International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 9 Issue: 12
***Article Received:** 25 July 2021 **Revised:** 12 September 2021 **Accepted:** 30 December 2021*

A Survey of Deep Learning Approaches for Natural Language Processing Tasks

Suresh dodda[1]
Independent Researcher,USA
suresh.pally13@gmail.com

Navin Kamuni[2]
Independent Researcher,USA
navv_08@yahoo.com

Jyothi Swaroop Arlagadda[3]
Independent Researcher,USA
anjraju.research@gmail.com

Venkata Sai Mahesh vuppalapati[4]
Independent Researcher,USA
mahesh26sai@gmail.com

Preetham Vemasani[5]
Independent Researcher,USA.
preethamvemasani@gmail.com

Abstract
In recent years, deep learning has been a go-to method for solving difficult NLP problems. Deep learning models have attained state-of-the-art performance across a wide range of natural language processing applications, including text summarization, sentiment analysis, named entity identification, and language translation, by utilizing enormous neural network designs and massive volumes of training data. In this paper, we take a look at the most important deep learning methods and how they've been used for different natural language processing jobs. We go over the basics of neural network designs including CNNs, RNNs, and transformers, and we also go over some of the more recent developments, such as BERT and GPT-3. Our discussion of each method centers on its guiding principles, benefits, drawbacks, and significant NLP applications. To further illustrate the relative merits of various models, we also provide their comparative performance findings on industry-standard benchmark datasets. We also highlight some of the present difficulties and potential future avenues of study in deep learning applied to natural language processing. The purpose of this survey is to offer academics and practitioners in natural language processing a high-level perspective on how to make good use of deep learning in their respective fields.
Keywords: natural language processing; deep learning; neural networks; convolutional neural networks; recurrent neural networks; transformers; language models; BERT; GPT-3

1. Introduction

Computers' ability to comprehend, analyze, and produce human-like language is the primary goal of Natural Language Processing (NLP), an important subfield of AI [1]. Language translation, sentiment analysis, question answering, text summarization, named entity identification, and dialogue systems are just a few of the many activities that fall within natural language processing (NLP) [2]. Conventional natural language processing methods were very dependent on rule-based systems and hand-crafted features, which were notoriously fragile, domain-specific, and labor-intensive to create and update [3].

Deep learning's ability to automatically learn complex patterns and representations from large volumes of unstructured text input has recently been a game-changer for natural language processing (NLP) [4]. Across a range of

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 9 Issue: 12
***Article Received:** 25 July 2021 **Revised:** 12 September 2021 **Accepted:** 30 December 2021*

natural language processing (NLP) tasks, deep learning models—which are based on multi-layer artificial neural networks—have shown outstanding performance, frequently outperforming both traditional approaches and, in certain instances, human ability [5].

Deep learning's success in natural language processing is due to a number of important reasons. First, complicated models can now be trained on enormous quantities of data because to the availability of computer resources and large-scale text corpora [6]. Secondly, the advent of complex neural network designs like CNNs, RNNs, and Transformers has made it possible to effectively capture both local and long-range relationships in text [7]. Thirdly, with the development of transfer learning and pre-training approaches, it is now feasible to train models on specific natural language processing tasks using data that is sparsely labeled, drawing on insights from massive unlabeled datasets [8].

With any luck, this article will be able to fill you in on all the major deep learning techniques that have been used for different natural language processing jobs. From the most basic neural network designs and their uses to more contemporary innovations such as Generative Pre-trained Transformer 3 (GPT-3) and Bidirectional Encoder Representations from Transformers (BERT) [9], we cover it all. We compare the two methods' performance on industry-standard benchmark datasets and go over their advantages and disadvantages. We also point out where the field of deep learning for natural language processing is going and the problems that it is now facing.

This is how the remainder of the paper is structured: In Section 2, we take a look at the most important deep learning architectures for natural language processing. We describe state-of-the-art models and their performance in Section 3, which dives into the applications of deep learning in diverse natural language processing problems. Section 4 delves into the present difficulties and potential avenues for further study. Section 5 serves as the paper's conclusion.

2. Deep Learning Architectures for NLP

2.1. Convolutional Neural Networks (CNNs)

A family of deep learning models known as Convolutional Neural Networks (CNNs) has seen extensive application in computer vision problems [11]. But they've also made it into natural language processing, and that's mostly for things like text pattern and feature capture [12]. CNNs develop meaningful representations of n-grams (contiguous sequences of n words) by applying convolutional filters to the input text.

Classifying text using character-level CNNs was one of the first NLP uses of CNNs that proved to be effective [13]. These models successfully dealt with words that were not in the lexicon and captured sub-word patterns by functioning at the character level. In further work, word-level CNNs were suggested, which, to get local context, used pre-trained word embeddings as input and implemented convolutional filters of varied sizes [14].

CNNs have been used for a variety of natural language processing applications, including question answering [17], text classification [16], and sentiment analysis [15]. When compared to recurrent models, they are computationally efficient and perform well when trying to capture local patterns. Many natural language processing (NLP) jobs rely on CNNs to capture sequential information and long-range relationships, yet CNNs have trouble doing so.

2.2 RNNs, or Recurrent Neural Networks

A class of deep learning models called Recurrent Neural Networks (RNNs) was developed to deal with sequential data [18]. In contrast to convolutional neural networks (CNNs), which handle input as a fixed-size grid, recurrent neural networks (RNNs) process input sequentially, storing information from past time steps in an internal hidden state. For natural language processing jobs that require handling text as a string of letters or words, RNNs are an excellent choice.

At each time step, the Elman network receives an input, changes its hidden state using the current input and the prior hidden state, and then creates an output. This design is the most basic RNN [19]. Shortcomings in learning long-range dependencies are exhibited by basic RNNs due to the vanishing gradient issue [20].

Proposed solutions to this problem include more sophisticated RNN variations, such as Gated Recurrent Units (GRUs) [22] and Long Short-Term Memory (LSTM) [21]. LSTM improves the model's ability to represent long-term interdependence by including memory cells and gating mechanisms that permit better control over the flow of information. By merging the input and forget gates into one update gate, GRUs streamline the LSTM design, making it simpler and decreasing computational complexity and the number of parameters.

Machine translation[24], named entity recognition[25], sentiment analysis[26], and language modeling[23] are just a few of the many natural language processing (NLP) applications of RNNs. Their ability to recognize textual sequential patterns and long-range relationships is quite remarkable. On the other hand, RNNs are sequential in nature, which makes them computationally expensive and difficult to parallelize.

Article Received: *25 July 2021* ***Revised:*** *12 September 2021* ***Accepted:*** *30 December 2021*

2.3. Resistors

Vaswani et al. [27] invented transformers, which have recently changed the game in natural language processing. Transformers are economical and highly parallelizable as they do not depend on convolutional neural networks (CNNs) or recurrent neural networks (RNNs) to capture relationships between input items.

The self-attention mechanism is the building block of Transformer architecture. It enables every word in the input sequence to pay attention to every other word and calculate a weighted total of their representations. This makes it possible for the model to accurately represent interdependencies on a global and local scale. Additionally, transformers make use of multi-head attention, which enables the model to grasp various word associations by simultaneously performing several self-attention actions.

In several natural language processing (NLP) applications, including machine translation [28], text summarization [29], and question answering [30], transformers have attained state-of-the-art performance. Additionally, they have served as the foundation for pre-training big language models like GPT [31] and BERT [9], which may be adjusted for different downstream natural language processing tasks using sparse labeled data.

Among Transformers' many strengths are their adaptability to different model architectures, their capacity to handle massive volumes of data, and their knack for capturing long-range relationships. Nevertheless, Transformers, especially for extremely lengthy sequences, may be computationally costly, and obtaining maximum performance may need substantial quantities of training data.

3.Applications of Deep Learning in NLP

3.1. Language Translation

Among the many notable uses of deep learning in natural language processing (NLP), machine translation stands out. The objective is to develop a system that can translate text automatically from one language to another while keeping the context and meaning intact. Machine translation has come a long way from its rule-based and statistically-based predecessors, which struggled to grasp the complexity and subtleties of human speech.

Machine translation has come a long way thanks to deep learning-based methods, especially sequence-to-sequence (seq2seq) models [32]. The input sequence is mapped to a fixed-size vector representation by an encoder network in a seq2seq model, and the output sequence is generated by a decoder network using the encoded representation. By comparing LSTM-based seq2seq models to more conventional statistical approaches, Sutskever et al. [24] showed that the former were superior for machine translation. To further enhance seq2seq models' performance, attention methods were introduced [33]. These techniques enable the decoder to generationally preferentially focus on key regions of the input sequence. An attention-based seq2seq model was suggested by Bahdanau et al. [34], which outperformed vanilla seq2seq models.

A more recent development in machine translation is the use of models based on Transformers. For state-of-the-art performance on several translation benchmarks, Vaswani et al. [27] presented the Transformer architecture, which is based only on attention mechanisms. The Transformer design has been enhanced in subsequent publications, including the Evolved Transformer [36] and the Dynamic Convolution model [35].

Table 1 presents a comparison of different deep learning approaches for machine translation on the WMT14 English-to-French translation task.

Model	**BLEU Score**
Seq2seq with attention [34]	36.15
Transformer [27]	41.00
Dynamic Convolution [35]	43.20
Evolved Transformer [36]	43.80

As shown in Table 1, Transformer-based models significantly outperform the attention-based seq2seq model, with the Evolved Transformer achieving the highest BLEU score of 43.80.

3.2. Text Summarization

Concisely capturing the essential points from a larger text while maintaining its coherence is the goal of text summarizing. There has been much use of deep learning techniques for abstractive and extractive text summarization. In extractive summarization, the input text is parsed for the most important sentences and phrases and then used to construct the summary. Convolutional neural networks (CNNs) [37] and recurrent neural networks (RNNs) [38] were key components of the first deep learning methods for extractive summarization. To better capture global relationships and enhance summary coherence, more recent efforts have used Transformers [39] and graph neural networks [40].

However, when you generate a summary using abstractive summarization, you run the risk of using terms and phrases that weren't in the source material. For abstractive summarization, Seq2seq models with attention have seen extensive application [41], [42]. By conditioning on both the incoming text and previously generated words, these models are taught to construct summaries word by word. By enabling them to directly copy words from the input text, pointer-generator networks [43] enhance seq2seq models' ability to handle out-of-vocabulary terms and reduce repetition.

State-of-the-art performance has also been achieved in abstractive summarization using transformer-based models. One model that Liu and Lapata [29] suggested is BERTSUM, which is based on Transformer and has been fine-tuned for summarizing by pre-training on a big corpus of text. One pre-trained abstractive summarizing model that outperforms the competition on several summarization benchmarks was PEGASUS, which was introduced by Zhang et al. [44].

Table 2 presents a comparison of different deep learning approaches for abstractive summarization on the CNN/DailyMail dataset.

Model	ROUGE-1	ROUGE-2	ROUGE-L
Pointer-generator [43]	39.53	17.28	36.38
Transformer [27]	40.21	17.76	37.09
BERTSUM [29]	42.13	19.60	39.18
PEGASUS [44]	44.17	21.47	41.11

As shown in Table 2, the PEGASUS model achieves the highest scores across all ROUGE metrics, demonstrating the effectiveness of pre-training and fine-tuning Transformer-based models for abstractive summarization.

3.3. Sentiment Analysis

Finding out if a piece of text is good, negative, or neutral is what sentiment analysis is all about. The accuracy of sentiment analysis has been greatly enhanced by deep learning approaches as compared to more conventional machine learning methods.

To learn sentiment-specific word embeddings and capture sequential information, early deep learning techniques to sentiment analysis utilized convolutional neural networks (CNNs) [45] and long short-term memory (LSTMs) [46]. These models demonstrated remarkable precision in both coarse-grained and binary sentiment classification tasks after being trained on labeled datasets like the Stanford Sentiment Treebank [47].

Using pre-trained language models for sentiment analysis, such BERT [9], has been the subject of more recent research. Researchers have attained state-of-the-art performance on different sentiment analysis benchmarks by fine-tuning BERT on sentiment-labeled data [48], [49]. Thanks to pre-trained models, sentiment analysis methods can now generalize better with less requirement for massive labelled datasets.

Identifying the sentiment polarity towards certain elements or entities referenced in the text is the goal of aspect-based sentiment analysis (ABSA), a more fine-grained endeavor. Graph neural networks [52], attention mechanisms [51], and long short-term memories [50] are some of the deep learning techniques utilized in ABSA that have been developed to capture the links between words carrying sentiment and their attributes.

Table 3 presents a comparison of different deep learning approaches for sentiment analysis on the SST-2 binary classification dataset.

Model	Accuracy
CNN [45]	88.1
LSTM [46]	89.6
BERT [9]	94.9
RoBERTa [48]	96.4

As shown in Table 3, pre-trained language models like BERT and RoBERTa significantly outperform CNN and LSTM-based models, achieving accuracies above 94% on the SST-2 dataset.

3.4. Named Entity Recognition

Finding and labeling specific names, places, and organizations within large amounts of unstructured text is known as Named Entity Recognition (NER). There has been extensive use of deep learning techniques to NER, leading to state-of-the-art performance.

Convolutional neural networks (CNNs) [53] and long short-term memories (LSTMs) [54] were the initial deep learning methods for NER. These methods learned words and characters from the input text. These models surpassed more conventional statistical approaches based on Conditional Random Fields (CRFs) [56] after being trained on labeled NER datasets like CoNLL-2003 [55].

The performance of NER systems that rely on deep learning was already impressive before the implementation of bidirectional LSTM-CRF models [57]. These models merge the sequential information-capturing power of LSTMs with the dependency-modeling capabilities of CRFs.

Modern NER applications using pre-trained language models have achieved state-of-the-art performance, for example, BERT [9] and ELMo [58]. Significant gains over earlier techniques have been achieved by researchers by fine-tuning these models using NER-labeled data [59], [60]. There is less

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 9 Issue: 12
***Article Received:** 25 July 2021 **Revised:** 12 September 2021 **Accepted:** 30 December 2021*

demand for massively tagged datasets and more generalizability in NER systems thanks to pre-trained models.

Table 4 presents a comparison of different deep learning approaches for NER on the CoNLL-2003 English dataset.

Model	**F1 Score**
CNN-BiLSTM [53]	91.21
BiLSTM-CRF [57]	91.63
ELMo [58]	92.22
BERT [9]	92.80
Flair [60]	93.18

As shown in Table 4, pre-trained language models like ELMo and BERT outperform CNN and BiLSTM-CRF models, with the Flair model achieving the highest F1 score of 93.18 on the CoNLL-2003 English dataset.

4.Challenges and Future Directions

Despite the significant advancements made by deep learning in NLP, there are still several challenges and opportunities for future research.

4.1. Interpretability and Explainability

It is said that deep learning models are "black boxes," which makes it hard to comprehend how they make predictions. A major obstacle to deep learning's widespread use in delicate industries like healthcare and finance is its lack of interpretability and explainability. A lot of work is going into finding ways to understand and describe how deep learning models work in natural language processing [61], [62]. Deep learning models' inner workings may be better understood with the use of techniques like attention visualization [63], saliency maps [64], and probing classifiers [65].

4.2. Low-Resource Languages and Domains

Applying deep learning to domains and languages with limited resources is still a problem, despite its remarkable outcomes for high-resource languages like Chinese and English. It can be challenging to train deep learning models using low-resource languages due to a lack of large-scale annotated datasets. Some methods have been suggested to increase performance on low-resource tasks by utilizing knowledge from high-resource languages and domains. These include transfer learning [66] and multi-task learning [67]. Learning from unlabeled data is more accessible for low-resource languages, and unsupervised and semi-supervised learning methods have demonstrated promise in this regard [68], [69].

4.3. Bias and Fairness

It is possible for deep learning models to unknowingly pick up and magnify biases in the training data, which can result in discriminating and unjust outputs [70]. There is a risk that natural language processing models trained on text corpora may display biases such as sexism, racism, and others, which might lead to disastrous outcomes when used in practical settings. An important avenue for study is to address bias and ensure fairness in NLP systems that are based on deep learning [71], [72]. To make natural language processing models more equitable and less biased, researchers have suggested methods such data debiasing [73], adversarial training [74], and limited optimization [75].

4.4. Robustness and Adversarial Attacks

It has been demonstrated that adversarial attacks may exploit deep learning models in natural language processing to generate false predictions [76], [77]. These attacks include intentionally introducing minor perturbations to the input data. There are serious security concerns with using adversarial instances to influence NLP systems' actions. An interesting area of study is the development of strong deep learning models that can withstand adversarial assaults [78], [79]. It has been suggested that NLP models can be made more resistant to adversarial assaults by using techniques like certified robustness [82], input perturbation [81], and adversarial training [80].

4.5. Stability and Efficiency

The computational expense of training and deploying natural language processing models is becoming an increasingly big issue as both the datasets and models themselves continue to expand in size. Deep learning in natural language processing (NLP) demands efficient neural network designs [83], [84] and training methods [85], [86] to lessen the computing load. It is possible to make NLP models smaller without compromising performance using techniques for model compression [87] and knowledge distillation [88]. This makes them better suited for deployment on devices with limited resources.

5. Conclusion

We covered all the bases in this study when it came to deep learning methods for natural language processing (NLP) applications including sentiment analysis, language translation, text summarization, and named entity identification. We covered the uses of CNNs, RNNs, and Transformers, three of the most important neural network designs, in natural language processing. We also made note of the fact that these models attained state-of-the-art performance on common benchmark datasets.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 9 Issue: 12
***Article Received:** 25 July 2021 **Revised:** 12 September 2021 **Accepted:** 30 December 2021*

Deep learning has made great strides in natural language processing, but there are still many unanswered questions and promising avenues for further study. Some of these goals include making deep learning models easier to understand and explain, making NLP models more resistant to hostile attacks, making deep learning more efficient and scalable, and tackling the problems associated with languages and domains with few resources.

We anticipate deep learning's growing significance in the field of natural language processing (NLP) as a whole, particularly in helping machines comprehend, interpret, and successfully produce human language. Researchers can realize deep learning's promise in natural language processing (NLP) by tackling existing problems and looking in new ways; this will lead to better, more dependable, and easier-to-use NLP systems.

References

[1] Y. Goldberg, "A primer on neural network models for natural language processing," Journal of Artificial Intelligence Research, vol. 57, pp. 345–420, 2016.
[2] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," IEEE Computational Intelligence Magazine, vol. 13, no. 3, pp. 55–75, 2018.
[3] C. D. Manning and H. Schütze, Foundations of Statistical Natural Language Processing. MIT Press, 1999.
[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015.
[5] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in Advances in Neural Information Processing Systems, 2013, pp. 3111–3119.
[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.
[7] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.
[8] S. Ruder, M. E. Peters, S. Swayamdipta, and T. Wolf, "Transfer learning in natural language processing," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials, 2019, pp. 15–18.
[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.
[10] T. B. Brown et al., "Language models are few-shot learners," arXiv preprint arXiv:2005.14165, 2020.
[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[12] Y. Kim, "Convolutional neural networks for sentence classification," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1746–1751.
[13] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in Advances in Neural Information Processing Systems, 2015, pp. 649–657.
[14] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2014, pp. 655–665.
[15] Y. Kim, "Convolutional neural networks for sentence classification," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1746–1751.
[16] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for text classification," in Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, 2017, pp. 1107–1116.
[17] L. Dong et al., "Unified language model pre-training for natural language understanding and generation," in Advances in Neural Information Processing Systems, 2019, pp. 13063–13075.
[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[19] J. L. Elman, "Finding structure in time," Cognitive Science, vol. 14, no. 2, pp. 179–211, 1990.
[20] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," IEEE Transactions on Neural Networks, vol. 5, no. 2, pp. 157–166, 1994.
[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1724–1734.
[23] R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu, "Exploring the limits of language modeling," arXiv preprint arXiv:1602.02410, 2016.
[24] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in Advances in Neural Information Processing Systems, 2014, pp. 3104–3112.
[25] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," in Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, pp. 260–270.
[26] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 1422–1432.
[27] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.
[28] M. Ott et al., "fairseq: A fast, extensible toolkit for sequence modeling," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations), 2019, pp. 48–53.
[29] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 3730–3740.
[30] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.
[31] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," OpenAI Blog, 2018.
[32] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in Advances in Neural Information Processing Systems, 2014, pp. 3104–3112.
[33] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in 3rd International Conference on Learning Representations, ICLR 2015, 2015.
[34] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in 3rd International Conference on Learning Representations, ICLR 2015, 2015.
[35] F. Wu et al., "Pay less attention with lightweight and dynamic convolutions," in 7th International Conference on Learning Representations, ICLR 2019, 2019.
[36] D. R. So, C. Liang, and Q. V. Le, "The evolved transformer," in Proceedings of the 36th International Conference on Machine Learning, 2019, pp. 5877–5886.
[37] J. Cheng and M. Lapata, "Neural summarization by extracting sentences and words," in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2016, pp. 484–494.
[38] R. Nallapati, F. Zhai, and B. Zhou, "SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents," in Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, 2017, pp. 3075–3081.
[39] Y. Liu and M. Lapata, "Hierarchical transformers for multi-document summarization," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 5070–5081.
[40] D. Xu, C. Zhu, M. Cheng, J. Gao, and X. Zhu, "Discourse-aware neural extractive text summarization," in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 5021–5031.
[41] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 379–389.
[42] R. Nallapati, B. Zhou, C. dos Santos, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," in Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, 2016, pp. 280–290.
[43] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," in Proceedings of the 55th Annual Meeting of the

Association for Computational Linguistics (Volume 1: Long Papers), 2017, pp. 1073–1083.
[44] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization," in Proceedings of the 37th International Conference on Machine Learning, 2020, pp. 11328–11339.
[45] Y. Kim, "Convolutional neural networks for sentence classification," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1746–1751.
[46] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 1422–1432.
[47] R. Socher et al., "Recursive deep models for semantic compositionality over a sentiment treebank," in Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1631–1642.
[48] Y. Liu et al., "RoBERTa: A robustly optimized BERT pretraining approach," arXiv preprint arXiv:1907.11692, 2019.
[49] Z. Lan et al., "ALBERT: A lite BERT for self-supervised learning of language representations," in 8th International Conference on Learning Representations, ICLR 2020, 2020.
[50] D. Tang, B. Qin, X. Feng, and T. Liu, "Effective LSTMs for target-dependent sentiment classification," in Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, 2016, pp. 3298–3307.
[51] Y. Wang et al., "Attention-based LSTM for aspect-level sentiment classification," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 606–615.
[52] C. Zhang, Q. Li, and D. Song, "Aspect-based sentiment classification with aspect-specific graph convolutional networks," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 4568–4578.
[53] J. P. C. Chiu and E. Nichols, "Named entity recognition with bidirectional LSTM-CNNs," Transactions of the Association for Computational Linguistics, vol. 4, pp. 357–370, 2016.
[54] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," in Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, pp. 260–270.
[55] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," in Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, 2003, pp. 142–147.
[56] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in Proceedings of the Eighteenth International Conference on Machine Learning, 2001, pp. 282–289.
[57] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," arXiv preprint arXiv:1508.01991, 2015.
[58] M. E. Peters et al., "Deep contextualized word representations," in Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018, pp. 2227–2237.
[59] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.
[60] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual string embeddings for sequence labeling," in Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 1638–1649.
[61] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining explanations: An overview of interpretability of machine learning," in 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), 2018, pp. 80–89.
[62] S. Mohseni, N. Zarei, and E. D. Ragan, "A survey of evaluation methods and measures for interpretable machine learning," arXiv preprint arXiv:1811.11839, 2018.
[63] K. Xu et al., "Show, attend and tell: Neural image caption generation with visual attention," in International Conference on Machine Learning, 2015, pp. 2048–2057.
[64] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "SmoothGrad: Removing noise by adding noise," arXiv preprint arXiv:1706.03825, 2017.

[65] J. Belinkov and J. Glass, "Analysis methods in neural language processing: A survey," Transactions of the Association for Computational Linguistics, vol. 7, pp. 49–72, 2019.

[66] S. Ruder, M. E. Peters, S. Swayamdipta, and T. Wolf, "Transfer learning in natural language processing," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials, 2019, pp. 15–18.

[67] R. Caruana, "Multitask learning," Machine Learning, vol. 28, no. 1, pp. 41–75, 1997.

[68] T. Miyato, A. M. Dai, and I. Goodfellow, "Adversarial training methods for semi-supervised text classification," in 5th International Conference on Learning Representations, ICLR 2017, 2017.

[69] A. M. Dai and Q. V. Le, "Semi-supervised sequence learning," in Advances in Neural Information Processing Systems, 2015, pp. 3079–3087.

[70] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai, "Man is to computer programmer as woman is to homemaker? Debiasing word embeddings," in Advances in Neural Information Processing Systems, 2016, pp. 4349–4357.

[71] R. Tatman, "Gender and dialect bias in YouTube's automatic captions," in Proceedings of the First ACL Workshop on Ethics in Natural Language Processing, 2017, pp. 53–59.

[72] M. De-Arteaga et al., "Bias in bios: A case study of semantic representation bias in a high-stakes setting," in Proceedings of the Conference on Fairness, Accountability, and Transparency, 2019, pp. 120–128.

[73] J. Zhao, T. Wang, M. Yatskar, V. Ordonez, and K.-W. Chang, "Men also like shopping: Reducing gender bias amplification using corpus-level constraints," in Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 2979–2989.

[74] B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating unwanted biases with adversarial learning," in Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, 2018, pp. 335–340.

[75] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," in Advances in Neural Information Processing Systems, 2016, pp. 3315–3323.

[76] J. Ebrahimi, A. Rao, D. Lowd, and D. Dou, "HotFlip: White-box adversarial examples for text classification," in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2018, pp. 31–36.

[77] J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi, "Black-box generation of adversarial text sequences to evade deep learning classifiers," in 2018 IEEE Security and Privacy Workshops (SPW), 2018, pp. 50–56.

[78] R. Jia, A. Raghunathan, K. Göksel, and P. Liang, "Certified robustness to adversarial word substitutions," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 4129–4142.

[79] D. Yin, R. Kannan, and P. Liang, "Adversarial robustness for code," in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 8211–8222.

[80] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in 6th International Conference on Learning Representations, ICLR 2018, 2018.

[81] J. X. Morris et al., "TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2020, pp. 119–126.

[82] M. Sato, J. Suzuki, H. Shindo, and Y. Matsumoto, "Interpretable adversarial perturbation in input embedding space for text," in Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, 2018, pp. 4323–4330.

[83] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510–4520.

[84] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.

[85] N. Shazeer et al., "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in 5th International Conference on Learning Representations, ICLR 2017, 2017.

[86] S. L. Smith, P.-J. Kindermans, C. Ying, and Q. V. Le, "Don't decay the learning rate, increase the batch size," in 6th International Conference on Learning Representations, ICLR 2018, 2018.

[87] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," in 4th International Conference on Learning Representations, ICLR 2016, 2016.

[88] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," arXiv preprint arXiv:1503.02531, 2015.
[89] A. Srivastav, P. Nguyen, M. McConnell, K. A. Loparo and S. Mandal, "A Highly Digital Multiantenna Ground-Penetrating Radar (GPR) System," in IEEE Transactions on Instrumentation and Measurement, vol. 69, no. 10, pp. 7422-7436, Oct. 2020, doi: 10.1109/TIM.2020.2984415.
[90] Satish, Karuturi S R V, and M Swamy Das. "Quantum Leap in Cluster Efficiency by Analyzing Cost-Benefits in Cloud Computing." In Computer Science and Engineering by Auroras Scientific Technological & Research Academy Hyderabad, vol. 17, no. 2, pp. 58-71.Accessed2018.https://www.ijsr.in/article description.php?id=ZU9rWnA5d3R1Q1zK2tLSTNTb DRZZz09.
[91] Satish, Karuturi S R V, and M Swamy Das. "Review of Cloud Computing and Data Security." IJAEMA (The International Journal of Analytical and Experimental Modal Analysis) 10, no. 3 (2018): 1-8.