain linguistic properties. This approach has two main limitations. First, it targets the whole vector representation and fails to analyze individual dimensions in the vector space. In contrast, previous work found meaningful individual neurons in computer vision (Zeiler & Fergus, 2014; Zhou et al., 2016; Bau et al., 2017, among others) and in a few NLP tasks (Karpathy et al., 2015; Radford et al., 2017; Qian et al., 2016a). Second, these methods require external supervision in the form of linguistic annotations. They are therefore limited by available annotated data and tools.

In this work, we make initial progress towards addressing these limitations by developing unsupervised methods for analyzing the contribution of *individual neurons* to NMT models. We aim to answer the following questions:

- How important are individual neurons for obtaining high-quality translations?
- Do individual neurons in NMT models contain interpretable linguistic information?
- Can we control MT output by intervening in the representation at the individual neuron level?

To answer these questions, we develop several unsupervised methods for ranking neurons according to their importance to an NMT model. Inspired by work in machine vision (Li et al., 2016b), we hypothesize that different NMT models learn similar properties, and therefore similar important neurons should emerge in different models. To test this hypothesis, we map neurons between pairs of trained NMT models using several methods: correlation analysis, regression analysis, and SVCCA, a recent method combining singular vectors and canonical correlation analysis (Raghu et al., 2017). Our mappings yield lists of candidate neurons containing shared information across models. We then evaluate whether these neurons carry important information to the NMT model by masking their activations during testing. We find that highly-shared neurons impact translation quality much more than unshared neurons, affirming our hypothesis that *shared information matters*.

Given the list of important neurons, we then investigate what linguistic properties they capture, both qualitatively by visualizing neuron activations and quantitatively by performing supervised classification experiments. We were able to identify neurons corresponding to several linguistic phenomena, including morphological and syntactic properties.

Finally, we test whether intervening in the representation at the individual neuron level can help *control the translation*. We demonstrate the ability to control NMT translations on three linguistic properties—tense, number, and gender—to varying degrees of success. This sets the ground for controlling NMT in desirable ways, potentially reducing system bias to properties like gender.

Our work indicates that not all information is distributed in NMT models, and that many human-interpretable grammatical and structural properties are captured by individual neurons. Moreover, modifying the activations of individual neurons allows controlling the translation output according to specified linguistic properties. The methods we develop here are task-independent and can be used for analyzing neural networks in other tasks. More broadly, our work contributes to the localist/distributed debate in neural cognitive science (Gayler & Levy, 2011) by investigating the important case of neural machine translation.

2 Related Work

Much recent work has been concerned with analyzing neural representations of linguistic units, such as word embeddings (Köhn, 2015; Qian et al., 2016b), sentence embeddings (Adi et al., 2016; Ganesh et al., 2017; Brunner et al., 2018), and NMT representations at different linguistic levels: morphological (Belinkov et al., 2017), syntactic (Shi et al., 2016b), and semantic (Hill et al., 2017). These studies follow a common methodology of evaluating learned representations on external supervision by training classifiers or measuring other kinds of correlations. Thus they are limited to the available supervised annotation. In addition, these studies also do not typically consider individual dimensions. In contrast, we propose intrinsic unsupervised methods for detecting important neurons based on correlations between independently trained models. A similar approach was used to analyze vision networks (Li et al., 2016b), but to the best of our knowledge these ideas were not used to study NMT or other NLP models before.

In computer vision, individual neurons were shown to capture meaningful information Zeiler & Fergus (2014); Zhou et al. (2016); Bau et al. (2017). Even though some doubts were cast on the importance of individual units (Morcos et al., 2018), recent work stressed their contribution to predicting specific object classes via masking experiments similar to ours (Zhou et al., 2018). A few studies analyzed individual neurons in NLP. For instance, neural language models learn specific neurons that activate on brackets (Karpathy et al., 2015), sentiment (Radford et al., 2017), and length (Qian et al., 2016a). Length-specific neurons were also found in NMT (Shi et al., 2016a), but generally not much work has been devoted to analyzing individual neurons in NMT. We aim to address this gap.

3 Methodology

Much recent work on analyzing NMT relies on supervised learning, where NMT representations are used as features for predicting linguistic annotations (see Section 2). However, such annotations may not be available, or constrain the analysis to a particular scheme.

Instead, we propose to use different kinds of correlations between neurons from different models as a measure of their importance. Suppose we have M such models and let $\mathbf{h}_t^m[i]$ denote the activation of the i-th neuron in the encoder of the m-th model for the t-th word.[1] These may be models from different training epochs, trained with different random initializations or datasets, or even different architectures—all realistic scenarios that researchers often experiment with. Let x_i^m denote a random variable corresponding to the i-th neuron in the m-th model. x_i^m maps words to their neuron activations: $\mathrm{x}_i^m : t \mapsto \mathbf{h}_t^m[i]$. Similarly, let $\mathbf{x}^m$ denote a random vector corresponding to the activations of all neurons in the m-th model: $\mathbf{x}^m : t \mapsto \boldsymbol{h}_t^m$.

We consider four methods for *ranking* neurons, based on correlations between pairs of models. Our hypothesis is that different NMT models learn similar properties, and therefore similar important neurons emerge in different models, akin to neural vision models (Li et al., 2016b). Our methods capture different levels of localization/distributivity, as described next. See Figure 1 for illustration.

[1] We only consider neurons from the top layer, although the approach can also be applied to other layers.

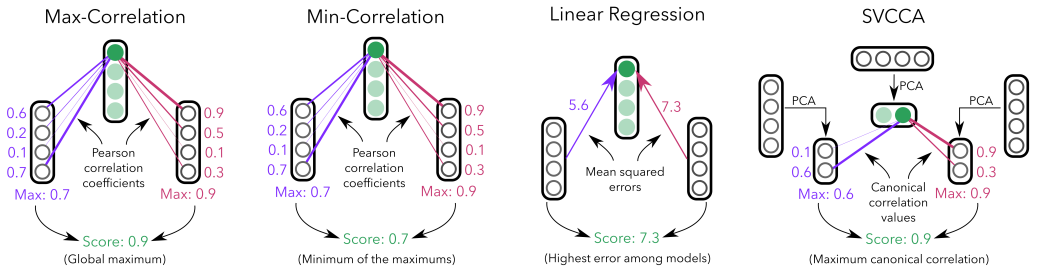


Figure 1: An illustration of the correlation methods methods, showing how to compute the score for one neuron using each of the methods. Here the number of models is $M = 3$.

3.1 Unsupervised correlation Methods

Maximum correlation The maximum correlation (`MaxCorr`) of neuron x_i^m looks for the highest correlation with any neuron in all other models:

$$\texttt{MaxCorr}(\mathrm{x}_i^m) = \max_{j, m' \neq m} |\rho(\mathrm{x}_i^m, \mathrm{x}_j^{m'})| \tag{1}$$

where $\rho(\mathrm{x}, \mathrm{y})$ is the Pearson correlation coefficient between x and y. We then rank the neurons in model m according to their `MaxCorr` score. We repeat this procedure for every model m. This score looks for neurons that capture properties that emerge strongly in two separate models.

Minimum correlation The minimum correlation (`MinCorr`) of neuron x_i^m looks for the neurons most correlated with X_i^m in each of the other models, but selects the one with the lowest correlation:

$$\texttt{MinCorr}(\mathrm{x}_i^m) = \min_{m' \neq m} \max_{j} |\rho(\mathrm{x}_i^m, \mathrm{x}_j^{m'})| \tag{2}$$

Neurons in model m are ranked according to their `MinCorr` score. This tries to find neurons that are well correlated with many other models, even if they are not the overall most correlated ones.

Regression ranking We perform linear regression (`LinReg`) from the full representation of another model $\mathbf{x}^{m'}$ to the neuron x_i^m. Then we rank neurons by the regression mean squared error. This attempts to find neurons whose information might be distributed in other models.

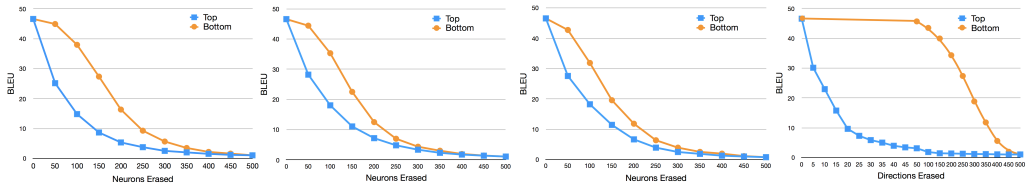
SVCCA Singular vector canonical correlation analysis (`SVCCA`) is a recent method for analyzing neural networks (Raghu et al., 2017). In our implementation, we perform PCA on each model's representations $\mathbf{x}^m$ and take enough dimensions to account for 99% of the variance. For each pair of models, we obtain the canonically correlated basis, and rank the basis directions by their CCA coefficients. This attempts to capture information that may be distributed in less dimensions than the whole representation. In this case we get a ranking of directions, rather than individual neurons.

3.2 Verifying Detected Neurons

We want to verify that neurons ranked highly by the unsupervised methods are indeed important for the NMT models. We consider quantitative and qualitative techniques for verifying their importance.

Erasing Neurons We test importance of neurons by erasing some of them during translation. Erasure is a useful technique for analyzing neural networks (Li et al., 2016a). Given a ranked list of neurons π, where $\pi(i)$ is the rank of neuron x_i, we zero-out increasingly more neurons according to the ranking π, starting from either the top or the bottom of the list. Our hypothesis is that erasing neurons from the top would hurt translation performance more than erasing from the bottom.

Concretely, we first run the entire encoder as usual, then zero out specific neurons from all source hidden states $\{\mathbf{h}_1, \ldots, \mathbf{h}_n\}$ before running the decoder. For `MaxCorr`, `MinCorr`, and `LinReg`, we zero out individual neurons. To erase k directions found by `SVCCA`, we instead project the embedding $\boldsymbol{E}$ (corresponding to all activations of a given model over a dataset) onto the space spanned by the non-erased directions: $\boldsymbol{E}' = \boldsymbol{E}(\boldsymbol{C}(\boldsymbol{C}^T\boldsymbol{C})^{-1}\boldsymbol{C}^T)$, where $\boldsymbol{C}$ is the CCA projection matrix with the first or last k columns removed. This corresponds to erasing from the top or bottom.



(a) MaxCorr (b) MinCorr (c) LinReg (d) SVCCA

Figure 2: Erasing neurons (or SVCCA directions) from the top and bottom of the list of most important neurons (directions) ranked by different unsupervised methods, in an English-Spanish model.

Supervised Verification While our focus is on unsupervised methods for finding important neurons, we also utilize supervision to verify our results. Since training a supervised classifier on every neuron is costly, we instead report simple metrics that can be easily computed. Specifically, we sometimes report the expected conditional variance of neuron activations conditioned on some property. In other cases we found it useful to estimate a Gaussian mixture model (GMM) for predicting a label and measure its prediction quality. We obtain linguistic annotations with Spacy: spacy.io.

Visualization Interpretability of machine learning models remains elusive (Lipton, 2016), but visualizing can be an instructive technique. Similar to previous work analyzing neural networks in NLP (Elman, 1991; Karpathy et al., 2015; Kádár et al., 2016), we visualize activations of neurons and observe interpretable behavior. We will illustrate this with example heatmaps below.

4 Experimental Setup

Data We use the United Nations (UN) parallel corpus (Ziemski et al., 2016) for all experiments. We train models from English to 5 languages: Arabic, Chinese, French, Russian, and Spanish, as well as an English-English auto-encoder. For each target language, we train 3 models on different parts of the training set, each with 500K sentences. In total, we have 18 models. This setting allows us to compare models trained on the same language pairs but different training data, as well as models trained on different language pairs. We evaluate on the official test set.

MT training We train 500 dimensional 2-layer LSTM encoder-decoder models with attention Bahdanau et al. (2014). In order to study both word and sub-word properties, we use a word representation based on a character convolutional neural network (charCNN) as input to both encoder and decoder, which was shown to learn morphology in language modeling and NMT (Kim et al., 2015; Belinkov et al., 2017).[2] While we focus here on recurrent NMT, our approach can be applied to other models like the Transformer (Vaswani et al., 2017), which we leave for future work.

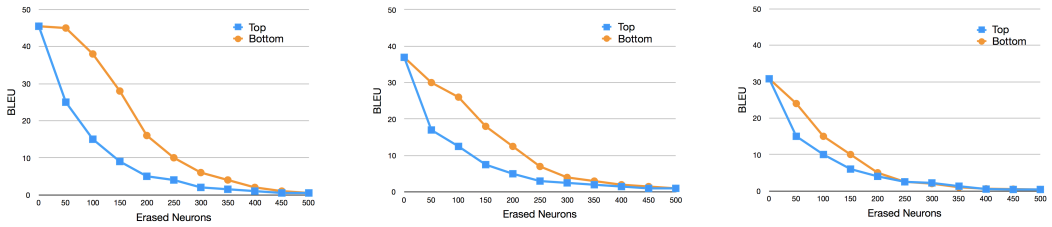
5 Results

5.1 Erasure Experiments

Figure 2 shows erasure results using the methods from Section 3.1, on an English-Spanish model. For all four methods, erasing from the top hurts performance much more than erasing from the bottom. This confirms our hypothesis that neurons ranked higher by our methods have a larger impact on translation quality. Comparing erasure with different rankings, we find similar patterns with MaxCorr, MinCorr, and LinReg: erasing the top ranked 10% (50 neurons) degrades BLEU by 15-20 points, while erasing the bottom 10% neurons only hurts by 2-3 points. In contrast, erasing SVCCA directions results in rapid degradation – 15 BLEU point drop when erasing 1% (5) of the top directions, and poor performance when erasing 10% (50). This indicates that top SVCCA directions capture very important information in the model. We analyze these top neurons and directions in the next section, finding that top SVCCA directions focus mostly on identifying specific words.

Figure 3 shows the results of MaxCorr when erasing neurons from top and bottom, using models trained on three language pairs. In all cases, erasing from the top hurts performance more than erasing from the bottom. We found similar trends with other language pairs and ranking methods.

[2] We used this representation rather than BPE sub-word units (Sennrich et al., 2016) to facilitate interpretability with respect to specific words. In the experiments, we report word-based results unless noted otherwise.



(a) English-Spanish (b) English-French (c) English-Chinese

Figure 3: Erasing neurons from the top or bottom of the `MaxCorr` ranking in three language pairs.

Table 1: Top 10 neurons (or `SVCCA` directions) in an English-Spanish model according to the four methods, and the percentage of explained variance by conditioning on position or token identity.

`MaxCorr`			`MinCorr`			`LinReg`			`SVCCA`	
ID	Pos	Tok	ID	Pos	Tok	ID	Pos	Tok	Pos	Tok
464	**92%**	10%	342	**88%**	7.9%	464	**92%**	10%	**86%**	26%
342	**88%**	7.9%	464	**92%**	10%	260	0.71%	**94%**	1.6%	**90%**
260	0.71%	**94%**	260	0.71%	**94%**	139	0.86%	**93%**	7.5%	**85%**
49	11%	6.1%	383	**67%**	6.5%	494	3.5%	**96%**	20%	**79%**
124	**77%**	48%	250	**63%**	6.8%	342	**88%**	7.9%	1.1%	**89%**
394	0.38%	22%	124	**77%**	47%	228	0.38%	**96%**	10%	**76%**
228	0.38%	**96%**	485	**64%**	10%	317	1.5%	**83%**	30%	**57%**
133	0.14%	**87%**	480	**70%**	12%	367	0.44%	**89%**	24%	**55%**
221	1%	30%	154	**63%**	15%	106	0.25%	**92%**	23%	**60%**
90	0.49%	28%	139	0.86%	**93%**	383	**67%**	6.5%	18%	**63%**

5.2 Evaluating Top Neurons

What kind of information is captured by the neurons ranked highly by each of our ranking methods? Previous work found specific neurons in NMT that capture position of words in the sentence (Shi et al., 2016a). Do our methods capture similar properties? Indeed, we found that many of the top neurons capture position. For instance, Table 1 shows the top 10 ranked neurons from an English-Spanish model according to each of the methods. The table shows the percent of variance in neuron activation that is eliminated by conditioning on position in the sentence, calculated over the test set. Similarly, it shows the percent of explained variance by conditioning on the current token identity.

We observe an interesting difference between the ranking methods. `LinReg` and especially `SVCCA`, which are both computed by using multiple neurons, tend to find information determined by the identity of the current token. `MaxCorr` and (especially) `MinCorr` tend to find position information. This suggests that information about the current token is often distributed in multiple neurons, which can be explained by the fact that tokens carry multiple kinds of linguistic information. In contrast, position is a fairly simple property that the NMT encoder can represent in a small number of neurons.

5.3 Linguistically Interpretable Neurons

Neurons that activate on specific tokens or capture position in the sentence are important, as shown in the previous section. But they are less interesting from the perspective of capturing language information. In this section, we investigate several linguistic properties by measuring predictive capacity and visualizing neuron activations. The supplementary material discusses more properties.

Parentheses Table 2 shows top neurons from each model for predicting that tokens are inside/outside of parentheses, quotes, or brackets, estimated by a GMM model. Often, the parentheses neuron is unique (low scores for the 2nd best neuron), suggesting that this property tends to be relatively localized. Generally, neurons that detect parentheses were ranked highly in most models by the `MaxCorr` method, indicating that they capture important patterns in multiple networks.

The next figure visualizes the most predictive neuron in an English-Spanish model. It activates positively (red) inside parentheses and negatively (blue) outside. Similar neurons were found in RNN language models (Karpathy et al., 2015). Next we consider more complicated linguistic properties.

`Private International Law ( " Hague Conference " ) requested the`

Table 2: F_1 scores of the top two neurons from each network for detecting tokens inside parentheses, and the ranks of the top neuron according to our intrinsic unsupervised methods.

Neuron	1st	2nd	Max	Min	Reg	Neuron	1st	2nd	Max	Min	Reg
en-es-1:232	0.59	0.3	14	44	26	en-ar-3:331	0.59	0.35	17	92	49
en-es-2:208	0.72	0.26	8	43	21	en-ru-1:259	0.64	0.33	10	47	44
en-es-3:47	0.57	0.29	11	34	23	en-ru-2:23	0.71	0.26	10	72	31
en-fr-1:499	0.6	0.27	37	41	14	en-ru-3:214	0.65	0.32	25	67	114
en-fr-2:361	0.61	0.35	28	44	60	en-zh-1:49	0.58	0.44	5	85	63
en-fr-3:253	0.37	0.35	140	122	68	en-zh-2:159	0.76	0.38	5	47	37
en-ar-1:383	0.38	0.36	119	195	228	en-zh-3:467	0.54	0.32	5	59	47
en-ar-2:166	0.63	0.25	4	117	67						

Table 3: Strongest correlations in all models relative to a tense neuron in an English-Arabic model.

Arabic	0.66, 0.57	French	-0.69, -0.58, -0.48	Chinese	-0.51, -0.30, -0.18
Spanish	0.56, 0.36, 0.22	Russian	-0.50, -0.39, -0.29	English	-0.33, -0.19, -0.03

Tense We annotated the test data for verb tense (with Spacy) and trained a GMM model to predict tense from neuron activations. The following figure shows activations of a top-scoring neuron (0.56 F_1) from the English-Arabic model on the first 5 test sentences. It tends to activate positively (red color) on present tense ("recognizes", "recalls", "commemorate") and negatively (blue color) on past tense ("published", "disbursed", "held"). These results are obtained with a charCNN representation, which is sensitive to common suffixes like "-ed", "-es". However, this neuron also detects irregular past tense verbs like "held", suggesting that it captures context in addition to sub-word information. The neuron also makes some mistakes by activating weakly positively on nouns ending with "s" ("videos", "punishments"), presumably because it gets confused with the 3rd person present tense.

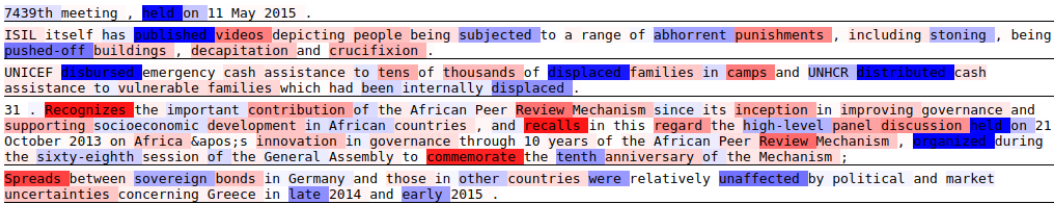
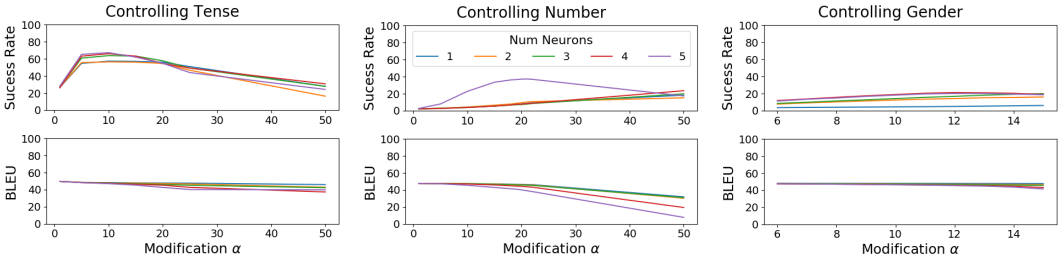


Table 3 shows correlations of neurons most correlated with this tense neuron, according to `MaxCorr`. All these neurons are highly predictive of tense: all are in the top 5 and 9 out of 15 (non-auto-encoder) neurons have the highest F_1 score for predicting tense. The auto-encoder English models are an exception, exhibiting much lower correlations with the English-Arabic tense neuron. This suggests that tense emerges in a "real" NMT model, but not in an auto-encoder that only learns to copy. Interestingly, English-Chinese models have somewhat lower correlated neurons with the tense neuron, possibly due to the lack of explicit tense marking in Chinese. The encoder does not need to pay as much attention to tense when generating representations for the decoder.

Other Properties We found many more linguistic properties by visualizing top neurons ranked by our methods, especially with `MaxCorr`. We briefly mention some of these here and provide more details and quantitative results in the appendix. We found neurons that activate on numbers, dates, adjectives, plural nouns, auxiliary verbs, and more. We also investigated noun phrase segmentation, a compositional property above the word level, and found high-scoring neurons (60-80% accuracy) in every network. Many of these neurons were ranked highly by the `MaxCorr` method. In contrast, other methods did not rank such neurons very highly. See Table 5 in the appendix for the full results.

Some neurons have quite complicated behavior. For example, when visualizing neurons highly ranked by `MaxCorr` we found a neuron that activates on numbers in the beginning of a sentence, but not elsewhere (see Figure 7 in the appendix). It would be difficult to conceive of a supervised prediction task which would capture this behavior a-priori, without knowing what to look for. Our supervised methods are flexible enough to find any neurons deemed important by the NMT model, without constraining the analysis to properties for which we have supervised annotations.



(a) Tense (b) Number (c) Gender

Figure 4: Success rates and BLEU scores for controlling NMT by modifying neuron activations.

6 Controlling Translations

In this section, we explore a potential benefit of finding important neurons with linguistically meaningful properties: controlling the translation output. This may be important for mitigating biases in neural networks. For instance, gender stereotypes are often reflected in automatic translations, as the following motivating examples from Google Translate demonstrate.[3]

(1) a. o bir doctor
b. he is a doctor

(2) a. o bir hemşire
b. she is a nurse

The Turkish sentences (1a, 2a) have no gender information—they can refer to either male or female. But the MT system is biased to think that doctors are usually men and nurses are usually women, so its generated translations (1b, 2b) represent these biases.

We conjecture that if a given neuron matters to the model, then we can control the translation in predictable ways by modifying its activations. To do this, we first encode the source sentence as usual. Before decoding, we set the activation of a particular neuron in the encoder state to a value α, which is a function of the mean activations over a particular property (defined below). To evaluate our ability to control the translation, we design the following protocol:

1. Tag the source and target sentences in the development set with a desired property, such as gender (masculine/feminine). We use Spacy for these tags.
2. Obtain word alignments for the development set with using an alignment model trained on 2 million sentences of the UN data. We use `fast_align` (Dyer et al., 2013) with default settings.
3. For every neuron in the encoder, predict the target property on the word aligned to its source word activations using a supervised GMM model.[4]
4. For every word having a desired property, modify the source activations of the top k neurons found in step 3, and generate a modified translation. The modification value is defined as $\alpha = \mu_1 + \beta(\mu_1 - \mu_2)$, where μ_1 and μ_2 are mean activations of the property we modify from and to, respectively (e.g. modifying gender from masculine to feminine), and β is a hyper-parameter.
5. Tag the output translation and word-align it to the source. Declare *success* if the source word was aligned to a target word with the desired property value (e.g. feminine).

6.1 Results

Figure 4 shows translation control results in an English-Spanish model. We report success rate—the percentage of cases where the word was aligned to a target word with the desired property–and the effect on BLEU scores, when varying α. Our tense control results are the most successful, with up to 67% success rate for changing past-to-present. Modifications generally degrade BLEU, but the loss at the best success rate is not large (2 BLEU points). We provide more tense results in Appendix 1.2.

Controlling other properties seems more difficult, with the best success rate for controlling number at 37%, using the 5 top number neurons. Gender is the most difficult to control, with a 21% success rate using the 5 top neurons. Modifying even more neurons did not help. We conjecture that these properties are more distributed than tense, which makes controlling them more difficult. Future work can explore more sophisticated methods for controlling multiple neurons simultaneously.

[3]For more biased examples, see `mashable.com/2017/11/30/google-translate-sexism`.

[4]This is different from our results in the previous section, where we predicted a source-side property, because here we seek neurons that are predictive of target-side properties to facilitate controlling the translation.

Table 4: Examples for controlling translation by modifying activations of different neurons on the *italicized* source words. α = modification value (–, no modification).

(a) Controlling number when translating "The interested *parties*" to Spanish.

α	Translation	Num	α	Translation	Num
-1	abiertas particulares	pl.	-0.25, -0.125, 0	La parte interesada	sing.
-0.5	Observaciones interesadas	pl.	0.25	Cuestion interesada	sing.
-0.25, -0.125, 0	Las partes interesadas	pl.	0.5, 1	Gran útil	sing.

(b) Controlling gender when translating "The interested *parties*" (left) and "*Questions* relating to information" (right) to Spanish.

α	Translation	Gen	α	Translation	Gen
-0.5, -0.25	Los partidos interados	ms.	-1	Temas relativos a la información	ms.
0, 0.25	Las partes interesadas	fm.	-0.5, 0, 0.5	Cuestiones relativas a la información	fm.

(c) Controlling tense when translating "The committee *supported* the efforts of the authorities".

	α	Translation	Tense
Arabic	–/+10	وأيدت\وتؤيد اللجنة {جهود\الجهود التي تبذلها} السلطات	past/present
French	–/-20	Le Comité a appuyé/appuie les efforts des autorités	past/present
Spanish	–/-3/0	El Comité apoyó/apoyaba/apoya los esfuerzos de las autoridades	past/impf./present
Russian	–/-1	Комитет поддержал/поддерживает усилия властей	past/present
Chinese	–/-50	委员会 支持 当局 的 努力 / 委员会 正在 支持 当局 的 努力	untensed/present

6.2 Example translations

We provide examples of controlling translation of number, gender, and tense. While these are cherry-picked examples, they illustrate that the controlling procedure can work in multiple properties and languages. Appendix B discusses these examples and language-specific behaviors in more detail.

Number Table 4a shows translation control results for a number neuron from an English-Spanish model, which activates negatively/positively on plural/singular nouns. The translation changes from plural to singular as we increase the modification α. We notice that using too high α values yields nonsense translations, but with correct number: transitioning from the plural adjective *particulares* ("particular") to the singular adjective *útil* ("useful"), with valid translations in between.

Gender Table 4b shows examples of controlling gender translation for a gender neuron from the same model, which activates negatively/positively on masculine/feminine nouns. The translations change from masculine to feminine synonyms as we increase the modification α. Generally, we found it difficult to control gender, as also suggested by the relatively low success rate.

Tense Table 4c shows examples of controlling tense when translating from English to five target languages. In all language pairs, we are able to change the translation from past to present by modifying the activation of the tense neurons from the previous section (Table 3). In Spanish, we find a transition from past to imperfect to present. Interestingly, in Chinese, we had to use a fairly large α value (in absolute terms), consistent with the fact that tense is not usually marked in Chinese.

7 Conclusion

We developed unsupervised methods for finding important neurons in NMT, and evaluated how these neurons impact translation quality. We analyzed several linguistic properties that are captured by individual neurons using quantitative prediction tasks and qualitative visualizations. We also designed a protocol for controlling translations by modifying neurons that capture desired properties.

Our analysis can be extended to other NMT components (e.g. the decoder) and architectures Gehring et al. (2017); Vaswani et al. (2017), as well as other tasks. We believe that more work should be done to analyze the spectrum of localized vs. distributed information in neural language representations. We would also like to develop more sophisticated ways to control translation output, for example by modifying representations in variational NMT architectures (Zhang et al., 2016; Su et al., 2018).

REFERENCES

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. Fine-grained Analysis of Sentence Embeddings Using Auxiliary Prediction Tasks. *arXiv preprint arXiv:1608.04207*, 2016.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Computer Vision and Pattern Recognition*, 2017.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. What do Neural Machine Translation Models Learn about Morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 861–872. Association for Computational Linguistics, 2017. doi: 10.18653/v1/P17-1080. URL http://www.aclweb.org/anthology/P17-1080.

Gino Brunner, Yuyi Wang, Roger Wattenhofer, and Michael Weigelt. Natural Language Multitasking: Analyzing and Improving Syntactic Saliency of Hidden Representations. *arXiv preprint arXiv:1801.06024*, 2018.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. A Simple, Fast, and Effective Reparameterization of IBM Model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 644–648. Association for Computational Linguistics, 2013. URL http://www.aclweb.org/anthology/N13-1073.

Jeffrey L. Elman. Distributed representations, simple recurrent networks, and grammatical structure. *Machine learning*, 7(2-3):195–225, 1991.

J. Ganesh, Manish Gupta, and Vasudeva Varma. Interpretation of Semantic Tweet Representations. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, ASONAM '17, pp. 95–102, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4993-2. doi: 10.1145/3110025.3110083. URL http://doi.acm.org/10.1145/3110025.3110083.

Ross W. Gayler and Simon D. Levy. Compositional connectionism in cognitive science ii: the localist/distributed dimension. *Connection Science*, 23(2):85–89, 2011. doi: 10.1080/09540091.2011.587505. URL https://doi.org/10.1080/09540091.2011.587505.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional Sequence to Sequence Learning. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1243–1252, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL http://proceedings.mlr.press/v70/gehring17a.html.

Felix Hill, Kyunghyun Cho, Sébastien Jean, and Yoshua Bengio. The representational geometry of word meanings acquired by neural machine translation models. *Machine Translation*, 31(1):3–18, Jun 2017. ISSN 1573-0573. doi: 10.1007/s10590-017-9194-2. URL https://doi.org/10.1007/s10590-017-9194-2.

Ákos Kádár, Grzegorz Chrupała, and Afra Alishahi. Representation of linguistic form and function in recurrent neural networks. *arXiv preprint arXiv:1602.08952*, 2016.

Andrej Karpathy, Justin Johnson, and Li Fei-Fei. Visualizing and understanding recurrent networks. *arXiv preprint arXiv:1506.02078*, 2015.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware Neural Language Models. *arXiv preprint arXiv:1508.06615*, 2015.

Arne Köhn. What's in an Embedding? Analyzing Word Embeddings through Multilingual Evaluation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 2067–2073, Lisbon, Portugal, September 2015. Association for Computational Linguistics. URL http://aclweb.org/anthology/D15-1246.

Jiwei Li, Will Monroe, and Dan Jurafsky. Understanding Neural Networks through Representation Erasure. *arXiv preprint arXiv:1612.08220*, 2016a.

Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John Hopcroft. Convergent Learning: Do different neural networks learn the same representations? In *International Conference for Learning Representations (ICLR)*, 2016b.

Zachary C Lipton. The Mythos of Model Interpretability. In *ICML Workshop on Human Interpretability in Machine Learning (WHI)*, 2016.

Ari S. Morcos, David G.T. Barrett, Neil C. Rabinowitz, and Matthew Botvinick. On the importance of single directions for generalization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=r1iuQjxCZ.

Peng Qian, Xipeng Qiu, and Xuanjing Huang. Analyzing Linguistic Knowledge in Sequential Model of Sentence. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 826–835, Austin, Texas, November 2016a. Association for Computational Linguistics. URL https://aclweb.org/anthology/D16-1079.

Peng Qian, Xipeng Qiu, and Xuanjing Huang. Investigating Language Universal and Specific Properties in Word Embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1478–1488, Berlin, Germany, August 2016b. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P16-1140.

Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. Learning to generate reviews and discovering sentiment. *arXiv preprint arXiv:1704.01444*, 2017.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. SVCCA: Singular Vector Canonical Correlation Analysis for Deep Learning Dynamics and Interpretability. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 6078–6087. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7188-svcca-singular-vector-canonical-correlation-analysis-for-deep-learning-dynamics-and-interpretability.pdf.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725. Association for Computational Linguistics, 2016. doi: 10.18653/v1/P16-1162. URL http://www.aclweb.org/anthology/P16-1162.

Xing Shi, Kevin Knight, and Deniz Yuret. Why Neural Translations are the Right Length. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2278–2282. Association for Computational Linguistics, 2016a. doi: 10.18653/v1/D16-1248. URL http://www.aclweb.org/anthology/D16-1248.

Xing Shi, Inkit Padhi, and Kevin Knight. Does String-Based Neural MT Learn Source Syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1526–1534, Austin, Texas, November 2016b. Association for Computational Linguistics. URL https://aclweb.org/anthology/D16-1159.

Jinsong Su, Shan Wu, Deyi Xiong, Yaojie Lu, Xianpei Han, and Biao Zhang. Variational Recurrent Neural Machine Translation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, 2018.

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 30*, pp. 5998–6008. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

Biao Zhang, Deyi Xiong, jinsong su, Hong Duan, and Min Zhang. Variational Neural Machine Translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 521–530. Association for Computational Linguistics, 2016. doi: 10.18653/v1/D16-1050. URL http://www.aclweb.org/anthology/D16-1050.

B. Zhou, A. Khosla, Lapedriza. A., A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. *CVPR*, 2016.

Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Revisiting the Importance of Individual Units in CNNs via Ablation. *arXiv preprint arXiv:1806.02891*, 2018.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. The United Nations Parallel Corpus v1.0. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis (eds.), *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may 2016. European Language Resources Association (ELRA). ISBN 978-2-9517408-9-1.

A ADDITIONAL RESULTS AND VISUALIZATIONS

1.1 NOUN PHRASE SEGMENTATION

Table 5 shows the top neurons from each network by accuracy when classifying interior, exterior, or beginning of a noun phrase. The annotations were obtained with Spacy. We found high-scoring neurons (60-80% accuracy) in every network. Many of these neurons were ranked highly by the `MaxCorr` ranking methods. In contrast, other correlation methods did not rank such neurons very highly. Thus there is correspondence between a high rank by our intrinsic unsupervised measure `MaxCorr` and the neuron's capacity to predict external annotation.

Table 5: Top neuron from each network by accuracy for classifying interior, exterior, or beginning of a noun phrase, as well as ranking of these neurons by our intrinsic unsupervised measures.

		Rank		
Neuron	Accuracy	`MaxCorr`	`MinCorr`	`LinReg`
en-es-1:221	0.79	8	54	57
en-es-2:158	0.77	11	59	65
en-es-3:281	0.73	24	39	111
en-fr-1:111	0.77	12	62	122
en-fr-2:85	0.73	32	45	86
en-fr-3:481	0.76	13	65	133
en-ar-1:492	0.69	32	59	161
en-ar-2:190	0.75	48	80	90
en-ar-3:288	0.69	25	84	157
en-ru-1:38	0.66	35	54	158
en-ru-2:130	0.67	34	64	134
en-ru-3:78	0.67	159	106	123
en-zh-1:427	0.64	22	75	240
en-zh-2:199	0.65	187	216	232
en-zh-3:28	0.68	63	32	42

1.2 CONTROLLING TRANSLATIONS

We provide additional translation control results. Table 6 shows the tense results using the best modification value from Figure 4a. We report the number of times the source word was aligned to a target word which is past or present, or to multiple words that include both or neither of these tenses. The success rate is the percentage of cases where the word was aligned to a target word with the desired tense. By modifying the activation of only one neuron (the most predictive one), we were able to change the translation from past to present in 67% of the times and vice-versa in 49% of the times. In many other cases, the tense was erased, that is, the modified source word was not aligned to any tensed word, which is a partial success.

Table 6: Results for controlling tense.

From \ To	Past	Present	Both	Neither	Success Rate
Past	85	820	9	311	67%
Present	1586	256	30	1363	49%

1.3 VISUALIZATIONS

Here we provide additional visualizations of neurons capturing different linguistic properties.

Noun phrases We visualize the top scoring neuron (79%) from an English-Spanish model in Figure 5. Notice how the neuron activates positively (red color) on the first word in the noun phrases, but negatively (blue color) on the rest of the noun phrase (e.g. "Regional" in "Regional Service Centre").

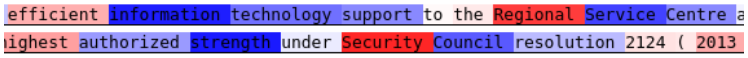
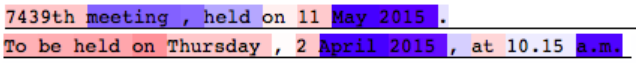
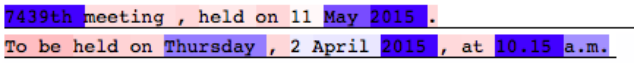


Figure 5: Visualization of a neuron from an English-Spanish model that activates positively (red color) on the first word in the noun phrase and negatively (blue) on the following words.

Dates and Numbers Figure 6 shows activations of neurons capturing dates and numbers. These neurons were ranked highly (top 30) by `MaxCorr` when ranking an English-Arabic model trained with charCNN representations. We note that access to character information leads to many neurons capturing sub-word information such as years (4-digit numbers). The first neuron is especially sensitive to month names ("May", "April"). The second neuron is an approximate year-detector: it is sensitive to years ("2015") as well as other tokens with four digits ("7439th", "10.15").



(a) Month neuron



(b) Approximate "year" neuron

Figure 6: Neurons capturing dates and numbers.

List items Figure 7 shows an interesting case of a neuron that is sensitive to the appearance of two properties simultaneously: position in the beginning of the sentence and number format. Notice that it activates strongly (negatively) on numbers when they open a sentence but not in the middle of the sentence. Conversely, it does not activate strongly on non-number words that open a sentence. This neuron aims to capture patterns of opening list items.

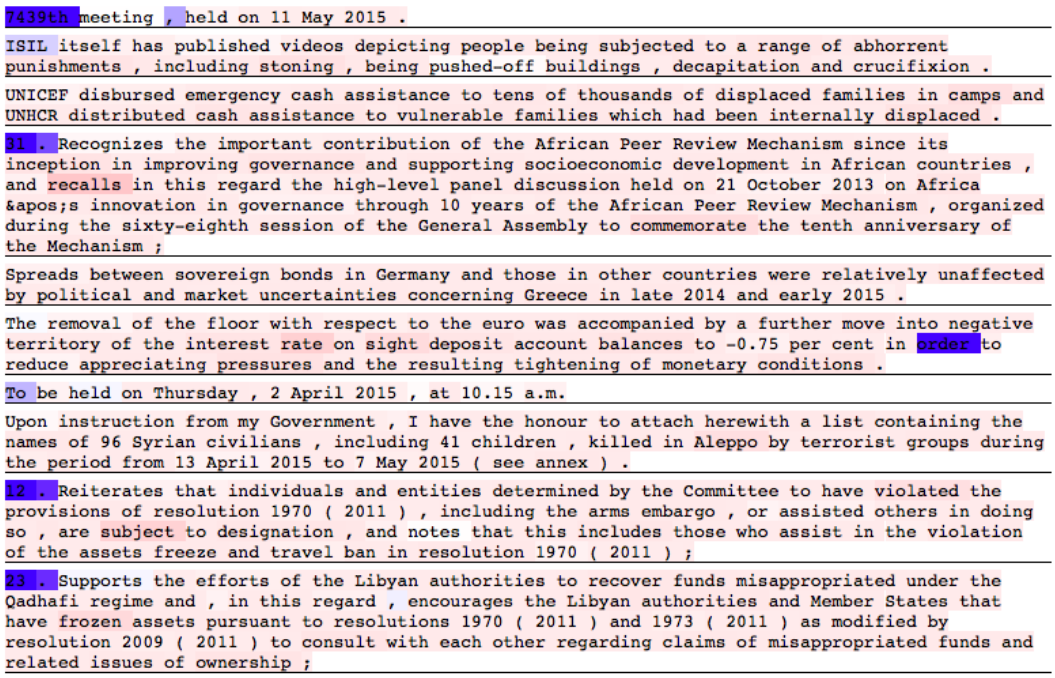


Figure 7: A neuron that activates on numbers in the beginning of sentences. The first 10 sentences in the test set are shown.

B EXAMPLES FOR CONTROLLING TRANSLATIONS

We provide here a more detailed discussion of the example translations from Section 6.

Number Table 4a shows translation control results for a number neuron from an English-Spanish model, which activates negatively/positively on plural/singular nouns. The table shows how the translation of the phrase "The interested *parties*" changes from plural to singular, as we increase the modification α. Notice that using too high α values yields nonsense translations, but with correct number: high negative values give the plural adjective *particulares* ("particular"); high positive values give the singular adjective *útil* ("useful"). In between, we see a nice transition between plural and singular translations. Interestingly, the translations exhibit correct agreement between the modified noun and its adjectives and determines, e.g., *Las partes interesadas* vs. *La parte interesada*. This is probably due to the strong language model in the decoder.

Gender Table 4b shows examples of controlling gender translation for a gender neuron from the same English-Spanish model, which activates negatively/positively on masculine/feminine nouns, for two phrases: "The interested *parties*" and "*Questions relating to information*". The translations of "parties" and "questions" change from masculine to feminine synonyms as we increase the modification α.

Tense Table 4c shows tense control results for the sentence "The committee *supported* the efforts of the authorities". In all cases, we are able to change the translation of "supported" from past to present by modifying the activation of the tense neurons (Table 3). Occasionally, modifying the neuron activation on a single word leads to a change in phrasing; in Arabic the translation changes to "the efforts that the authorities invest". In Spanish, we find a transition from past (*apoyó*) to imperfect (*apoyaba*) to present (*apoya*). Finally, note that in Chinese, we had to use a fairly large α value (in absolute terms) to generate a manipulation. This is consistent with the fact that tense is not usually marked in Chinese. In fact, our modification generates a Chinese expression (正在) that is used to express an action in progress, similar to English "-ing", resulting in the meaning "is supporting".

C A CATALOG OF TOP RANKED NEURONS

In order to illustrate the range of linguistic phenomena captures by individual neurons, we provide here a list of the top 20 neurons (or projected directions, in the case of SVCCA) found by each of our methods, for an example English-Spanish model. For each neuron, we give the percentage of variance that is eliminated by conditioning on position in the sentence or identity of the current token. We also comment on what properties each neuron appears to capture, based on visualizations of neuron activation. Where possible, we give F_1 scores of a GMM model predicting certain properties such as detecting noun phrase segmentation, parenthetical phrases, adjectives, and plural nouns. Annotations are obtained with Spacy: `https://spacy.io`.

Table 7: Top 20 ranked neurons by `MaxCorr`.

Neuron	Position	Token	Comments
464	92%	10%	Position.
342	88%	7.9%	Position.
260	0.71%	94%	Conjunctions: ”and”, ”or”, ”well”, ”addition”.
49	11%	6.1%	Activates for several words after ”and” or ”or”.
124	77%	48%	Position.
394	0.38%	22%	Noun phrase segmentation. 3rd-best F1-score (0.56) for finding interiors of noun phrases. 8th-best IOB accuracy (0.64).
228	0.38%	96%	Unknown: ”’s”, ”the”, ”this”, ”on”, ”that”.
133	0.14%	87%	Adjective detector. Best F1-score (0.56) for finding adjectives.
221	1%	30%	Noun phrase segmentation. Best F1-score for finding interiors (0.72) and for finding beginnings (0.66). Best IOB accuracy (0.80).
90	0.49%	28%	Noun phrase segmentation. Best F1-score (0.59) for finding beginnings on noun phrases. Second-best IOB accuracy (0.73).
383	67%	6.5%	Position.
494	3.5%	96%	Punctuation/conjunctions: ”,”, ”;”, ”Also”, ”also”, ”well”.
120	0.094%	84%	Plural noun detector. Best F1-score (0.87) for retrieving plural nouns.
269	0.1%	80%	Spanish noun gender detector. Very positive for ”islands”, ”activities”, ”measures” – feminine. Very negative for ”states”, ”principles”, ”aspects” – masculine.
232	0.63%	31%	Parentheses. Best F1-score (0.60) for retrieving tokens inside parentheses/quotes/brackets.
332	0.13%	83%	Unknown.
324	0.18%	81%	Unknown.
210	0.61%	45%	Date detector. Third-best F1-score (0.39) for retrieving tokens inside dates.
339	0.48%	39%	Activates for a verb and also surrounding inflection words/auxiliary verbs.
139	0.86%	93%	Punctuation/conjunctions: ”,”, ”.”, ”–”.

Table 8: Top 20 ranked neurons by `MinCorr`.

Neuron	Position	Token	Comments
342	88%	7.9%	Position.
464	92%	10%	Position.
260	0.71%	94%	Conjunctions: ”and”, ”or”, ”well”, ”addition”.
383	67%	6.5%	Position.
250	63%	6.8%	Position.
124	77%	48%	Position.
485	64%	10%	Position.
480	70%	12%	Position.
154	63%	15%	Position.
139	0.86%	93%	Punctuation/conjunctions: ”,”, ”.”, ”–”, ”alia”.
20	60%	9.2%	Position.
494	3.5%	96%	Punctuation/conjunctions: ”,”, ”;”, ”also”, ”well”.
199	67%	6%	Position.
126	42%	9.4%	Unknown.
348	50%	13%	Position.
46	48%	8.6%	Unknown.
196	60%	8.5%	Position.
367	0.44%	89%	Prepositions: ”of”, ”or”, ”United”, ”de”.
186	1.6%	69%	Conjunctions: ”also”, ”therefore”, ”thus”, ”alia”.
244	54%	15%	Position.

Table 9: Top 20 ranked neurons by `LinReg`.

Neuron	Position	Token	Comments
464	92%	10%	Position.
260	0.71%	94%	Conjunctions: ”and”, ”or”, ”well”, ”addition”.
139	0.86%	93%	Punctuation/conjunctions: ”,”, ”.”, ”–”, ”alia”.
494	3.5%	96%	Punctuation/conjunctions: ”,”, ”;”, ”also”, ”well”.
342	88%	7.9%	Position.
228	0.38%	96%	Possibly determiners: ””s”, ”the”, ”this”, ”on”, ”that”.
317	1.5%	83%	Indefinite determiners: ”(”, ”one”, ”a”, ”an”.
367	0.44%	89%	Prepositions. ”of”, ”for”, ”United”, ”de”, ”from”, ”by”, ”about”.
106	0.25%	92%	Possibly determiners: ”that”, ”this”, ”which”, ”the”.
383	67%	6.5%	Position.
485	64%	10%	Position.
186	1.6%	69%	Conjunctions. ”also”, ”therefore”, ”thus”, ”alia”.
272	2%	73%	Tokens that mean ”in other words”: ”(”, ”namely”, ”i.e.”, ”see”, ”or”.
124	77%	48%	Position.
480	70%	12%	Position.
187	1.1%	87%	Unknown: ”them”, ”well”, ”be”, ”would”, ”remain”.
201	0.14%	73%	Tokens that mean ”regarding”: ”on”, ”in”, ”throughout”, ”concerning”, ”regarding”.
67	0.27%	71%	Unknown: ”united”, ”’s”, ”by”, ”made”, ”from”.
154	63%	17%	Position.
72	0.32%	89%	Verbs suggesting equivalence: ”is”, ”was”, ”are”, ”become”, ”constitute”, ”represent”.

Table 10: Top 20 ranked directions by SVCCA.

Position	Token	Comments
86%	26%	Position
1.6%	90%	Detects ”the”.
7.5%	85%	Conjunctions: ”and”, ”well”, ”or”.
20%	79%	Determiners: ”the”, ”this”, ”these”, ”those”.
1.1%	89%	Possibly conjunctions: negative for ”and”, ”or”, ”nor”, positive for ”been”, ”into”, ”will”.
10%	76%	Punctuation/conjunctions: positive for ”,”, ”;”, ”.” ”–”, negative for ”and”.
30%	57%	Possibly verbs: ”been”, ”will”, ”be”, ”shall”.
24%	55%	Possibly date detector.
23%	60%	Possibly adjective detector.
18%	63%	Unknown.
4.5%	88%	Punctuation: ”.”, ”,”, ”;”
9.8%	69%	Forms of ”to be”: ”is”, ”will”, ”shall”, ”would”, ”are”.
1.7%	77%	Combined dates/prepositions/parentheses: negative for ”in”, ”at”, ”.”, positive for dates and in quotes/parentheses/brackets. Noisy.
16%	25%	Activates for a few words after ”and”.
14%	63%	Possibly plural noun detector.
0.8%	73%	Spanish noun gender detector.
11%	61%	Possibly singular noun detector.
13%	58%	Possibly possessives: ”its”, ”his”, ”their”.
1.4%	73%	Spanish noun gender detector.
5.6%	53%	Unknown.