2017-06-12

# Back to the Future: Logic and Machine Learning

Simon Dobnik
*CLASP, University of Gothenburg, Sweden*

John D. Kelleher
*Technological University Dublin*, john.d.kelleher@tudublin.ie

## Recommended Citation

Dobnik, S. & Kelleher, J. (2017). Back to the future: logic and machine learning. *LaML:Conference on Logic and Machine Learning in Natural Language* , Gothenburg, Sweden, 12th-14th June, 2017. doi:10.21427/D7041Z

Funder: ADAPT Research Centre

# Back to the Future: Logic and Machine Learning

**Simon Dobnik**
FLOV & CLASP
University of Gothenburg, Sweden
simon.dobnik@gu.se

**John D. Kelleher**[*]
School of Computing
Dublin Institute of Technology, Ireland
john.d.kelleher@dit.ie

## Abstract

In this paper we argue that since the beginning of the natural language processing or computational linguistics there has been a strong connection between logic and machine learning. First of all, there is something logical about language or linguistic about logic. Secondly, we argue that rather than distinguishing between logic and machine learning, a more useful distinction is between top-down approaches and data-driven approaches. Examining some recent approaches in deep learning we argue that they incorporate both properties and this is the reason for their very successful adoption to solve several problems within language technology.

## 1 Introduction

At a surface level, logic and machine learning represent two distinct methodologies for analysing (or building models of) the world. Logic based theories can be characterised as qualitative, symbolic and driven by domain theory,[1] whereas machine learning may be characterised as quantitative, numeric and driven by computational learning theory. The focus of this paper is to examine and frame the (potentially synergistic) relationship between these distinct analytic methods for natural language processing (NLP). In historic terms this discussion is a recurrent throughout the history of NLP, for example in the mid-1990s the rise of statistical learning methods in NLP inspired several discussions on this topic (e.g., (Gazdar, 1996; Jones et al., 2000)). However, the dramatic recent advances in the NLP based on deep neural network approaches have made the question of how these two methodologies should be used/related/integrated in NLP research apposite.

A first step in such a study is to understand the goal of the task for which the methods are being used. NLP can loosely be defined as the field of research that studies how computers can best be used to process natural language. This definition is generic enough to be generally acceptable but lacks clarity in terms of what is the goal of this processing activity.

One goal for processing natural language is to develop useful applications that help humans in their daily life, e.g.: machine translation, and speech transcription. In application scenarios where a rough (shallow) analysis is useful (e.g., in some situations, even a rough translation that provides the gist of the message can be helpful) and large (annotated/structured) corpora are available machine learning is the ideal methodology to address this goal. However, where a deeper or precise analysis is required or where there is a scarcity of data a pure machine learning approach may not be suitable. Furthermore, if the goal of processing language is rather motivated by the desire to better understand the cognitive foundations of natural language than a machine learning methodology, particularly one based on an unconstrained (e.g. fully connected) deep neural network, are not appropriate. The criticisms of unconstrained neural network based models (typically characterised by fully-connected feed-forward multi-layer networks) in cognitive science has a long history (see (Massaro, 1988) *inter alia*) and often focuses on: (1) the difficultly in analysing in a domain theoretic sense how the model works, and (2) the, somewhat ironic scientific short-coming, that neural networks are such powerful and general learning mechanisms that demonstrating the ability of a network to learn a particular mapping/function is

[*] Both authors contributed equally.

[1] In this way we use the term "logic" in a loose sense, not only to refer to formal logics.

scientifically useless from a cognitive science perspective. In particular, as Massaro (1988) argues, a neural network model is so adaptable that—given the appropriate dataset and sufficient time and computing power—it is likely to be able to learn mappings that not only support a cognitive theory but also ones that contradict that theory. One approach to addressing this problem is to introduce domain relevant structural constraints into the model via the network architecture, early examples of this approach include (Feldman et al., 1988; Feldman, 1989; Regier, 1996). Indeed, we argue in this paper that one of the important (and somewhat overlooked) factors driving the success of DL research is the specificity of DL architectures to the tasks they are applied too.

**Contribution:** In this paper we evaluate the relation between logic and machine learning and argue that although it appears that logic has lost its significance in computational models and applications it is still very much present in the form of formal language modelling that underlines most of the current work with machine learning. Moreover, we highlight that many of the recent advances in deep learning for NLP are not based on unconstrained neural networks but rather that these networks have domain/task specific architectures that encode domain theoretic considerations. In this light, the relationship between logic and machine learning can be viewed as potentially more synergistic. Given that many logical theories are defined in terms of *functions* and *compositional* operations and neural networks learn and compose functions, a logic based domain theory of linguistic performance can naturally inform the structural design of deep learning architectures and thereby have benefits both in terms of model interpretability and performance.

**Overview:** In Section 2 we review the historic context of the logic versus machine learning for NLP debate; next, in Section 3, we concentrate on recent developments in NLP (e.g., deep learning approaches) and situate these developments within the broader debate; then, in Section 4, we use the computational modelling of spatial language as an NLP case study to frame the possible synergies between logic and machine learning; finally, in Section 5 we set out our thoughts for potential approaches to developing a more synergistic understanding of the logic and machine learning for NLP research.

## 2 A brief history of logic and machine learning

The groundwork for the application of logical techniques in the computational linguistics has been set by (Montague, 1974) with his description of English as a formal language. This introduced first-order logic, lambda calculus, model building and theorem proving to linguistics and later with the development of computational approaches to NLP. The application of logic also coincided with the view of language being a formal system driven by rules (Chomsky, 1968). However, this is not the only view: there existed a view of examining linguistic data from which one could extract generalisations which developed in the work following (Firth, 1957) and (Harris, 1954) who mark the beginning of distributional semantics. The work in computational linguistics took off with the development of computers in the late 1980s and 1990s with development of several approaches based on formal rules (Shieber, 1986; Alshawi, 1992) as well as well as approaches to semantics (Blackburn and Bos, 2005; Copestake et al., 2005). In parallel, there has been also development of machine learning which in several respects also involves the development on learning formal rules from data (Mitchell, 1997), in many respects similar to the rules used in modelling language. An interesting and promising approach was Inductive Logic Programming (ILP) (Muggleton, 1991) that learned specialised or generalised rules in a subset of first-order logic (Prolog clauses) from positive and negative examples, also in a subset of first-order logic. Attempts were made to use this framework to learn the missing grammar rules from a set of existing rules and linguistic data (Pulman and Liakata, 2003; Liakata and Pulman, 2004; Kazakov and Dobnik, 2003). ILP worked well on small and well-defined domains. However, working on real data introduced inconsistencies that could not be captured in a pure logical way and therefore the approach has been extended with non-logical techniques that better captured the variation in data (cf. (Liakata and Pulman, 2004)).

With the availability of large corpora in early 2000s (Manning and Schütze, 1999) (possibly related with the expansion of the internet where large amounts of regular everyday language has become available in computer-readable form) there has been a shift in focus from designing rules that generate representations to inducing such rules from datasets (Turney et al., 2010), thus from formalism to the processes, and hence machine learning has become the focus of the field. However, both approaches were somehow in a complimentary distribution as shown in Table 1:[2] Symbolic, rule-based approaches provided deep coverage but of a limited domain; outside the domain they proved brittle and therefore limited. On the other hand, data-based approaches were wide-coverage and robust to variation but provided shallow representation of language.

| *tech/cov* | wide | narrow |
|---|---|---|
| deep | **our goal** | symbolic |
| shallow | data-based | **useless** |

Table 1: Properties of rule-based and data-based approaches to NLP

Our desiderata is a wide-coverage system with deep analyses and it was considered that this could be achieved by a hybrid system but this was not an easy undertaking (cf. (Gazdar, 1996) and (Jones et al., 2000)). The work on ML and language from data between 2000–2010 has exceeded expectations and it has become progressively deeper (learning probabilistic hierarchical language models, for example (Tenenbaum et al., 2011) for a general probabilistic approach to cognition), but a few problems remain which are linked to the "logical" nature of language and include interpretation of quantification, negation, different kinds of semantic modifications under compositionality of expressions and others. Another difficulty is that the learned theories are not interpretable. ML methods learn only one of the possible theories that covers the data, not necessarily the one that would correspond to human intuitions and therefore as stated earlier their applicability for cognitive modelling has been considered limited. Overall, it follows that rule-based and logic based systems are not opposing but they are different approaches to modelling language: top down vs. bottom up.

## 3 Deep Learning and Handcrafted Network Architectures

In recent years deep learning (DL) models have improved (and in some cases markedly improved) the state of the art across a range of NLP tasks. Some of the drivers of DL success include: the availability of large datasets, more powerful computers, and the powerful learning and adaptability of connectionist neural networks. However, another and less obvious driver of DL is the fact that DL network models often have architectures that are specifically tailored or structured to the needs of a specific domain or task. This fact becomes more obvious when one considers the variety of DL architectures that are currently being researched (see Figure 1 for some examples).

This diversity of network architectures is not a given. For example, given the flexibility of neural networks one approach to accommodating structure into the processing of a network is to apply minimal constraints on the architecture and to rely on the ability of the learning algorithm to induce (and encode) the relevant structure constraints by adjusting the network's weights. However, it has long been known that pre-structuring a neural network by the careful design of its architecture to fit the requirements of the task results in better generalisation of the model beyond the training dataset (LeCun and others, 1989). Understood in this context, DL is assisted (dare we say supervised) by the task designer which decides what kind of networks they are going to build, the number of layers, the connectivity of the layers and other parameters. DL is not using fully connected layers, instead it developed several kinds of layered networks tailored to the task. In this respect it captured top-down specification that we have seen with the logic/rule-based systems.

The designer of the learning task brings significant background knowledge to learning: for example if language models are to be learned then the system should capture sequence learning and RNNs (LTSMs and GRUs) will be used. The inputs (and outputs) to such networks can be either characters of words, the latter represented as word embeddings in vector spaces. ConvNet have their origin in image processing where convolutions are meant as filters that encode a region of pixels into a single neural unit. Additionally, to decrease the effects of spatial continuum, operations such as pooling are used that encode convolved representations from various parts of the image. ConvNets are also used for language processing to capture different patterns of words or strings. The size of the network and the depth of the layers, the sizes of the matrices passed between the layers, activation

[2] Adapted from the slides of Stephen Pulman.

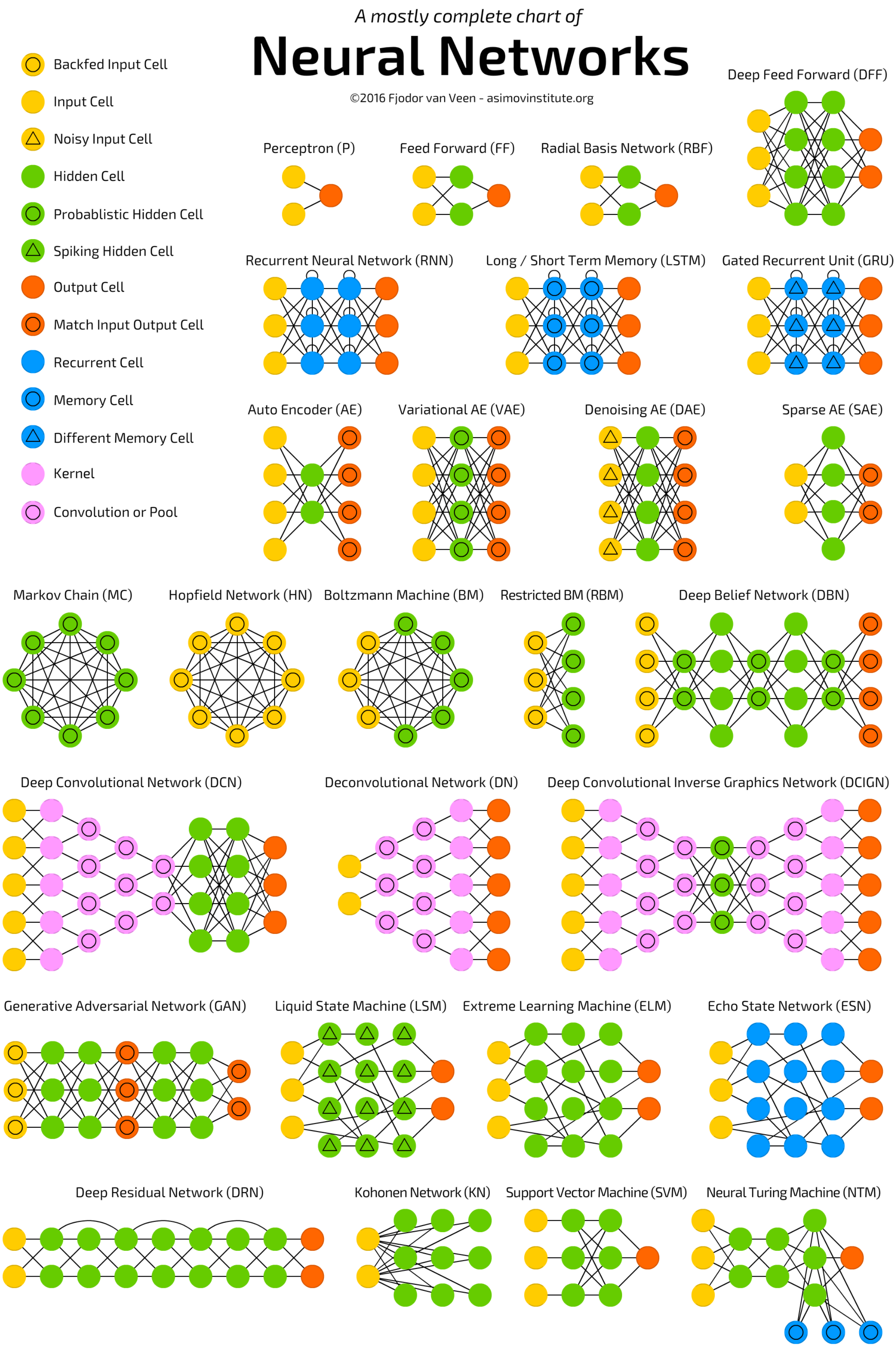
A mostly complete chart of
Neural Networks
©2016 Fjodor van Veen - asimovinstitute.org
Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Radial Basis Network (RBF)
Deep Feed Forward (DFF)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Gated Recurrent Unit (GRU)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)
Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)
Deep Residual Network (DRN)
Kohonen Network (KN)
Support Vector Machine (SVM)
Neural Turing Machine (NTM)

Figure 1: An illustration of some popular neural network architectures. Image sourced from (van Veen, 2016).

functions and optimiser are also relevant parameters that appear highly task dependent and are normally determined through an empirical trial-and-error process that is informed by designer intuition (Jozefowicz et al., 2016).

Another design aspect of DL architectures is the treatment of DL networks themselves as modular components within larger DL networks. In these modular DL architecture networks may be sequenced or stacked on top (defining compositional operations) of each other. For example, currently the standard Neural Machine Translation architecture is the encoder-decoder architecture (e.g., (Bahdanau et al., 2015; Luong et al., 2015)). This architecture uses one RNN (known as the encoder) to fully process the input sentence and generate a vector based representation of this sentence that is then passed to a second RNN (the decoder, essentially implementing a language model in the target language) that generates the translation word by word. Domain theoretic considerations have affected the design and development of this architecture in a number of ways. For example an understanding that enabling the decoder to look both back and forward along the input sentence during translation is one of the reasons why the input is fully processed prior to translation beginning. However, the understanding of the need for local alignments between different sections of the translation (and somewhat contrary requirement to the need for a potentially global perspective on the input) has resulted in the development of attention mechanisms within the NMT framework. A variant of the NMT encoder-decoder architecture that replaces the encoder RNN with a ConvNet has revolutionised the field of image captioning research (e.g., (Xu et al., 2015)). It is noteworthy, however, that sometimes the design of the network architecture constrain what the model can learn in undesirable ways. For example, Kelleher (2016) argues that these image captioning networks have been configured in a way that they capture visual properties of objects rather than relations between them. Consequently, within the captions generated by these systems the relation between the preposition and the object is not grounded in space but only in the linguistic sequences through the decoder language model where the co-occurrence of particular words in a sequence is estimated. Another dimension in DL network architecture design is to stack (as opposed to sequences) network modules. A fundamental NLP task where module DL design has result in significant breakthroughs is in language models. For example, the language model proposed by (Models, 2016) uses a ConvNet to generate word representations, that factor the international (character level) structure of the word, which are then passed to a RNN model to predict the next word in the context of preceding words in the sequence. This example illustrates how the DL architecture design can guide the network to process and integrate different levels of linguistic information. In summary, the design of a DL architectures, where DL networks are treated as composable modules, can constrain (and guide) a number of factors that are important to NLP, in particular the (hierarchical) composition of features and the sequencing of the processing.

## 4 Spatial language

Our focus is computational modelling of spatial language such as ("the chair is to the left and close to the table" or "go down the corridor until the large painting on your right, then turn left") which requires integration of different sources of knowledge that affect the semantics of spatial descriptions: scene geometry, knowledge about dynamic kinematic routines of objects, and language coordination with dialogue partners. Furthermore, because situated agents are located within dynamic linguistic and perceptual environments they need to continuously adapt their understanding and representations of the environment.

It follows that an appropriate computational model of spatial language should consist of several connected modalities (for which individual neural network architectures would be specified) but also of a general network that connects these modalities, thus akin to the specialised regions and their interconnections in the brain (Roelofs, 2014). Furthermore, two areas in DNN research that are particular relevant for such stratified modelling of DNNs are *active learning* (Olsson, 2009) and *transfer learning* (Pan and Yang, 2010). In an active learning framework the learning algorithm is able to query an oracle for labels on examples chosen by the algorithm and incrementally improving its understanding of the concept it is learning through interaction (in language). *Transfer learning* describes learning methods that can transfer knowledge learned in one task to improve learning on a new (but related) task. A common thread to both of these approaches is that they involve supervision in the form of the design of the network and information transfer from other modalities.

## 5 Conclusion and future research

DNNs provide a platform for machine learning that allow us a great flexibility in combining top-down specification (in terms of hand-designed structures and rules) and data driven approaches. This way we can tailor the learning algorithm to each individual learning problem and therefore effectively reach the goal of combining symbolic and data-driven approaches: a problem that has been investigated in NLP for several decades. The strength of DNNs is their compositionality of perceptrons/neural units which represent individual classification functions that can be combined in novel ways. This was not possible with other approaches in ML that worked more as black boxes. Finally, it is important to note that DNNs take inspiration from neural connections in human brain and hence at some abstract level share similarities with models of human cognition that NLP models are trying to capture at least at the level of output forms. Relating and understanding the performance of DNNs to models of language and cognition in general provides an interesting research question for the future.

## References

Hiyan Alshawi. 1992. *The Core Language Engine*. ACL-MIT Press series in natural language processing. MIT Press, Cambridge, Mass.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*, volume abs/1409.0473v6.

Patrick Blackburn and Johan Bos. 2005. *Representation and inference for natural language. A first course in computational semantics*. CSLI Publications.

Noam Chomsky. 1968. *Syntactic structures*, volume 4. Mouton, The Hague.

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A. Sag. 2005. Minimal recursion semantics: An introduction. *Research on Language and Computation*, 3(2-3):281–332.

J. A. Feldman, M. A. Fanty, and N. H. Goodard. 1988. Computing with structured neural networks. *Computer*, 21(3):91–103, March.

Jerome A. Feldman. 1989. Structured neural networks in nature and in computer science. In Rolf Eckmiller and Christoph v.d. Malsburg, editors, *Neural Computers*, pages 17–21. Springer, Berlin, Heidelberg.

John R. Firth. 1957. A synopsis of linguistic theory 1930–1955. *Studies in linguistic analysis*, pages 1–32.

Gerald Gazdar. 1996. Paradigm merger in natural language processing. In Ian Wand and Robin Milner, editors, *Computing Tomorrow*, pages 88–109. Cambridge University Press, New York, NY, USA.

Zellig S. Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Karen I. B. Spärck Jones, Gerald J. M. Gazdar, and Roger M. Needham. 2000. Introduction: combining formal theories and statistical data in natural language processing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 358(1769):1227–1238.

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*.

Dimitar Kazakov and Simon Dobnik. 2003. Inductive learning of lexical semantics with typed unification grammars. In Esther Grabe and David G. S. Wright, editors, *Topics in Phonetics and Computational Linguistics*, volume 8 of *Oxford Working Papers in Linguistics, Philology and Phonetics*, pages 113–133. Committee for General Linguistics and Comparative Philology, Oxford, United Kingdom.

John D. Kelleher. 2016. What is not where: the challenge of integrating spatial representations into deep learning architectures. In *Presented at the* $1^{st}$ *International Workshop on Models and Representations in Spatial Cognition*, Hanse Wissenschaftskolleg Delmenhorst, March.

Yann LeCun et al. 1989. Generalization and network design strategies. *Connectionism in perspective*, pages 143–155.

Maria Liakata and Stephen Pulman. 2004. Learning theories from text. In *Proceedings of Coling 2004*, pages 183–190, Geneva, Switzerland, Aug 23–Aug 27. COLING.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of statistical natural language processing*. The MIT Press.

Dominic Massaro. 1988. Some criticisms of connectionist models of human performance. *Journal of Memory and Language*, 27:213–234.

Tom M. Mitchell. 1997. *Machine learning*. McGraw-Hill Series in Computer Science. McGraw-Hill.

Character-Aware Neural Language Models. 2016. Yoon kim and yacine jernite and david sontag and alexander m. rush. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*.

Richard Montague. 1974. *Formal Philosophy: Selected Papers of Richard Montague*. Yale University Press, New Haven. ed. and with an introduction by Richmond H. Thomason.

Stephen Muggleton. 1991. Inductive logic programming. *New Generation Computing*, 8:295–318. 10.1007/BF03037089.

Fredrik Olsson. 2009. A literature survey of active machine learning in the context of natural language processing. Technical Report T2009:06, Swedish Institute of Computer Science.

Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Stephen G. Pulman and Maria Liakata. 2003. Learning domain theories. In Nicolas Nicolov, Kalina Bontcheva, Galia Angelova, and Ruslan Mitkov, editors, *RANLP*, volume 260 of *Current Issues in Linguistic Theory (CILT)*, pages 29–44. John Benjamins, Amsterdam/Philadelphia.

Terry Regier. 1996. *The human semantic potential: Spatial language and constrained connectionism*. MIT Press.

Ardi Roelofs. 2014. A dorsal-pathway account of aphasic language production: The weaver++/arc model. *Cortex*, 59:33–48.

Stuart Shieber. 1986. *An Introduction to Unification-Based Approaches to Grammar*. CSLI Publications, Stanford.

Joshua B. Tenenbaum, Charles Kemp, Thomas L. Griffiths, and Noah D. Goodman. 2011. How to grow a mind: Statistics, structure, and abstraction. *Science*, 331(6022):1279–1285.

Peter D Turney, Patrick Pantel, et al. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37(1):141–188.

Fjodor van Veen. 2016. The neural network ZOO. The Asimov Institute Blog, http://www.asimovinstitute.org/neural-network-zoo, September 14.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*.