Natural Language Detectors Emerge in Individual Neurons

Anonymous Author(s)
Affiliation
Address
email

Abstract

Although deep convolutional networks have achieved improved performance in many natural language tasks, they have been treated as black boxes because they are difficult to interpret. Especially, little is known about how they represent language in their intermediate layers. In an attempt to understand the representations of deep convolutional networks trained on language tasks, we show that individual units are selectively responsive to specific morphemes, words, and phrases, rather than responding to arbitrary and uninterpretable patterns. In order to quantitatively analyze such intriguing phenomenon, we propose a concept alignment method based on how units respond to replicated text. We conduct analyses with different architectures on multiple datasets for classification and translation tasks and provide new insights into how deep models understand natural language.

1 Introduction

Understanding and interpreting how deep neural networks process natural language is a crucial and challenging problem. While deep neural networks have achieved state-of-the-art performances in neural machine translation (NMT) [20, 4, 8, 22], sentiment classification tasks [24, 5] and many more, the sequence of non-linear transformations makes it difficult for users to make sense of any part of the whole model. Because of their lack of interpretability, deep models are often regarded as hard to debug and unreliable for deployment, not to mention that they also prevent the user from learning about how to make better decisions based on the model's outputs.

An important research direction toward interpretable deep networks is to understand what their hidden representations learn and how they encode informative factors when solving the target task. Among them, studies including Bau et al. [2], Fong & Vedaldi [7], Olah et al. [16, 17] have researched on what information is captured by individual or multiple units in visual representations learned for image recognition tasks. These studies showed that some of the individual units are selectively responsive to specific visual concepts, as opposed to getting activated in an uninterpretable manner. By analyzing individual units of deep networks, not only were they able to obtain more fine-grained insights about the representations than analyzing representations as a whole, but they were also able to find meaningful connections to various problems such as generalization of network [14] or generating explanations for the decision of the model [25, 17, 26].

Since these studies of unit-level representations have mainly been conducted on models learned for computer vision-oriented tasks, little is known about the representation of models learned from natural language processing (NLP) tasks. Several studies that have previously analyzed individual units of natural language representations assumed that they align a predefined set of specific concepts, such as sentiment present in the text [18], text lengths, quotes and brackets [9]. They discovered the emergence of certain units that selectively activate to those specific concepts. Building upon these

Submitted to 32nd Conference on Neural Information Processing Systems (NIPS 2018). Do not distribute.

lines of research, we consider the following question: *What natural language concepts are captured by each unit in the representations learned from NLP tasks?*

To answer this question, we newly propose a simple but highly effective concept alignment method that can discover which natural language concepts are aligned to each unit in the representation. Here we use the term *unit* to refer to each channel in convolutional representation, and *natural language concepts* to refer to the grammatical units of natural language that preserve meanings; *i.e.* morphemes, words, and phrases. Our approach first identifies the most activated sentences per unit and breaks those sentences into these natural language concepts. It then aligns specific concepts to each unit by measuring activation value of replicated text that indicates how much each concept contributes to the unit activation. This method also allows us to systematically analyze the concepts carried by units in diverse settings, including depth of layers, the form of supervision, and data-specific or task-specific dependencies.

The contributions of this work can be summarized as follows:

- We show that the units of deep CNNs learned in NLP tasks could act as a natural language concept detector. Without any additional labeled data or re-training process, we can discover, for each unit of the CNN, natural language concepts including morphemes, words and phrases that are present in the training data.
- We systematically analyze what information is captured by units in representation in multiple settings by varying network architectures, tasks, and datasets. We use VDCNN [5] for sentiment and topic classification tasks on Yelp Reviews, AG News [24], and DBpedia ontology dataset [13] and ByteNet [8] for translation tasks on Europarl [12] and News Commentary [21] datasets.

2 Related Work

2.1 Analysis of deep representations learned for NLP tasks

Most previous work that analyzes the learned representation of NLP tasks focused on constructing downstream tasks that predict concepts of interest. A common approach is to measure the performance of a regression/classification model that predicts the concept of interest to see whether those concepts are encoded in representation of a input sentence. For example, Conneau et al. [6], Adi et al. [1], Zhu et al. [27] proposed several probing tasks to test whether the (non-)linear regression model can predict well the syntactic or semantic information from the representation learned on translation tasks or the skip-thought or word embedding vectors. Shi et al. [19], Belinkov et al. [3] constructed regression tasks that predict labels such as voice, tense, part-of-speech tag, and morpheme from the encoder representation of the learned model in translation task.

Compared with previous work, our contributions can be summarized as follows. (1) By identifying the role of the individual units, rather than analyzing the representation as a whole, we provide more fine-grained understanding of how the representations encode informative factors in training data. (2) Rather than limiting the linguistic features within the representation to be discovered, we focus on covering concepts of fundamental building blocks of natural language (morphemes, words, and phrases) present in the training data, providing more flexible interpretation results without relying on a predefined set of concepts. (3) Our concept alignment method does not need any additional labeled data or re-training process, so it can always provide deterministic interpretation results using only the training data.

3 Approach

We focus on convolutional neural networks (CNNs), particularly their character-level variants. CNNs have shown great success on various natural language applications, including translation, language modeling, and sentence classification [8, 10, 24, 5]. Compared to deep architectures based on fully connected layers, CNNs are natural candidates for unit-level analysis because their channel-level representations are reported to work as templates for detecting concepts [2].

Our approach for aligning natural language concepts to units is summarized as follows. We first train a CNN model for each natural language task and retrieve training sentences that highly activate specific

Dataset	Task	Model	# of Layers	# of Units
AG News	Ontology Classification	VDCNN	4	[64, 128, 256, 512]
DBpedia	Topic Classification	VDCNN	4	[64, 128, 256, 512]
Yelp Review	Polarity Classification	VDCNN	4	[64, 128, 256, 512]
WMT17' EN-DE	Translation	ByteNet	15	[1024] for all
WMT14' EN-FR	Translation	ByteNet	15	[1024] for all
WMT14' EN-CS	Translation	ByteNet	15	[1024] for all
EN-DE Europarl-v7	Translation	ByteNet	15	[1024] for all

Table 1: Datasets and model descriptions used in our analysis.

units. Interestingly, we discover morphemes, words, and phrases that appear dominantly within these retrieved sentences, implying that those concepts have a significant impact on the activation value of the unit. Then, we find a set of concepts which attribute a lot to the unit activation by measuring activation value of each replicated candidate concept, and align them to unit.

3.1 The Model and The Task

We analyze representations learned on three classification and four translation datasets shown in Table 1. Training details for each dataset are available in Appendix **??**. We then focus on the representations in each encoder layer of ByteNet and convolutional layer of VDCNN, because as Mou et al. [15] pointed out, the representation of the decoder (the output layer in the case of classification) is specialized for predicting the output of the target task rather than for learning the semantics of the input text.

3.2 Top K Activated Sentences Per Unit

Once we train a CNN model for a given task, we feed again all sentences in the training data to the CNN and measure the activation in the unit of interest. The dimension of sentence representation is $l \times d$, where l is the length of the activation map and d is the number of units per layer. That is, the activation of each of d units is l-dimensional. For each unit, we retrieve top K training sentences with the highest mean activation over the l entries of the vector. Interestingly, some natural language patterns such as morphemes, words, phrases frequently appear in the retrieved sentences, implying that those concepts might have a large attribution to the activation value of that unit.

3.3 Concept Alignment with Replicated Text

We propose a simple approach for identifying the concepts as follows. For constructing candidate concepts, we parse each of top K sentences with a constituency parser [11]. Within the constituency-based parse tree, we define candidate concepts as all terminal and non-terminal nodes (*e.g.* from sentence *John hit the balls*, we obtain candidate concepts as $\{$*John, hit, the, balls, the balls, hit the balls, John hit the balls*$\}$). We also break each word into morphemes using a morphological analysis tool [23] and add them to candidate concepts (*e.g.* from word *balls*, we obtain morphemes $\{$*ball, s*$\}$). We repeat this process for every top K sentence and build a set of candidate concepts for unit u, which is denoted as $\mathcal{C}_u = \{c_1, ..., c_N\}$, where N is the number of candidate concepts of the unit.

Next, we measure how each candidate concept attributes to the unit's activation value. We create a synthetic sentence by replicating each candidate concept so that its length is identical to the average length of all training sentences (*e.g.* candidate concept *the ball* is replicated as *the ball the ball the ball...*). Replicated sentences are denoted as $\mathcal{R} = \{r_1, ..., r_N\}$, and each $r_n \in \mathcal{R}$ is forwarded to CNN, and their activation value of unit u is measured as $a_u(r_n) \in \mathbb{R}$, which is averaged over l entries. Finally, the degree of alignment (DoA) between a candidate concept c_n and a unit u is defined as follows:

$$\mathsf{DoA}_{u,c_n} = a_u(r_n) \tag{1}$$

In short, the DoA measures the extent to which unit u's activation is sensitive to the presence of candidate concept c_n. If a candidate concept c_n appears in the top K sentences and unit's activation value a_u is responsive to c_n a lot, then DoA_{u,c_n} gets large, suggesting that candidate concept c_n is strongly aligned to unit u.

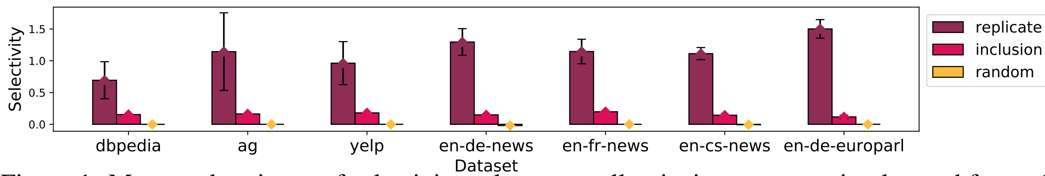


Figure 1: Mean and variance of selectivity values over all units in representation learned for each dataset. Sentences including the concepts that our alignment method discovers always activate units significantly more than random sentences. See section 4.1 for details.

Finally, for each unit u, we define a set of its aligned concepts $\mathcal{C}_u^* = \{c_1^*, ..., c_M^*\}$ as M candidate concepts with the largest DoA values in C_u. Depending on how we set M, we can detect different numbers of concepts per unit. In this experiments, we set M to 3.

4 Experiments

4.1 Evaluation of concept alignment

To quantitatively evaluate how well our approach aligns concepts, we measure how selectively each unit responds to the aligned concept. Motivated by Morcos et al. [14], we define the **concept selectivity** of a unit u, to which a set of concepts $\mathcal{C}_u^*$ that our alignment method detects, as follows:

$$\mathsf{Sel}_u = \frac{\mu_+ - \mu_-}{\max_{s \in \mathcal{S}} a_u(s) - \min_{s \in \mathcal{S}} a_u(s)} \tag{2}$$

where $\mathcal{S}$ denotes all sentences in training set, and $\mu_+ = \frac{1}{|\mathcal{S}_+|} \sum_{s \in \mathcal{S}_+} a_u(s)$ is the average value of unit activation when forwarding a set of sentences $\mathcal{S}_+$, which is defined as one of the following:

- *replicate*: $\mathcal{S}_+$ contains the sentences created by replicating each concept in $\mathcal{C}_u^*$. As before, the sentence length is set as the average length of all training sentences for fair comparison.
- *inclusion*: $\mathcal{S}_+$ contains the training sentences that include at least one concept in $\mathcal{C}_u^*$.
- *random*: $\mathcal{S}_+$ contains randomly sampled sentences from the training data.

In contrast, $\mu_- = \frac{1}{|\mathcal{S}_-|} \sum_{s \in \mathcal{S}_-} a_u(s)$ is the average value of unit activation when forwarding $\mathcal{S}_-$, which consists of sentences that do *not* include any concept in $\mathcal{C}_u^*$.

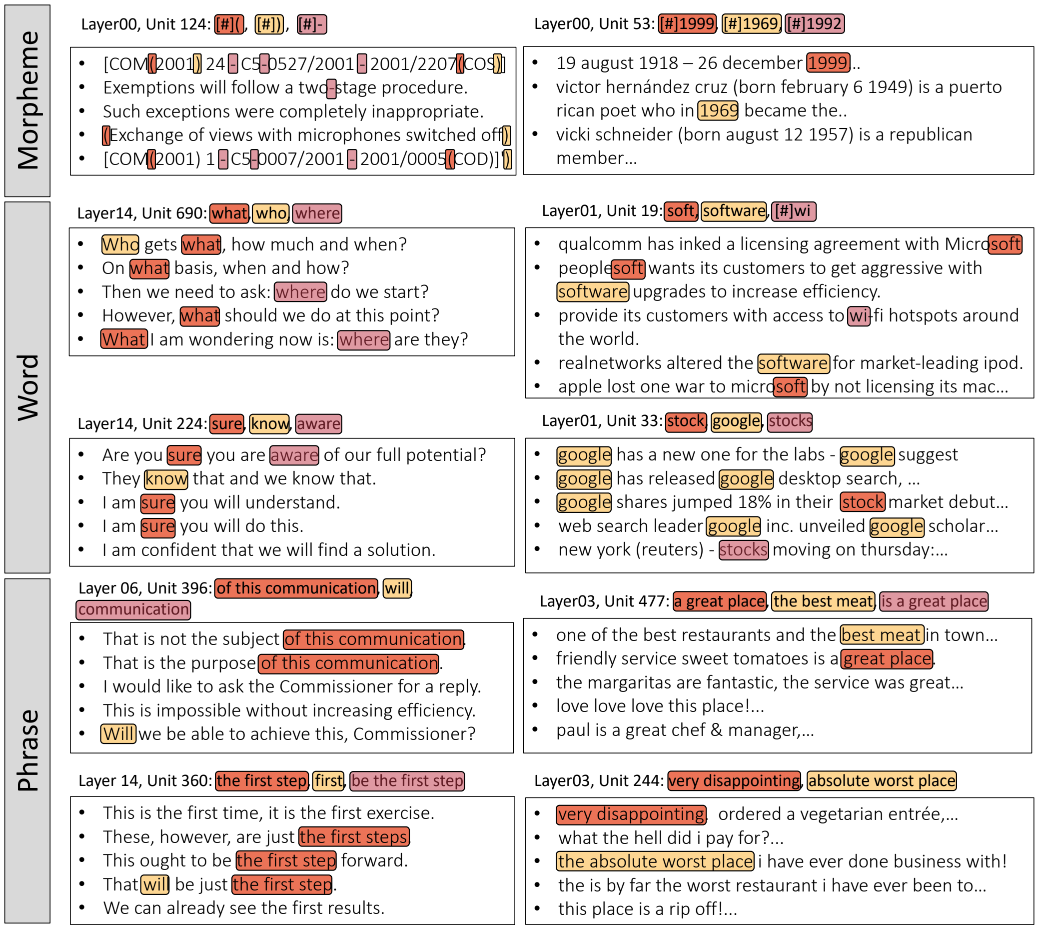
Intuitively, if unit u's activation is highly sensitive to $\mathcal{C}_u^*$ (*i.e.* those found by our alignment method) and if it is not to other factors, then Sel_u gets large; otherwise, Sel_u is near 0.

Figure 1 shows the mean and variance of selectivity values for all units learned in each dataset for the three $\mathcal{S}_+$ categories. Consistent with our intuition, in all datasets, the mean selectivity of the *replicate* set is the highest with a significant margin, that of *inclusion* set is the runner-up, and that of the *random* set is the lowest. These results support our claim that our method is successful to align concepts in which the unit responds selectively.

4.2 Concept Alignment of Units

Figure 2 shows examples of the top K sentences and the aligned concepts that are discovered by our method, for selected units. For each unit, we find the top $K = 10$ sentences that activate the most in the several encoding layer of ByteNet and VDCNN, and select some of them (only up to five sentences are shown due to space constraints). We observe that some patterns appear frequently within the top K sentences. For example, in the top K sentences that activate unit 124 of 0th layer of ByteNet, the concepts of *'(', ')', '-'* appear in common, while the concepts of *soft, software, wi* appear frequently in the sentences for unit 19 of 1st layer of VDCNN. These results qualitatively show that individual units are selectively responsive to specific natural language concepts.

More interestingly, we discover that many units could capture specific meanings or syntactic roles beyond superficial, low-level patterns. For example, unit 690 of the 14th layer in ByteNet captures (*what, who, where*) concepts, all of which play the similar grammatical role. On the other hand, unit 224 of the 14th layer in ByteNet and unit 53 of the 0th layer in VDCNN each captures semantically



(a) Translation (ByteNet) (b) Classification (VDCNN)

Figure 2: Examples of top activated sentences and aligned concepts for some units in the several encoding layers of ByteNet and VDCNN. For each unit, aligned concept and it's presence in top K sentences are painted by the same color. $[\#]$ symbol denotes morpheme concept. See section 4.2 for details.

similar concepts, with the ByteNet unit detecting the meaning of certainty in knowledge (*sure, know, aware*) and the VDCNN unit detecting years (*1999, 1969, 1992*). This suggests that, although we train character-level CNNs with feeding sentences as the form of discrete symbols (*i.e.* character indices), individual units could capture natural language concepts sharing similar semantic or grammatical role.

We note that there are units that detect concepts more abstract than just morphemes, words, or phrases, and for these units our method tends to align relevant lower-level concepts. For example, in units 477 and 244 of the 3rd layer in VDCNN, while each aligned concept emerges only once in the top K sentences, all top K sentences have similar *nuances* like positive and negative sentiments. In these cases, our method does capture relevant phrase-level concepts (e.g., *very disappointing, absolute worst place*), indicating that the higher-level *nuance* (e.g., negativity) is indirectly captured.

We also note that, because the number of morphemes, words and phrases present in training corpus is usually much greater than the number of units per layer, we do not expect to always align any natural language concepts in the corpus to one of the units. Our approach thus tends to find concepts that are considered as more important than others for solving the target task.

Overall, these results suggest how input sentences are represented in the hidden representation of the CNN as follows:

- Several units in the CNN learned on NLP tasks respond selectively to specific natural language concepts, rather than getting activated in an uninterpretable way. This means that these units can serve as detectors for specific natural language concepts.
- There are units capturing syntactically or semantically related concepts, suggesting that they model the *meaning or grammatical role shared between those concepts*, as opposed to superficially modeling each natural language *symbol*.

5 Conclusion

We proposed a simple but highly effective concept alignment method for character-level CNNs to confirm that each unit of the hidden layers serves as detectors of natural language concepts. Using this method, we analyzed the characteristics of units with multiple datasets on classification and translation tasks.

References

[1] Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *ICLR*, 2017.

[2] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *CVPR*, 2017.

[3] Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. What do neural machine translation models learn about morphology? In *ACL*, 2017.

[4] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *EMNLP*, 2014.

[5] Alexis Conneau, Holger Schwenk, Loïc Barrault, and Yann Lecun. Very deep convolutional networks for text classification. In *EACL*, 2017.

[6] Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. What you can cram into a single \$&!#∗ vector: Probing sentence embeddings for linguistic properties. In *ACL*, 2018.

[7] Ruth Fong and Andrea Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *CVPR*, 2018.

[8] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. Neural machine translation in linear time. *arXiv preprint arXiv:1610.10099*, 2016.

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

[10] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware neural language models. In *AAAI*, 2016.

[11] Nikita Kitaev and Dan Klein. Constituency parsing with a self-attentive encoder. *ACL*, 2018.

[12] Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pp. 79–86, 2005.

[13] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

[14] Ari S. Morcos, David G.T. Barrett, Neil C. Rabinowitz, and Matthew Botvinick. On the importance of single directions for generalization. In *ICLR*, 2018.

[15] Lili Mou, Zhao Meng, Rui Yan, Ge Li, Yan Xu, Lu Zhang, and Zhi Jin. How transferable are neural networks in nlp applications? In *EMNLP*, 2016.

[16] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. doi: 10.23915/distill.00007. https://distill.pub/2017/feature-visualization.

[17] Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. The building blocks of interpretability. *Distill*, 2018. doi: 10.23915/distill.00010. https://distill.pub/2018/building-blocks.

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

[19] Xing Shi, Inkit Padhi, and Kevin Knight. Does string-based neural mt learn source syntax? In *EMNLP*, 2016.

[20] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014.

[21] Jörg Tiedemann. Parallel data, tools and interfaces in opus. In *LREC*. ELRA, 2012. ISBN 978-2-9517408-7-7.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Llion Jones, Jakob Uszkoreit, Aidan N Gomez, and Ł ukasz Kaiser. Attention is all you need. In *NIPS*, 2017.

[23] Sami Virpioja, Peter Smit, Stig-Arne Gronroos, and Mikko Kurimo. Morfessor 2.0: Python implementation and extensions for morfessor baseline. In *Aalto University publication series*. Department of Signal Processing and Acoustics, Aalto University, 2013.

[24] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *NIPS*, 2015.

[25] Bolei Zhou, David Bau, Aude Oliva, and Antonio Torralba. Interpreting deep visual representations via network dissection. *IEEE TPAMI*, 2018.

[26] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Interpretable basis decomposition for visual explanation. In *ECCV*, 2018.

[27] Xunjie Zhu, Tingfeng Li, and Gerard Melo. Exploring semantic properties of sentence embeddings. In *ACL*, 2018.