Assessing the Linguistic Productivity of Unsupervised Deep Neural Networks

Lawrence Phillips (Lawrence.Phillips@pnnl.gov)
Pacific Northwest National Laboratory

Nathan Hodas (Nathan.Hodas@pnnl.gov)
Pacific Northwest National Laboratory

Abstract

Increasingly, cognitive scientists have demonstrated interest in applying tools from deep learning. One use for deep learning is in language acquisition where it is useful to know if a linguistic phenomenon can be learned through domain-general means. To assess whether unsupervised deep learning is appropriate, we first pose a smaller question: Can unsupervised neural networks apply linguistic rules productively, using them in novel situations? We draw from the literature on determiner/noun productivity by training an unsupervised, autoencoder network measuring its ability to combine nouns with determiners. Our simple autoencoder creates combinations it has not previously encountered and produces a degree of overlap matching adults. While this preliminary work does not provide conclusive evidence for productivity, it warrants further investigation with more complex models. Further, this work helps lay the foundations for future collaboration between the deep learning and cognitive science communities.

Keywords: Deep Learning; Language Acquisition; Linguistic Productivity; Unsupervised Learning; Determiners

Introduction

Computational modeling has long played a significant role within cognitive science, allowing researchers to explore the implications of cognitive theories and to discover what properties are necessary to account for particular phenomena (J. L. McClelland, 2009). Over time, a variety of modeling traditions have seen their usage rise and fall. While the 1980s saw the rise in popularity of connectionism (Thomas & McClelland, 2008), more recently symbolic Bayesian models have risen to prominence (Chater & Oaksford, 2008; Lee, 2011). While the goals of cognitive modelers have largely remained the same, increases in computational power and architectures have played a role in these shifts (J. L. McClelland, 2009). Following this pattern, recent advances in the area of deep learning (DL) have led to a rise in interest from the cognitive science community as demonstrated by a number of recent workshops dedicated to DL (Saxe, 2014; J. McClelland, Hansen, & Saxe, 2016; J. McClelland, Frank, & Mirman, 2016).

As with any modeling technique, DL can be thought of as a tool which is best suited to answering particular types of questions. One such question is that of *learnability*, whether an output behavior could ever be learned from the types of input given to a learner. These types of questions play an integral role in the field of language acquisition where researchers have argued over whether particular aspects of language could ever be learned by a child without the use of innate, language-specific mechanisms (Smith, 1999; C. D. Yang, 2004; Chater & Christiansen, 2010; Pearl, 2014). The success of a domain general learner does not necessarily imply that human learners acquire the phenomenon in a similar fashion, but it does open the possibility that we need not posit innate, domain-specific knowledge.

The crux of these learning problems typically lies in making a particular generalization which goes beyond the input data. One major type of generalization that DL models would need to capture is known as *linguistic productivity*. A grammatical rule is considered productive when it can be applied in novel situations. For example, as a speaker of English you may never have encountered the phrase *a gavagai* before, but you now know that *gavagai* must be a noun and can therefore combine with other determiners to produce a phrase such as *the gavagai*. Before DL might be applied to larger questions within language acquisition, the issue of productivity must first be addressed. If DL models are not capable of productivity, then they cannot possibly serve to model the cognitive process of language acquisition. On the other hand, if DL models demonstrate basic linguistic productivity, we must explore what aspects of the models allow for this productivity.

The Special Case of Determiners

For decades, debate has raged regarding the status of productive rules among children acquiring their native language. On the one hand, some have argued that children seem hardwired to apply rules productively and demonstrate this in their earliest speech (Valian, Solt, & Stewart, 2009; C. Yang, 2011). On the other, researchers have argued that productivity appears to be learned, with children's early speech either lacking productivity entirely or increasing with age (Pine & Martindale, 1996; Pine, Freudenthal, Krajewski, & Gobet, 2013; Meylan, Frank, Roy, & Levy, 2017). Of particular interest to this debate has been the special case of English determiners. In question is whether or not English-learning children have acquired the specific linguistic rule which allows them to create a noun phrase (NP) from a determiner (DET) and noun (N) or if they have simply memorized the combinations that they have previously encountered. This linguistic rule, NP → DET N, is productive in two senses. First, it can be applied to novel nouns, e.g. *a gavagai*. Second, consider the determiners *a* and *the*. If a singular noun can combine with one of these determiners, it may also combine with the other, e.g. *the wug*.

This type of rule seems to be acquired quite early in acquisition, making it appropriate to questions of early productivity, and provides an easy benchmark for a DL model. Yet answering such a simple question first requires addressing how one might measure productivity. Most attempts to measure productivity have relied on what is known as an *overlap score*, intuitively what percentage of nouns occur with both *a* and *the* (C. Yang, 2011). This simple measure has been the source of some controversy. C. Yang (2011) argues that early attempts failed to take into account the way in which word frequencies affect the chance for a word to "overlap". Because word frequency follows a Zipfian distribution, with a long tail of many infrequent words, many nouns are unlikely to ever appear with both determiners. He proposes a method to calculate an expected level of overlap which takes into account these facts. Alternatively, Meylan et al. (2017) propose a Bayesian measure of productivity which they claim takes into account the fact that certain nouns tend to prefer one determiner over another. For instance, while one is more likely to hear *a bath* than the phrase *the bath*, the opposite is true of the noun *bathroom* which shows a preference for the determiner *the* (Meylan et al., 2017).

The literature is quite mixed regarding whether or not children show early productivity. Differences in pre-processing have lead researchers to draw opposite conclusions from similar data, making interpretation quite difficult (C. Yang, 2011; Pine et al., 2013). Indeed, most corpora involving individual children are small enough that Meylan et al. (2017) argue it is impossible to make a statistically significant claim as to child productivity. For analyzing whether or not text generated by a DL model is productive or not, we thankfully do not need to fully address the problem of inferring child productivity. Ideally, the model would demonstrate a similar level of overlap to the data it was exposed to. We make use of the overlap statistic from Yang because it is more easily comparable to other works and has been better studied than the more recent Bayesian metric of Meylan et al. (2017).

Deep Learning for Language Acquisition

Deep learning, or deep neural networks, are an extension of traditional artificial neural networks (ANN) used in connectionist architectures. A "shallow" ANN is one that posits a single hidden layer of neurons between the input and output layers. Deep networks incorporate multiple hidden layers allowing these networks in practice to learn more complex functions. The model parameters can be trained through the use of the backpropogation algorithm. The addition of multiple hidden layers opens up quite a number of possible architectures, not all of which are necessarily applicable to problems in cognitive science or language acquisition more specifically.

While the most common neural networks are discriminative, i.e. categorizing data into specific classes, a variety of techniques have been proposed to allow for truly generative neural networks. These generative networks are able to take in input data and generate complex outputs such as images or text which makes them ideal for modeling human behavior. We focus on one generative architecture in particular known as a deep *autoencoder* (AE) (Hinton & Salakhutdinov, 2006).

While AEs have been used for a variety of input data types, most prominently images, we describe their use here primarily for text. The first half, the *encoder*, takes in sentences and transforms them into a condensed representation. This condensed representation is small enough that the neural network cannot simply memorize each sentence and instead is forced to encode only the aspects of the sentence it believes to be most important. The second half, the *decoder*, learns to take this condensed representation and transform it back into the original sentence. Backpropogation is used to train model weights to reduce the loss between the original input and the reconstructed output. Although backpropagation is more typically applied to supervised learning problems, the process is in fact unsupervised because the model is only given input examples and is given no external feedback.

AEs have been shown to successfully capture text representations in areas such as paragraph generation (Li, Luong, & Jurafsky, 2015), part-of-speech induction (Vishnubhotla, Fernandez, & Ramabhadran, 2010), bilingual word representations (Chandar et al., 2014), and sentiment analysis (Socher, Pennington, Huang, Ng, & Manning, 2011), but have not been applied to modeling language acquisition. While any number of DL architectures could be used to model language acquisition, the differences between ANNs and actual neurons in the brain make any algorithmic claims difficult. Instead, DL models might be used to address computational-level questions, for instance regarding whether or not a piece of knowledge is learnable from the data encountered by children. Before this can be done, however, it remains to be seen whether DL models are even capable of creating productive representations. If they cannot, then they do not represent useful models of language acquisition. This work attempts to address this not by creating a model of how children acquire language, but by using methods from the psychological literature on productivity to assess the capability of DL to learn productive rules.

Methods

Corpora

To train our neural network, we make use of child-directed speech taken from multiple American-English corpora in the CHILDES database (MacWhinney, 2000). In particular, we make use of the CDS utterances in the Bloom 1970, Brent, Brown, Kuczaj, Providence, Sachs, and Suppes corpora (Bloom, 1970; Brent & Siskind, 2001; Brown, 1973; Kuczaj, 1977; Demuth & McCullough, 2009; Sachs, 1983; Suppes, 1974). The combined corpora contain almost 1 million utterances and span a wide age range, including speech directed to children as young as 6 months and as old as 5 years. Relevant information about the used corpora can be found in Table 1.

Because we are interested in seeing what the AE can learn from data similar to that encountered by children, we train the model only on child-*directed* utterances. These can be produced by any adult in the dataset, including parents and researchers. Although a comparison with child-produced text

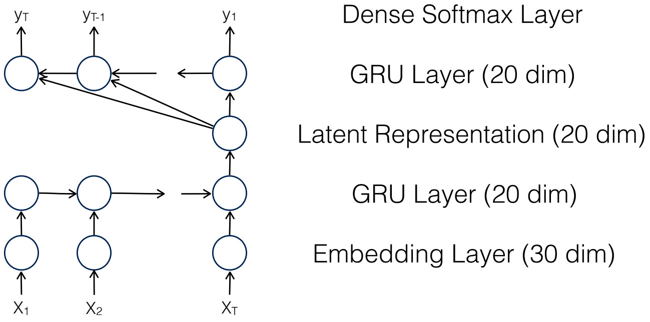

Figure 1: Visual representation of the autoencoder model.

holds great interest, it is not clear whether child-produced speech is rich enough to support robust language learning on its own. It therefore provides a poor basis upon which to train the AE.

Text from the various corpora is processed as a single document. Child-directed utterances are cleaned from the raw files using the CHILDESCorpusReader function of the Python Natural Language Toolkit (NLTK). Utterances from all non-children speakers are included and not limited just to the primary caregiver. Each utterance is split into words according to the available CHILDES transcription and then made lowercase. The model represents only the most frequent 3000 words, while the remainder are represented as a single *out-of-vocabulary* (OOV) token. This step is taken both to reduce computational complexity but also to mimic the fact that young children are unlikely to store detailed representations of all vocabulary items encountered. Because the neural networks require each input to be of the same length, sentences are padded to a maximum length of 10 words. Sentences that are longer than this are truncated, while short sentences are prepended with a special *PAD* token.

Corpora	Age Range	N. Utterances
Bloom 1970	1;9 - 3;2	62,756
Brent	0;6 - 1;0	142,639
Brown	1;6 - 5;1	176,856
Kuczaj	2;4 - 4;1	57,719
Providence	1;0 - 3;0	394,800
Sachs	1;1 - 5;1	28,200
Suppes	1;11 - 3;3	67,614
Overall	0;6 - 5;1	930,584

Table 1: Descriptive statistics of CHILDES corpora. Ages are given in (year;month) format and indicate the age of the child during corpus collection.

Neural Network Architecture

Our autoencoder model was implemented using Keras and Tensorflow. The words in each sentence are input to the model as a one-hot vector, a vector of 0s with a single 1 whose placement indicates the presence of a particular word. This is an inefficient representation because it assumes all words are equally similar, e.g. that *dog* is equally similar to *dogs* as it is to *truck*. To deal with this, the model passes the one-hot vector to an embedding layer. Neural word embeddings, as popularized by the word2vec algorithm (Mikolov, Chen, Corrado, & Dean, 2013), are a way to represent words in a low-dimensional space without requiring outside supervision. Words are placed within the space such that words that are predictive of neighboring words are placed closer to one another. Because our training data is relatively small, we keep the embedding dimensionality low, at only 30. Standard embeddings trained on much larger NLP corpora tend to use 100 or 200 dimensions.

Once each word has been transformed into a 30-dimensional embedding vector, the sequence of words is passed into a gated-recurrent unit (GRU) layer (Cho et al., 2014). The GRU is a type of recurrent (RNN) layer which we choose because it can be more easily trained. RNN layers read in their inputs sequentially and make use of hidden "memory" units that pass information about previous inputs to later inputs, making them ideal for sequence tasks such as language. As such, the model creates a representation of the sentence which it passes from word to word. The final representation is the output of the encoder, a latent representation of the full sentence.

This 20-dimensional latent vector serves as the input to the decoder unit. The first layer of the decoder is a GRU layer of the same shape as in the encoder. For each timestep, we feed into the GRU the latent vector, similar to the model proposed in Cho et al. (2014). Rather than producing a single output, as in the encoder, the decoder's GRU layer outputs a vector at each timestep. Each of these vectors is fed into a shared dense softmax layer which produces a probability distribution over vocabulary items. The model then outputs the most likely word for each timestep.

The model loss is calculated based on the model's ability to reconstruct the original sentence through categorical crossentropy. Model weights are trained using the Adam optimzer over 10 epochs. During each epoch the model sees the entire training corpus, updating its weights after seeing a batch of 64 utterances. While this process does not reflect that used by a child learner, it is a necessary component of training the neural network on such a small amount of data. If the network had access to the full set of speech that a child encounters such a measure likely would not be necessary. Future work might also investigate whether optimizing the dimensionality of the network might lead to better text generation with higher levels of productivity.

Baseline Models

Because the AE is learning to reproduce its input data, one might wonder whether similar results might be achieved by a simpler, distributional model. To assess this, we also measure the performance of an n-gram language model. We train bigram and trigram language models using the modified Kneser-Ney smoothing (Heafield, Pouzyrevsky, Clark,

& Koehn, 2013) implemented in the KenLM model toolkit to estimate the distributional statistics of the training corpus. Sentences are generated from the n-gram language model by picking a seed word and then sampling a new word from the set of possible n-grams. The smoothing process allows for the model to generate previously unseen n-grams. Sampling of new words continues for each utterance until the end-of-sentence token is generated or a maximum of 10 tokens is reached (the same maximum size as for the AE).

Since the AE is able to generate sentences from a latent representation, it would be inappropriate to generate n-gram sentences from random seed words. Instead, for every sentence in the test set we begin the n-gram model with the first word of the utterance. While this allows the model to always generate its first token correctly, this does not directly impact our measure of productivity as it relies on combinations of tokens.

Productivity Measures

We measure the productivity of our autoencoders through the overlap score described in C. Yang (2011). Words both in the child-directed corpus and the autoencoder-generated output are tagged using the default part-of-speech tagger from NLTK. The empirical overlap scores are simply calculated as a percentage of unique nouns that appear immediately after both the determiners *a* and *the*. The expected overlap score is calculated based off of three numbers from the corpus under consideration, the number of unique nouns N, the number of unique determiners D, and the total number of noun/determiner pairs S. The expected overlap is defined as in Equation 1:

$$O(N,D,S) = \frac{1}{N}\sum_{r=1}^{N} O(r,N,D,S) \tag{1}$$

where $O(r,N,D,S)$ is the expected overlap of the noun at frequency rank r:

$$O(r,N,D,S) = 1 + (D-1)(1-p_r)^S - \sum_{i=1}^{D}[(d_i p_r + 1 - p_r)^S] \tag{2}$$

d_i represents the probability of encountering determiner i, for which we use the relative frequencies of *a* and *the* calculated from the training corpus (39.3% and 60.7%, respectively). The probability p_r represents the probability assigned to a particular word rank. The Zipfian distribution takes a shape parameter, *a* which C. Yang (2011) set equal to 1 and which we optimize over the training corpus using least squares estimation and set at 1.06:

$$p_r = \frac{1/r^a}{\sum_{n=1}^{N}(\frac{1}{n^a})} \tag{3}$$

It should be noted that Zipfian distributions are not perfect models of word frequencies (Piantadosi, 2014), but assigning empirically-motivated values to the determiner probabilities and Zipfian parameter *a* represents an improvement upon the original measure.

Results

We analyze our overlap measures for the adult-generated (i.e. child-directed) as well as the autoencoder and n-gram model-generated text and present these results in Figure 2. We analyze overlap scores across 10 training epochs with three levels of dropout, 10%, 20%, and 30%. Dropout is typically included in neural models to encourage the model to better generalize. We hypothesized that a certain level of dropout would encourage the model to generate novel combinations of words that might lead to higher overlap scores. We find that with only two training epochs the AEs have already begun to near their maximum overlap performance. The 30% dropout AE achieves the highest level of performance, matching the empirical overlap score of the original corpus. The 10% and 20% dropout models perform somewhat worse suggesting that high levels of dropout may be necessary for good text generation.

In Table 2, we present the results for the final epoch of the AE models as well as for the adult-generated and n-gram generated text. We note that the expected overlap measure consistently overestimates the productivity of all learners, including the adult-generated text. It is unclear why this should be the case, but could be a result of capping the model vocabularies, resulting in lower N values. In particular, the autoencoders tend to produce a relatively limited set of nouns. Looking at empirical overlap measures, the worst-performing models are the bigram and trigram models with overlap scores below 30%. The AEs fair much better all producing overlap scores over 50%. The 30% dropout AE is actually able to match the overlap score of the original adult-generated corpus (59.4% vs. 59.3%).

Looking at the number of unique nouns following a determiner (N) and the total number of determiner-noun pairs (S), it becomes clear there are large differences between the n-gram and AE models. The n-gram models tend to produce very few determiner-noun pairs (low S) but are likely to choose from any of the nouns in the corpus, leading to high N. This fact accounts for the low overlap scores that they achieve. In contrast, the AEs follow a pattern which mirrors the adult corpus with few unique nouns but a large number of noun-determiner pairs. In all cases, however, the AEs produce both fewer unique nouns and fewer noun-determiner pairs than the original corpus.

One possible problem for calculating the expected overlaps comes from the difficulty of part-of-speech tagging text generated by the neural network. Whereas adult-generated speech follows set patterns that machine taggers are built to recognize, the neural network does not necessarily generate well-formed language. Examples of AE-generated text can be found in Table 3. In some cases, the tagger treats items that occur after a determiner as a noun regardless of its typical usage. For example, in the generated sentence *let put*

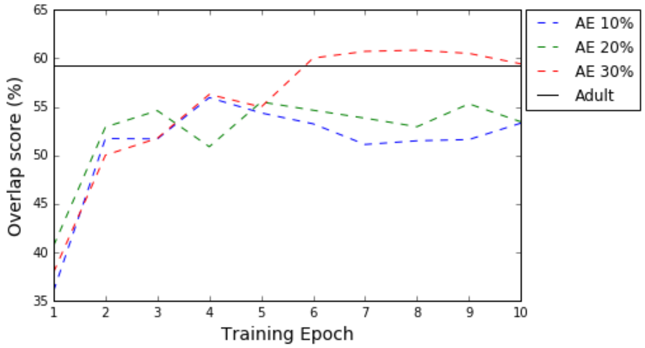

Figure 2: Empirical overlap scores. Adult-generated speech is marked by the solid black line while autoencoder-generated speech is marked by the dashed colored lines. Results are presented for three levels of dropout, 10%, 20%, and 30%. The x-axis represents the training epoch of the model.

	N	***S***	**Exp. Over.**	**Emp. Over.**
Adult	1,390	34,138	77.5%	59.3%
AE 10%	861	29,497	88.4%	53.3%
AE 20%	870	28,817	87.6%	53.4%
AE 30%	816	31,181	90.8%	59.4%
Bigram	1,780	5,177	17.6%	28.6%
Trigram	2,506	4,595	11.2%	22.1%

Table 2: Expected and empirical overlap scores for adult- and autoencoder-generated language with varying levels of dropout. Expected overlap scores were calculated as in Yang (2011). Empirical overlap was calculated as the percent of unique nouns that appeared immediately following both *a* and *the*.

put the over over here, the phrase *the over* is tagged as a DET+N pair. These type of errors are further evidenced by the fact that the trigram language model produces a larger set of words tagged as nouns than the original adult-generated corpus (2,506 vs. 1,390).

Another explanation for the difference between expected and empirical overlaps may come from deviation from a true Zipfian distribution of word frequencies. If word frequencies are Zipfian, we should expect a perfect correlation between log ranks and log counts. C. Yang (2011) report a correlation of 0.97, while our larger corpus deviates from this with $r^2 = 0.86$. Although we attempt to take this into account by fitting the Zipfian distribution's shape parameter, this divergence clearly indicates that further work is needed.

The success of the AE model in generating productive text serves as a confirmation that unsupervised neural models might be used in future work to investigate other cognitive phenomena. This work does not directly address the question of how infants might learn to produce productive speech, it does represent one possible approach. AEs can, for instance, be thought of as information compression algorithms which learn to represent high-dimensional data into a low-dimensional latent space (Hinton & Salakhutdinov, 2006). If the brain likewise attempts to find efficient representations of the stimuli it encounters then it may prove fruitful to investigate how these representations compare to one another.

Adult	**Autoencoder**
falling down	down down
you're playing with your bus	you're playing with the head
why did OOV say what's wrong with these apples	what what you say say say with **the dada**

Table 3: Example adult and AE-generated language. The AE-generated text is from the final epoch of the AE with 20% dropout. In bold is a DET+N combination that does not appear in the AEs input.

Conclusion

While there is great interest regarding the inclusion of deep learning methods into cognitive modeling, a number of major hurdles remain. For the area of language acquisition, deep learning is poised to help answer questions regarding the learnability of complex linguistic phenomena without access to innate, linguistic knowledge. Yet it remains unclear whether unsupervised versions of deep learning models are capable of capturing even simple linguistic phenomena. In this preliminary study, we find that a simple autoencoder with sufficient levels of dropout is able to mirror the productivity of its training data, although it is unclear whether this proves productivity in and of itself.

Future work will need to investigate whether more complex models might be able to generate text with higher productivity as well as further investigating how particular model choices impact performance. It would also be worthwhile to compare AEs against simpler models such as a basic LSTM language model. While additional work needs to be done to motivate the use of deep learning models as representations of how children might learn, this preliminary work shows how one might combine techniques from deep learning and developmental psychology.

Acknowledgments

The authors thank the reviewers for their thoughtful comments and Lisa Pearl for initial discussion regarding productivity.

References

Bloom, L. (1970). *Language development: Form and function in emerging grammars.* MIT Press.

Brent, M., & Siskind, J. (2001). The role of exposure to isolated words in early vocabulary development. *Cognition*, *81*, 31–44.

Brown, R. (1973). *A first language: The early stages*. Harvard University Press.

Chandar, S., Lauly, S., Larochelle, H., Khapra, M., Ravindran, B., Raykar, V. C., & Saha, A. (2014). An autoencoder approach to learning bilingual word representations. In *Advances in Neural Information Processing Systems* (pp. 1853–1861).

Chater, N., & Christiansen, M. H. (2010). Language acquisition meets language evolution. *Cognitive Science*, *34*(7), 1131–1157.

Chater, N., & Oaksford, M. (2008). *The probabilistic mind: Prospects for bayesian cognitive science*. Oxford University Press.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Demuth, K., & McCullough, E. (2009). The prosodic (re)organization of children's early english articles. *Journal of Child Language*, *36*, 173–200.

Heafield, K., Pouzyrevsky, I., Clark, J. H., & Koehn, P. (2013). Scalable modified kneser-ney language model estimation. In *Proceedings of the Association of Computational Linguistics conference* (pp. 690–696).

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*(5786), 504–507.

Kuczaj, S. (1977). The acquisition of regular and irregular past tense forms. *Journal of Verbal Learning and Verbal Behavior*, *16*, 589–600.

Lee, M. D. (2011). How cognitive modeling can benefit from hierarchical bayesian models. *Journal of Mathematical Psychology*, *55*(1), 1–7.

Li, J., Luong, M.-T., & Jurafsky, D. (2015). A hierarchical neural autoencoder for paragraphs and documents. *arXiv preprint arXiv:1506.01057*.

MacWhinney, B. (2000). *The CHILDES project: The database* (Vol. 2). Psychology Press.

McClelland, J., Frank, S., & Mirman, D. (Eds.). (2016). *Contemporary neural network models: Machine learning, artificial intelligence, and cognition.*

McClelland, J., Hansen, S., & Saxe, A. (Eds.). (2016). *Tutorial workshop on contemporary deep neural network models.*

McClelland, J. L. (2009). The place of modeling in cognitive science. *Topics in Cognitive Science*, *1*(1), 11–38.

Meylan, S. C., Frank, M. C., Roy, B. C., & Levy, R. (2017). The emergence of an abstract grammatical category in children's early speech. *Psychological Science*, 0956797616677753.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Pearl, L. (2014). Evaluating learning-strategy components: Being fair (commentary on Ambridge, Pine, and Lieven). *Language*, *90*(3), e107–e114.

Piantadosi, S. (2014). Zipf's word frequency law in natural language: A critical review and future directions. *Psychonomic Bulletin & Review*, *21*(5), 1112-1130.

Pine, J. M., Freudenthal, D., Krajewski, G., & Gobet, F. (2013). Do young children have adult-like syntactic categories? Zipf's law and the case of the determiner. *Cognition*, *127*(3), 345–360.

Pine, J. M., & Martindale, H. (1996). Syntactic categories in the speech of young children: The case of the determiner. *Journal of Child Language*, *23*(02), 369–395.

Sachs, J. (1983). Talking about the there and then: The emergence of displaced reference in parent-child discourse. In K. Nelson (Ed.), *Children's language* (Vol. 4). Lawrence Erlbaum Associates.

Saxe, A. (Ed.). (2014). *Workshop on deep learning and the brain.*

Smith, L. B. (1999). Do infants possess innate knowledge structures? The con side. *Developmental Science*, *2*(2), 133–144.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., & Manning, C. D. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 151–161).

Suppes, P. (1974). The semantics of children's language. *American Psychologist*, *29*, 103–114.

Thomas, M. S., & McClelland, J. L. (2008). Connectionist models of cognition. *Cambridge handbook of computational cognitive modelling*, 23–58.

Valian, V., Solt, S., & Stewart, J. (2009). Abstract categories or limited-scope formulae? The case of children's determiners. *Journal of Child Language*, *36*(04), 743–778.

Vishnubhotla, S., Fernandez, R., & Ramabhadran, B. (2010). An autoencoder neural-network based low-dimensionality approach to excitation modeling for HMM-based text-to-speech. In *IEEE International Conference on Acoustics Speech and Signal Processing* (pp. 4614–4617).

Yang, C. (2011). A statistical test for grammar. In *Proceedings of the 2nd workshop on Cognitive Modeling and Computational Linguistics* (pp. 30–38).

Yang, C. D. (2004). Universal grammar, statistics or both? *Trends in Cognitive Sciences*, *8*(10), 451–456.