Sentiment Analysis

The sentiment analyzer can be built using different methods. Existing approaches to sentiment analyzer can be grouped into three main categories: knowledge-based techniques, statistical methods and deep learning approaches.

Knowledge-based approach

Knowledge-based techniques look for unambiguous affect words such as “good”, “bad”, etc. to determine the sentiment polarity of the text. This is probably the simplest way of implementing a sentiment analyzer but this is also the least accurate because reviews contains complex grammatical structures like negations, idioms, and sarcasms that cannot be all listed explicitly to develop a knowledge-based sentiment classifier. Therefore, knowledge-based approach only works for simple sentences.

Statistical Bag-of-words Methods

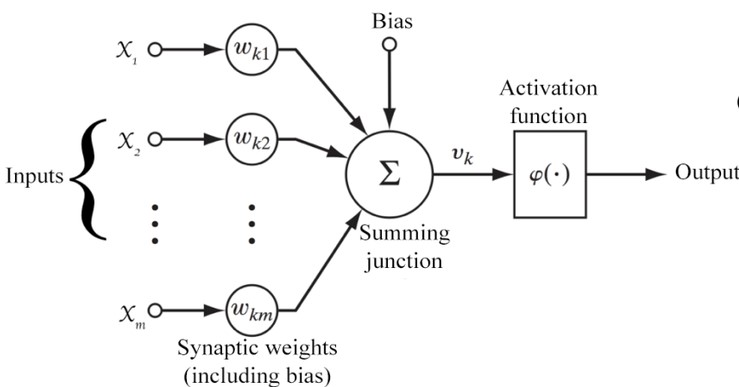
Statistical methods use elements of machine learning such as latent semantic analysis, Support Vector Machines (SVM), bag-of-words (BOW), etc. to calculate the probabilistic polarity of a text. This model is similar to the knowledge-based approach in a way that it also looks at words lexically in isolation, i.e. it analyzes the character composition of words and not the true meaning of the word. For example, lexically, the words “swam” is more similar to “swan” than “swimming”.

The only difference between this BOW method and knowledge-based approach is that in this method, the knowledge is learned from the label dataset whereas in the knowledge-based approach, the fact that word “good” carries positive sentiment and the word “bad” carries negative sentiment is explicitly fed to the algorithm. In the BOW method, the sentiment of the word “good” is learned from the statistics of its appearance in the labeled dataset. For example, if there are 100 positive examples and 100 negative examples in the dataset and 60 positive examples contain the word “good” whereas 20 negative examples contain the word “good”, then the model learns that the word “good” must be 75% positive because out of 80 training examples containing the word “good”, 60 are positive and 20 are negative, and therefore positive equals 0.75 (60/80).

If single word is taken into account at a time, it is called unigram BOW model. The order of the words in the sentence is ignored in such model and important information may be lost. If a pair of consecutive words is taken from the dataset and calculated jointly, then bigram BOW model is obtained. Similarly, if three consecutive words are taken, the trigram BOW model is obtained. By using these n-gram BOW models, the accuracy of the model can be boosted further, but it requires more computer memory and processing resources.

Deep learning approach

Sentiment analysis is a subclass of NLP. Specifically, it is a form of text classification problem where one classifies a piece of text into one of the sentiment classes. The modern approach for NLP is deep learning using Artificial Neural Networks (ANN). ANNs are machine learning models inspired from the structure of the biological brain. An ANN is based on collection of connected units called artificial neurons, analogous to axons in a biological brain. Each connection (synapse) between neurons can transmit a signal to another neuron. The receiving (postsynaptic) neuron can process the signal(s) and then signal downstream neurons connected to it. Neurons may have state, generally represented by real numbers, typically 0 and 1. Neurons and synapses may also have weight that varies as learning proceeds, which can increase or decrease the signal strength as it sends downstream. Further, they may have a threshold such that only if the aggregate signal is below (or above) that level is the downstream signal sent.



Let,

= number of inputs

= vector of input parameters

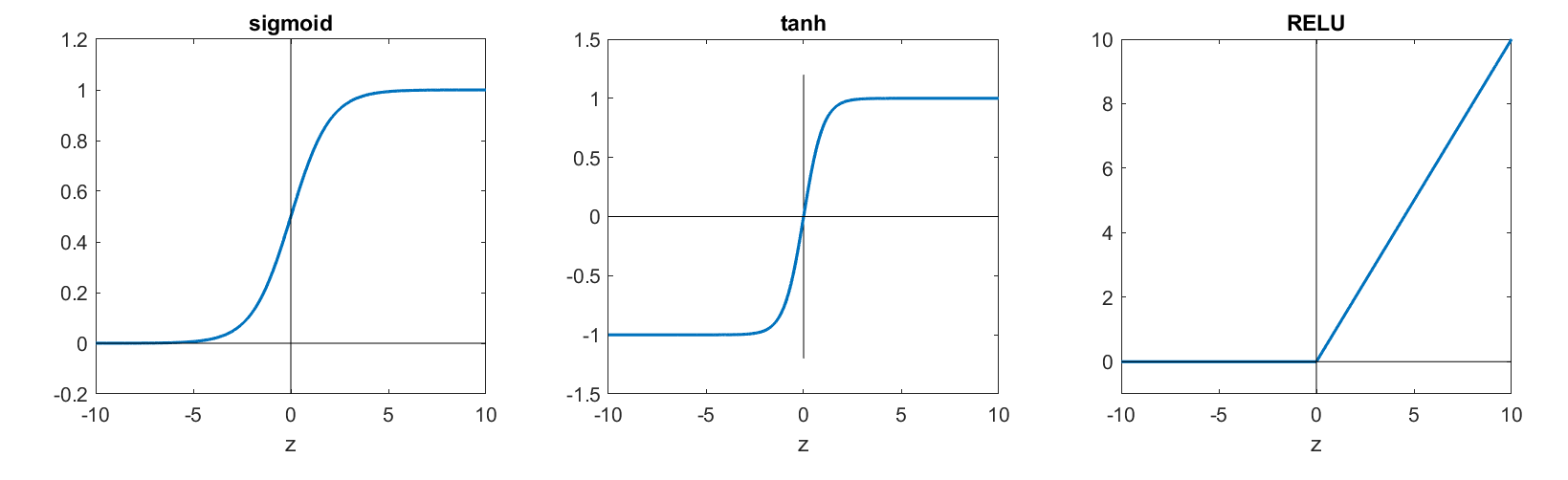
= vector of weights associated with the corresponding inputs

= bias term

Then, the output is,

It can be written in vector form as,

Where, is a monotonically increasing, continuous, differentiable and bounded non-linear function such as:



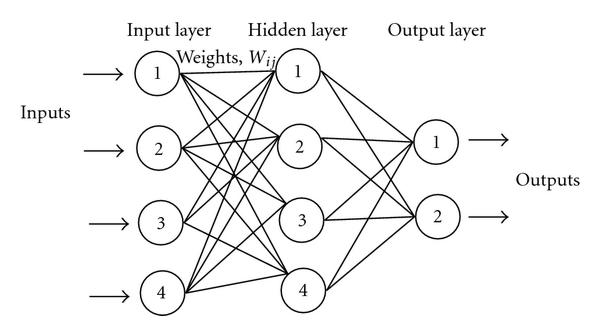


Figure 1.1 (**FIX THIS**) show a simple neural network. All the arrows shown in the figure have weights associated with them. Those weights are adjusted during training phase, which is done using an algorithm called backpropagation. Backpropagation is an algorithm used to calculate the gradient of the cost function with respect to the weights in an ANN. It is commonly used as a part of algorithms that optimize the performance of the network by adjusting the weights. It is also called backward propagation of errors. The cost function is a measure of the error of the neural network model. Most common cost functions are squared error, cross-entropy error, etc. The neural network weights are adjusted to minimize the cost function using any of the optimization algorithms such as gradient descent, BFGS, etc. For instance, in gradient descent, the weight updating step is:

for = 1, 2, 3, ...,

Where,

= number of parameters to learn

= parameter vector to learn

= cost function to minimize

= learning rate

The term “deep learning” basically refers to ANN models having many hidden layers. They could be simple ANNs with multiple hidden layers, or ANNs with special architectures such as Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and so on.

In deep learning models, there are lots of parameters to learn (adjust) and usually the training set is not sufficient to learn those parameters with great certainty. Therefore, deep learning models are prone to overfitting. Hence, regularizing the cost function is vital for good performance of the model.