# What is TensorFlow?

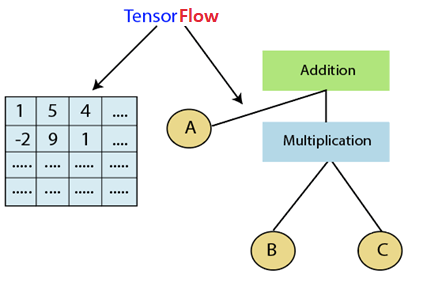
**TensorFlow** is a popular framework of **machine learning** and **deep learning**. It is **a free** and **open-source** library which is released on **9 November 2015** and developed by **Google Brain Team**. It is entirely based on Python programming language and use for numerical computation and data flow, which makes machine learning faster and easier.

TensorFlow can train and run the deep neural networks for image recognition, handwritten digit classification, recurrent neural network, **word embedding**, **natural language processing**, video detection, and many more. TensorFlow is run on multiple **CPU**s or **GPU**s and also mobile operating systems.

**The word TensorFlow is made by two words, i.e., Tensor and Flow**

1. **Tensor** is a multidimensional array
2. **Flow** is used to define the flow of data in operation.

TensorFlow is used to define the flow of data in operation on a multidimensional array or Tensor.



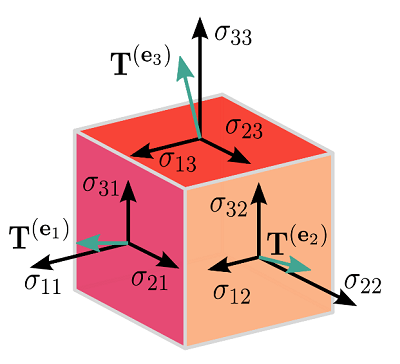
## Components of TensorFlow

### Tensor

The name TensorFlow is derived from its core framework, "**Tensor**." A tensor is a vector or a matrix of n-dimensional that represents all type of data. All values in a tensor hold similar data type with a known shape. The shape of the data is the dimension of the matrix or an array.

A tensor can be generated from the input data or the result of a computation. In

TensorFlow, all operations are conducted inside a graph. group is a set of calculation that takes place successively. Each transaction is called an op node are connected.



### Graphs

TensorFlow makes use of a graph framework. The chart gathers and describes all the computations done during the training.

### Advantages

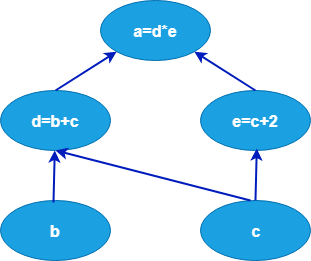
* It was fixed to run on multiple CPUs or GPUs and mobile operating systems.
* The portability of the graph allows to conserve the computations for current or later use. The graph can be saved because it can be executed in the future.
* All the computation in the graph is done by connecting tensors together.

Considernfollowing exp a= (b+c)\*(c+2).

We can break func into components given below:

d=b+c  
e=c+2  
a=d\*e

**Now, we can represent these operations graphically below:**

****

### Session

A session can execute the operation from the graph. To feed the graph with the value of a tensor, we need to open a session. Inside a session, we must run an operator to create an output.

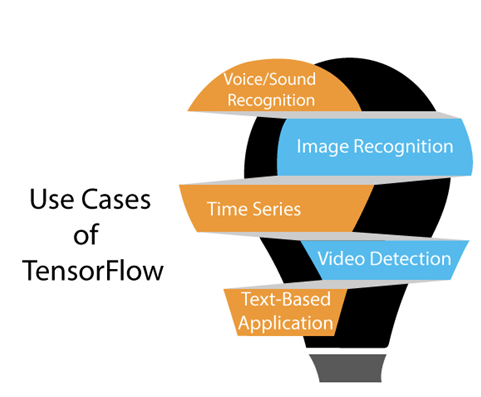
## Why is TensorFlow popular?

TensorFlow is the better library for all because it is accessible to everyone. TensorFlow library integrates different **API** to create a scale deep learning architecture like **CNN (Convolutional Neural Network)** or **RNN (Recurrent Neural Network)**.

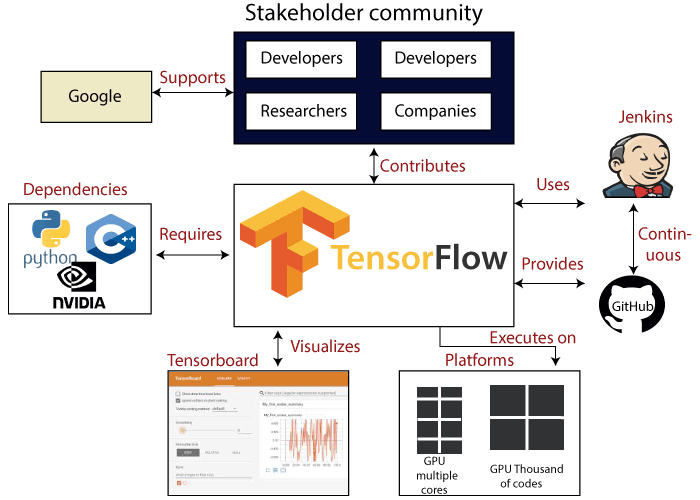
TensorFlow is based on graph computation; it can allow the developer to create the construction of the neural network with Tensorboard. This tool helps debug our program. It runs on CPU (Central Processing Unit) and GPU (Graphical Processing Unit).

***“TensorFlow*** *attracts the most considerable popularity on* ***GitHub*** *compare to the other deep learning framework.”*

## Use Cases/Applications of TensorFlow



TensorFlow provides amazing functionalities and services when compared to other popular deep learning frameworks. TensorFlow is used to create a large-scale **neural network** with many layers.



It is mainly used for deep learning or machine learning problems such as **Classification**, **Perception**, **Understanding**, **Discovering Prediction**, and **Creation**.

### 1. Voice/Sound Recognition

Voice and sound recognition applications are the most-known use cases of deep-learning. If the neural networks have proper input data feed, neural networks are capable of understanding audio signals.

**For example:**

**Voice recognition** is used in the Internet of Things, automotive, security, and UX/UI.AD

**Sentiment Analysis** is mostly used in customer relationship management **(CRM)**.

**Flaw Detection (engine noise)** is mostly used in automotive and Aviation.

**Voice search** is mostly used in customer relationship management (CRM)

### 2. Image Recognition

Image recognition is the first application that made deep learning and machine learning popular. Telecom, Social Media, and handset manufacturers mostly use image recognition. It is also used for face recognition, image search, motion detection, machine vision, and photo clustering.

**For example,** image recognition is used to recognize and identify people and objects in from of images. Image recognition is used to understand context & content of any img.

For object recognition, TensorFlow helps to classify and identify arbitrary objects within larger images.

This is also used in engineering application to identify shape for modeling purpose (**3d** reconstruction from **2d** image) and by Facebook for photo tagging.

**For example,** deep learning uses TensorFlow for analyzing thousands of photos of cats. So a deep learning algorithm can learn to identify a cat because this algorithm is used to find general features of objects, animals, or people.

### 3. Time Series

Deep learning is using Time Series algorithms for examining the time series data to extract meaningful statistics. For example, it has used the time series to predict the stock market.

A recommendation is the most common use case for Time Series. **Amazon**, **Google**, **Facebook**, and **Netflix** are using deep learning for the suggestion. So, the deep learning algorithm is used to analyze customer activity and compare it to millions of other users to determine what the customer may like to purchase or watch.

**For example,** it can be used to recommend us TV shows or movies that people like based on TV shows or movies we already watched.

### 4. Video Detection

The deep learning algorithm is used for video detection. It is used for motion detection, real-time threat detection in gaming, security, airports, and UI/UX field.

**For ex:-,** NASA is developing a deep learning network for object clustering of asteroids and orbit classification. So, it can classify and predict NEOs (**Near Earth Objects**).

### 5. Text-Based Applications

Text-based application is also a popular deep learning algorithm. Sentimental analysis, social media, threat detection, & fraud detection, are the ex. of Text-based applications.

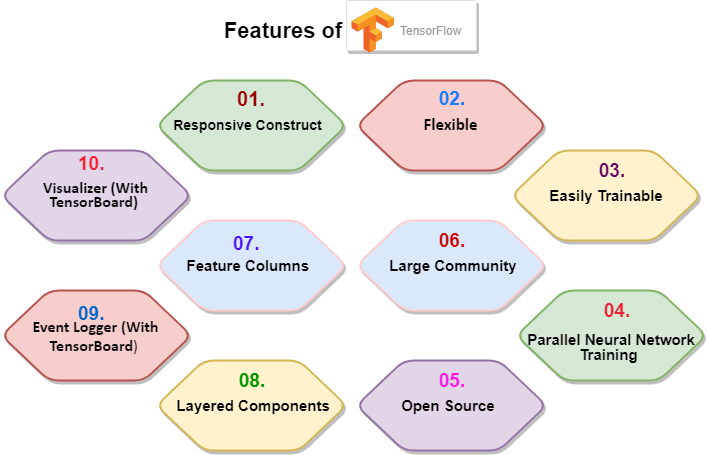
**For example,** Google Translate supports over 100 languages.

Some **companies** who are *currently using TensorFlow* are Google, AirBnb, eBay, Intel, DropBox, Deep Mind, Airbus, CEVA, Snapchat, SAP, Uber, Twitter, Coca-Cola, and IBM.

## Features of TensorFlow

TensorFlow has an interactive **multiplatform** programming interface which is scalable and reliable compared to other deep learning libraries which are available.

These features of TensorFlow will tell us about the popularity of TensorFlow.



### 1. Responsive Construct

We can visualize each part of the graph, which is not an option while using **Numpy** or **SciKit**. To develop a deep learning application, firstly, there are two or three components that are required to create a deep learning application & need a programming language.

### 2. Flexible

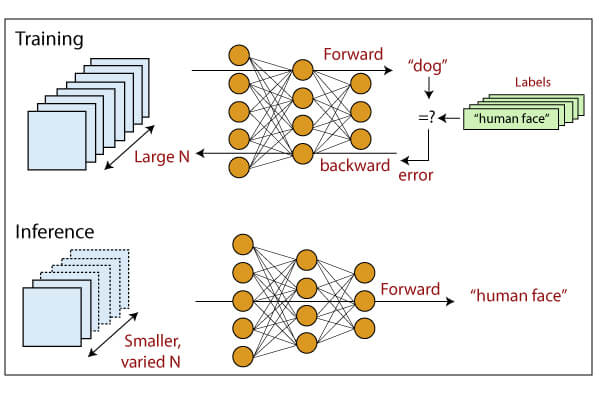
It is one of the essential TensorFlow Features according to its operability. It has modularity and parts of it which we want to make standalone.

### 3. Easily Trainable

It is easily trainable on CPU and for GPU in distributed computing.

### 4. Parallel Neural Network Training

TensorFlow offers to the pipeline in the sense that we can train multiple neural networks and various **GPUs**, which makes the models very efficient on large-scale systems.



### 5. Large Community

Google has developed it, and there already is a large team of software engineers who work on stability improvements continuously.

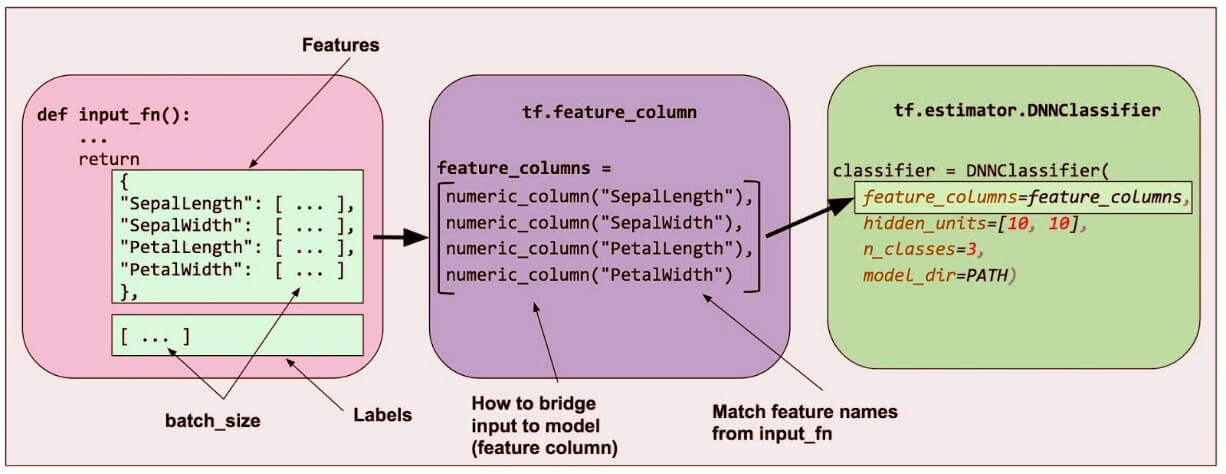
### 6. Open Source

The best thing about the machine learning library is that it is open source so anyone can use it as much as they have internet connectivity. So, people can manipulate the library and come up with a fantastic variety of useful products. And it has become another **DIY** community which has a massive forum for people getting started with it and those who find it hard to use it.

### 7. Feature Columns

TensorFlow has feature columns which could be thought of as intermediates between raw data and estimators; accordingly, **bridging** input data with our model.

The feature below describes how the feature column is implemented.



### 8. Availability of Statistical Distributions

This library provides distributions functions including Bernoulli, Beta, Chi2, Uniform, Gamma, which are essential, especially where considering probabilistic approaches such as Bayesian models.

### 9. Layered Components

TensorFlow produces layered operations of weight and biases from the function such as tf.contrib.layers and also provides batch normalization, convolution layer, and dropout layer. So **tf.contrib.layers.optimizers** have optimizers such as **Adagrad**, **SGD**, **Momentum** which are often used to solve optimization problems for numerical analysis.

### 10. Visualizer (With TensorBoard)

We can inspect a different representation of a model and make the changed necessary while debugging it with the help of TensorBoard.

### 11.Event Logger (With TensorBoard)

It is just like UNIX, where we use **tail - f** to monitor the output of tasks at the cmd. It checks, logging events and summaries from graph & production with the TensorBoard.

# Architecture of TensorFlow

The TensorFlow runtime is a cross-platform library. The system architecture which makes this combination of scale flexible. We have basic familiarity with TensorFlow programming concepts such as the computation graph, operations, and sessions.

Some terms need to be understood first to understand TensorFlow architecture. The terms are TensorFlow Servable, servable Streams, TensorFlow Models, Loaders, Sources, Manager, and Core. The term and their functionality in the architecture of TensorFlow are described below.

TensorFlow architecture is appropriate to read and modify the core TensorFlow code.

### 1. TensorFlow Servable

These are the central uncompleted units in TensorFlow serving. Servables are the objects that the clients use to perform the computation.

The size of a servable is flexible. A single servable may consist of anything from a lookup table to a unique model in a tuple of interface models. Servable should be of any type and interface, which enabling flexibility and future improvements such as:

* Streaming results
* Asynchronous modes of operation.
* Experimental APIs

### 2. Servable Versions

TensorFlow server can handle one or more versions of the servables, over the lifetime of any single server instance. It opens the door for new algorithm configurations, weights, and other data can be loaded over time. They also can enable more than one version of a servable to be charged at a time. They also allow more than one version of a servable to be loaded concurrently, supporting roll-out and experimentation gradually.

### 3. Servable Streams

A sequence of versions of any servable sorted by increasing version of numbers.

### 4. TensorFlow Models

A serving represents a model in one or more servables. A machine-learned model includes one or more algorithm and lookup the embedding tables. A servable can also serve like a fraction of a model; for example, an example, a large lookup table be served as many instances.

### 5. TensorFlow Loaders

Loaders manage a servable's life cycle. The loader API enables common infrastructure which is independent of the specific learning algorithm, data, or product use-cases involved.

### 6. Sources in TensorFlow Architecture

In simple terms, sources are modules that find and provide servable. Each reference provides zero or more servable streams at a time. For each servable stream, a source supplies only one loader instance for every servable.

Each source also provides zero or more servable streams. For each servable stream, a source supplies only one loader instance and makes available to be loaded.

### 7. TensorFlow Managers

TensorFlow managers handle the full lifecycle of a Servables, including:

* Loading Servables
* Serving Servables
* Unloading Servables

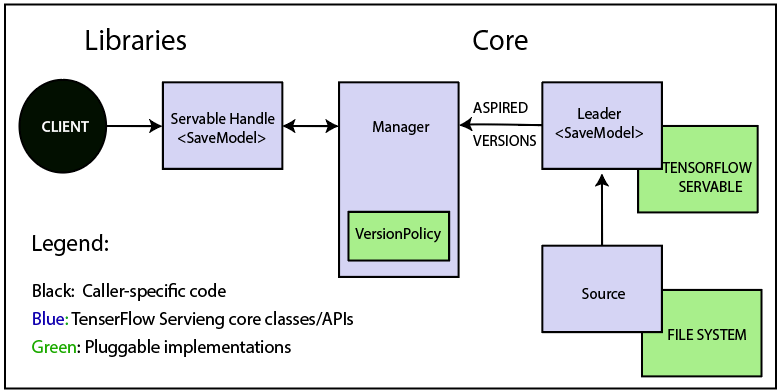
Manager observes to sources and tracks all versions. The Manager tries to fulfill causes, but it can refuse to load an Aspired version.

Managers may also postpone an "**unload**." For example, a manager can wait to unload as far as a newer version completes loading, based on a policy to assure that at least one version is loaded all the times.

**For example,** GetServableHandle (), for clients to access the loaded servable instances.

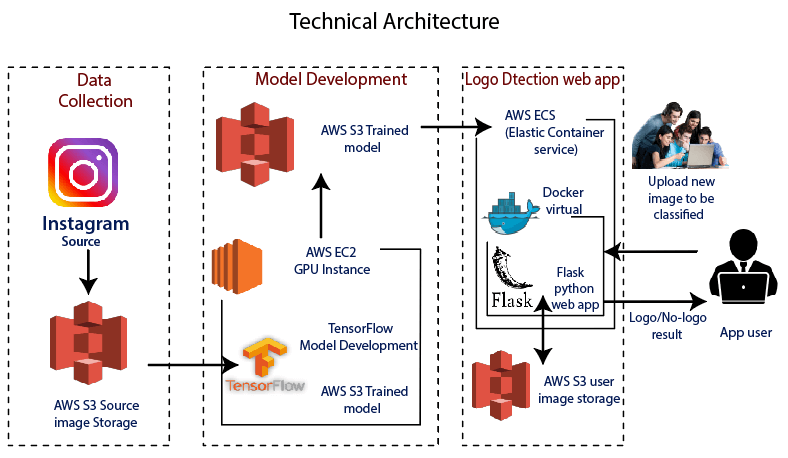
### 8. TensorFlow Core

This manages the below aspects of servables:



* Lifecycle
* Metrics
* TensorFlow serving core satisfaction servables and loaders as opaque objects.

### 9. Life of a Servable



**TensorFlow Technical Architecture:**

* Sources create loaders for Servable Versions, and then loaders are sent as Aspired versions to Manager, which will load and serve them to client requests.
* The Loader contains metadata, and it needs to load the servable.
* The source uses a callback to convey the Manager of Aspired version.
* The Manager applies effective version policy to determine the next action to take.
* If the Manager determines that it gives the Loader to load a new version, clients ask the Manager for the servable, and specifying a version explicitly or requesting the current version. The Manager returns a handle for servable. The dynamic Manager applies the version action and decides to load the newer version of it.
* The dynamic Manager commands the Loader that there is enough memory.
* A client requests a handle for the latest version of the model, and dynamic Manager returns a handle to the new version of servable.

### 10. TensorFlow Loaders

TensorFlow is one such algorithm backend. For example, we will implement a new loader to load, provide access, and unload an instance of a new type of servable of the machine learning model.

### 11. Batcher in TensorFlow Architecture

Batching of TensorFlow requests into a single application can significantly reduce the cost f performing inference, especially in the presence of hardware accelerators and GPUs. TensorFlow serving has a claim batching device that approves clients to batch their type-specific assumption beyond request into batch quickly. And request that algorithm systems can process more efficiently.

# Advantage and Disadvantage of TensorFlow

TensorFlow is an open-source machine learning concept which is designed and developed by Google. It offers a very high level and abstract approach to organizing low-level numerical programming. And supporting libraries that can allow our software to run without changes on regular CPU.

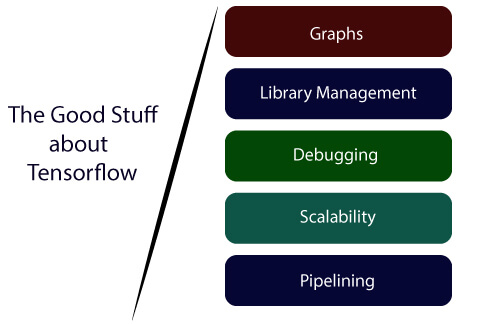
It supported platforms include **Linux**, **macOS**, **Windows**, and **Android**.

**TensorFlow** models can also be run without a traditional computer platform in the Google Cloud Machine Learning Engine.



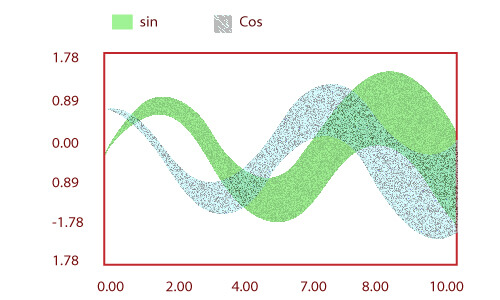
The more advanced technology, and the more useful it can be, but everything has its downside and also this machine learning library. When comparing TensorFlow with other libraries like **Torch**, **SciKit**, **Theano**, **Neon**, there are drawbacks in several features that the library lets us manipulate. This library is designed and updated by Google, so needless to say, and it has come a far way since its initial release.

## **Advantages of TensorFlow**

****

**1) Graphs:**

TensorFlow has better computational graph visualizations. Which are inherent when compared to other libraries like Torch and Theano.



**2) Library management:**

Google backs it. And has the advantages of seamless performance, quick updates, and frequent new releases with new features.

**3) Debugging:**

It helps us execute subpart of a graph which gives it an upper hand as we can introduce and retrieve discrete data

**4) Scalability:**

The libraries are deployed on a hardware machine, which is a cellular device to the computer with a complex setup.

**5) Pipelining:**

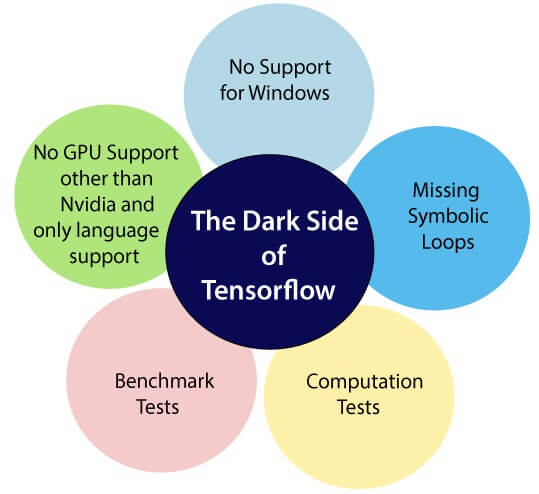
TensorFlow is designed to use various backend software (GPUs, ASIC), etc. and also highly parallel.

**6)** It has a unique approach that allows monitoring the training progress of our models and tracking several metrics.

**7)** TensorFlow has excellent community support.

**8)** Its performance is high and matching the best in the industry.

## Disadvantages of TensorFlow



**1) Missing Symbolic loops:**

When we say about variable-length sequence, feature is more required. Unfortunately, TensorFlow does not offer functionality, but finite folding is the right solution to it.

**2) No supports for windows:**

There is a wide variety of users who are comfortable in a window environment rather than Linux, and TensorFlow doesn't satisfy these users. But we need not worry about that if we are a window user we can also install it through conda or python package library (pip).

TensorFlow lacks in both speed and usage when it is compared to its competitors.

**4) No GPU support for Nvidia and only language support:**

Currently, the single supported GPUs are **NVIDIA** and the only full language support of Python, which makes it a drawback as there is a hike of other languages in deep learning as well as the **Lau**.

**5) Computation Speed:**

This is a field where TF is delaying behind, but we focus on the production environment ratherish than the performance, it is still the right choice.

**6)** No support for OpenCL.

**7)** It requires fundamental knowledge of advanced calculus and **linear algebra** along with a good understanding of **machine learning** also.

**8)** TensorFlow has a unique structure, so it's hard to find an error and difficult to debug.

**9)** There is no need for any super low-level matter.

**10)** It is a very low level with a steep learning curve.