Do neural networks learn filler-gap dependencies?

Satoru Ozaki (ikazos@cmu.edu)
Carnegie Mellon University

4 Aug 2021

1 Neural networks (NNs) work wonders

Neural networks are used everywhere. In machine translation (Google Translate, DeepL, etc.), speech recognition (Siri, Alexa, etc.), and they even fix your typos (Word, Gmail, etc.).

Neural networks are function approximators. E.g. you can think of translation from language A to language B as a function that takes a sentence in A as an input and outputs its translation in B:

$$\text{translate}_{\text{en}\to\text{ja}}(\text{The story made him laugh.}) = \text{Sono hanasi o kiite, kare wa waratta.}$$

Then you can train a neural network that tries to learn (i.e. approximate) the function $\text{translate}_{\text{en}\to\text{ja}}$.

2 Recurrent neural networks (RNNs) are trained to predict next-word probabilities

RNNs are one kind of neural networks. Typically, they learn this function:

$$\mathbb{P}(w_k|w_1,\ldots,w_{k-1})$$

where $w_1,\ldots,w_{k-1}$ is the context defined by the first $k-1$ words in some sentence, and w_k is the kth word. This objective is also known as **language modelling**.

In other words, **if you give an RNN a context and a word, it tells you how likely that the next-word that follows the context is going to be that word.**

RNN is an umbrella term for various kinds of NNs that adopt a **recurrent** architecture. Some examples are standard RNNs, GRUs (Gated Recurrent Units), LSTMs (Long Short-Term Memory), etc.

3 Filler-gap dependencies (technically) have unbounded lengths

In a filler-gap dependency construction like (1), there is a **gap** after (the 2nd occurrence of) the verb *took*, the **filler** is *more notes*, and the **licensor** of the gap is the usage of the comparative *more* and *than*.

(1) Harald took **more notes** in my class than Astrid thought he took ___ in your class.

It's interesting to study filler-gap dependencies like (1) in the context of NNs because...

They represent a (non-linear) syntactic dependency. The sentence is grammatical iff [licensor] XOR [gap], i.e. either there is a licensor and a gap, or there is neither a licensor nor a gap.

(2) a. Harald took **more notes than** Astrid thought he took ___.
b. *Harald took **more notes than** Astrid thought he took a lot of notes.
c. *Harald took **a lot of notes and** Astrid thought he took ___.

d. Harald took **a lot of notes and** Astrid thought he took a lot of notes.

Their unboundedness breaks statistical language modelling. NNs that take fixed-length input cannot represent dependencies longer than their maximum input length. On the other hand, RNNs are special because they 1) take a variable-length input, i.e. a sequence and 2) "remember" previous input.

4 Lit review: NNs (don't) learn filler-gap dependencies

What do neural networks really know about language? Some studies have looked at filler-gap dependencies in particular. They ask: do neural networks really learn filler-gap dependencies?

Chowdhury & Zamparelli 2018: Kind of, but models can't tease apart processing factors and grammaticality. They train GRU and LSTM (2 types of RNNs) on a large English corpus. When presented with *wh*-questions like:

(3) a. Which candidate should the students discuss ___?
b. *Which candidate should the students discuss him?

Their NNs produce higher **perplexity** (≈ lower probability) for the grammatical, gapped variant.

Their NNs also produce higher perplexity for certain island violations (*wh*-extraction from subject, *wh*-extraction from relative clauses), but the same pattern obtains for yes/no questions and declaratives of the same structures. So it's hard to say if 1) their NNs are actually learning island constraints or 2) their NNs just find island configurations harder to parse.

(4) a. *wh*-questions
(i) Who has Tim seen a portrait of ___? (object extraction)
(ii) *Who has a portrait of ___ scared Tim? (subject extraction)
b. Yes/no questions
(i) Has Tim seen a portrait of John? (like (4-a-i))
(ii) Has a portrait of John scared Tim? (like (4-a-ii))
c. Declaratives
(i) Tim has seen a portrait of John. (like (4-a-i))
(ii) A portrait of John has scared Tim. (like (4-a-ii))

Wilcox et al. 2018, 2019: Yes, LSTMs learn several structural properties of filler-gap dependencies and several island constraints. They take 2 pre-trained LSTMs. They look at embedded *wh*-questions, and focus on a metric they call *wh*-licensing interaction, which is the interaction between [licensor] and [gap]:

(5) a. I know that the lion devoured a gazelle at sunrise. [-licensor, -gap]
b. *I know what the lion devoured a gazelle at sunrise. [+licensor, -gap]
c. *I know that the lion devoured ___ at sunrise. [-licensor, +gap]
d. I know what the lion devoured ___ at sunrise. [+licensor, +gap]

Their metric tells you: **how much does having a licensor improve your [+gap] sentence?** They take this metric to indicate "how well" the model has learned filler-gap dependencies.

Chaves 2020, Da Costa & Chaves 2020: No. RNNs become less sensitive to agreement violations over deeper dependencies, nor do they learn certain exceptions to island constraints. They use the same LSTMs as Wilcox et al. 2018, 2019[1]. When you *wh*-extract a phrase from a clause, the extracted phrase and the main verb in the clause have to agree in number.

(6) a. Filler ... 0 embeddings ... gap

[1] They also check Transformers, but that is a completely different problem...

(i) Which lawyer was smart?
(ii) *Which lawyers was smart?

b. Filler ... 4 embeddings ... gap
(i) Which lawyer I think [you said [he claimed [she believed [was smart?
(ii) *Which lawyers I think [you said [he claimed [she believed [was smart?

The agreement should be observed regardless of the number of embeddings separating the filler from the gap. However, LSTM RNNs produce a higher **surprisal** (≈ lower probability) for (6-b-i) (grammatical sentence with a long-distance dependency) than for (6-a-ii) (ungrammatical sentence with a short-distance dependency).

5 Comparatives

We take the pre-trained LSTMs (called Google and Gulordava models) just like Wilcox et al. 2018, 2019, Chaves 2020 and Da Costa & Chaves 2020. We generate a bunch of comparative sentences like (7), conditioned by [licensor] and [gap]:

(7) a. Harald took **more notes** in my class **than** Astrid thought he took ___ in your class.
b. H. took **more notes** in my class **than** A. thought he took **a lot of notes** in your class.
c. H. took **a lot of notes** in my class **and** A. thought he took ___ in your class.
d. H. took **a lot of notes** in my class **and** A. thought he took **a lot of notes** in your class.

We measure the average surprisal (negative log-likelihood) of the LSTM RNN over the post-gap material, e.g.

$$\begin{aligned}\text{Avg post-gap surprisal}(\text{(7-a)}) = \frac{1}{4}\,(\; &-\log \mathbb{P}(\text{in} \quad &&|\text{Harald ... took} &&)\\ &-\log \mathbb{P}(\text{your} &&|\text{Harald ... took in} &&)\\ &-\log \mathbb{P}(\text{class} &&|\text{Harald ... took in your} &&)\\ &-\log \mathbb{P}(\text{.} &&|\text{Harald ... took in your class} &&)\;)\end{aligned}$$

Since surprisal is **negative** likelihood, the higher the probability, the lower the surprisal.

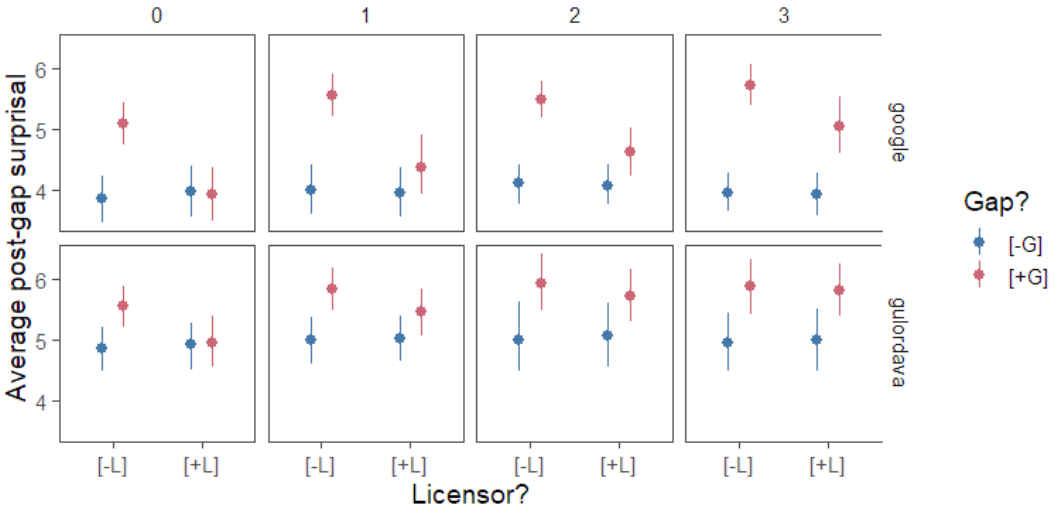

Figure 1: 0/1/2/3 = number of embeddings. google/gulordava = 2 LSTMs we tested.

Observations so far:

Having a licensor makes having a gap better. Looking at the leftmost column (0 embeddings), [+licensor, +gap] has a much lower surprisal than [-licensor, +gap].

This effect diminishes when you have more embeddings. The same difference between [+licensor, +gap] and [-licensor, +gap] diminishes over increasing levels of embeddings (they completely collapse in [gulordava, 3]).

Both models prefer gapless sentences over gapped sentences. The [+gap] sentences always has a higher surprisal than [-gap] sentences. [+gap] has a significant effect on surprisal.

Conclusions so far:

What counts as having acquired the knowledge of filler-gap dependencies? The NNs certainly learn that having a licensor makes a gap better, which you can interpreted roughly as "you need a licensor for a gap". But they clearly haven't learned the other direction, i.e. "you need a gap for a licensor".

Grammaticality = probability? Clearly, there is not a single surprisal threshold you can use to correctly classify the surprisal results into categorical grammaticality judgments. So what? Is this a reasonable expectation (NNs do show this for some phenomena, e.g. subject-verb agreement and NPI licensing in English)? Is there a way we can make NNs show this behavior?