**CHAPTER  
INTRODUCTION**

Depression as a common mental health disorder has long been defined as a single disease with a set of diagnostic criteria. It often co-occurs with anxiety or other psychological and physical disorders; and has an impact on feelings and behaviour of the affected individuals. According to the WHO study, there are 322 million people estimated to suffer from depression, equivalent to 4.4% of the global population. Nearly half of the in-risk individuals live in the South-East Asia (27%) and Western Pacific region (27%) including China and India. In many countries’ depression is still under-diagnosed and left without any adequate treatment which can lead into a serious self-perception and at its worst, to suicide

In addition, the social stigma surrounding depression prevents many affected individuals from seeking an appropriate professional assistance. As a result, they turn to less formal resources such as social media. With the development of Internet usage, people have started to share their experiences and challenges with mental health disorders through online forums, micro-blogs or tweets. Their online activities inspired many researchers to introduce new forms of potential health care solutions and methods for early depression detection systems. Using different Natural Language Processing (NLP) techniques and text classification approaches, they tried to succeed in a higher performance improvement. Some studies use single set features, such as bag of words (BOW), N-grams, LIWC or LDA, to identify depression in their posts. Some other papers compare the performance of individual features with various machine learning classifiers. Recent studies examine the power of single features and their combinations such as N-grams+LIWC or BOW+LDA and TF-IDF+LDA to improve the accuracy results.

They experiment with a smarter text pre-processing, and introduce different substitute words depending on the nature of the original string. In addition, he applied multiple feature combinations to increase the performance using Convolutional Neural Networks (CNN) which consist of neurons with learnable weights and differ in terms of their layers. CNNs are very similar to simple feed-forward neural networks and state of the art method in the text and sentence classification tasks.

Our study has four specific contributions: first, to examine the relationship between depression and user’s language usage; second, to design three LIWC features for our specific research problem; third, to evaluate the power of N-grams probabilities, LIWC and LDA as single features for performance accuracy; fourth, to show the predictive power of both single and combined features with proposed classification approaches to achieve a higher performance in depression identification tasks. The rest of the paper is organized as follows. In section I, we discuss related work in depression detection. In section II, we define the properties of the Twitter dataset. In section III, we introduce the methodology and conduct data pre-processing followed by feature extraction. In section IV, we compare and analyze the feature sets and examine the results as well as the most powerful machine learning technique for depression detection. We conclude our study and provide a direction for future work in section V.

* 1. **ARTIFICIAL INTELLINGENCE:**

Artificial intelligence (AI) is the ability of a computer program or a machine to think and learn. It is also a field of study which tries to make computers "smart". As machines become increasingly capable, mental facilities once thought to require intelligence are removed from the definition. AI is an area of computer sciences that emphasizes the creation of intelligent machines that work and reacts like humans. Some of the activities computers with artificial intelligence are designed for include: Face recognition, Learning, Planning, Decision making etc.,

Artificial intelligence is the use of computer science programming to imitate human thought and action by analysing data and surroundings, solving or anticipating problems and learning or self-teaching to adapt to a variety of tasks.

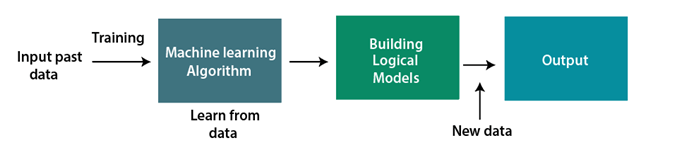
* 1. **MACHINE LEARNING**

Machine learning is a growing technology which enables computers to learn automatically from past data. Machine learning uses various algorithms for **building mathematical models and making predictions using historical data or information**. Currently, it is being used for various tasks such as **image recognition**, **speech recognition**, **email filtering**, **Facebook auto-tagging**, **recommender system**, and many more.

Machine Learning is said as a subset of **artificial intelligence** that is mainly concerned with the development of algorithms which allow a computer to learn from the data and past experiences on their own. The term machine learning was first introduced by **Arthur Samuel** in **1959**. We can define it in a summarized way as: “Machine learning enables a machine to automatically learn from data, improve performance from experiences, and predict things without being explicitly programmed”.

A Machine Learning system **learns from historical data, builds the prediction models, and whenever it receives new data, predicts the output for it**. The accuracy of predicted output depends upon the amount of data, as the huge amount of data helps to build a better model which predicts the output more accurately.

Suppose we have a complex problem, where we need to perform some predictions, so instead of writing a code for it, we just need to feed the data to generic algorithms, and with the help of these algorithms, machine builds the logic as per the data and predict the output. Machine learning has changed our way of thinking about the problem. The below block diagram explains the working of Machine Learning algorithm:



* + 1. **Features of Machine Learning:**
* Machine learning uses data to detect various patterns in a given dataset.
* It can learn from past data and improve automatically.
* It is a data-driven technology.
* Machine learning is much similar to data mining as it also deals with the huge amount of the data.
  + 1. **Classification of Machine Learning**

At a broad level, machine learning can be classified into three types:

1. Supervised learning
2. Unsupervised learning
3. Reinforcement learning

### 1) Supervised Learning

Supervised learning is a type of machine learning method in which we provide sample labeled data to the machine learning system in order to train it, and on that basis, it predicts the output.

The system creates a model using labeled data to understand the datasets and learn about each data, once the training and processing are done then we test the model by providing a sample data to check whether it is predicting the exact output or not.

The goal of supervised learning is to map input data with the output data. The supervised learning is based on supervision, and it is the same as when a student learns things in the supervision of the teacher. The example of supervised learning is **spam filtering**.

Supervised learning can be grouped further in two categories of algorithms:

* **Classification**
* **Regression**

### 2) Unsupervised Learning

Unsupervised learning is a learning method in which a machine learns without any supervision. The training is provided to the machine with the set of data that has not been labeled, classified, or categorized, and the algorithm needs to act on that data without any supervision. The goal of unsupervised learning is to restructure the input data into new features or a group of objects with similar patterns.

In unsupervised learning, we don't have a predetermined result. The machine tries to find useful insights from the huge amount of data.

It can be further classifieds into two categories of algorithms:

* **Clustering**
* **Association**
  1. **NATURAL LANGUAGE PROCESSING (NLP):**

Natural language processing (NLP) is a branch of [artificial intelligence](https://www.sas.com/en_in/insights/analytics/what-is-artificial-intelligence.html) that helps computers understand, interpret and manipulate human language. NLP draws from many disciplines, including computer science and computational linguistics, in its pursuit to fill the gap between human communication and computer understanding.

While natural language processing isn’t a new science, the technology is rapidly advancing thanks to an increased interest in human-to-machine communications, plus an availability of big data, powerful computing and enhanced algorithms.

Natural language processing includes many different techniques for interpreting human language, ranging from statistical and machine learning methods to rules-based and algorithmic approaches. We need a broad array of approaches because the text- and voice-based data varies widely, as do the practical applications. Basic NLP tasks include tokenization and parsing, lemmatization/stemming, part-of-speech tagging, language detection and identification of semantic relationships. If you ever diagrammed sentences in grade school, you’ve done these tasks manually before. In general terms, NLP tasks break down language into shorter, elemental pieces, try to understand relationships between the pieces and explore how the pieces work together to create meaning.

These underlying tasks are often used in higher-level NLP capabilities, such as:

* **Content categorization**. A linguistic-based document summary, including search and indexing, content alerts and duplication detection.
* **Topic discovery and modelling.** Accurately capture the meaning and themes in text collections, and apply [advanced analytics](https://www.sas.com/en_in/solutions/analytics.html) to text, like optimization and forecasting.
* **Contextual extraction.** Automatically pull structured information from text-based sources.
* **Sentiment analysis.** Identifying the mood or subjective opinions within large amounts of text, including average sentiment and opinion mining.
* **Speech-to-text and text-to-speech conversion.** Transforming voice commands into written text, and vice versa.
* **Document summarization.** Automatically generating synopses of large bodies of text.
* **Machine translation.**Automatic translation of text or speech from one language to another.

In all these cases, the overarching goal is to take raw language input and use linguistics and algorithms to transform or enrich the text in such a way that it delivers greater value.

* 1. **BAG OF WORDS**

The bag-of-words model is a way of representing text data when modeling text with machine learning algorithms.

The bag-of-words model is simple to understand and implement and has seen great success in problems such as language modeling and document classification.

In this tutorial, you will discover the bag-of-words model for feature extraction in [natural language processing](https://machinelearningmastery.com/natural-language-processing/).

After completing this tutorial, you will know:

* What the bag-of-words model is and why it is needed to represent text.
* How to develop a bag-of-words model for a collection of documents.
* How to use different techniques to prepare a vocabulary and score words.

The approach is very simple and flexible, and can be used in a myriad of ways for extracting features from documents.

A bag-of-words is a representation of text that describes the occurrence of words within a document. It involves two things:

1. A vocabulary of known words.
2. A measure of the presence of known words.

It is called a “*bag*” of words, because any information about the order or structure of words in the document is discarded. The model is only concerned with whether known words occur in the document, not where in the document.

Let’s make the bag-of-words model concrete with a worked example.

### Step 1: Collect Data

Below is a snippet of the first few lines of text from the book “[A Tale of Two Cities](https://www.gutenberg.org/ebooks/98)” by Charles Dickens, taken from Project Gutenberg.

It was the best of times,  
it was the worst of times,  
it was the age of wisdom,  
it was the age of foolishness,

For this small example, let’s treat each line as a separate “document” and the 4 lines as our entire corpus of documents.

### Step 2: Design the Vocabulary

Now we can make a list of all of the words in our model vocabulary. The unique words here (ignoring case and punctuation) are:

* “it”
* “was”
* “the”
* “best”
* “of”
* “times”
* “worst”
* “age”
* “wisdom”
* “foolishness”

That is a vocabulary of 10 words from a corpus containing 24 words.

### Step 3: Create Document Vectors

The next step is to score the words in each document. The objective is to turn each document of free text into a vector that we can use as input or output for a machine learning model. Because we know the vocabulary has 10 words, we can use a fixed-length document representation of 10, with one position in the vector to score each word. The simplest scoring method is to mark the presence of words as a Boolean value, 0 for absent, 1 for present.

Using the arbitrary ordering of words listed above in our vocabulary, we can step through the first document (“It was the best of times “) and convert it into a binary vector.

The scoring of the document would look as follows:

* “it” = 1
* “was” = 1
* “the” = 1
* “best” = 1
* “of” = 1
* “times” = 1
* “worst” = 0
* “age” = 0
* “wisdom” = 0
* “foolishness” = 0

As a binary vector, this would look as follows:



The other three documents would look as follows:



All ordering of the words is nominally discarded and we have a consistent way of extracting features from any document in our corpus, ready for use in modeling.

New documents that overlap with the vocabulary of known words, but may contain words outside of the vocabulary, can still be encoded, where only the occurrence of known words is scored and unknown words are ignored.

You can see how this might naturally scale to large vocabularies and larger documents.

* + 1. **Managing Vocabulary**

As the vocabulary size increases, so does the vector representation of documents. In the previous example, the length of the document vector is equal to the number of known words. You can imagine that for a very large corpus, such as thousands of books, that the length of the vector might be thousands or millions of positions. Further, each document may contain very few of the known words in the vocabulary.

This results in a vector with lots of zero scores, called a sparse vector or sparse representation.

Sparse vectors require more memory and computational resources when modeling and the vast number of positions or dimensions can make the modeling process very challenging for traditional algorithms.

As such, there is pressure to decrease the size of the vocabulary when using a bag-of-words model.

There are simple text cleaning techniques that can be used as a first step, such as:

* Ignoring case
* Ignoring punctuation
* Ignoring frequent words that don’t contain much information, called stop words, like “a,” “of,” etc.
* Fixing misspelled words.
* Reducing words to their stem (e.g., “play” from “playing”) using stemming algorithms.

A more sophisticated approach is to create a vocabulary of grouped words. This both changes the scope of the vocabulary and allows the bag-of-words to capture a little bit more meaning from the document.

In this approach, each word or token is called a “gram”. Creating a vocabulary of two-word pairs is, in turn, called a bigram model. Again, only the bigrams that appear in the corpus are modeled, not all possible bigrams.

For example, the bigrams in the first line of text in the previous section: “It was the best of times” are as follows:

* “it was”
* “was the”
* “the best”
* “best of”
* “of times”

A vocabulary then tracks triplets of words is called a trigram model and the general approach is called the n-gram model, where n refers to the number of grouped words. Often a simple bigram approach is better than a 1-gram bag-of-words model for tasks like documentation classification

## **Scoring Words**

Once a vocabulary has been chosen, the occurrence of words in example documents needs to be scored.

In the worked example, we have already seen one very simple approach to scoring: a binary scoring of the presence or absence of words.

Some additional simple scoring methods include:

* **Counts**. Count the number of times each word appears in a document.
* **Frequencies**. Calculate the frequency that each word appears in a document out of all the words in the document.

### Word Hashing

You may remember from computer science that a [hash function](https://en.wikipedia.org/wiki/Hash_function) is a bit of math that maps data to a fixed size set of numbers. For example, we use them in hash tables when programming where perhaps names are converted to numbers for fast lookup.

We can use a hash representation of known words in our vocabulary. This addresses the problem of having a very large vocabulary for a large text corpus because we can choose the size of the hash space, which is in turn the size of the vector representation of the document.

Words are hashed deterministically to the same integer index in the target hash space. A binary score or count can then be used to score the word. This is called the “hash trick” or “feature hashing “.

The challenge is to choose a hash space to accommodate the chosen vocabulary size to minimize the probability of collisions and trade-off sparsity.

### TF-IDF

A problem with scoring word frequency is that highly frequent words start to dominate in the document (e.g., larger score), but may not contain as much “informational content” to the model as rarer but perhaps domain specific words.

One approach is to rescale the frequency of words by how often they appear in all documents, so that the scores for frequent words like “the” that are also frequent across all documents are penalized.

This approach to scoring is called Term Frequency – Inverse Document Frequency, or TF-IDF for short, where:

* **Term Frequency**: is a scoring of the frequency of the word in the current document.
* **Inverse Document Frequency**: is a scoring of how rare the word is across documents.

The scores are a weighting where not all words are equally as important or interesting.

The scores have the effect of highlighting words that are distinct (contain useful information) in a given document.

* 1. **LITERATURE SURVEY**

**[1] Title:** Depression detection from social network data using machine learning techniques

**Authors:** [Md. Rafiqul Islam](javascript:;), [Muhammad Ashad Kabir](javascript:;), [Ashir Ahmed](javascript:;) - 2018

**Description:**

Social networks have been developed as a great point for its users to communicate with their interested friends and share their opinions, photos, and videos reflecting their moods, feelings and sentiments. This creates an opportunity to analyze social network data for user’s feelings and sentiments to investigate their moods and attitudes when they are communicating via these online tools. In this study, we aim to perform depression analysis on Facebook data collected from an online public source. To investigate the effect of depression detection, we propose machine learning technique as an efficient and scalable method.

# **[2] Title:** Depression detection using emotion artificial intelligence

# **Authors:** Mandar Deshpande, Vignesh Rao - 2017

**Description:**

Depression is a leading cause of mental ill health, which has been found to increase risk of early death. Moreover, it is a major cause of suicidal ideation and leads to significant impairment in daily life. Emotion artificial intelligence is a field of ongoing research in emotion detection, specifically in the field of text mining. The advent of internet-based media sources has resulted in significant user data being available for sentiment analysis of text and images. This paper aims to apply natural language processing on Twitter feeds for conducting emotion analysis focusing on depression. Individual tweets are classified as neutral or negative, based on a curated word-list to detect depression tendencies. In the process of class prediction, support vector machine and Naive-Bayes classifier have been used. The results have been presented using the primary classification metrics including F1-score, accuracy and confusion matrix.

**[3] Title:** Deep Learning for Depression Detection of Twitter Users

**Authors:** Ahmed Husseini Orabi, Prasadith Buddhitha, Mahmoud Husseini Orabi, Diana Inkpen - 2018

**Description:**

Mental illness detection in social media can be considered a complex task, mainly due to the complicated nature of mental disorders. In recent years, this research area has started to evolve with the continuous increase in popularity of social media platforms that became an integral part of people’s life. This close relationship between social media platforms and their users has made these platforms to reflect the users’ personal life on many levels. In such an environment, researchers are presented with a wealth of information regarding one’s life. In addition to the level of complexity in identifying mental illnesses through social media platforms, adopting supervised machine learning approaches such as deep neural networks have not been widely accepted due to the difficulties in obtaining sufficient amounts of annotated training data. Due to these reasons, we try to identify the most effective deep neural network architecture among a few of selected architectures that were successfully used in natural language processing tasks. The chosen architectures are used to detect users with signs of mental illnesses (depression in our case) given limited unstructured text data extracted from the Twitter social media platform.

# **[4] Title:** A text classification framework for simple and effective early depression detection over social media streams

**Authors:**[Sergio G.Burdisso](https://www.sciencedirect.com/science/article/abs/pii/S0957417419303525#!), [MarceloErrecalde](https://www.sciencedirect.com/science/article/abs/pii/S0957417419303525" \l "!) - 2019

**Description:**

With the rise of the Internet, there is a growing need to build intelligent systems that are capable of efficiently dealing with early risk detection (ERD) problems on social media, such as early depression detection, early rumor detection or identification of sexual predators. These systems, nowadays mostly based on machine learning techniques, must be able to deal with data streams since users provide their data over time. In addition, these systems must be able to decide when the processed data is sufficient to actually classify users. Moreover, since ERD tasks involve risky decisions by which people’s lives could be affected, such systems must also be able to justify their decisions. However, most standard and state-of-the-art supervised machine learning models (such as SVM, MNB, Neural Networks, etc.) are not well suited to deal with this scenario. This is due to the fact that they either act as black boxes or do not support incremental classification/learning. In this paper we introduce SS3, a novel supervised learning model for text classification that naturally supports these aspects. SS3 was designed to be used as a general framework to deal with ERD problems. We evaluated our model on the CLEF’s eRisk2017 pilot task on early depression detection. Most of the 30 contributions submitted to this competition used state-of-the-art methods. Experimental results show that our classifier was able to outperform these models and standard classifiers, despite being less computationally expensive and having the ability to explain its rationale.

# **[5] Title:** Using machine learning-based analysis for behavioral differentiation between anxiety and depression

# **Authors:** [Thalia Richter](javascript:;), [Barak Fishbain](javascript:;), [Andrey Markus](javascript:;), [Gal Richter-Levin](javascript:;) & [Hadas Okon-Singer](javascript:;) - 2020

# **Description:**

Anxiety and depression are distinct—albeit overlapping—psychiatric diseases, currently diagnosed by self-reported-symptoms. This research presents a new diagnostic methodology, which tests rigorously for differences in cognitive biases among subclinical anxious and depressed individuals. 125 participants were divided into four groups based on the levels of their anxiety and depression symptoms. A comprehensive behavioral test battery detected and quantified various cognitive–emotional biases. Advanced machine-learning tools, developed for this study, analyzed these results. These tools detect unique patterns that characterize anxiety versus depression to predict group membership. The prediction model for differentiating between symptomatic participants (i.e., high symptoms of depression, anxiety, or both) compared to the non-symptomatic control group revealed a 71.44% prediction accuracy for the former (sensitivity) and 70.78% for the latter (specificity). 68.07% and 74.18% prediction accuracy were obtained for a two-group model with high depression/anxiety, respectively. The analysis also disclosed which specific behavioral measures contributed to the prediction, pointing to key cognitive mechanisms in anxiety versus depression. These results lay the ground for improved diagnostic instruments and more effective and focused individually-based treatment.