

Deep Learning Based Truth Discovery Algorithm for Research the Genuineness of Given Text Corpus

Adilakshmi Vadavalli, R Subhashini

ABSTRACT--- Lot of research has gone into Natural language processing and the state of the art algorithms in deep learning that unambiguously helps in converting an English text into a data structure without loss of meaning. Also with the advent of neural networks for learning word representations as vectors has helped a lot in revolutionizing the automatic feature extraction from text data corpus. A combination of word embedding and the use of a deep learning algorithm like a convolution neural network helped in better accuracy for text classification. In this era of Internet of things and the voluminous amounts of data that is overwhelming the users determining the veracity of the data is a very challenging task. There are many truth discovery algorithms in literature that help in resolving the conflicts that arise due to multiple sources of data. These algorithms help in estimating the trustworthiness of the data and reliability of the sources. In this paper, a convolution based truth discovery with multitasking is proposed to estimate the genuineness of the data for a given text corpus. The proposed algorithm has been tested on analysing the genuineness of Quora questions dataset and experimental results showed an improved accuracy and speed over other existing approaches.

Index Terms — Truth Discovery, Natural Language Processing, Text Classification, Deep learning, Word embedding

I. INTRODUCTION

With so much of information overflow determining the genuineness of the data is a very challenging task. Many algorithms exist in literature that determine the trustworthiness of the data and reliability of the sources[1]. But the problems associated with the existing literature is that it does not work on unstructured data[2]. Also there is no standardized models that exist for ascertaining the data veracity[3]. With the entry of deep learning algorithms the analysis of unstructured data has become lot simpler. The good thing about deep learning is that the performance of algorithm increases with increase in amount of data. As we know the deep learning algorithm have a wide variety of applications in image processing, image classification, speech recognition, language translation, computer vision and time series force casting. Before the entry of deep learning algorithms the feature extraction was manual. It was very tedious and cumbersome. There are many methods in literature which propose different feature extraction techniques[4]. As we can see in Fig 1 which list the most

commonly used methods like the Iterative methods, Probabilistic methods and optimization methods. But these methods are not good in case of extracting features from unstructured data. For unstructured data feature extraction we can use a simple neural network model or Bag of words model. Simple neural network cannot learn generalized features and requires more time. A bag of words has a disadvantage of having sparse representations and others which are detailed later. Among all of these method the deep neural network methods like the convolution neural network has an innate capability of automatic extraction of features in very less time.

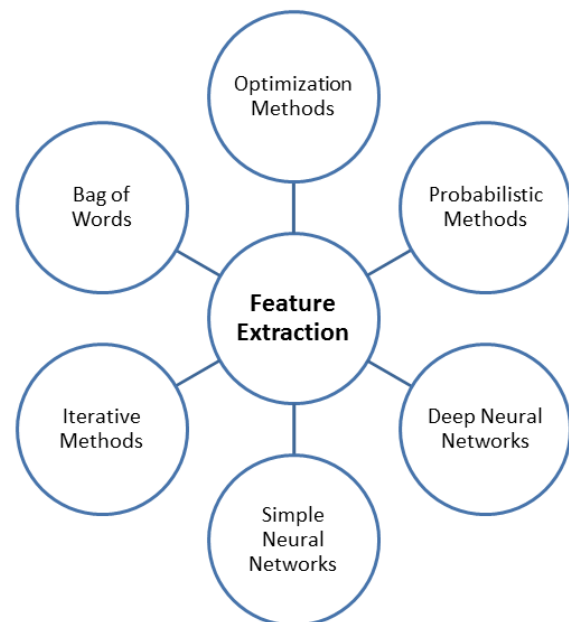


Fig 1. Automatic Feature extraction methods

A. Problem Identification

In this paper we are using a truth discovery based convolution neural network model for predicting the genuineness of Quora questions data set. The problem is that given a dataset of Quora questions one has to identify the toxic content in the text and classify them into genuine question or not a genuine question. Toxic or misleading content in the above problem is to identify questions that have neutral tone, disparaging information or isn't grounded in reality. Identification and classification of such content is

Revised Manuscript Received on July 10, 2019.

Adilakshmi Vadavalli, Research Scholar, School of Computing, Sathyabama Institute of Science and Technology, Chennai, T.N, India. (E-mail: adilakshmi.it@sathyabama.ac.in)

R Subhashini, School of Computing, Sathyabama Institute of Science and Technology, Chennai, T.N, India. (E-mail: subhashini.it@sathyabama.ac.in)

our main criteria.

B. Objectives

Based on the problem identification, this research work has the following objectives:

- To use a word embedding layer to obtain vocabulary from given dataset.
- To define a convolutional neural network model and compile it
- To fit the model to the existing problem
- Evaluate the model based on certain parameters
- Make predictions on the model

C. Organization

The remaining sections in this paper are organized as follows: the existing Natural language processing techniques for deep learning are surveyed in Section II. The clear description about the proposed Truth discovery based convolution neural network model algorithm is presented in Section III. The experimental results and comparative analysis of proposed algorithm is presented in Section IV. Finally, the overall outcomes of the proposed work and the future enhancement that can be performed to improve the results are stated in Section V.

II. RELATED WORKS

In this section, the existing design techniques, strategies, applications and algorithms related to truth discovery for unstructured data are surveyed with its clear merits and demerits.

In [5] the authors proposed a character level Convolution neural network which use small convolutions and pooling operations. It has used around 29 convolution layers which increases the depth of the network and there by performance. It has been applied on 8 freely available large scale datasets which belong to object detection, image classification, sentiment analysis and topic classification etc. It is inspired by the philosophy of VGG and Resnets that two design rules have been applied. For the same output temporal resolution same number of feature maps exist for layers. Also, when temporal resolution is halved the number of feature maps is doubled. In [6] the authors have explained about the speech recognition strategies for sequential data using deep neural networks. They have used recurrent neural networks with connectionist temporal classification and have trained them end to end. The network has been trained on TIMIT phoneme recognition benchmark. The concept here is that the RNNs learn to map directly from acoustic to phonetic sequences. It removes the need for a predefined error alignment. Four main steps that are involved in the process are connectionist temporal classification, RNN transducer, Decoding and Regularization. An extension of their work aims to combine frequency domain convolutional neural networks with deep RNN. In [7] the authors propose a methodology to generate the description of an image in sentences very clearly using a convolution neural network followed by recurrent neural network. It uses a neural probabilistic framework for generating the likelihood of words from image and then training it on recurrent neural network for generating sentences. They have designed a framework in place where given an input using a Long Short

term memory networks with a forget gate to generate a softmax and then a word prediction model. Experimentally the model has been evaluated on the following standard datasets with improved performance Pascal VOC 2008, Flickr8k, Flickr30k, MSCOCO, SBU etc. In [8] the authors proposed a general end to end approach to sequence learning with minimal assumption approach. This conception has been tested on machine translation tasks. Experiments have been conducted on WTM 14 dataset where it is performing an English to french translation task. Reversing the order of the words in all source sentences also improved the performance of the system to a great extent. The two main concepts used are decoding and rescoring. Also the flow of the procedures are to reverse the source of sentences training and parallelization of the data. The experimental results and metrics like the length of the text sentences and the average word frequency rank had great improvement over existing methods. In [9] the authors proposed a connectionist language model a variant of recurrent neural network for on speech recognition systems. These models prove superior to the n gram techniques. But the disadvantage is that it takes a high computational training complexity. This language model proposed also closely connects with other techniques such as machine learning, data compression and cognitive science research. The experiments have been conducted on real time datasets like DARPA WSJ92 and 93. In [10] the author explains about the execution of Natural Language processing programs dealing with lot of important concepts. They proposed the Long Short term memory networks for executing the sequence to sequence framework. Character by character reading and processing for New curriculum learning has been experimented. Four curriculum learning strategies have been discussed namely:

1. No curriculum learning
2. Naïve curriculum strategy
3. Mixed Strategy
4. Combining the mixed strategy with the naïve curriculum learning strategy

Also, Inverting input and doubling input strategies have been adopted to increase performance. Hidden state allocation hypothesis technique has been used for getting better performance as stochastic gradient descent can have poor performance. The combined strategy reduces the need to for excessive memorization. In [11] the author proposes an encoder decoder neural networks which has two recurrent neural networks. It is used for a statistical machine translation system. They are trained together to increase the probability of a target sequence given a source sequence. This model has the advantage of increasing the semantic and syntactic meaning of the linguistic phrases. The main principle of this model is that the hidden unit adaptively remembers and forgets. There are generally three types of models in this:

1. The Translation model
2. Language model
3. Log-Linear model

The model is evaluated on an English to French Translation dataset. The source and the target vocabulary has been limited to 15,000 words. The out of vocabulary words are treated as special tokens. The word and phrase representations have many advantages like they can learn semantically meaningful word embedding. Also, it maps the sequence of words into a continuous space vector. Also, semantically similar words can be clustered together. In [12] In this paper the authors propose a framework with which we can build a deep recurrent neural network. The proposed model outperforms the shallow recurrent neural networks. It can be achieved in two steps:

1. Deep input to hidden function
2. Deep hidden to output function
3. Hidden to Hidden transition function

The four different neural networks that have been illustrated are:

1. A conventional RNN
2. Deep transition RNN
3. Deep output RNN
4. Stacked RNN

They have also used different neural operators which were implemented with a multi-layer perceptron. The model has been used on the below datasets namely:

1. Polyphonic Music Prediction
2. Language modelling

In [13] the authors propose a vector space representation of words which have captured the semantics and syntax using a vector arithmetic. A new global log bilinear regression model is the key aspect behind this representation. The two major model families discussed and on top of which the model is built are:

1. global matrix factorization
2. Local context window

The following are the procedures followed in order to use the statistics to extract the semantic meaning of words.

1. Ratios of Co-occurrence probabilities of words
2. Word to word co-occurrence matrix
3. A weighted least square regression model

The other models that were discussed are the skip gram and ivLBL model. This model proposed is by far the most successfully used method for word embedding. Syntactic and semantic word analogies have also been discussed. In [14] In this paper the authors show recurrent neural networks can be used for handwriting synthesis and predictions on text sequence. A deep neural network prediction architecture has been defined. Experiments have been conducted on Penn Treebank Test Set and results show the perplexity measures which is the usual metric for calculating the performance of the Natural language processing models. Real Wikipedia data has been used to synthesize the handwriting data which is highly realistic and in different styles. In [15] the authors propose language understanding models which can accurately place sentences over distributions and yet encode complexities of the grammatical structure. The fundamental models deal with machine translation, question answering system, text summarization. The Language modelling improvements that are suggested as:

1. Relationship between noise contrastive estimation and Importance sampling.

2. The convolution neural network softmax
3. Character LSTM predictions

Three architecture language models have been discussed as:

1. A standard LSTM language model
2. A language model where both input and softmax embedding have been replaced by a character CNN
3. Next character prediction LSTM network

The parameters on which the models have been evaluated are as follows:

1. Dimensionality of embedding layers
2. The state
3. Projection sizes
4. No of LSTM layers to use
5. Test Perplexity

The disadvantage is that as the dataset size increases and with complex data, the training performance degrades.

In [16] the authors suggested model architecture for computing continuous vector space representation of words in a word similarity task. The main advantage of this proposed approach is that it is very time efficient and the algorithm designed takes less than a day to learn word vectors from a 1.6 billion words dataset. It is also a good test set for measuring syntactic and semantic word similarities. It can further be extended to automatically extend to knowledge bases. In [17] the authors introduced a continuous skip gram model that can efficiently capture a large number of syntactic and semantic representations. The paper proposes how to train distributed representation of words and phrases. Also, models have been trained on large data with subsampling of frequent words using negative sampling algorithm with an advantage of faster training. In [18] the authors propose an attack on the perspective toxic detection system based on adversarial methods. The approaches suggested for improving the robustness of the system are:

1. Adversarial Training
2. Spell checking
3. Blocking

The system has a high false alarm rate in detecting huge toxicity. The highlights of the observations that have been made are:

1. Susceptibility to false alarm
2. Robustness to random misspellings
3. Vulnerability to poisoning attack

In [19] a convolution based neural network text classification algorithm with emphasis on structure of words has been proposed. Toxic comments on large pool of documents provided by Kaggle called Wikipedia's talk page edits have been identified. The type of layers in convolution neural network architectures has been discussed and given as

1. Convolutional layer
2. Pooling layer
3. Embedding layer
4. Fully connected layer

DEEP LEARNING BASED TRUTH DISCOVERY ALGORITHM FOR RESEARCH THE GENUINENESS OF GIVEN TEXT CORPUS

It has been explained that these trainings rely on back propagation training algorithm. Data encoding via vocabulary has also been explained. Two dimensional representation of the Document Text Matrix (DTM) using PCA Principle Component analysis and visualizing using t-SNE t-Distributed Stochastic Neighbor embedding a dimensionality reduction technique. However adaptive learning and n-gram based approaches could have been used for motivating results. In [20] the authors have automated the process of identifying whether a comment or a post if deemed toxic should be flagged. This kind of comment abuse classification has two benefits. One is it saves time and the other is it protects user's safety. This concept has been tested with deep learning models. It has been tested with two models as baseline i.e. general machine learning model and deep learning perspective. The following three models have been employed:

1. An RNN with an LSTM cell and word embeddings
2. A CNN with word embeddings
3. A CNN with character embeddings

However, qualitative features can be evaluated by using learned weight vectors and learned character embeddings. In [21] propose an empirical explanation of ConvNets for text classification. The concepts proposed are:

1. Key module convolution using stochastic gradient descent
2. Character quantization and encoding
3. 2 layered ConvNet model using max pooling and fully connected layers
4. Data Augmentation

The following parameters are deemed to be important based on the results

1. Dataset size
2. Choice of alphabet makes a difference
3. Work well with user generated data
4. Semantics of tasks may not matter
5. Bag of words is a misuse of word2vec

However, these models does not work well when broader range of language processing tasks with structured outputs are needed. In [22] proposes a model where character inputs are sufficient for neural language model. The model takes the output from a single-layer character-level convolutional neural network with max-over-time pooling. Highway network model is also employed to obtain improving results. However, it has to be identified if this model can be employed for encoder/decoder in machine translation. In [23] the authors propose a deep neural networks with multi task learning for obtaining a unified architecture for Natural language processing tasks. They introduce two models namely multi task learning and semi supervised learning for improving the performance. Six standard NLP tasks have been considered as:

1. Part of speech tagging
2. Chunking
3. Named Entity Recognition
4. Semantic Role labeling
5. Language Model
6. Semantically related words

Given an input sequence the output probabilities of the words are choose and a classical window approach is used and the Neural Network layer outputs and max layer

performs an identity. Thus learning tasks can simultaneously improve the syntax feature understanding for semantic extraction. In [24] the authors propose the sensitivity analysis for sentence classification. The work has two major contributions. One is improving the training time and second one is improving the space of model architectures and hyper parameter settings like the input word vector representation, filter regions, number of feature maps, pooling strategy and regularization terms. However, random search and Bayesian optimization framework can help for further hyper parameter simplification. In [25] the author proposes a model for eliminating the curse of dimensionality. They employ a learning distributed representation of words which allows to train sentences about an exponential number of semantically neighboring sentences. The advantage of the model is that it proposes a Parallelized data and feature reading mechanisms in forward and backward phases. It employs an energy minimization work. However, generalization and training speed can be further improved. In [26] the authors propose a neural conversational model for machine learning tasks. The model can generate simple conversations on a large conversational dataset. It can also reason out on a domain specific movie dataset. It can also find solution to a technical problem on an domain specific IT helpdesk dataset. The advantage of this model is that it can be trained end to end but however the model lacks consistency. In [27] the authors propose an anatomy of online hate in online news media. The model automatically detects and classifies online hate. The main concepts proposed as part of the model are as follows:

1. Create a granular taxonomy for hateful online hate
2. Experimenting with machine learning models to automatically detect and categorize hateful comments

The challenges involved in detecting online hate are as follows:

1. Linguistic diversity
2. Contextuality of hate
3. Gaming the system
4. Freedom of speech

However, the following still remain as challenges for classification of online hate:

1. Interpretation problem
2. Linguistic variety
3. Limits of automation
4. Danger of over-moderation

III. PROPOSED METHODOLOGY & RESULTS

In this section, the clear description about the proposed methodology for truth discovery on convolution neural network for equity classification of quora questions has been provided. The motive of this scheme is to increase the level of accuracy when predicting the genuineness of the given text corpus of data from kaggle quora insincere question dataset. The following algorithm strategies have been employed for truth discovery and will be discussed in detail



A. Algorithm for Clean Select and filter vocabulary

B. Algorithm for CNN Truth Discovery Model

C. Algorithm for Evaluation

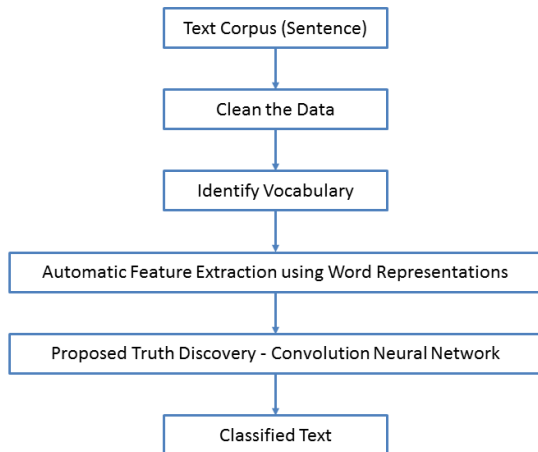


Fig 2. Flow chart of the proposed methodology

A. Algorithm for Clean Select and filter vocabulary

1. Clean the dataset
2. Load the document into memory
3. Convert the document into clean tokens
4. Filename = 'txt_kaggle/sincere/sc000_0000.txt'
5. Select the vocabulary
6. Load the documents and add to vocabulary
7. Load the entire documents
8. Define vocabulary
9. Add all documents to vocabulary
10. Process_the_documents('txt_kaggle/sincere', vocab)
11. Process_the_documents('txt_kaggle/insincere', vocab)
12. Filter the vocabulary
13. Print the size of the vocab
14. print the top words from the vocabulary
15. Print the most common 50 words from vocabulary
16. Keep all the tokens with a minimum occurrence count
17. Print the length of the tokens
18. Save all the tokens to a vocabulary file

In the Algorithm for clean select and filter vocabulary the following things are taken care. The tokens are split on white space. The punctuations from all the words will be removed. Removed the words that are not pure alphabetical characters. Get rid of the stop words. We can remove words whose length is less than or equal to one character.

In Fig 2. The flowchart for the proposed methodology has been given. The entire process starts with data preparation. Then comes splitting of the data into train and test sets. Then a vocabulary of words has been defined. Then an automatic feature extraction of words using word representations has been defined. Then comes the proposed convolution based truth discovery leading finally to the text classification as sincere or insincere.

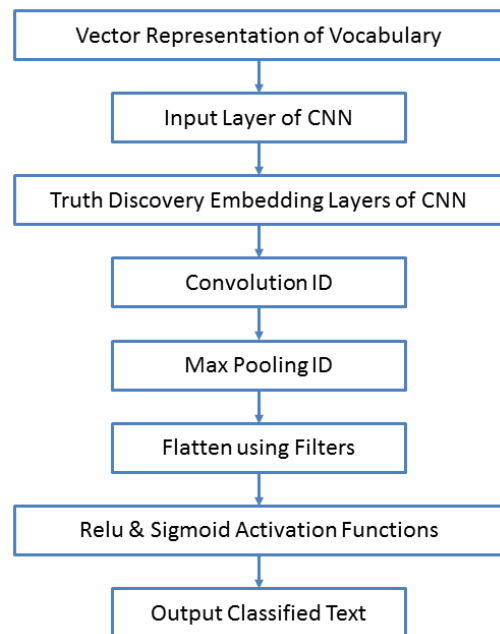


Fig 3. Proposed Truth discovery convolution neural network

In Fig 3. The proposed truth discovery based Convolution neural network model has been explained in detail. First the vector representation of the vocabulary is presented to the input layer of CNN. Then the truth discovery algorithm running the embedding layers of CNN. Then followed by the convolution I, maxpooling ID. Flatten the layers using filters. Relu and sigmoid functions are used in the model. Finally we get classified text.

B. Algorithm for CNN Truth Discovery Model

1. Make a path to open file
2. return documents
3. load and clean a dataset for training
4. load documents both sincere and insincere
5. prepare labels
6. fit a tokenizer on lines of text
7. return tokenizer
8. integer encode and pad sequences of documents
9. return padded sequences
10. Define the cnn model
11. load the vocabulary
12. load training data
13. create the tokenizer
14. define vocabulary size
15. print Vocabulary size
16. calculate the maximum sequence length
17. encode data
18. define model
19. fit network
20. save the model

The algorithm for CNN truth discovery model is as follows: First we load the document into memory. Then we

DEEP LEARNING BASED TRUTH DISCOVERY ALGORITHM FOR RESEARCH THE GENUINENESS OF GIVEN TEXT CORPUS

try to load the vocabulary. We will clean all the train and test data. The embedding layer is used as a specific embedding layer for vector representation of data. These are random in the beginning but will later become more adept. Then using a tokenizer class in an API training documents are encoded. It will generate all tokens from the training dataset and map them to integers. The size is defined and maximum sequence length is calculated. A model is defined and made fit to the The model created is thus saved.

The algorithm for the definition of the model is as follows: The complete model is listed in the algorithm. A CNN embedding layer with 32 filters. Then followed by a pooling layer. 2D output from CNN is made as a 2D vector. Behind the scenes runs the multilayer perceptron's to know the features. Sigmoid activation function is used as output.

C. Algorithm for Evaluation of Model

1. classify a Output as insincere or sincere
2. predict_genuity(Output, vocab, tokenizer, max_length, model):
3. clean Output
4. encode and pad Output
5. predict genuinity of yhat model
6. load the vocabulary
7. load all Outputs train and test
8. create the tokenizer
9. define vocabulary size
10. calculate the maximum sequence length
11. encode data
12. load the model
13. evaluate model on both train and test data

The algorithm for the Evaluation of the model is as follows: The complete evaluation model is listed in the algorithm. The output is classified as sincere or insincere. The output is encoded and padded. All outputs are loaded for training and testing. The data is encoded and the model is evaluated on both training and testing datasets.

Sample test Output

Sample test output

test sincere text

text = 'What are the job prospects after doing an MSc in Chemistry or Biology?'

percent, sincere = predict_genuinity(text, vocab, tokenizer, max_length, model)

print('Output: [%s]\ngenuinity: %s (%.3f%%)' % (text, genuinity, percent*100))

test insincere text

text = 'How do I sell Pakistan? I need lots of money so I decided to sell Pakistan any one wanna buy?'

percent, genuinity = predict_genuinity(text, vocab, tokenizer, max_length, model)

print('Output: [%s]\ngenuinity: %s (%.3f%%)' % (text, genuinity, percent*100))

The sample test output and a particular output are shown as follows: It can be seen that the model

produces 100% accuracy on the training data and 87% accuracy on the testing data. The model makes a correct prediction on two sample quora questions and the confidence percentages are 76% and 69% respectively on sincere and insincere claims

Output

Vocabulary size: 26768

Maximum length: 1562

Train Accuracy: 100.00

Test Accuracy: 87.12

Text: [What are the job prospects after doing an MSc in Chemistry or Biology?]

Genuinity Prediction: POSITIVE (76.7%)

Text: [How do I sell Pakistan? I need lots of money so I decided to sell Pakistan any one wanna buy?]

Genuinity prediction: NEGATIVE (69.415%)

Vocabulary size: 25768

Maximum length: 1317

IV. CONCLUSION AND FUTURE WORK

The genuinity of the given text corpus has been predicted using the proposed CNN truth discovery model. The model makes correct predictions for the quora questions classification dataset. We can see that the percentage confidence for the prediction is closed to 60% for both positive as well as negative texts. However this is a very basic level truth discovery that has been proposed. Further many enhancements can be carried out in different aspects to the existing work like the Better data cleaning like leaving punctuations intact. Exploring for further reduction of the size of the vocabulary for better effect on skill model. Filters and kernel size parameters have to be tuned further. This in this proposed approach the model fits the training data very quickly but we could explore different epochs and batch sizes for picking a better point to stop for training. We can explore a deeper network to train.

REFERENCES

1. D. A. Waguih and L. Berti-Equille, "Truth discovery algorithms: An experimental evaluation," ArXiv Prepr. ArXiv14096428, 2014.
2. "A survey on Truth Discovery.pdf."
3. L. Berti-Equille, "Data veracity estimation with ensembling truth discovery methods," in 2015 IEEE International Conference on Big Data (Big Data), 2015, pp. 2628–2636.
4. B. Zhao and J. Han, "A probabilistic model for estimating real-valued truth from conflicting sources," Proc QDB, 2012.
5. A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for text classification," ArXiv Prepr. ArXiv160601781, 2016.
6. A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in Acoustics, speech and signal processing (icassp), 2013 IEEE international conference on, 2013, pp. 6645–6649.
7. O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3156–3164.



8. 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.
9. T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, "Recurrent neural network based language model," in Eleventh Annual Conference of the International Speech Communication Association, 2010.
10. W. Zaremba and I. Sutskever, "Learning to execute," ArXiv Prepr. ArXiv14104615, 2014.
11. K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," ArXiv Prepr. ArXiv14061078, 2014.
12. R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," ArXiv Prepr. ArXiv13126026, 2013.
13. J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.
14. A. Graves, "Generating sequences with recurrent neural networks," ArXiv Prepr. ArXiv13080850, 2013.
15. R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu, "Exploring the limits of language modeling," ArXiv Prepr. ArXiv160202410, 2016.
16. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," ArXiv Prepr. ArXiv13013781, 2013.
17. 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.
18. H. Hosseini, S. Kannan, B. Zhang, and R. Poovendran, "Deceiving Google's Perspective API Built for Detecting Toxic Comments," ArXiv Prepr. ArXiv170208138, 2017.
19. S. V. Georgakopoulos, S. K. Tasoulis, A. G. Vrahatis, and V. P. Plagianakos, "Convolutional Neural Networks for Toxic Comment Classification," ArXiv Prepr. ArXiv180209957, 2018.
20. T. Chu, K. Jue, and M. Wang, "Comment Abuse Classification with Deep Learning," Von Httpsweb Stanf. Educlasscs224nreports2762092 Pdf Abgerufen, 2016.
21. Anu, V. Maria, M. I. Deepika, and L. Mary Gladance. "Animal identification and data management using RFID technology." In International Conference on Innovation Information in Computing Technologies, pp. 1-6. IEEE, 2015.
22. Y. Bevish Jinila, K. Komathy (2015), "Rough Set Based Fuzzy Scheme for Clustering and Cluster Head Selection in VANET", ELEKTRONIKA IR ELEKTROTECHNIKA, Vol.21, No.1, pp.54-59, ISSN : 1392-1215
23. Gladance, L. Mary, M. Karthi, and V. Maria Anu. "A statistical comparison of logistic regression and different Bayes classification methods for machine learning." ARPN Journal of Engineering and Applied Sciences 10, no. 14 (2015): 5947-5953.
24. Anu, V. Maria, GS Anandha Mala, and K. Mathi. "Comparison of RFID data processing using dimensionality reduction techniques." In 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), pp. 265-268. IEEE, 2014.
25. Sanjanna, Y. Bevish Jinila (2015), "An Approach on Automated Rescue System with Intelligent Traffic Lights for Emergency Services", International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS 2015), pp. 1-5, IEEE
26. 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.
27. Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-Aware Neural Language Models," in AAAI, 2016, pp. 2741–2749.
28. R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in Proceedings of the 25th international conference on Machine learning, 2008, pp. 160–167.
29. Y. Zhang and B. Wallace, "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification," ArXiv Prepr. ArXiv151003820, 2015.
30. Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," J. Mach. Learn. Res., vol. 3, no. Feb, pp. 1137–1155, 2003.
31. O. Vinyals and Q. Le, "A neural conversational model," ArXiv Prepr. ArXiv150605869, 2015.
32. J. Salminen et al., "Anatomy of Online Hate: Developing a Taxonomy and Machine Learning Models for Identifying and Classifying Hate in Online News Media.," in ICWSM, 2018, pp. 330–339.