# Chapter 2: Literature Review

Alexander Tan Ka Jin MCS241002

14 December 2024

# Literature Review

## Polarity of language within politics

Politics heavily uses language. It is mainly communicated using language, politicians use certain words, languages and mannerism to shape and influence public support for policies and discourse within politics is frequently a site of analysis for many journalists and analysts. This language can have various sentiments, ranging from anger to hope, from praise to condemnation, parties, politicians and citizens can have strong opinions on these subjects and is communicated through textual opinions on social media, newspapers and speeches. Given this, the analysis of these opinions is done through the analysis of the language of politics like in (Orellana & Bisgin, 2023) where opinions can be classified in terms of importance to a given political group. A significant amount of these opinions are communicated through online social media platforms and newspaper like Twitter (now also known as X), which provides an opportunity for textual analysis using computers. (Németh, 2023)'s review of political polarity research using Natural Language Processing (NLP) notes that language polarity often uses political polarity as a basis for it's measurement, even though political polarity is not a well-defined term. Such political polarity measures include the four measures by (DiMaggio, Evans, & Bryson, 1996), 1. variance of opinion, 2. bimodality of opinion, 3. ideological constraint of opinion, 4. differences in in-group opinion. From there, language polarity can be measured as the propensity for political groups to hold to a certain group. That being said, most papers utilize polarity as the strength and attitude of a given opinion between negative opinion and positive opinion such as (Alshutayri et al., 2022).

| Sentence | Polarity |
|---|---|
| Stupid storm. No river for us tonight | Negative |
| that's great!! weee!! visitors! | Positive |
| looking forward to body works today | Neutral |

Table 1: Examples of sentiment from tweets on Twitter. Gathered from Kaggle Dataset: (`https://www.kaggle.com/datasets/abhi8923shriv/sentiment-analysis-dataset` Accessed 22:50 14/12/2024)

## Use of NLP in Text Mining

(Nadkarni, Ohno-Machado, & Chapman, 2011) notes in their introduction that NLP originated as the intersection of artificial intelligence (AI). (Liddy, 1998) describes NLP as a range of techniques for processing natural text in hopes of achieving human-like text comprehension. This subject was initially distinct from Information Retrieval (IR), which was concerned with efficiently identifying and querying large corpora of text. It's integration with IR occurred with a need for identifying underlying meanings of text, often in regards to text that contain non-conventional uses of language such as sarcasm or irony that contain semantics outside of it's literal meaning.

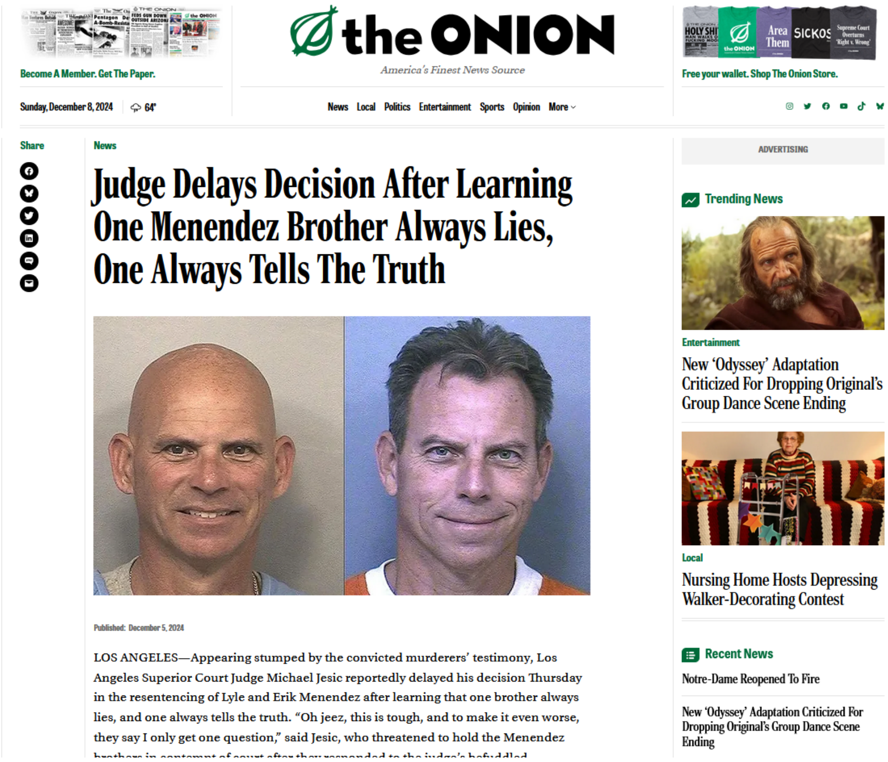


Figure 1: Screenshot of a headline from The Onion, a satirical news site. Satire often employs use of ironic language that can be mistaken to be literal. A machine learning model may falsely label this as genuine news. (`https://theonion.com/judge-delays-decision-after-learning-one-menendez-brother-always-lies-one-always-tells-the-truth/` Accessed 16:00, 9/12/2024)

### Text Mining

Similar to IR, Text Mining is the discovery of patterns within unstructured textual data, it deals with the processing and comprehension of complex textual patterns in order to uncover hidden trends. Like IR, Text Mining deals with large corporea of text and is concerned with the complexities of language.(Stavrianou, Andritsos, & Nicoloyannis, 2007) The rules of language can be complex and change frequently to fit the cultural context of it's speakers. (AbuSa'aleek, 2015) notes that internet users have a change in words and expression in English internet discourse ranging from shortening of words to the use of emoticons while (Bengio, Ducharme, & Vincent, 2000) mentions that traditional language models suffer greatly from the curse of dimensionality and have decreases in efficiency owing to the number of words in a given vocabulary. The analysis of text has to adjust to these changes in both morphology and semantics while also accommodating the diversity of expressions and words.

Due to the complexities of text, much of NLP have moved away from statistical models (n-gram, maximum entropy) into the domain of Machine Learning utilizing the presence of larger corpora with the development of models like BERT and GPT-3, achieving higher results than traditional models. (Wei, Wang, Wang, & Kuo, 2023) The use of Machine Learning helps tremendously with pattern recognition and mining of textual data. Still, more research is to be done on more complicated use of words in different languages and context (such as humour or disgust and hate) (Farabi, Ranasinghe, Kanojia, Kong, & Zampieri, 2024).

**Semantics**

This thesis concerns mainly with the semantics of language. Semantics is one of the main disciplines of language along with phonetics (individual sounds), phonology (the organization of sounds), morphology (the formation of sound into words), syntax (the structure of sentences), pragmatics (how information is conveyed through language), discourse (use of language in conversations) and socio-linguistics (how society influences language). These components of language are listed in the Handbook of Linguistics (Aronoff & Rees-Miller, 2020), where semantics is listed as the study of meaning developed through words and sentences. Though semantics is mainly thought to come from words and phrases, the meaning of a given sentence can change depending on any of the other components of language like discourse, phonology and socio-linguistics.

(a) Thanks again for the full fries!

(b) Another perfect pizza from <user>!

Figure 2: Examples of how social media users may post sarcastic messages. The semantics of the sentence may be literal but is rendered not due to the juxtaposition of the images. Similarly, the discussion surrounding a conversation may render a message sarcastic. (Farabi et al., 2024)

## Sentiment Analysis

Sentiment is the emotional reaction of a given person. In text, we often communicate our emotions and opinions using words with specific connotations we learn. Examples of words that communicate strong sentiment may include direct emotions (Happy, Sad, Afraid), objects that evoke emotions (Baby, Rain, Rainbow) or adjectives (Cool, Warm, Bitter). These opinions can be all sorts of different emotions ranging from anger, sadness to joy and surprise. Important to the analysis is how subjectively or objectively said are the opinions, the emotion/polarity classification of a text and the strength of a specific opinion (Taboada, 2016).

### Goal of Sentiment Analysis

The purpose of Sentiment Analysis (also known as Opinion Mining) is to uncover and summarize the various opinions hidden within large corpora of text in order to discover cultural and societal opinions on subjects. Papers such as (Singh, Imam, Wibowo, & Grandhi, 2022) classified tweets on Twitter during the Covid-19 pandemic with the labels of joy,sad,fear and anger using various machine learning models in order to help organizations adapt to current world events. From sentiment analysis, research can uncover the trends in opinion within given subjects like pandemics or opinions on political/cultural issues such as dementia.(Kong et al., 2022)


Figure 3: Examples of various emotional sentiments in Weibo messages and their sentiment labels. (Li & Li, 2023)

### Machine Learning based Sentiment Analysis

Of the methods in sentiment analysis, two are most significant, Machine Learning approaches and Lexicon-Based approaches.(Taboada, 2016) A majority of Sentiment Analysis utilizes Machine Learning/Deep Learning methods. These methods utilize a textual dataset that is transformed by Feature Engineering to emphasize relevant information. The machine learning models then train and classify based on the transformed dataset. Machine learning approaches are more popular within the domain. As (Rodríguez-Ibánez, Casánez-Ventura, Castejón-Mateos, & Cuenca-Jiménez, 2023) notes, machine learning became more popular when vector embedding was introduced where words are transformed into vector space, allowing a mathematical representation of language. Deep learning, SVMs and Bayesian methods are some of the most commonly used techniques for sentiment analysis with more recent deep learning models like Long Short-term Memory (LSTM) and Transformers like Generative Pre-trained Transformers (GPT) and Bidirectional Encoder Representations from Transformers (BERT) achieving higher results. This thesis will use this branch of sentiment analysis and several deep learning and machine learning models will be discussed in detail later in the thesis.

**Lexicon-based Sentiment Analysis**

These methods differs from lexicon based approaches that develop rules based on dictionaries and text and separate the data according to the rules. Mentioned in (Saberi & Saad, 2017), a lexicon-based method like semantic orientation would use k-means clustering to group documents and sentences based on a set of positive/negative words to form a dictionary, the text would be then be weighted upon using term frequency-inverse document frequency (TF-IDF) and a voting technique to measure it's importance and fine-tune the dictionary. The key advantage that Lexicon-based approaches have was their ability to be applied to a wide variety of corpora without changes to the dictionary (Taboada, Brooke, Tofiloski, Voll, & Stede, 2011) allowing for easy reuse. But with increases in computational capabilities, recent research is heavily skewed towards Machine Learning methods.

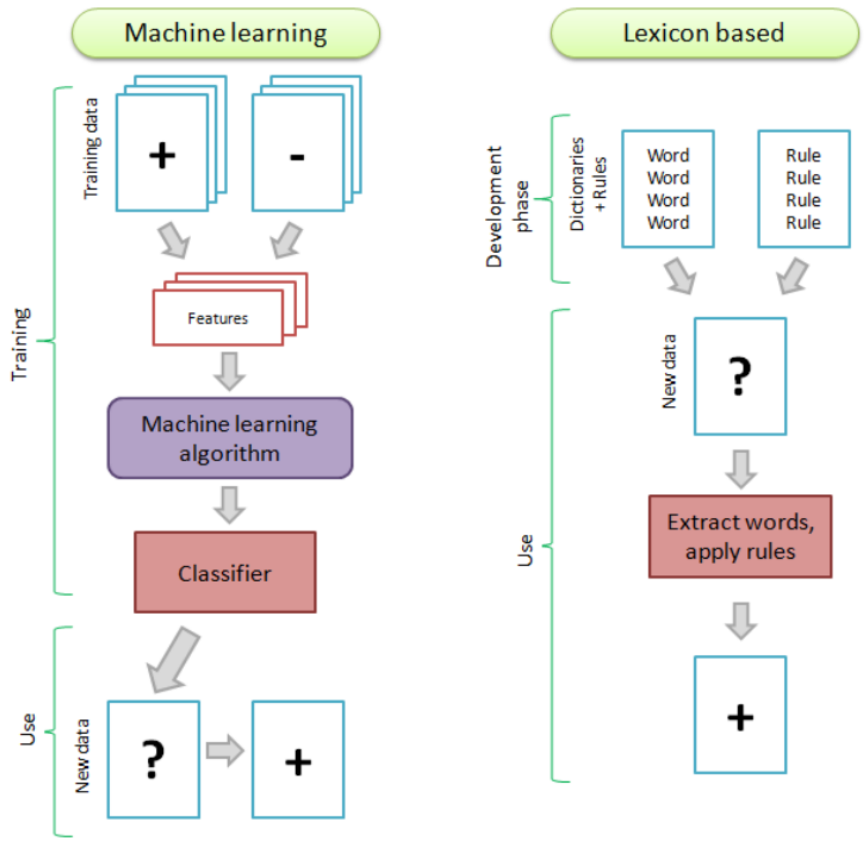


Figure 4: Difference between Machine Learning methods and Lexicon-based methods. (Taboada, 2016)

**Challenges and Research Gaps of Sentiment Analysis**

The main challenge to all sentiment analysis is non-literal use of language. As stated above, sarcasm, irony and satire all use language in a way that communicates the opposite of it's literal meaning. (Maynard & Greenwood, 2014) analysed tweets using labels of polarity, strength of opinion and sarcasm and showed that accounting for sarcasm could have an improvement on Text Mining systems. In terms of political humour, (Frenda et al.,

2018) noted that aspects of bigotry like misogyny are often transmitted through sarcastic jokes, identification of sarcasm and correct labelling of sentiment may be important for political jokes that demean groups of people.

## Supervised Machine Learning/Deep Learning Classification Models

The domain of Artificial Intelligence (AI) is largely split between two types of classifications. (Janiesch, Zschech, & Heinrich, 2021) distinguishes these two classifications as the model type (Machine Learning (ML) methods which includes Artificial Neural Networks (ANN) which includes Deep Learning (DL) methods) and the learning type (supervised, unsupervised and reinforcement learning).

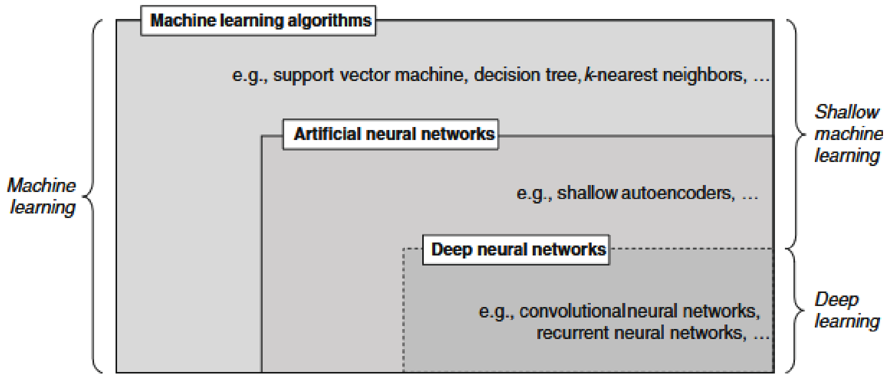


Figure 5: Venn diagram grouping on ML Methods. DL is a subset of ML that utilizes complicated neural networks. (Janiesch et al., 2021)

Much of sentiment analysis is done through supervised classification ML/DL such that the model is able to recognize the concept of polarity and emotional labels. Due to this, the focus of this section is mostly on supervised learning, this type of learning occurs when models are first trained on a set of data with labelled outputs (positive/negative, emotional labels, etc) and then utilized for use and evaluation in unlabelled data. In general, both supervised ML and DL follow the same basic methodology:

1. Labelled Data Input

   Data is labelled with it's correct output and fed to the model. This directs the model so that it is useful for classification.

2. Model Training

   The model is trained to identify patterns within the data provided. The results of the model are transformed by the loss function into a value that is gradually optimized. For deep learning models this is usually a form of gradient descent. (Andrychowicz et al., 2016) The next sections will go into detail on how ML/DL models are gradually trained to identify patterns and the specific architecture of these models.

3. Model Evaluation

   The model is evaluated based on it's performance. Common evaluation metrics for classification models include Precision, Recall, Accuracy, F1-score and Receiver Operating Characteristic(ROC)-Curve

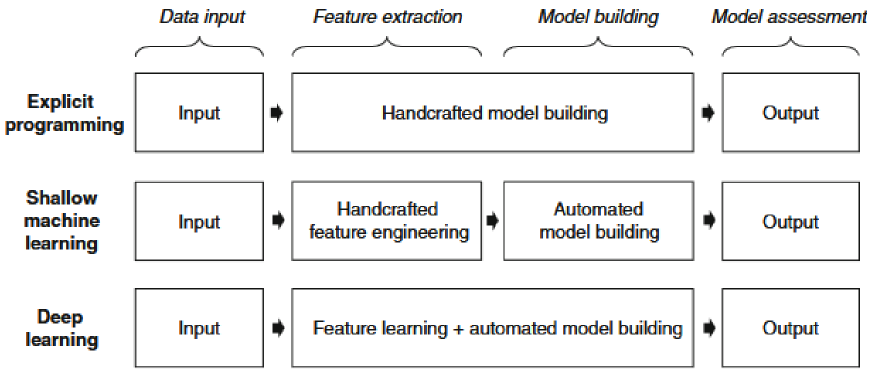


Figure 6: Comparison of explicit programming methods, machine learning and deep learning. (Janiesch et al., 2021)

## Neural Networks

Artificial Neural Networks (ANN) are at the heart of deep learning. They are composed of several layers of artificial neurons within hidden layers. Each neuron has weight inputs and transforms them via an activation function of choice (sigmoid, tanh, softmax, so on.) The output weight is then fed to the next layer, continuing until it reaches the end of network where the value of the prediction is calculated by a loss function (usually Mean Square Error) against it's training dataset. (Zakaria, Mabrouka, & Sarhan, 2014) The network "learns" and self-corrects by performing back-propagation. Starting from the output layer, back-propagation derives every layer in order to minimize the loss of the model.

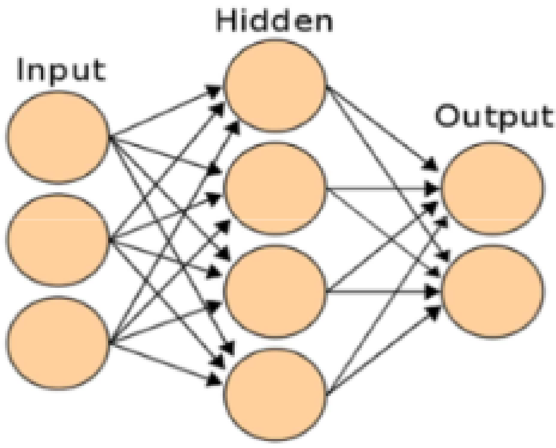


Figure 7: Illustration of a simple ANN model. (Islam, Chen, & Jin, 2019)

## Feature Extraction Methods

The main distinction of DL methods from other ML methods like Support Vector Machines (SVM) is the reduced need for feature engineering, that is the preprocessing of data in order to reduce the dimensionality of it such that the ML model can achieve suitable accuracy. (Zebari, Abdulazeez, Zeebaree, Zebari, & Saeed, 2020)

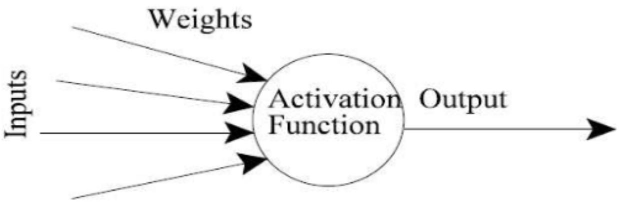


Figure 8: Illustration of an artificial neuron. The neuron is fed weights from previous layers and transforms it into an output by an activation function. (Zakaria et al., 2014)

DL Models do not suffer from the requirement of feature engineering due to it's ability to find features by itself. As (LeCun, Bengio, & Hinton, 2015) notes, DL models operate on successive feature extraction layers which can find significant features of data by continuously abstracting from higher representative patterns. Given that textual data and language contains high dimensional data such that multiple factors can affect the semantics of a given sentence, it is easy to see how the self extraction of significant features in DL can result in improved performances in both Sentiment Analysis and NLP. That being said, DL methods still require the use of feature extraction in order to convert textual data into mathematical representations before the DL model can extract more abstract features. (Chaudhary et al., 2023)'s overview of Machine Learning and Deep Learning method in Sentiment Analysis for Twitter data noted that a majority of Deep Learning methods utilize Word2Vec, TF-IDF and GloVe in order to transform text into mathematical representations.

**Term Frequency-Inverse Document Frequency (TF-IDF)**

TF-IDF weights words based on their importance within a given document. This is based on the frequency of the word within the document and within the corpus. The weight of the word is increased based on it's frequency of within the document while simultaneously decreased by it's frequency within the wider corpus.(Lubis, Nasution, Sitompul, & Zamzami, 2021) The total equation is given as:

$$TF - IDF_{t,d} = tf_{t,d} \cdot idf_t$$

For word $t$ in document $d$, $tf_{t,d}$ is the weight of importance based on it's frequency within document $d$ while $idf_t$ is the inverse weight of a term based on $df_t$, the frequency of documents that contain word $t$ against the number of documents within the corpus, $N$. There are a variety of different methods to calculate the two terms:

| Term Frequency $tf_{t,d}$ | | Inverse Document Frequency $idf_t$ | |
|---|---|---|---|
| natural | $f_{t,d}$ | no frequency | 1 |
| logarithmic | $1 + \log(f_{t,d})$ | idf | $log \frac{N}{df_t}$ |
| augmented | $0.5 + \frac{0.5 f_{t,d}}{mode_d}$ | probabilistic | $max\{0, log \frac{N - df_t}{df_t}\}$ |
| boolean | $f(x) = \begin{cases} 1 & \text{if } f_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$ | | |

Table 2: List of tf-idf methods documented by (Schütze, Manning, & Raghavan, 2008)

For $f_{t,d}$ being the frequency of term $t$ in document $d$, $mode_d$ being the count of the most frequent term in document $d$. $N$ being the total number of documents in the corpus, $df_t$ being the number of documents containing term $t$.

A common method of calculating $tf_{t,d}$ is the logarithmic method, where the importance of words is not measured linearly like in natural count. Another method is by augmented $tf_{t,d}$ where the weight of importance is measured against the frequency of the most common word in the document.

**Word2Vec**

Until Word2Vec, most techniques for extracting textual features ignore the surrounding context of a given word like TF-IDF. Word2Vec is a technique for transforming text into representational vectors that utilizes one of two simple neural network models, Continuous Bag of Words (CBOW) and Skip-gram. (Ma & Zhang, 2015) notes that CBOW predicts the appropriate words given adjacent words (word order does not matter) and Skip-gram does the opposite, predicting the adjacent words using the words given. Using either of the two models, Word2Vec constructs a lexicon of words from the dataset and then assigns weight to each word to a point in vector space, such that similar words are placed closed to each other while dissimilar words are placed further apart. Due to it's ability to consider the context of a word, Word2Vec has been popularly adopted as a feature extraction method for many NLP tasks.

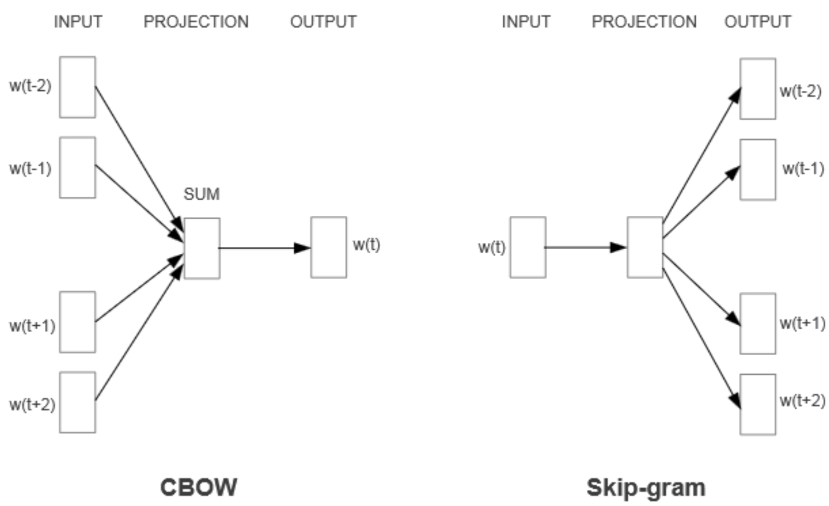


Figure 9: Illustration of Word2Vec's two models, CBOW and Skip-gram. CBOW takes words and predicts the most appropriate word in the context. Skip-gram takes a word and generates the most appropriate words outside of the input word. Both of these methods will produce a vector of weights that represents it's similarity to other words in vector space. (Mikolov, 2013)

**Global Vectors (GloVe)**

Word2Vec is a powerful method for feature extraction of text, but it focuses on small windows of words and ignore the global view of how words interact. As introduced in (Pennington, Socher, & Manning, 2014), GloVe is a method that uses a large global matrix to represent the probability of any given word co-occurrence represented by the conditional probability:

$$g_{i,j} = P(w_i|w_j) = \frac{P(w_i \cap w_j)}{P(w_j)}$$

For $i$-th row, $j$-th column within the GloVe matrix, $g_{i,j}$ is represented by the probability that words $w_i$ and $w_j$ co-occur in a sentence/phrase over the probability that word $w_j$ is present with a sentence. The matrix is then factorized by the equation to extract the weights of the words in a given sentence. This allows for extraction of the weights through a simple mathematical operation at the cost of utilizing large matrices that scale quickly with the number of words in a given corpus.

$$\begin{array}{l} \text{The} \\ \text{Cat} \\ \text{Sat} \\ \text{On} \\ \text{Mat} \end{array} \begin{bmatrix} 5 \\ 3 \\ 4 \\ 3 \\ 1 \end{bmatrix} = \begin{array}{|c|c|c|c|c|c|} \hline & \text{the} & \text{cat} & \text{sat} & \text{on} & \text{mat} \\ \hline \text{the} & 2 & 1 & 2 & 1 & 1 \\ \hline \text{cat} & 1 & 1 & 1 & 1 & 0 \\ \hline \text{sat} & 2 & 1 & 1 & 1 & 0 \\ \hline \text{on} & 1 & 1 & 1 & 1 & 1 \\ \hline \text{mat} & 1 & 0 & 0 & 1 & 1 \\ \hline \end{array} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Figure 10: Demonstration of matrix factorization for GloVe Matrices. The importance of each word is extracted given that words "the" "cat" "sat" (represented as the vector on the right) appear within a sentence. This example may be misleading as it is preferable to understand co-occurrence matrices as a matrix of conditional probabilities. Example GloVe co-occurrence matrix given by (Hindocha et al., 2019)

## Recurrent Neural Networks (RNN)

Although the utilization of ANNs within NLP and consequently, sentiment analysis, have been massive benefits in identifying patterns within unstructured textual data. That being said, ANNs do not have the ability to remember information from previous contexts. This makes the recognition of patterns in highly dynamical systems of information (such as language or speech recognition) achieve less than desirable accuracy. Due to this, the use of RNN is deemed to be an improvement on these subjects. The main modification of RNNs is the presence of prior knowledge from previous layers, which allows RNNs to have limited knowledge of previous words and phrases and making use of the order of sentences and words. (Tarwani & Edem, 2017)

Some recent studies have been using bidirectional RNNs that promised increased results compared to single directional RNNs, at the expense of increased computational time during back-propagation. (Mahadevaswamy & Swathi, 2023)

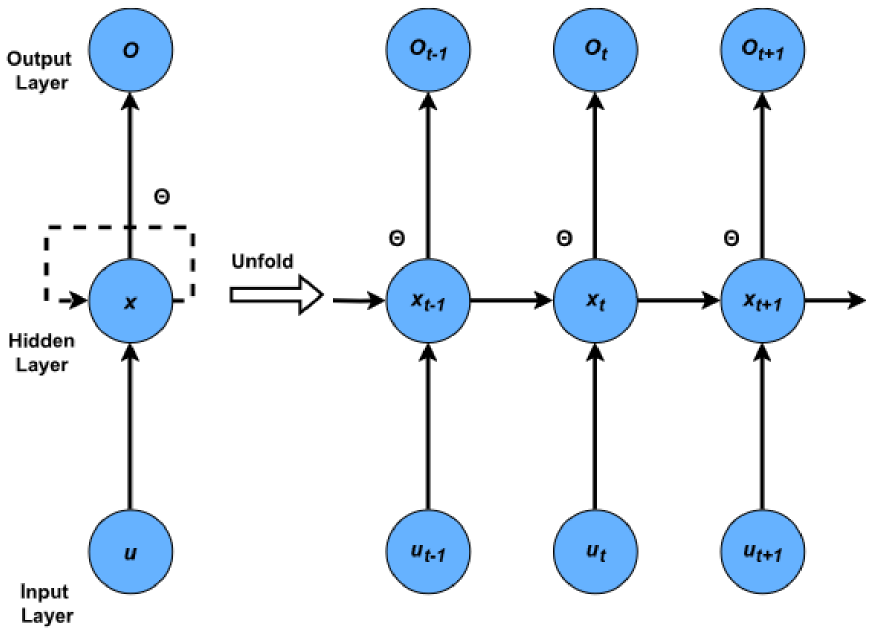


Figure 11: Illustration of RNN architecture. The left image depicts the neural network containing a hidden layer that loops back information from the previous step. When unfolded, information from time $t-1$ is fed to the RNN at time $t$. Information at time $t$ is fed to time $t+1$ and so on. (Alhajeri et al., 2024)

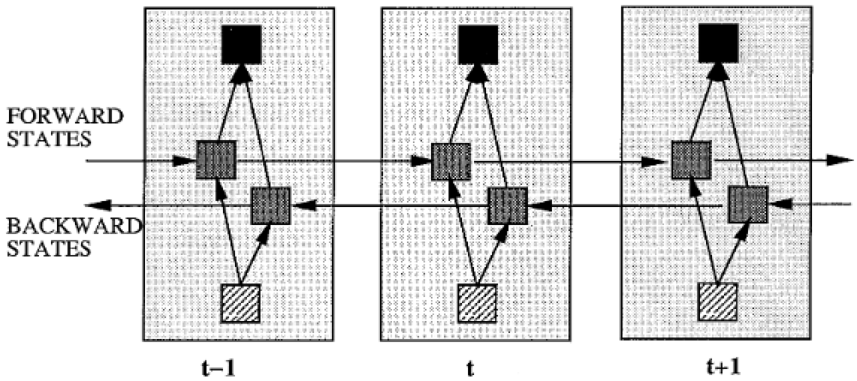


Figure 12: Illustration of Bidirectional RNNs, this applies to subsequent models like LSTM and GRU. The information is passed through two RNNs understanding context from two directions. (Schuster & Paliwal, 1997)

**Long Short-Term Memory (LSTM)**

(Noh, 2021) noted that the standard RNN model has difficulty learning information in the long run. The training improvement will gradually vanish and long term training of RNNs will bring about increasingly diminishing returns. Similarly, the RNN may explode in learning gradient that equally leads to a lack of training progress. These two issues were described by (Bengio, Simard, & Frasconi, 1994). The problem comes with the use of back-propagation, the gradient during back-propagation. (Pascanu, Mikolov, & Bengio, 2013) examined the gradient of back-propagation in RNNs and highlighted that the cause of vanishing/exploding gradients was due to repeated multiplications in the differential between RNN states. A common solution to this is to utilize a variant of RNN called LSTM (Hochreiter, 1997). LSTM does not address with the exploding gradient, again as

$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \le t \le T} \frac{\partial \mathcal{E}_t}{\partial \theta} \tag{3}$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \le k \le t} \left( \frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \tag{4}$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \ge i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \ge i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \tag{5}$$

Figure 13: The issue with back-propagation in RNNs as highlighted by (Pascanu et al., 2013). In the general back-propagation chain, $\frac{\partial \epsilon}{\partial \theta}$, it is required that it also account for the gradient between two states of a RNN, $\frac{\partial x_t}{\partial x_k}$. This differential is comprised of only multiplications of gradient between the two states, making it susceptible to greatly amplifying/diminishing the gradient of the network over time.

noted by (Pascanu et al., 2013), but it does adequately address the vanishing gradient problem by providing a long term memory channel, allowing the results of older states to leak into new states. The model utilizes two channels, a long term memory channel $c_t$ and a short term memory channel, $h_t$ (illustrated in Figure 13 as the top channel and bottom channel respectively). The model is designed such that the results of long term memory will leak into short term memory and short term memory will modify long term memory. This allows for a broader view than RNNs and allows significant words and phrases in the beginning of a sentence/document to have greater effect on the prediction overall. (Lakretz et al., 2019) notes that LSTMs are able to capture specific linguistic information, that models contain units that are able to capture information such as subject-verb dependency.

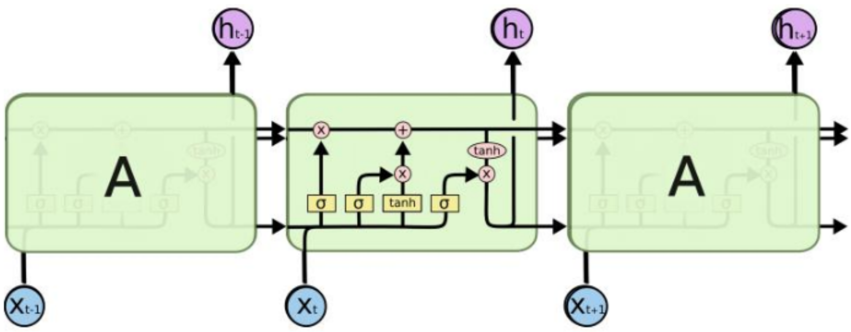


Figure 14: Simplified illustration of the LSTM model. (Tarwani & Edem, 2017)

### Gated Recurrent Units (GRU)

GRUs are a subset of LSTMs that are simpler than the traditional LSTM model. The main difference is the combination of the long-term memory channel and short-term memory channel into a single channel. The model is worse than LSTM by a bit, but has demonstrated improved running times. (Shewalkar, Nyavanandi, & Ludwig, 2019)

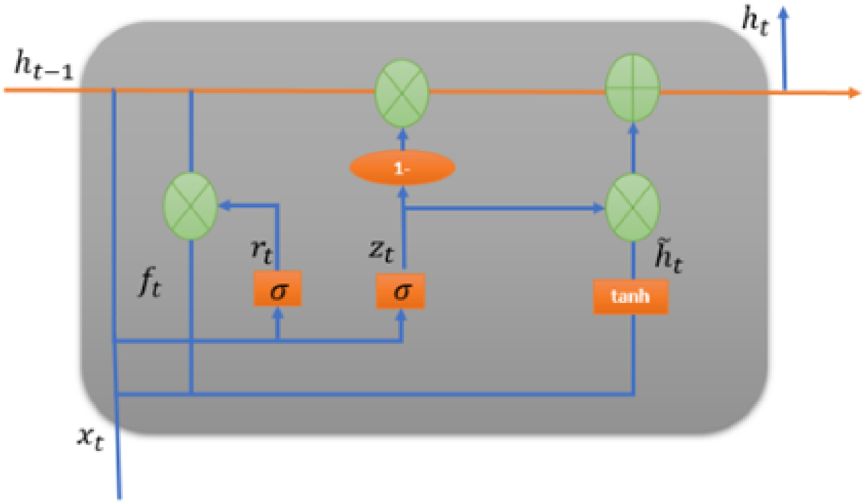


Figure 15: Simplified illustration of the GRU model. (Mateus, Mendes, Farinha, Assis, & Cardoso, 2021)

### Literature of LSTM use in sentiment analysis

| Paper | Dataset Language | Model | Accuracy | Precision | Recall |
|---|---|---|---|---|---|
| (Singh et al., 2022) | English | LSTM-RNN with attention mechanism | 0.8456 | 0.8234 | 0.8212 |
| (Said, Tawfik, & Makhlouf, 2023) | English | LSTM-2BiGRU | 0.9246 | 0.9297 | 0.9194 |
| (Mu et al., 2024) | English | IDBO-CNN-BiLSTM | 0.8033 | 0.8235 | 0.8207 |
| (Eyvazi-Abdoljabbar, Kim, Feizi-Derakhshi, Farhadi, & Abdulameer Mohammed, 2024) | Farsi | Ensemble of CNN and LSTM Hybrid models | 0.7234 | 0.72069 | 0.72198 |
| (Halawani, Mashraqi, Badr, & Alkhalaf, 2023) | English | ABiLSTM | 0.8425 | 0.8583 | 0.8637 |
| (Yin et al., 2023) | English | DPG-LSTM | | 0.842 | 0.845 |
| (Li & Li, 2023) | Chinese | CNN-BiLSTM | 0.8778 | 0.8946 | 0.8841 |
| (Nguyen, 2024) | English | BiLSTM, with GloVe | 0.8218 | 0.8219 | 0.8218 |
| (Kastrati, Kastrati, Shariq Imran, & Biba, 2024) | English with Emojis | BiLSTM with attention mechanism | 0.7022 | 0.7093 | 0.7091 |
| (Vernikou, Lyras, & Kanavos, 2022) | English | LSTM with BERT tokenizer | 0.91 | 0.91 | 0.91 |
| (Vanam & Raj, 2023) | English | BET Tokenizer and H-LSTM | 0.99 | 0.98 | 0.97 |
| (Lin, Zhang, Wu, & Chen, 2023) | Chinese | BiGRU-Att | 0.9758 | 0.98 | 0.98 |
| (Tabinda Kokab, Asghar, & Naz, 2022) | English | BERT embedding and BiLSTM | 0.95 | 0.96 | 0.97 |

Table 3: List of LSTM/GRU models used in literature.

From the table, we can see that LSTMs are effective at classifying text into sentiment with most models reaching 80% to 90% in evaluation metrics. That being said most of the recent models utilize a hybrid model with the only non-hybrid model being BiGRU and BiLSTM. From the table, some of the best performing models are relatively simple LSTM/BiLSTM and BiGRU models.

## References


AbuSa'aleek, A. O. (2015). Internet linguistics: A linguistic analysis of electronic discourse as a new variety of language. *International journal of English linguistics*, *5*(1), 135.

Alhajeri, M. S., Ren, Y. M., Ou, F., Abdullah, F., & Christofides, P. D. (2024). Model predictive control of nonlinear processes using transfer learning-based recurrent neural networks. *Chemical Engineering Research and Design*, *205*, 1–12.

Alshutayri, A., Alghamdi, A., Nassibi, N., Aljojo, N., Aldhahri, E., & ABOULOLA, O. (2022). Sentiment analysis for arabic tweet about the covid-19 worldwide epidemic. *Romanian Journal of Information Technology and Automatic Control*, *32*(2), 127–136.

Andrychowicz, M., Denil, M., Gomez, S., Hoffman, M. W., Pfau, D., Schaul, T., ... De Freitas, N. (2016). Learning to learn by gradient descent by gradient descent. *Advances in neural information processing systems*, *29*.

Aronoff, M., & Rees-Miller, J. (2020). *The handbook of linguistics.* John Wiley & Sons.

Bengio, Y., Ducharme, R., & Vincent, P. (2000). A neural probabilistic language model. *Advances in neural information processing systems*, *13*.

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, *5*(2), 157–166.

Chaudhary, L., Girdhar, N., Sharma, D., Andreu-Perez, J., Doucet, A., & Renz, M. (2023). A review of deep learning models for twitter sentiment analysis: Challenges and opportunities. *IEEE Transactions on Computational Social Systems*.

DiMaggio, P., Evans, J., & Bryson, B. (1996). Have american's social attitudes become more polarized? *American journal of Sociology*, *102*(3), 690–755.

Eyvazi-Abdoljabbar, S., Kim, S., Feizi-Derakhshi, M.-R., Farhadi, Z., & Abdulameer Mohammed, D. (2024). An Ensemble-Based Model for Sentiment Analysis of Persian Comments on Instagram Using Deep Learning Algorithms. *IEEE ACCESS*, *12*, 151223–151235.

Farabi, S., Ranasinghe, T., Kanojia, D., Kong, Y., & Zampieri, M. (2024, August). A survey of multimodal sarcasm detection. In *Proceedings of the thirty-thirdinternational joint conference on artificial intelligence* (p. 8020–8028). International Joint Conferences on Artificial Intelligence Organization.

Frenda, S., et al. (2018). The role of sarcasm in hate speech. a multilingual perspective. In *Proceedings of the doctoral symposium of the xxxivinternational conference of the spanish society for natural language processing (sepln 2018)* (pp. 13–17).

Halawani, H., Mashraqi, A., Badr, S., & Alkhalaf, S. (2023). Automated sentiment analysis in social media using Harris Hawks optimisation and deep learning techniques. *Alexandria Engineering Journal*, *80*, 433–443. (Publisher: Elsevier B.V.)

Hindocha, E., Yazhiny, V., Arunkumar, A., & Boobalan, P. (2019). Short-text semantic similarity using glove word embedding. *International Research Journal of Engineering and Technology*, *6*(4).

Hochreiter, S. (1997). Long short-term memory. *Neural Computation MIT-Press*.

Islam, M., Chen, G., & Jin, S. (2019). An overview of neural network. *American Journal of Neural Networks and Applications*, *5*(1), 7–11.

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, *31*(3), 685–695.

Kastrati, M., Kastrati, Z., Shariq Imran, A., & Biba, M. (2024). Leveraging distant supervision and deep learning for twitter sentiment and emotion classification. *Journal of Intelligent Information Systems*, *62*(4), 1045–1070. (Publisher: Springer)

Kong, D., Chen, A., Zhang, J., Xiang, X., Lou, W. V., Kwok, T., & Wu, B. (2022). Public discourse and sentiment toward dementia on chinese social media: Machine learning analysis of weibo posts. *Journal of medical Internet research*, *24*(9), e39805.

Lakretz, Y., Kruszewski, G., Desbordes, T., Hupkes, D., Dehaene, S., & Baroni, M. (2019). The emergence of number and syntax units in lstm language models. *arXiv preprint arXiv:1903.07435*.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

Li, C., & Li, F. (2023). Emotion recognition of social media users based on deep learning. *PeerJ Computer Science*, *9*, e1414.

Liddy, E. D. (1998). Enhanced text retrieval using natural language processing. *Bulletin of the American Society for Information Science and Technology*, *24*(4), 14–16.

Lin, W., Zhang, Q., Wu, Y., & Chen, T.-C. (2023). Running a Sustainable Social Media Business: The Use of Deep Learning Methods in Online-Comment Short Texts. *Sustainability (Switzerland)*, *15*(11). (Publisher: MDPI)

Lubis, A. R., Nasution, M. K., Sitompul, O. S., & Zamzami, E. M. (2021). The effect of the tf-idf algorithm in times series in forecasting word on social media. *Indones. J. Electr. Eng. Comput. Sci*, *22*(2), 976.

Ma, L., & Zhang, Y. (2015). Using word2vec to process big text data. In *2015 ieee international conference on big data (big data)* (pp. 2895–2897).

Mahadevaswamy, U., & Swathi, P. (2023). Sentiment analysis using bidirectional lstm network. *Procedia Computer Science*, *218*, 45–56.

Mateus, B. C., Mendes, M., Farinha, J. T., Assis, R., & Cardoso, A. M. (2021). Comparing lstm and gru models to predict the condition of a pulp paper press. *Energies*, *14*(21), 6958.

Maynard, D. G., & Greenwood, M. A. (2014). Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Lrec 2014 proceedings*.

Mikolov, T. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, *3781*.

Mu, G., Li, J., Li, X., Chen, C., Ju, X., & Dai, J. (2024). An Enhanced IDBO-CNN-BiLSTM Model for Sentiment Analysis of Natural Disaster Tweets. *Biomimetics*, *9*(9). (Publisher: Multidisciplinary Digital Publishing Institute (MDPI))

Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, *18*(5), 544–551.

Németh, R. (2023). A scoping review on the use of natural language processing in research on political polarization: trends and research prospects. *Journal of computational social science*, *6*(1), 289–313.

Nguyen, H.-H. (2024). Enhancing Sentiment Analysis on Social Media Data with Advanced Deep Learning Techniques. *International Journal of Advanced Computer Science and Applications*, *15*(5), 970–980. (Publisher: Science and Information Organization)

Noh, S.-H. (2021). Analysis of gradient vanishing of rnns and performance comparison. *Information*, *12*(11), 442.

Orellana, S., & Bisgin, H. (2023). Using natural language processing to analyze political party manifestos from new zealand. *Information*, *14*(3), 152.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. *arXiv preprint arXiv:1211.5063*.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)* (pp. 1532–1543).

Rodríguez-Ibánez, M., Casánez-Ventura, A., Castejón-Mateos, F., & Cuenca-Jiménez, P.-M. (2023). A review on sentiment analysis from social media platforms. *Expert Systems with Applications*, *223*, 119862.

Saberi, B., & Saad, S. (2017). Sentiment analysis or opinion mining: A review. *Int. J. Adv. Sci. Eng. Inf. Technol*, *7*(5), 1660–1666.

Said, H., Tawfik, B., & Makhlouf, M. (2023). A Deep Learning Approach for Sentiment Classification of COVID-19 Vaccination Tweets. *International Journal of Advanced Computer Science and Applications*, *14*(4), 530–538. (Publisher: Science and Information Organization)

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, *45*(11), 2673–2681.

Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval* (Vol. 39). Cambridge University Press Cambridge.

Shewalkar, A., Nyavanandi, D., & Ludwig, S. A. (2019). Performance evaluation of deep neural networks applied to speech recognition: Rnn, lstm and gru. *Journal of Artificial Intelligence and Soft Computing Research*, *9*(4), 235–245.

Singh, C., Imam, T., Wibowo, S., & Grandhi, S. (2022). A Deep Learning Approach for Sentiment Analysis of COVID-19 Reviews. *Applied Sciences (Switzerland)*, *12*(8). (Publisher: MDPI)

Stavrianou, A., Andritsos, P., & Nicoloyannis, N. (2007). Overview and semantic issues of text mining. *ACM Sigmod Record*, *36*(3), 23–34.

Tabinda Kokab, S., Asghar, S., & Naz, S. (2022). Transformer-based deep learning models for the sentiment analysis of social media data. *Array*, *14*. (Publisher: Elsevier B.V.)

Taboada, M. (2016). Sentiment analysis: An overview from linguistics. *Annual Review of Linguistics*, *2*(1), 325–347.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational linguistics*, *37*(2), 267–307.

Tarwani, K. M., & Edem, S. (2017). Survey on recurrent neural network in natural language processing. *Int. J. Eng. Trends Technol*, *48*(6), 301–304.

Vanam, H., & Raj, J. (2023). Novel Method for Sentiment Analysis in Social Media Data Using Hybrid Deep Learning Model. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, *32*(1), 272–289. (Publisher: Penerbit Akademia Baru)

Vernikou, S., Lyras, A., & Kanavos, A. (2022). Multiclass sentiment analysis on COVID-19-related tweets using deep learning models. *Neural Computing and Applications*, *34*(22), 19615–19627. (Publisher: Springer Science and Business Media Deutschland GmbH)

Wei, C., Wang, Y.-C., Wang, B., & Kuo, C.-C. J. (2023). An overview on language models: Recent developments and outlook. *arXiv preprint arXiv:2303.05759*.

Yin, Z., Shao, J., Hussain, M., Hao, Y., Chen, Y., Zhang, X., & Wang, L. (2023). DPG-LSTM: An Enhanced LSTM Framework for Sentiment Analysis in Social Media Text Based on Dependency Parsing and GCN. *Applied Sciences (Switzerland)*, *13*(1). (Publisher: MDPI)

Zakaria, M., Mabrouka, A., & Sarhan, S. (2014). Artificial neural network: a brief overview. *neural networks*, *1*, 2.

Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., & Saeed, J. (2020). A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *Journal of Applied Science and Technology Trends*, *1*(1), 56–70.