Linguistic generalization and compositionality in modern artificial neural networks

Marco Baroni[1,2,3]

[1]Catalan Institute for Advanced Studies and Research, Barcelona, Catalunya, Spain
[2]Department of Translation and Language Sciences, Universitat Pompeu Fabra, Carrer Roc Boronat 138, Barcelona 08018, Spain
[3]Facebook Artificial Intelligence Research, Paris, France

(iD) MB, 0000-0001-5066-3580

In the last decade, deep artificial neural networks have achieved astounding performance in many natural language-processing tasks. Given the high productivity of language, these models must possess effective generalization abilities. It is widely assumed that humans handle linguistic productivity by means of algebraic compositional rules: are deep networks similarly compositional? After reviewing the main innovations characterizing current deep language-processing networks, I discuss a set of studies suggesting that deep networks are capable of subtle grammar-dependent generalizations, but also that they do not rely on systematic compositional rules. I argue that the intriguing behaviour of these devices (still awaiting a full understanding) should be of interest to linguists and cognitive scientists, as it offers a new perspective on possible computational strategies to deal with linguistic productivity beyond rule-based compositionality, and it might lead to new insights into the less systematic generalization patterns that also appear in natural language.

This article is part of the theme issue 'Towards mechanistic models of meaning composition'.

1. Introduction

Neural networks have been a prominent tool to model cognitive phenomena at the mechanistic level since at least the mid-80s [1]. In the last decade, under their 'deep learning' re-branding, neural networks have also proven their worth as astonishingly successful general-purpose, large-scale machine-learning algorithms [2]. In the domain of natural language, today neural networks are core components of effective machine-translation engines such as Google Translate (https://translate.google.com) and DeepL (https://www.deepl.com/translator). OpenAI recently caused controversy when it announced that it would not make its new language modelling network publicly available, as it generates novel text about arbitrary topics that is so realistic and coherent that it could easily be deployed in malicious applications, such as bulk creation of faked or abusive content (See https://openai.com/blog/better-language-models/). The debate on the linguistic abilities of neural networks of the 80s and 90s involved small experiments and theoretical speculation about whether neural networks would ever be able to process language at scale (e.g. [3–7], among many others). Given the impressive empirical results achieved by modern neural networks, the interesting question today is not *whether*, but *how* neural networks achieve their language skills, and what causes the surprising and sometimes dramatic failures that still affect them [8,9].

Many early neural networks were developed with the specific purpose of understanding mental processes, and thus cognitive or biological plausibility was a central concern. Modern deep networks are instead optimized for practical goals, such as better translation quality or information extraction. It is thus unlikely that their behaviour will closely mimic human cognitive processes. I contend, however, that their high natural language-processing performance makes

them very worth studying from the perspective of cognitive science. Following an early proposal by McCloskey [10], we should treat this as comparative psychology. Just like the communication systems of primates and other species can shed light on the unique characteristics of human language (e.g. [11,12]), studying how artificial neural networks accomplish (or fail to accomplish) sophisticated linguistic tasks can provide important insights on the nature of such tasks, and the possible ways in which a computational device can (or cannot) solve them. This is the perspective I adopt here in looking at linguistic productivity and compositionality in deep networks.

Natural languages are characterized by immense *productivity*, in the sense that they license a theoretically infinite set of possible expressions. Linguists almost universally agree that *compositionality*, the ability to construct larger linguistic expressions by combining simpler parts, subtends productivity. The focus is typically on *semantic* compositionality, the principle whereby the meaning of a linguistic expression is a function of the meaning of its components and the rules used to combine them [13,14]. When studying the generalization properties of neural networks, I believe it is more useful to consider a broader notion of compositionality, also encompassing, for example, the syntactic derivation rules allowing us to judge the grammaticality of nonce sentences independently of their meaning [15,16]. Indeed, compositionality is conjectured to be a landmark not only of language but of human thought in general [5,8,17], and the compositional abilities of neural networks have been tested on tasks that are not semantic [7] or even linguistic in nature [18]. If a system is not compositional in this more general sense, it will not, *a fortiori*, be able to build complex semantic representations by parallel composition of syntactic and semantic constituents.

Compositional operations in language (and thought) are argued to constitute a rule-based algebraic system, of the sort that can be formally captured by symbolic functions with variable slots. It follows that compositionality is 'systematic', in the sense that a function must apply in the same way to all variables of the right type. As famously put by Fodor, if you know the correct compositional rules to understand *John loves Mary*, you *must* also understand *Mary loves John* [5] (Fodor and colleagues distinguish systematicity and compositionality: simplifying somewhat, they see compositionality, in the stricter semantic sense presented above, as the natural consequence of applying systematic rules in the domain of natural language). Neural networks are not thought to be capable of acquiring systematic rules. Their linguistic generalization abilities have thus been the focus of much research in the past (e.g. [19–23] among many others). Note that productivity *per se* does not entail systematic compositionality. Some forms of generalization outside language are not rule-based (and not systematic). For example, similarity-driven reasoning about concept instances is probably too fuzzy and prototype-based to be accounted for by systematic rules [24]. One could also imagine a language that is productive but not (systematically) compositional. For example, Hockett [25], reflecting about the origins of language, conjectured a stage in which new expressions are formed not by systematic composition of smaller parts, but by blending unanalysed wholes in inconsistent ways. Modern languages also exhibit many corners of non-systematic, partial productivity, a point I will return to in the conclusion. Still, systematic composition rules are an extremely powerful generalization mechanism. Once you know that *super-* attaches to adjectives to form other adjectives, you can in principle understand an infinite (if rather contrived) set of words: *super-good*, *super-super-good*, etc. In this context, it has been argued that lack of compositionality is one reason why modern neural networks, in striking contrast to humans, require huge amounts of data to induce correct generalizations [8].

In this article, I introduce researchers interested in compositionality from a cognitive perspective to some relevant recent work about linguistic productivity in modern deep networks. After briefly reviewing the main novelties characterizing current deep language-processing architectures, I will present experimental evidence that these systems are at the same time able to capture subtle syntactic generalizations about novel forms (thus handling a sophisticated form of grammatical productivity), and failing to show convincing signs of rule-based compositionality. I will conclude with some considerations about the significance of these results for the general study of linguistic productivity.

2. Modern deep networks for language processing: what has changed

Much of the last-decade improvements in neural network performance are due to the availability of larger training datasets that, together with computational power and better optimization methods, have enabled large-scale data-driven training of complex, multi-layer architectures [2,26]. In the domain of language, large corpora made it possible to train deep networks with the simple *language modelling* method [27,28]. In this set-up, the weights of a sequence-processing network are set by optimizing the objective of predicting the next word in a text, given the previous context. This is schematically illustrated for a recurrent network in figure 1a. Nowadays, language modelling is used as a general-purpose way to pre-train networks to perform linguistic tasks [29]. It is also an interesting training regime from a cognitive point of view, since humans in many cultures are also exposed to large amounts of raw language data during acquisition (e.g. [30]), and predicting what comes next plays a central role in cognition [31–34]. Of course, prediction is not the only task humans perform when learning a language, and how to design more varied and human-like training environments is an open research issue.

Important advances have also been made in architectural terms. The original sequence-processing recurrent network schematically illustrated in figure 1a reads some input (e.g. a word), and produces an output (e.g. a guess about the next word) at each time step. The output of the network at time t is a nonlinear function of the input at time t, as well as of the state of the network itself at step $t-1$ (weighted by *recurrent connections* that propagate activations across time) [35]. *Gated* recurrent networks, such as long short-term memory networks [36] and gated recurrent units [37], possess mechanisms regulating the dynamics of information processing across time, whose parameters are jointly induced with the rest of the network in the training phase. In particular, the network gates can 'decide', at each time step and for each unit, how much the input should be updated with information about the current input (versus preserving currently stored information), and how much it should contribute to (the hidden representation determining) the current output. Such gating mechanisms, schematically illustrated in figure 1b, allow longer-term and more nuanced control of the information flow. Gates have
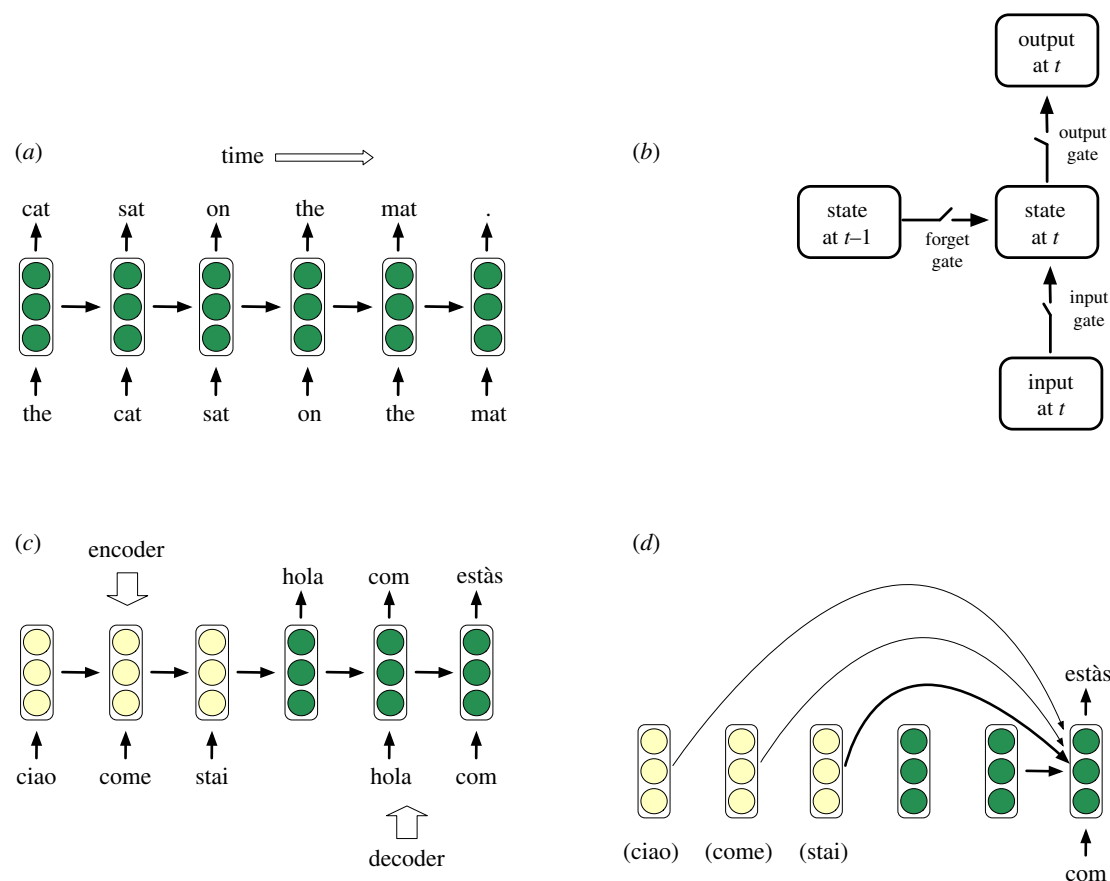
Figure 1. Architectural features of modern sequence-processing networks. (*a*) A traditional **recurrent network** processing a sequence of words in multiple time steps. The black arrows represent sets of weighted connections (a single arrow stands for multiple unit-to-unit connections). The three green (darker-shaded) circles represent the same network at different time steps, which might be structured into multiple layers (not depicted in the figure). The output at time t is a (nonlinear) function of the current input as well as the state of the network at time $t-1$ (information is carried through time by the *recurrent* connections, represented by vertical arrows in the figure). (*b*) In modern sequence-processing networks, a set of **gates** modulate the amount of information flowing through the connections of the network. (*c*) The **encoder–decoder** architecture is modular, with separate subnetworks (in yellow (lighter shading) and green (darker shading) in the figure) trained to process input (encoder) and generate output (decoder). The decoder is typically initialized from the last state of the encoder. (*d*) In **attention**-enhanced architectures, the state of the encoder at each time step is separately stored in memory, and at each decoding step the network dynamically determines how much information to read from each of the memory slots. The diagram schematically depicts the step in which the decoder makes a prediction based on the last word it produced (*com*), its previous state and attention-mediated memorized states from the decoder (with line thickness symbolizing the relative weight assigned to each memorized state). Unlike the previous diagrams, this one only shows the connections that are active in the last depicted processing step. (Online version in colour.)

proven empirically extremely effective, and are standard in modern language-processing networks [38].

Another important innovation consisted in decoupling input and output processing through *encoder–decoder* architectures [39]. As sketched in figure 1*c*, separate subnetworks are trained to process the input and generate the output, with the last state of the first network (the encoder) used to initialize the second (the decoder). Input–output decoupling allows effective handling of *sequence-to-sequence* tasks, in which a sequence (e.g. a sentence in a language) has to be mapped onto another sequence (e.g. a sentence in another language), especially where input and output sequences are very different. This approach is by now standard in machine translation (where, for example, it permits flexible mapping between languages with different word orders), but it is extremely general, and it has also been employed to convert linguistic instructions to actions and sentences to semantic representations [40,41].

A (learned) *attention* mechanism, in its original form [42], automatically allows the decoder to read more or less information from different encoder states (on the basis of similarity computations between vectors representing current and past states). As schematically illustrated in figure 1*d*, when it

is about to translate the word following *com* ('*how*' in Catalan), an attention-augmented network might decide to read more from the encoder state corresponding to the word that immediately follows *come* ('*how*') in the Italian source. Attention plays an increasingly important role in modern encoder–decoder architectures, to the point that the most successful contemporary models dispense with recurrent connections altogether, and rely instead on a rich attention mechanism to keep track of relevant past information [43].

Modern sequence-processing networks are complex systems, equipped with strong structural priors such as gates, encoding and decoding modules and attention. They should not be thought of as 'tabulae rasae', as they often were in early debates on connectionism. At the same time, the 'innate' biases they encode are rather different from those assumed to shape human linguistic competence. Some researchers *are* trying to inject into modern networks priors closer to those traditionally postulated by linguists, such as a preference for hierarchical tree structures, (e.g. [44,45]; see also Brennan's contribution to this issue [46]). Models of this latter kind have however not yet proven their worth as generic language-processing devices, and I will not delve further into them. Intriguingly, Williams *et al.* [47] recently found that, when such models are not provided

with explicit information about conventional compositional derivations, they come up with tree structures that do not resemble those posited by linguists at all. This is in line with the basic tenet of this paper, that neural networks might solve complex linguistic tasks, but not in the way we expect them to be solved.

3. Colourless green grammatical generalization in deep networks

There is no doubt that modern language-processing neural networks can generalize beyond their training data. Without such ability, their astounding performance in machine translation [43,48] would remain unexplained, as most sentences, or even long word sequences, in any text to be translated are extremely unlikely to have ever been produced before [49]. Recently, there has been widespread interest in understanding whether this performance depends on shallow heuristics, or whether the networks are indeed capturing grammar-based generalizations, of the sort that would be supported by symbolic compositional rules (e.g. [50–52]).

Gulordava *et al.* [53] test the networks' grammatical 'intuitions' in a set-up that is strictly controlled to insure they are tapping into their productive competence (as opposed to memorized patterns). They train a gated recurrent network on large Wikipedia-derived corpora, using the language modelling objective. They then feed it minimal pairs of sentences respecting/violating long-distance number agreement. Crucially, the test items are semi-randomly generated nonsense sentences. For example, one minimal pair is: *I realize the **wars** on which I should revise your hunt **understand/understands*** (here, the plural verb variant with *understand* is the grammatical one). The model, without further task-specific tuning, is asked to compute the probability of the two variants, and it is said to have produced the correct judgement if it assigns a higher probability to the grammatical one. Gulordava's non-sensical twist strips off possible semantic, lexical and collocational confounds, focusing on the abstract grammatical generalization.

Gulordava and colleagues run the experiment in English, Hebrew, Italian and Russian. In all cases, neural networks display a preference for the grammatical sentences that is well above chance level and competitive baselines (the lowest performance occurs in English, where the network still guesses correctly 74% of the cases, where chance level is at 50%). Moreover, for Italian, they compare the network performance with human subjects taking the same test. Human accuracy turns out to be only marginally above that of the network (88.4% versus 85.5%).

These results suggest that neural networks capture abstract, structure-based grammatical generalizations. However, the evidence is indirect, and others [54,55] have suggested that the networks are really capitalizing on shallow heuristics (such as: 'percolate the number of the first noun in a sentence to all verbs'). Lakretz and colleagues [56] conducted extensive ablation and connectivity studies of the Gulordava network. They found that the network specialized very few units to the task of carrying long-distance number information. For example, when the activation of these units is fixed to 0, the network performance on agreement tasks slides towards chance level. Importantly, these units are strongly connected to a subnetwork of nodes that can be independently shown to be sensitive to hierarchical syntactic constituency.

Unveiling this circuit in the network suggests that the latter has indeed developed genuine grammatical processing mechanisms, and it is not simply relying on surface heuristics when computing agreement.

The kind of productivity that was probed in these studies is grammatical in nature. Just like in Chomsky's famous '*colorless green ideas*' example [15], Gulordava's network can tell apart subtly different grammatical and ungrammatical non-sensical sentences that are certainly very far from anything it was exposed to during training. Lakretz's analysis of the network [56] further suggests that its behaviour relies on genuine sensitivity to grammatical structure. The traditional linguistic story about a cognitive device displaying this behaviour would be that it possesses a compositional rule-based system, allowing it to reliably process novel linguistic input (such as Gulordava's stimuli). We have no evidence, yet, about whether Gulordava's network possesses something akin to such a system, but I will now turn to experiments with similar networks directly probing their compositionality through a miniature language designed for this purpose that suggest that they do not.

4. Compositional generalization: can deep networks *dax* twice?

Lake & Baroni [57] introduced SCAN, a benchmark to test the compositional abilities of sequence-processing networks, later extended in [58]. The SCAN miniature language is characterized by a grammar generating a large but finite number of linguistic navigation commands, and an interpretation function associating a semantic representation (a sequence of action symbols) to each possible command. The primitives of the language are verbs such as *jump* and *run*, mapped to the corresponding actions (e.g. JUMP, RUN). Primitives are combined with a set of adverb-like modifiers and conjunctions, resulting in composite expressions denoting action sequences. For example, if [[x]] is the action associated with expression 'x' by the interpretation function, then 'x *and* y' maps to [[x]] [[y]] and 'x *twice*' maps to [[x]] [[x]]. Consequently, '*jump twice and run*' is compositionally mapped to the action sequence: JUMP JUMP RUN.

The general evaluation paradigm is as follows. An encoder–decoder network is trained on a set of SCAN commands for long enough that the network learns to accurately execute them (that is, to map them to the corresponding action sequences). The network performance is then evaluated on executing a set of test commands that were not encountered during training. Note that, unlike in the experiments reviewed in the previous section, where it had to assign a probability to pre-determined sentences, here the network has to actively produce an output action sequence; thus its generative abilities are more directly probed.

By splitting the possible SCAN commands into different training and testing partitions, we can gain insights into what the network is actually learning. I will focus here on the results obtained with three of the proposed splits, and their implications. In the *random* split, 80% of the commands are used for training, the remaining 20% for testing. This split checks the network ability to handle generic productivity (of the sort that might occur in standard machine-translation benchmarks), since all test expressions are new for the network. However, there is no *controlled* difference between the two sets, and the network will, in general, have seen a number of

examples quite similar to those it has to execute. For example, the test set contains the command 'look around left twice and jump right twice'. This is new, but the training data contain examples of look around left twice and 'jump right twice', both on their own and in conjoined expressions (e.g. '**look around left twice** and turn left', 'run twice and **jump right twice**').

In the *jump* split, the training set contains all possible commands with all primitives except *jump*. During training, *jump* is presented multiple times, but only in isolation. The test set is then made of all composite commands containing *jump*. For example, at training the network is exposed to 'run twice' and 'walk and look', and at test time it must execute 'jump twice' and 'walk and jump'. The split is straightforward for a system possessing composition rules such as: 'x *twice*' maps to [[x]] [[x]]. This is akin to a human subject learning a new verb *daxing*, and being immediately able to understand what '*dax twice*' means.

However, since *jump* only occurs in isolation at training time, the tested specimen could also reasonably conclude that the latter has a different distribution from the other verbs, and refuse to generalize it to novel composite contexts [59]. Loula *et al.* [58] introduce different partitions that control for this factor. In particular, in the *around–right* split, the training data contain all possible commands, except those where *around* is combined with *right*. Still, the network is given plenty of distributional evidence that *right* and *left* function identically otherwise, and it is exposed to many examples illustrating the behaviour of *around* in combination with *left*. The training data contain, among others, the commands 'run left', 'run right', 'jump opposite left', 'jump opposite right', 'look around left'. The test data include 'look around right'.

The results in the original papers, and the further experiments of [60] with more carefully-tuned models, tell a simple story. Modern recurrent sequence-processing networks, just as conjectured by Jerry Fodor and Gary Marcus (e.g. [5,7]), are able to generalize in a fuzzy, similarity-based way that allows them to succeed in the *random* split (100% average test accuracy and s.d. $\approx$ 0.0% across multiple runs with different initializations). However, they utterly fail at the *jump* and *around–right* splits, which require inducing systematic compositional rules (12.5% accuracy with 6.6% s.d. and 2.5% accuracy with 2.7% s.d., respectively).

In very recent work, Dessì & Baroni [61] found a somewhat more intriguing pattern. They replaced the gated recurrent network architectures used in earlier SCAN work with an out-of-the-box convolutional network that dispenses with recurrence by heavily relying on attention, and that has independently been shown to achieve competitive results in machine translation ([62]; Dessì & Baroni were not able, for the time being, to trace the difference in performance between this network and the previously tried models back to their architectural differences). The new model is still able to perfectly generalize in the random split (100% test accuracy, s.d. $\approx$ 0.0%), but now it reaches a surprising middle ground in the 'compositional' splits: 69.2% accuracy (8.2% s.d.) with *jump* and 56.7% (10.2% s.d.) with *around–right*. As chance level accuracy in these tasks is practically 0%, the results show that the network does get some important generalizations right. Still, if it extracted the correct rules, we would expect it to be perfectly accurate, which is not the case. Dessì & Baroni initially conjectured that the network learned a subset of compositional rules, explaining its partial success. For example, it could be that the network learned how to map 'x *twice*' to [[x]] [[x]], but failed to learn the corresponding 'x *thrice*' rule. However, a follow-up error analysis showed this not to be the case. The network makes errors relatively uniformly across composition frames, and, qualitatively, it does not display any trace of systematicity. For example, in the 'jump' split the network executes 'jump left after walk' correctly, but fails 'jump left after run'. In the around–right split, the network can execute 'run around right', but not 'walk around right'.

Similar conclusions were drawn by Andreas [63] from a very different experiment. Andreas trained sender and receiver sequence-processing networks to play a 'communication game': the sender must describe the properties of a set of objects to the receiver through a discrete communication channel, and the receiver must reconstruct the correct target objects. The agents were trained by rewarding successful communication. Across random initializations, the sender agent developed more or less compositional codes for its messages to the receiver. Interestingly, at least some codes with low degrees of compositionality were as good at generalizing to new object descriptions as more compositional ones. Again, neural networks can be productive without being compositional.

5. Conclusion

When critics of classic connectionism argued that neural networks are intrinsically incapable of inducing symbolic composition rules, they probably assumed that this would severely limit their practical ability to handle natural language. The empirical evidence concerning modern deep networks is surprising, as it suggests that they are extremely proficient at language, while indeed not being compositional. Note in particular that the experiments by Gulordava, Lakretz and others I reviewed in §3 above suggest that the linguistic proficiency of neural networks extends beyond shallow pattern recognition, to competence about structure-dependent generalizations of the sort traditionally attributed to the command of systematic compositional rules. Our current understanding of the strategies learned by these networks is very limited, and our highest priority should be to develop better analytical tools to uncover the mechanisms that lead to the detected dissociation of productive grammatical competence and systematic compositionality.

From the perspective of AI research, one central question is whether making neural networks more compositional, for example by means of more structured modular architectures [64], will also make them more adaptive and faster at learning, while not costing in terms of generality. Current deep networks are also brittle in surprising ways. For example, they are easily fooled by 'adversarial' examples (e.g. words with a few characters shifted) that would be trivially handled by humans (e.g. [65–67]). Explicitly compositional architectures might provide added robustness to similar attacks, or at least afford better insights into the often mysterious failings of the networks. In turn, this might lead to progress in ambitious natural language-processing tasks where the success of modern deep networks is less clear-cut, such as machine reading and natural language inference [e.g. 68,69].

Still, the way in which current models generalize without possessing anything resembling compositional rules also offers an intriguing opportunity for comparative studies to linguists and cognitive scientists. Classic and modern criticism of neural networks emphasizes the aspects of human language that are best characterized by clear-cut, algebraic rules.

Language, however, is also host to plenty of productive phenomena that obey less systematic, fuzzier laws, ranging from phonologically driven generalizations of irregular inflections [70], to partial semantic transparency in derivational morphology [71], to semi-lexicalized constraints in syntax [72], to the early stages of grammaticalization in language change [73]. Progress in understanding the linguistic capabilities of neural networks might help us to make precise predictions about the origin, scope and mechanics of these phenomena, and ultimately to develop a more encompassing account of the amazing productivity and malleability of human language.

References

1. Rumelhart DE, McClelland J, PDP Research Group (eds) 1986 Foundations. *Parallel distributed processing: explorations in the microstructure of cognition*, vol. 1. Cambridge, MA: MIT Press. (doi:10.7551/mitpress/5236.001.0001)

2. LeCun Y, Bengio T, Hinton G. 2015 Deep learning. *Nature* **521**, 436–444. (doi:10.1038/nature14539)

3. Smolensky P. 1990 Tensor product variable binding and the representation of symbolic structures in connectionist networks. *Artif. Intell.* **46**, 159–216. (doi:10.1016/0004-3702(90)90007-M)

4. Elman J. 1991 Distributed representations, simple recurrent networks, and grammatical structure. *Mach. Learn.* **7**, 195–225. (doi:10.1007/BF00114844)

5. Fodor J, Lepore E. 2002 *The compositionality papers*. Oxford, UK: Oxford University Press.

6. Pinker S, Ullman M. 2002 The past and future of the past tense. *Trends Cogn. Sci.* **6**, 456–463. (doi:10.1016/S1364-6613(02)01990-3)

7. Marcus G. 2003 *The algebraic mind*. Cambridge, MA: MIT Press.

8. Lake BM, Ullman TD, Tenenbaum JB, Gershman SJ. 2017 Building machines that learn and think like people. *Behav. Brain Sci.* **40**, e1. (doi:10.1017/S0140525X16001837)

9. Marcus G. 2018 Deep learning: a critical appraisal. https://arxiv.org/abs/1801.00631.

10. McCloskey M. 1991 Networks and theories: the place of connectionism in cognitive science. *Psychol. Sci.* **2**, 387–395. (doi:10.1111/j.1467-9280.1991.tb00173.x)

11. Schlenker P *et al.* 2016 Formal monkey linguistics. *Theor. Linguist.* **42**, 1–90. (doi:10.1515/tl-2016-0001)

12. Townsend S, Engesser S, Stoll S, Zuberbühler K, Bickel B. 2018 Compositionality in animals and humans. *PLoS Biol.* **16**, e2006425. (doi:10.1371/journal.pbio.2006425)

13. Frege G. 1892 Über Sinn und Bedeutung [Sense and reference]. *Z. Philos. Phil. Kritik* **100**, 25–50 (in German).

14. Montague R. 1970 Universal grammar. *Theoria* **36**, 373–398. (doi:10.1111/j.1755-2567.1970.tb00434.x)

15. Chomsky N. 1957 *Syntactic structures*. Berlin, Germany: Mouton.

16. Chomsky N. 1965 *Aspects of the theory of syntax*. Cambridge, MA: MIT Press. (doi:10.21236/AD0616323)

17. Fodor J, Pylyshyn Z. 1988 Connectionism and cognitive architecture: a critical analysis. *Cognition* **28**, 3–71. (doi:10.1016/0010-0277(88)90031-5)

18. Lake B, Salakhutdinov R, Tenenbaum J. 2015 Human-level concept learning through probabilistic program induction. *Science* **350**, 1332–1338. (doi:10.1126/science.aab3050)

19. Christiansen M, Chater N. 1994 Generalization and connectionist language learning. *Mind Lang.* **9**, 273–287. (doi:10.1111/j.1468-0017.1994.tb00226.x)

20. Marcus G. 1998 Rethinking eliminative connectionism. *Cognit. Psychol.* **282**, 243–282. (doi:10.1006/cogp.1998.0694)

21. Phillips S. 1998 Are feedforward and recurrent networks systematic? Analysis and implications for a connectionist cognitive architecture. *Connect. Sci.* **10**, 137–160. (doi:10.1080/095400998116549)

22. van der Velde F, van der Voort van der Kleij GT, de Kamps M. 2004 Lack of combinatorial productivity in language processing with simple recurrent networks. *Connect. Sci.* **16**, 21–46. (doi:10.1080/09540090310001656597)

23. Brakel P, Frank S. 2009 Strong systematicity in sentence processing by simple recurrent networks. In *Proc. 31st Annu. Meeting Cogn. Sci. Soc. (CogSci 2009), Amsterdam, The Netherlands, 29 July–1 August 2009* (eds N Taatgen, H van Rijn), pp. 1599–1604. Red Hook, NY: Curran Associates. See http://toc.proceedings.com/06199webtoc.pdf.

24. Murphy G. 2002 *The big book of concepts*. Cambridge, MA: MIT Press. (doi:10.7551/mitpress/1602.001.0001)

25. Hockett C. 1960 The origin of speech. *Scient. Am.* **203**, 88–111. (doi:10.1038/scientificamerican0960-88)

26. Goodfellow I, Bengio Y, Courville A. 2016 *Deep learning*. Cambridge, MA: MIT Press.

27. Graves A. 2012 *Supervised sequence labelling with recurrent neural networks*. Berlin, Germany: Springer. (doi:10.1007/978-3-642-24797-2)

28. Mikolov T. 2012 *Statistical language models based on neural networks*. PhD dissertation, Brno University of Technology.

29. Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I. 2019 Language models are unsupervised multitask learners. See https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

30. Landauer T, Dumais S. 1997 A solution to Plato's problem: the latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychol. Rev.* **104**, 211–240. (doi:10.1037/0033-295X.104.2.211)

31. Bar M. 2007 The proactive brain: using analogies and associations to generate predictions. *Trends Cogn. Sci.* **11**, 280–289. (doi:10.1016/j.tics.2007.05.005)

32. Levy R. 2008 Expectation-based syntactic comprehension. *Cognition* **106**, 1126–1177. (doi:10.1016/j.cognition.2007.05.006)

33. Pickering M, Garrod S. 2013 An integrated theory of language production and comprehension. *Behav. Brain Sci.* **36**, 329–347. (doi:10.1017/S0140525X12001495)

34. Clark A. 2016 *Surfing uncertainty*. Oxford, UK: Oxford University Press.

35. Elman J. 1990 Finding structure in time. *Cogn. Sci.* **14**, 179–211. (doi:10.1207/s15516709cog1402_1)

36. Hochreiter S, Schmidhuber J. 1997 Long short-term memory. *Neural Comput.* **9**, 1735–1780. (doi:10.1162/neco.1997.9.8.1735)

37. Chung J, Gulcehre C, Cho K, Bengio Y. 2014 Empirical evaluation of gated recurrent neural networks on sequence modeling. In *Proc. NIPS Deep Learning and Representation Learning Workshop, Montreal, Canada, 12 December 2014*, poster no. 47. See http://www.dlworkshop.org/accepted-papers.

38. Goldberg Y. 2017 *Neural network methods for natural language processing*. San Francisco, CA: Morgan & Claypool. (doi:10.2200/S00762ED1V01Y201703HLT037)

39. Sutskever I, Vinyals O, Le Q. 2014 Sequence to sequence learning with neural networks. *Adv. Neur. Inform. Process. Syst.* **27**, 3104–3112.

40. Dong L, Lapata M. 2016 Language to logical form with neural attention. In *Proc. 54th Annu. Meeting ACL, Berlin, Germany, August 2016*, vol. 1 (eds K Erk, NA Smith), pp. 33–43. Association for Computational Linguistics. (doi:10.18653/v1/P16-1004)

41. Mei H, Bansal M, Walter M. 2016 Listen, attend, and walk: neural mapping of navigational instructions to action sequences. In *Proc. 30th AAAI Conf. Artificial Intelligence and 28th Innovative Applications of Artificial Intelligence Conf., Phoenix, Arizona, 12–17 February 2016*. See https://openreview.net/forum?id=H1BLjgZCb. pp. 2772–

2778. See http://www.aaai.org/Conferences/AAAI/aaai16.php.

42. Bahdanau D, Cho K, Bengio Y. 2015 Neural machine translation by jointly learning to align and translate. *arχiv*, 1409.0473v7. See https://arxiv.org/pdf/1409.0473.pdf.

43. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A, Kaiser L, Polosukhin I. 2017 Attention is all you need. In *Adv. Neur. Inform. Process. Syst.* **30**, 5998–6008. See https://papers.nips.cc/book/advances-in-neural-information-processing-systems-30-2017.

44. Socher R, Perelygin A, Wu J, Chuang H, Manning C, Ng A, Potts C. 2013 Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. 2013 Conf. Empirical Methods in Natural Language Processing, Seattle, Washington, October 2013*, pp. 1631–1642. Association for Computational Linguistics. See https://www.aclweb.org/anthology/D13-1170/.

45. Dyer C, Kuncoro A, Ballesteros M, Smith N. 2016 Recurrent neural network grammars. In *Proc. Conf. NAACL: Human Language Technologies, San Diego, California, June 2016*, pp. 199–209. Association for Computational Linguistics. (doi:10.18653/v1/N16-1024)

46. Brennan JR, Martin AE. 2019 Phase synchronization varies systematically with linguistic structure composition. *Phil. Trans. R. Soc. B* **375**, 20190305. (doi:10.1098/rstb.2019.0305)

47. Williams A, Drozdov A, Bowman S. 2018 Do latent tree learning models identify meaningful structure in sentences? *Trans. Assoc. Comput. Linguist.* **6**, 253–267. (doi:10.1162/tacl_a_00019)

48. Edunov S, Ott M, Auli M, Grangier D. 2018 Understanding back-translation at scale. In *Proc. 2018 Conf. Empirical Methods in Natural Language Processing, Brussels, Belgium, November 2018* (eds E Riloff, D Chiang, J Hockenmaier, J Tsujii), pp. 489–500. Association for Computational Linguistics. (doi:10.18653/v1/D18-1045)

49. Baroni M. 2009 Distributions in text. In *Corpus linguistics: an international handbook*, vol. 2 (eds A Lüdeling, M Kytö), pp. 803–821. Berlin, Germany: Mouton de Gruyter.

50. Linzen T, Dupoux E, Goldberg Y. 2016 Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Trans. Assoc. Comput. Linguist.* **4**, 521–535. (doi:10.1162/tacl_a_00115)

51. Chowdhury S, Zamparelli R. 2018 RNN simulations of grammaticality judgments on long-distance dependencies. In *Proc. 27th Int. Conf. Computational Linguistics, Santa Fe, New Mexico, August 2018* (eds EM Bender, L Derczynski, P Isabelle), pp. 133–144. Association for Computational Linguistics. See https://www.aclweb.org/anthology/C18-1012.

52. Wilcox E, Levy R, Morita T, Futrell R. 2018 What do RNN language models learn about filler-gap dependencies? In *Proc. 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium, November 2018* (eds T Linzen, G Chrupała, A Alishahi), pp. 211–221. Association for Computational Linguistics. (doi:10.18653/v1/W18-5423)

53. Gulordava K, Bojanowski P, Grave E, Linzen T, Baroni M. 2018 Colorless green recurrent networks dream hierarchically. In *Proc. 2018 Conf. NAACL: Human Language Technologies, New Orleans, LA, June 2018*, vol. 1 (eds M Walker, H Ji, A Stent), pp. 1195–1205. Association for Computational Linguistics. (doi:10.18653/v1/N18-1108)

54. Kuncoro A, Dyer C, Hale J, Blunsom P. 2018 The perils of natural behavioral tests for unnatural models: the case of number agreement. Poster presented at Learning Language in Humans and in Machines, Paris, France, 5–6 July 2018. See https://osf.io/view/L2HM/.

55. Linzen T, Leonard B. 2018 Distinct patterns of syntactic agreement errors in recurrent networks and humans. In *Proc. CogSci. 2018, Austin, Texas*, pp. 692–697. See https://mindmodeling.org/cogsci2018/.

56. Lakretz Y, Kruszewski G, Desbordes T, Hupkes D, Dehaene S, Baroni M. 2019 The emergence of number and syntax units in LSTM language models. In *Proc. 2019 Conf. NAACL: Human Language Technologies, Minneapolis, Minnesota, June 2019*, vol. 1 (eds J Burstein, C Doran, T Solorio), pp. 11–20. Association for Computational Linguistics. (doi:10.18653/v1/N19-1002)

57. Lake B, Baroni M. 2018 Generalization without systematicity: on the compositional skills of sequence-to-sequence recurrent networks. In *Proc. 35th Int. Conf. Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018* (eds J Dy, A Krause), pp. 2879–2888. See http://proceedings.mlr.press/v80/lake18a.html.

58. Loula J, Baroni M, Lake B. 2018 Rearranging the familiar: testing compositional generalization in recurrent networks. In *Proc. 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium, November 2018* (eds T Linzen, G Chrupała, A Alishahi), pp. 108–114. Association for Computational Linguistics. (doi:10.18653/v1/W18-5413)

59. Wonnacott E, Newport E, Tanenhaus M. 2008 Acquiring and processing verb argument structure: Distributional learning in a miniature language. *Cognit. Psychol.* **56**, 165–209. (doi:10.1016/j.cogpsych.2007.04.002)

60. Bastings J, Baroni M, Weston J, Cho K, Kiela D. 2018 Jump to better conclusions: SCAN both left and right. In *Proc. 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium, November 2018* (eds T Linzen, G Chrupała, A Alishahi), pp. 47–55. Association for Computational Linguistics. (doi:10.18653/v1/W18-5407)

61. Dessì R, Baroni M. 2019 CNNs found to jump around more skillfully than RNNs: compositional generalization in seq2seq convolutional networks. In *Proc. 57th Annu. Meeting ACL, Firenze, Italy, July 2019* (eds A Korhonen, D Traum, L Màrquez), pp. 3919–3923. Association for Computational Linguistics. (doi:10.18653/v1/P19-1381)

62. Gehring J, Auli M, Grangier D, Yarats D, Dauphin Y. 2017 Convolutional sequence to sequence learning. In *Proc. Int. Conf. Machine Learning, Sydney, Australia, 6–11 August 2017* (eds D Precup, YW Teh), pp. 1243–1252. See http://proceedings.mlr.press/v70/gehring17a.html.

63. Andreas J. 2019 Measuring compositionality in representation learning. In *Proc. Int. Conf. Learning Representations, New Orleans, Louisiana, 6–9 May 2019*. See https://openreview.net/forum?id=HJz05o0qK7.

64. Andreas J, Rohrbach M, Darrell T, Klein D. 2016 Learning to compose neural networks for question answering. In *Proc. 2016 Conf. NAACL: Human Language Technologies, San Diego, California, June 2016* (eds K Knight, A Nenkova, O Rambow), pp. 1545–1554. Association for Computational Linguistics. (doi:10.18653/v1/N16-1181)

65. Belinkov Y, Bisk Y. 2018 Synthetic and natural noise both break neural machine translation. In *Proc. 6th Int. Conf. Learning Representations, Vancouver, Canada, 30 April–3 May 2018*. See https://openreview.net/forum?id=BJ8vJebC-.

66. Ebrahimi J, Lowd D, Dou D. 2018 On adversarial examples for character-level neural machine translation. In *Proc. 27th Int. Conf. Computational Linguistics, Santa Fe, New Mexico, August 2018* (eds EM Bender, L Derczynski, P Isabelle), pp. 653–663. See https://www.aclweb.org/anthology/C18-1055/.

67. Zhao Z, Dua D, Singh S. 2018 Generating natural adversarial examples. In *Proc. 6th Int. Conf. Learning Representations, Vancouver, Canada, 30 April–3 May 2018*. See https://openreview.net/group?id=ICLR.cc/2018/Conference.

68. Jia R, Liang P. 2017 Adversarial examples for evaluating reading comprehension systems. In *Proc. 2017 Conf. Empirical Methods in Natural Language Processing, Copenhagen, Denmark, September 2017* (eds M Palmer, R Hwa, S Riedel), pp. 2021–2031. Association for Computational Linguistics. (doi:10.18653/v1/D17-1215)

69. McCoy T, Pavlick E, Linzen T. 2019 Right for the wrong reasons: diagnosing syntactic heuristics in natural language inference. In *Proc. 57th Annu. Meeting ACL, Firenze, Italy, July 2019* (eds A Korhonen, D Traum, L Màrquez), pp. 3428–3448. Association for Computational Linguistics. (doi:10.18653/v1/P19-1334)

70. Albright A, Hayes B. 2003 Rules vs. analogy in English past tenses: a computational/experimental study. *Cognition* **90**, 119–161. (doi:10.1016/S0010-0277(03)00146-X)

71. Marelli M, Baroni M. 2015 Affixation in semantic space: modeling morpheme meanings with compositional distributional semantics. *Psychol. Rev.* **122**, 485–515. (doi:10.1037/a0039267)

72. Goldberg A, Jackendoff R. 2004 The English resultative as a family of constructions. *Language* **80**, 532–568. (doi:10.1353/lan.2004.0129)

73. Hopper P, Traugott E. 1990 *Grammaticalization*. Cambridge, UK: Cambridge University Press.