Deep Learning: Approaches and Challenges

Ahmad Akl[1], Ahmed Moustafa[2], Ibrahim El-Henawy[3]
[1,2,3]*Department of Computer Sciences, faculty of Computers and Informatics, Zagazig University, El-ZeraaSquare, Zagazig, Sharqiyah, Egypt, Postal code: 44519*

Abstract
Deep learning(DL) has gained increasing research interests since Hinton propose a fast learning algorithm in 2006 [11], because of its potential capability to outperform the drawbacks of traditional techniques that depend on engineering-based features in many fields such as computer vision, pattern recognition, speech recognition, natural language processing, and recommendation systems. In this paper, a brief review of various deep learning architectures, and then bierfly describe their challenges such as training data shortage. In addition, Some trends of deep learning optimization using traditional machine learning models are described such as K-nearest neighbor. Finally, a brief section for the most used deep learning toolkits and libraries.

Keywords *deep learning, architectures, challenges, new trends, libraries, and toolkits.*

I Introduction

Deep learning, also referred to as representation learning [2], is a subfield of machine learning, capable of learning a high level of abstract representations, which are created over multiple layers. In every layer, more abstract view of the input image is obtained. The basic idea depends on a series of transformations on the input image. On each transformation, we learn abstract representations of the image. We then take this representation, apply the transformation and learn more abstract representations. Using this procedure, we learn more and more abstract feature that represents the original image. The study on the artificial neural networks (ANN) has a great impact on originated the concept of deep learning [14]. Training standard neural networks (NN) is very expensive and takes a lot of computational stages. Backpropagation is used to train ANN since 1980 in a teacher-based supervised learning approach. The algorithm faced the problem of overfitting, as the training accuracy high, but when the algorithm is applied to test data the accuracy might be satisfactory. A new training method proposed by Hinton in 2006 called layer-wised-greeding-learning that marked the birth of deep learning techniques [13]. In the past decade, deep learning techniques have been developed with significant impacts on information and signal processing. The research in deep learning drew attention and a series of exciting results have been reported. Both Baidu and Google have updated their image search engines based on Hinton's deep learning architectures. In 2016, Google AI player (AlphaGo), that adopts deep learning techniques, beat one of the world's strongest player Lee se-dol by 4:1.

It has been applied and obtained great results in various challenges problems such as speech recognition, image classification, natural language processing, computer vision, recommendation systems, semantic parsing and more. It falls in four architectures namely, autoencoder(AE), convolutional neural networks(CNN), deep belief networks(DBN), restricted Boltzmann machine(RBM).

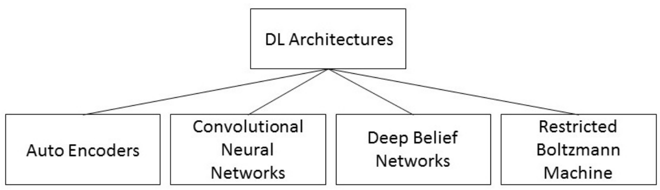

Figure 1: Deep Learning Architectures

Since 2006, a new area of machine learning research has emerged. There are three important factors for the advances in deep learning today: data availability (e.g. millions of photos uploaded to Facebook daily), increased chips processing abilities (e.g. GPU units), advances in machine learning algorithms, these factors make traditional tasks more interesting. Deep learning makes use of data, the more data the more performance. This paper [24] is widely regarded as one of the most influential publications in the field. Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton created a "large, deep convolutional neural network" that won the 2012 ILSVRC (ImageNet Large-Scale Visual Recognition Challenge)[41]. Researchers have been widely adopted deep learning architectures to achieve top accuracy scores [1].

II Deep Learning Architectures

In the next four sections, we will breifly review the four deep learning architectures.

A *Autoencoders*

Autoencoder is an unsupervised learning algorithm that was introduced by Hinton [40] to address the problem of backpropagation without a teacher. It is used to efficiently code the dataset for dimensionality reduction purposes [14][17][44]. Autoencoder is trained to reconstruct its own inputs X so that the output vector has the same dimensionality as the input vector. The autoencoder network can be viewed as two parts: an encoder function $h = f(x)$ where the AE converts the input x into a hidden representation using a weight matrix w; and a decoder that produced a reconstruction $r = g(h)$ where the AE map h back to the original format to obtain x with another weight matrix. There are three important variants of autoencoders: sparse autoencoder (SAE), denoising autoencoder (DAE) and contractive autoencoder (CAE).

In sparse autoencoder, sparse features are extracted from raw data, as the sparsity of the representation can be achieved by penalizing the hidden bias units [38][30][8], or by penalizing the output hidden unit activations [27][50]. Denoising autoencoder aims to recover the corrent input from a corrupted version, and this makes the model able to capture the structure of the input distribution [47][48]. Contractive autoencoder, proposed by Rifai et al.[39], it achieves robustness by adding a penalty term to the cost function in the reconstruction stage.

Table 1: AE Applications

Application	Method
Data Denoising	is the ability to map noisy data into its clean data version. a denoising autoencoder is used for data denoising such as images [47].
Dimensionality Reduction	is the process of converting high-dimensional data to low-dimensional representation, especially when the data are correlated and redundant. Dimensionality Reduction is useful for data visualization. Hinton introduced a model that is able to reduce dimensions better than the classic PCA [14].
Object Detection	is the ability to detect some specific-class features. In this [26]an autoencoder is used as a face detector using only unlabeled data
Word Embedding	in NLP, word embedding refers to representing the words that have the same meaning with the same representation. AE is used in [28] for word embedding.

B *Convolutional Neural Networks*

CNN is one of the most important deep learning architectures where it depends on multiple layers representation [29][24]. it achieves a satisfactory performance in processing two-dimensional data such as images and videos. CNNs are different than traditional Neural Networks, as in NN the relationship between the input and the output units are derived by matrix multiplication, CNN uses sparse interaction to reduce the computational burden where the kernels are smaller than the inputs. The training process is similar to the standard NN using backpropagation [3]. CNN has three main layers, which are convolutional layers, pooling layers, and fully connected layers. In the convolutional layers, various kernels are used to convolve the whole image and the intermediate feature maps. Convolution is introduced to replace the fully connected layers for learning process accleration[46]. Convolution has three main advantages: weight sharing, local connectivity, and invariance to the location of the object. Pooling layer used to reduce the dimensions of the feature maps and network parameters. Pooling makes the network more invariant to translation. The most commonly used strategies for pooling, max pooling, and average pooling. Fully connected layer converts the 2D feature maps to 1D feature vector for the classification process. As fully connected layers contain many parameters, they take large computational effort for training.

C *Restricted Boltzmann Machines*

In this section, a brief review of Restricted Boltzmann Machines is given. RBM is a generative stochastic neural network. It first proposed in 1986 [45][16], as variant of Boltzmann machine. It has

Table 2: CNN Applications

Application	Method
Face Recognition	is the ability to identity faces in an image [37].
Image Classification	CNN achieves the state-of-the-art- results in image classifications tasks [18].
Action Recognition	is creating a deep learning model to automatically recognize human actions for surveillance systems, such as this 3D model [22].
Human Pose Estimation	is a method to detect human figures in images or videos. In 2014, Toshev et al. introduced the first deep learning model for human pose classification.
Speech Recognition	refers to the ability of the system to identify word or phrases in spoken a language. CNN achieve results better than DNN [21].
Test Classification	is a method that automatically classifies documents into one or more predefined classes. Kim et. al introduced CNN for text classification [23].

a restriction that the visible and the hidden units must form a bipartite graph. RBM has a full connection between the visible and the hidden units, while there is no connection between units from the same layer. RBM models can learn the probability distribution with respect to their inputs. RBMs consists of two layers, the first layer is visible units, that correspond to the components of the input (e.g. one visible unit for each pixel of a digital input image), and the second layer is the hidden units, that model the dependencies between the components of the input (e.g. dependencies between pixels in images).

There is a detailed explanation and a practical way to train RBMs in [12], and further work in [4] to discuss the difficulties of training RBMs and propose a new algorithm that is consists of an adaptive learning rate and an enhanced gradient. RBMs are optimized in [35] by approximating the binary units with noisy rectified linear units.

RBMs play an important role in a lot of applications such as topic modeling, dimensionality reduction, collaborative filtering, classification, and feature learning.

Table 3: RBMs Applications

Application	Method
Topic Modeling	is a method that is used for purposes of document clustering, organizing, and summarize large collections of textual information. Hinton et. al applied a different-sized RMB to extract distributed semantic representation from large collections of textual data [15].
Classification	Although RBM is a generative model that learns a probability distribution from the input, it can be used as a discriminative model to classification process [25].
Collaborative Filtering	predicts user interests based on the user's past behavior. A binary RBM model is proposed by Hinton for Collaborative Filtering process in [42].
Feature Learning	RBMs are used for extracting feature in the pretraining process for classification tasks [5].
Dimensionality Reduction	Hinton used RBMs to create a model called autoencoder that achieved results better than PCA [14].

D *Deep Belief Networks*

In 2006, Hinton stacked a bank of RBMs to construct the deep belief networks. It is a Bayesian probabilistic generative model that gives a joint probability distribution over observable data and labels [9]. In DBN, every two adjacent layers for an RBM. Many variants of DBN were created by researchers [32, 31, 30, 3]. It is a very computational expensive task to create a DBN model, where several RBMs have to be trained. The training process can be divided into two stages: the pretraining stage, where an unsupervised learning is carried out a down-up direction to extract features, and the fine-tuning stage, where a supervised learning carried out an up-down direction to fine-tune the network parameters [34]. The Convolutional Deep Belief Networks (CDBN) [31] was introduced to overcome the drawback of DBN, that they do not consider the 2D structure of an input image. CDBN was further extended to achieve excellent performance in face verification [20].

Table 4: DBN Applications

Application	Method
Image Classification	Hinton proposed a generative model of DBN to classify digits [13].
Image reconstruction	DBN is applied to reconstruct images in three-dimensional space. This method has higher transparency and lowers complexity of time [36].
Image Recognition	DBNs are proposed as an unsupervised learning method for detecting sparse features [19].
Prediction of Disruption	DBNs has a powerful ability to detect complex patterns and extract features. Thus, it used in a real application to predict potential operational disruptions caused by the rail vehicle door system [7].
Text Classification	DBNs have better results in text classification than support vector machines, boosting and maximum entropy [43].

III Deep Learning New Trends

Although Deep learning outperforms all traditional techniques achieving state-of-the-art in most of the challenging problems, a lot of research focuses on improving this great accuracy by modifying its architecture or the mathematics behind it. A new trend of increasing deep learning accuracy is to combine it with traditional techniques such as preprocessing before deep learning model and fusion of features extracted.

One of the most challenging problems of CNN is its training time, it takes days even weeks to train a model for a very huge dataset. Instead of training for all the data that are redundant and noisy, Liang et. al. [64] proposed a new effective sample selection method for large-scale images to speed up the training process of CNN. They propose a clustering-based Condensed Nearest Neighbor (CBCNN), where condensed NN is able to condense a large number of original samples, then uses a k-mean clustering algorithm to optimize and select the high-quality samples that will be the input to the convolutional neural networks. The training sample subset got from the CBCNN is used as the initial input for CNN. It is named CBCNN-CNN. Experimental results on MNIST dataset show that selected samples by CBCNN, 29000 as indicated, get better accuracy than the total 60000. By this fusion, the input number is reduced by half and the test accuracy gets higher.

Another type of preprocessing of data is converting the images from the spatial domain to the wavelet domain. This preprocessing technique is proposed by Williams et.al. [65] where, instead of applying CNN to raw pixels of images they convert the images to wavelet domain and process it at lower dimension, with faster processing time. After converting the images to wavelet domain, all subbands are normalized, then perform CNN on selected subbands, and combine all the results using OR operator to get the final classification. Experimental results show that the proposed method has more accurate classification results than CNN and Stacked Denoising Autoencoders (SDA), because of the versatility of the wavelet subbands.

Zhao and Shihong [66] proposed a classification framework, for hyperspectral images, that combines the spectral and spatial features extracted from dimension reduction and deep learning, respectively. In this method spectral-spatial feature based classification (SSFC), spectral features are extracted using the dimension reduction algorithm Balanced Local Discriminant Embedding (BLDE), also deep spatial features are extracted by CNN. The extracted spectral and spatial features are combined together and a LR classifier is trained for classification. Experimental results show that SSFC achieves better performance than any fusion methods. In terms of computational complexity, it costs more time to implement SSFC to classify HIS images.

IV Deep Learning Challenges

Although deep learning achieved the state-of-the-art results in most of the challenging problems, researchers have indicated several important challenges such as theoretical understanding, training with limited data, time complexity, more powerful models.

Deep learning achieves promising results in most challenging problems, but the theory behind it is still not well understood, and there is no obvious understanding of which architectures should achieve better results than others. It is difficult to determine how many layers, number of nodes in each layer, structure of the model, and also values such as learning rate.

As the deep learning model becomes deep, the model capacity to extract features and accuracy increases, but the shortage of training data affect the learning ability. There are two commonly used solutions to solve this problem. First, using various data augmentation such as scaling, rotation, and cropping. Second, collecting more training data with weak learning algorithms.

Deep learning models require a lot of computational resources, which make them not applicable for real-time applications. Thus researchers focus to develop new architectures that are able to run in real-time. He et.al conducted a series of experiments under constrained time cost and proposed new models that are applicable to real time[10]. Also, Li et. al eliminated all the redundant competitions in the forward and the backward propagation, which result in a speedup of over 1500 times [33].

V Deep Learning Toolkits and libraries

This section provides a brief overview of the most popular deep learning tools and libraries that are available to construct and execute efficiently deep learning models. There is no standard to decide which toolkit is better than the others. As each toolkit is designed to address the problems perceived by the developer. Some criteria must be considered to choose a toolkit such as **Programming Language**, the toolkit programming language written in has an impact of using it. **Toolkit Documentation**, easy and coverage documentation that provides examples, similar to real case problem the developers will work on, is beneficial to develop solutions easliy and effectively. **Development Environment**, deep learning toolkits provide a development environment that makes the task of developer easy, such as using a graphical integrated development or some visualization tools to ensure that the construction of the network and monitoring the learning process. **Execution Speed**, deep learning models treated with very high dimensional data such as images or videos, so it requires very high execution speed for calculating the mathematical operations. **Training Speed**, training deep learning networks require huge dataset may be thousands and millions of images, so training speed is an important issue in deep learning, which depend on the toolkit's mathematical operations libraries are implemented to handle such huge calculations.**GPU Support,** deep learning have witnessed a big advance after using Graphical Processing Units in training deep models, so the toolkit that supports one or more GPUs will gain a better performance than others [6].

The most common used libraries and toolkits are **Tensorflow**, is an open source library written in C++ and Python and developed by Google. Tensorflow supports the using of multiple GPUs and CPUs for training deep models. Also, it provides TensorBoard to monitor and tune the performance of the network. **Keras**, a python toolkit that utilizes either Theano or Tensorflow as backend. Keras is very to build and create deep learning models, where each line of code is a layer. Keras provide pre-trained models of the state-of-the-art deep learning architecture for the process of finetuning or transfer learning. **Caffe** is one of the most advanced toolkits, it is written in C++ by Berkeley Vision and Learning Center. Caffe supports multiple GPUs and uses JSON text file to describe the models' architectures. **MXNet** is a tool that is written in C++, it supports multiple GPUs, low-level and high-level API construction. **Theano** is a deep learning toolkit that uses symbolic logic to create networks. Theano has written in python and make use of Numpy. **Cognitive Network Toolkit** (CNTK) is a deep learning tool developed by Microsoft for machine learning developers and researchers. **Lasagne** is a library written in python on top of Theano as a method for simplifying the work of Theano in building deep learning models. **DeepLearning4j** is a toolkit written in Java and Scala by Andrej Karpathy and supports GPU. Also, there are other libraries such as PyTorch, Chainer, Torch7, DIGITS, TFLearn, Caffe2, dlib [49]. Thus selecting the best toolkit depends on the skills and background of the researcher or the developer.

VI Conclusion

This paper considered a brief review of the deep learning architecture, the most important applications and challenges that face deep learning networks. Deep learning is used in many applications such as computer vision, pattern recognition, speech recognition, and text classification. It divides the deep learning models into four categories based on the structure of the model: Restricted Boltzmann Machines, Autoencoder, Convolutional Neural Networks, and Deep Belief Networks. The most important applications of each deep learning model are also reviewed in this literature. Also, the challenges that face deep learning networks are mentioned such as theoretical understanding, limited data, and computational resources. Most notably, there are some experiments to combine deep learning models with classical machine learning models, some of these experiments are mentioned in this literature. And finally, most deep learning toolkits and libraries are comprehensively covered to facilitate to any researcher or programmer to choose the most proper deep learning toolkit.

References

[1] *ILSVRC2017. Large Scale Visual Recognition Challenge 2017 Results.*

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[3] Yoshua Bengio et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[4] KyungHyun Cho, Tapani Raiko, and Alexander T Ihler. Enhanced gradient and adaptive learning rate for training restricted boltzmann machines. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 105–112, 2011.

[5] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223, 2011.

[6] Bradley J Erickson, Panagiotis Korfiatis, Zeynettin Akkus, Timothy Kline, and Kenneth Philbrick. Toolkits and libraries for deep learning. *Journal of digital imaging*, 30(4):400–405, 2017.

[7] Olga Fink, Enrico Zio, and Ulrich Weidmann. Development and Application of Deep Belief Networks for Predicting Railway Operation Disruptions. *International Journal of Performability Engineering*, 11(2):121–134, March 2015.

[8] Ian Goodfellow, Honglak Lee, Quoc V Le, Andrew Saxe, and Andrew Y Ng. Measuring invariances in deep networks. In *Advances in neural information processing systems*, pages 646–654, 2009.

[9] Yanming Guo, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu, and Michael S Lew. Deep learning for visual understanding: A review. *Neurocomputing*, 187:27–48, 2016.

[10] Kaiming He and Jian Sun. Convolutional neural networks at constrained time cost. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5353–5360, 2015.

[11] Geoffrey E Hinton. Learning multiple layers of representation. *Trends in cognitive sciences*, 11(10):428–434, 2007.

[12] Geoffrey E Hinton. A practical guide to training restricted boltzmann machines. In *Neural networks: Tricks of the trade*, pages 599–619. Springer, 2012.

[13] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[14] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[15] Geoffrey E Hinton and Ruslan R Salakhutdinov. Replicated softmax: an undirected topic model. In *Advances in neural information processing systems*, pages 1607–1614, 2009.

[16] Geoffrey E Hinton, Terrence J Sejnowski, et al. Learning and relearning in boltzmann machines. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1:282–317, 1986.

[17] Geoffrey E Hinton and Richard S. Zemel. Autoencoders, minimum description length and helmholtz free energy. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems 6*, pages 3–10. Morgan-Kaufmann, 1994.

[18] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. *arXiv preprint arXiv:1709.01507*, 7, 2017.

[19] Fu Jie Huang, Y-Lan Boureau, Yann LeCun, et al. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[20] Gary B Huang, Honglak Lee, and Erik Learned-Miller. Learning hierarchical representations for face verification with convolutional deep belief networks. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2518–2525. IEEE, 2012.

[21] Jui-Ting Huang, Jinyu Li, and Yifan Gong. An analysis of convolutional neural networks for speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 4989–4993. IEEE, 2015.

[22] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2013.

[23] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[25] Hugo Larochelle and Yoshua Bengio. Classification using discriminative restricted boltzmann machines. In *Proceedings of the 25th international conference on Machine learning*, pages 536–543. ACM, 2008.

[26] Quoc V Le. Building high-level features using large scale unsupervised learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8595–8598. IEEE, 2013.

[27] Quoc V Le, Jiquan Ngiam, Adam Coates, Abhik Lahiri, Bobby Prochnow, and Andrew Y Ng. On optimization methods for deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 265–272. Omnipress, 2011.

[28] Rémi Lebret and Ronan Collobert. " the sum of its parts": Joint learning of word and phrase representations with autoencoders. *arXiv preprint arXiv:1506.05703*, 2015.

[29] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[30] Honglak Lee, Chaitanya Ekanadham, and Andrew Y Ng. Sparse deep belief net model for visual area v2. In *Advances in neural information processing systems*, pages 873–880, 2008.

[31] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th annual international conference on machine learning*, pages 609–616. ACM, 2009.

[32] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y Ng. Unsupervised learning of hierarchical representations with convolutional deep belief networks. *Communications of the ACM*, 54(10):95–103, 2011.

[33] Hongsheng Li, Rui Zhao, and Xiaogang Wang. Highly efficient forward and backward propagation of convolutional neural networks for pixelwise classification. *arXiv preprint arXiv:1412.4526*, 2014.

[34] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 234:11–26, 2017.

[35] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[36] AMIN EMAMZADEH ESMAEILI NEJAD. An application of deep belief networks for 3-dimensional image reconstruction. 2014.

[37] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al. Deep face recognition. In *BMVC*, volume 1, page 6, 2015.

[38] Christopher Poultney, Sumit Chopra, Yann L Cun, et al. Efficient learning of sparse representations with an energy-based model. In *Advances in neural information processing systems*, pages 1137–1144, 2007.

[39] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 833–840. Omnipress, 2011.

[40] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[41] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[42] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. Restricted boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning*, pages 791–798. ACM, 2007.

[43] Ruhi Sarikaya, Geoffrey E Hinton, and Anoop Deoras. Application of deep belief networks for natural language understanding. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 22(4):778–784, 2014.

[44] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[45] P Smolensky. Foundations of harmony theory: Cognitive dynamical systems and the subsymbolic theory of information processing. *Parallel distributed processing: Explorations in the microstructure of cognition*, 1:191–281, 1986.

[46] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[47] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.

[48] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(Dec):3371–3408, 2010.

[49] Jan Zacharias, Michael Barz, and Daniel Sonntag. A survey on deep learning toolkits and libraries for intelligent user interfaces. *arXiv preprint arXiv:1803.04818*, 2018.

[50] Will Y Zou, Andrew Y Ng, and Kai Yu. Unsupervised learning of visual invariance with temporal coherence. In *NIPS 2011 workshop on deep learning and unsupervised feature learning*, volume 3, 2011.