Towards Understanding Learning Representations: To What Extent Do Different Neural Networks Learn the Same Representation

Liwei Wang  Lunjia Hu  Jiayuan Gu  Yue Wu  Zhiqiang Hu  Kun He  John Hopcroft

NeurIPS 2018 Spotlight
Motivation

- It’s widely believed that deep nets learn particular features/representations in their intermediate layers, and people design architectures in order to learn these representations better (e.g. CNN).
Motivation

- It’s widely believed that deep nets learn particular features/representations in their intermediate layers, and people design architectures in order to learn these representations better (e.g. CNN).

- However, there is a lack of theory on what these representations really are.
Motivation

- It’s widely believed that deep nets learn particular features/representations in their intermediate layers, and people design architectures in order to learn these representations better (e.g. CNN).

- However, there is a lack of theory on what these representations really are.

- One fundamental question: are the representations learned by deep nets robust? In other words, are the learned representations commonly shared across multiple deep nets trained on the same task?
Motivation

▶ In particular, suppose we have two deep nets with the same architecture trained on the same training data but from different initializations.

▶ When layer $i$ is the input layer, the similarity is high because both deep nets take the same test examples as input.

▶ When layer $i$ is the final output layer that predicts the classification labels, the similarity is also high assuming both deep nets have tiny test error.

▶ How similar are intermediate layers?

▶ Do some groups of neurons in an intermediate layer learn features/representations that both deep nets share in common? How large are these groups?
Motivation

▶ In particular, suppose we have two deep nets with the same architecture trained on the same training data but from different initializations.
▶ Given a set of test examples, do the two deep nets share similarity in their output of layer $i$?
Motivation

- In particular, suppose we have two deep nets with the same architecture trained on the same training data but from different initializations.
- Given a set of test examples,

**do the two deep nets share similarity in their output of layer $i$?**

- When layer $i$ is the input layer, the similarity is high because both deep nets take the same test examples as input.
Motivation

- In particular, suppose we have two deep nets with the same architecture trained on the same training data but from different initializations.
- Given a set of test examples, do the two deep nets share similarity in their output of layer \( i \)?
  - When layer \( i \) is the input layer, the similarity is high because both deep nets take the same test examples as input.
  - When layer \( i \) is the final output layer that predicts the classification labels, the similarity is also high assuming both deep nets have tiny test error.
Motivation

- In particular, suppose we have two deep nets with the same architecture trained on the same training data but from different initializations.
- Given a set of test examples,

  **do the two deep nets share similarity in their output of layer $i$?**

  - When layer $i$ is the input layer, the similarity is high because both deep nets take the same test examples as input.
  - When layer $i$ is the final output layer that predicts the classification labels, the similarity is also high assuming both deep nets have tiny test error.

- How similar are intermediate layers?
Motivation

▶ In particular, suppose we have two deep nets with the same architecture trained on the same training data but from different initializations.
▶ Given a set of test examples, do the two deep nets share similarity in their output of layer $i$?
  ▶ When layer $i$ is the input layer, the similarity is high because both deep nets take the same test examples as input.
  ▶ When layer $i$ is the final output layer that predicts the classification labels, the similarity is also high assuming both deep nets have tiny test error.
▶ How similar are intermediate layers?
▶ Do some groups of neurons in an intermediate layer learn features/representations that both deep nets share in common? How large are these groups?
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1; a_d$, there exist $A$ and $B$ such that for all $i$, $X(a_i) = Y(a_i) = Z(a_i)$ = span (activation vector of $X$; activation vector of $Y$) = span (activation vector of $Z$; activation vector of $W$).

We say $(f_X; Y; f_Z; W)$ form an exact match!
Two Groups of Neurons Learning the Same Representation: Exact Matches

Output of layer \(i\) after ReLU | Layer \(i+1\) | Output of layer \(i\) after ReLU | Layer \(i+1\)
--- | --- | --- | ---
\[X\] | \[\color{red}Z\] | \[\color{red}W\] | \[\color{red}V\]
\[Y\] | | | |

For test examples \(a_1, a_d\), there exist \(A\) and \(B\) such that for all \(i\),
\[
X(a_i) = Y(a_i) = Z(a_i) = W(a_i)
\]

We say \((f_X, g_Y, f_Z, g_W)\) form an exact match!
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples \(a_1, \cdots, a_d\), there exist \(A\) and \(B\) such that for all \(i\),

\[
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = A
\]

\[
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = B
\]
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \cdots, a_d$, there exist $A$ and $B$ such that for all $i$,

$$\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = A$$

and

$$\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = B$$
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \cdots, a_d$, there exist $A$ and $B$ such that for all $i$,

$$
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix}
= A
$$

$$
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix}
= B
$$
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \ldots, a_d$, there exist $A$ and $B$ such that for all $i$,

$$\begin{bmatrix} X(a_i) \\ Y(a_i) \end{bmatrix} = A \begin{bmatrix} Z(a_i) \\ W(a_i) \end{bmatrix}$$

$$\begin{bmatrix} X(a_i) \\ Y(a_i) \end{bmatrix} = B \begin{bmatrix} Z(a_i) \\ W(a_i) \end{bmatrix}$$
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \cdots, a_d$, there exist $A$ and $B$ such that for all $i$,

$$
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = A
\begin{bmatrix}
Z(a_i) \\
W(a_i)
\end{bmatrix}
\quad \text{and} \quad
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = B
\begin{bmatrix}
X(a_i) \\
Y(a_i)
\end{bmatrix}
$$
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \ldots, a_d$, there exist $A$ and $B$ such that for all $i$,

\[
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = A \begin{bmatrix}
Z(a_i) \\
W(a_i) \\
X(a_i) \\
Y(a_i)
\end{bmatrix}
\]

\[
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = B \begin{bmatrix}
Z(a_i) \\
W(a_i) \\
X(a_i) \\
Y(a_i)
\end{bmatrix}
\]
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \cdots, a_d$, there exist $A$ and $B$ such that for all $i$,

$$
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = A \begin{bmatrix}
Z(a_i) \\
W(a_i) \\
X(a_i) \\
Y(a_i)
\end{bmatrix} = B
$$
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \cdots, a_d$, there exist $A$ and $B$ such that for all $i$,

$$
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix}
= A
\quad \text{and} \quad
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix}
= B
$$
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \cdots, a_d$, there exist $A$ and $B$ such that for all $i$,

$$
\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = A \\
\begin{bmatrix}
Z(a_i) \\
W(a_i) \\
X(a_i) \\
Y(a_i)
\end{bmatrix} = B
$$
Two Groups of Neurons Learning the Same Representation: Exact Matches

For test examples $a_1, \cdots, a_d$, there exist $A$ and $B$ such that for all $i$,

$$\text{span}( [X(a_1), \cdots, X(a_d)], [Y(a_1), \cdots, Y(a_d)]) = \text{span}( [Z(a_1), \cdots, Z(a_d)], [W(a_1), \cdots, W(a_d)])$$
For test examples $\mathbf{a}_1, \cdots, \mathbf{a}_d$, there exist $A$ and $B$ such that for all $i$,

$$\begin{bmatrix}
X(a_i) \\
Y(a_i) \\
Z(a_i) \\
W(a_i)
\end{bmatrix} = A$$

$$\begin{bmatrix}
Z(a_i) \\
W(a_i) \\
X(a_i) \\
Y(a_i)
\end{bmatrix} = B$$

We say $\{\{X, Y\}, \{Z, W\}\}$ form an exact match!
Exact/Approximate Matches between Two Groups of Neurons

Suppose $a_1, a_2, \cdots, a_d$ are the test examples. The output of neuron $X$ on these test examples form a vector $(X(a_1), X(a_2), \cdots, X(a_d))$ called the activation vector [Raghu et al., 2017].
Exact/Approximate Matches between Two Groups of Neurons

- Suppose $a_1, a_2, \cdots, a_d$ are the test examples. The output of neuron $X$ on these test examples form a vector $(X(a_1), X(a_2), \cdots, X(a_d))$ called the activation vector [Raghu et al., 2017].
- If the activation vectors of two groups of neurons span the same linear subspace, we say the two groups of neurons form an exact match.
**Exact/Approximate Matches between Two Groups of Neurons**

- Suppose \( a_1, a_2, \cdots, a_d \) are the test examples. The output of neuron \( X \) on these test examples form a vector \((X(a_1), X(a_2), \cdots, X(a_d))\) called the activation vector [Raghu et al., 2017].

- If the activation vectors of two groups of neurons span the same linear subspace, we say the two groups of neurons form an exact match.

- If the activation vector of every neuron in each group is \( \varepsilon \)-close to the linear subspace spanned by the other group, we say the two groups form an \( \varepsilon \)-approximate match.
  - Vector \( u \) is \( \varepsilon \)-close to linear subspace \( S \) if the sine of the angle between \( u \) and \( S \) is at most \( \varepsilon \), or equivalently, \( \min_{v \in S} \| u - v \|_2 \leq \varepsilon \| u \|_2 \).
Maximum Matches and Simple Matches

Matches are closed under union, so there is a unique maximum match.
Maximum Matches and Simple Matches

- Matches are closed under union, so there is a unique maximum match.
- We define simple matches to be matches that are not the union of smaller matches.
Maximum Matches and Simple Matches

- Matches are closed under union, so there is a unique maximum match.
- We define simple matches to be matches that are not the union of smaller matches.
- Any match is a union of simple matches.
Maximum Matches and Simple Matches

- Matches are closed under union, so there is a unique maximum match.
- We define simple matches to be matches that are not the union of smaller matches.
- Any match is a union of simple matches.
- We designed algorithms for finding the maximum match and the simple matches, and we implemented the algorithms to conduct experiments.
Experimental Findings: Few Matches in Intermediate Layers

Figure: Size of maximum match / number of neurons across layers
Experimental Findings: Few Matches in Intermediate Layers

Figure: Size of maximum match / number of neurons across layers

Low similarity in intermediate layers!
Thank you!

Come to the poster for more details!

05:00 – 07:00 PM @ Room 210 & 230 AB #26