ERICSOFT TECHNOLOGY

What is Artificial Intelligence?

AI IS "A CORE, TRANSFORMATIVE WAY BY WHICH WE'RE RETHINKING HOW WE'RE DOING EVERYTHING."

Google CEO Sundar Pichai

"The science and engineering of making intelligent machines, especially intelligent computer programs". **-John McCarthy**.

It is believes that for getting a sapling of a Bamboo tree, it takes at least 5 years. So is the story for building any AI model. For making smart automated solutions for any real-time problems which is linked with AI takes certain time. Let me give an example, *imagine a problem where missing children or people are to be found in **Kumbh Mela** using CCTV camera. The solution is simple use **face recognition.** Well, the data of faces can be taken from registered IDs. But the real challenge is to make the system understand that "Boss! This face is Ram, this face is Krishna……so on…" To create this experience to the system, it takes lot of time, but the authorities doesn't have that much time, right? They need immediate assistance so that the camera can find the missing people since no one knows where the particular person is.*

Before getting into the real picture, is Artificial Intelligence really necessary in India? Is AI cutting-edge technology for the government and industries in India?

AI Adoption in India

Artificial Intelligence is the present and future for most of the problems. Coming to India, we are a civilisation of the ages, but with significant socioeconomic and technological challenges. These challenges are taken by AI, which is poised to disrupt the present and future India, with enabling machines to cognitive thinking, decision-making, automating, coupled with data collection, data annotation, training and deploying, powered by AI, tends to compliment and supplement the human-intelligence and enrich the art of living and way of work. It is phenomenal that India's technological potential has a significant stake in AI provided with rapid growth of economy and second largest population, the technology is on its wheels to push to greater heights to meet the requirements. Recognising AI's potential in transforming and innovating economies, many technological companies are hence strategizing their approach to put a benchmark in this AI revolution.

How does AI solve the problems in real-time?

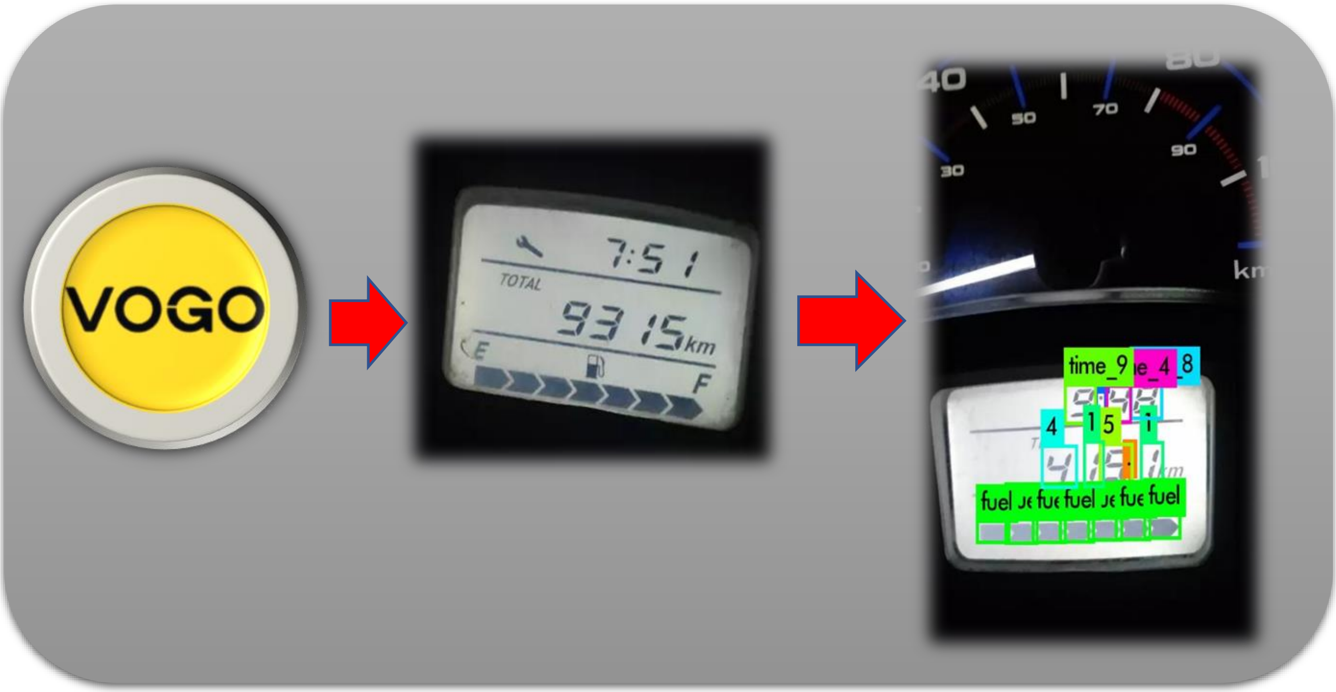
Let's see the ground situation on necessity of Artificial Intelligence in real-time scenarios and the pains faced by the Enterprises due to lack of AI adoption:

"Here is a case study of our own experience due to lack of Smart tools in AI and the story behind the creation of our product"

Couple of moons ago, we were given a project by **Vogo** which is Banglore based Company. The task was to detect the information present on the Odometer on the scooty. Initially, we were asked to detect only distance travelled by automatically detecting the distance in the odometer through their mobile app by clicking the picture of the odometer. Initially we were given **1 lac images** dataset and were asked to detect only "time" from the meter. So, we were using available existing solution for annotating the images under one class 'time' where the class names were numbers: "1-10" and "km".

We were 15 in team and everyone had to stop their ongoing work and had to take up the labelling job as we were given certain deadline.

Although we are a strong developer's team, we had to force stop our work and had to label all the 1 lac images. We divided the images among ourselves and were doing the mundane task of labelling day and night which prolonged till 10 days. Then we trained the model and finally deployed it in Android app with perfect output. This whole process took almost a month. The next day we were requested by the company that to perform the same outcome in addition with detection of time. Then, we again had to re-label the labelled one along with adding extra class of time and as expected it prolonged till 10 days and again a month time for deploying it in Android app. Then again, we were requested for detection of fuel. The same process was repeated for another month.



Then, the real challenge was installing dependencies of different versions that too on different platform. The annotated data is on one platform and the training and coding part is on another. The whole data must be transferred and combined since 15 of them did annotation on different systems. The training is done on another platform more likely GPU enabled. After training, edge deployment was key challenge for shifting the model file that too compatible version with the selected edge device. All necessary installations of dependencies were done on edge device and model file generated was shifted to edge device for testing.

This whole process of trail and error for testing for one trail took a whole month and imagine if you have to test it with different frameworks on different data with different classes on different edge devices.

That’s unimaginable hectic process for anyone. The excitement at starting of the process that the user would feel to look at the output, would fade away at the end of completion.

The excitement during Day-1 would incrementally fade away to the minimum due to lack of sufficient AI resources and mundane manual labelling

So, the lack of sufficient AI resources under a single platform and automated labelling tools, the excitement of the users is degrading while performing new research on any new AI models.

For Enterprises, there is a lack of consistent monitoring of the tasks of the employees under a single platform. The error operation is merely impossible to detect in the present ocean like data. The operational reports of the employees of a particular task is difficult to prolong due to lack of availability of smart monitoring tool, all on a single channel.

Developing AI models can be treated as *"playing a new game"*. The players get excited while playing a game. If he/she loses the game, he/she without any time delay, looks on for another approach and never give up his/her excitement until he/she wins. If that enthusiasm and excitement of gaming experience can be applied to AI model providing sufficient AI resources where the user has to just play in his/her own approach, possible exciting new outcomes can be seen in the AI field.

	List
Summary of Problems faced	Lack of AI resources under a single channel
	Lack of automated annotating tool and hence end up in manual labelling of all images
	Lack of changing classes on labelled data
	Lack of easy edge deployment
	Lack of proper monitoring on operational activities

What if a single tool provides a single pin-point solution for all these problems that too automatically?

What if a single tool eliminates the concern of data privacy?

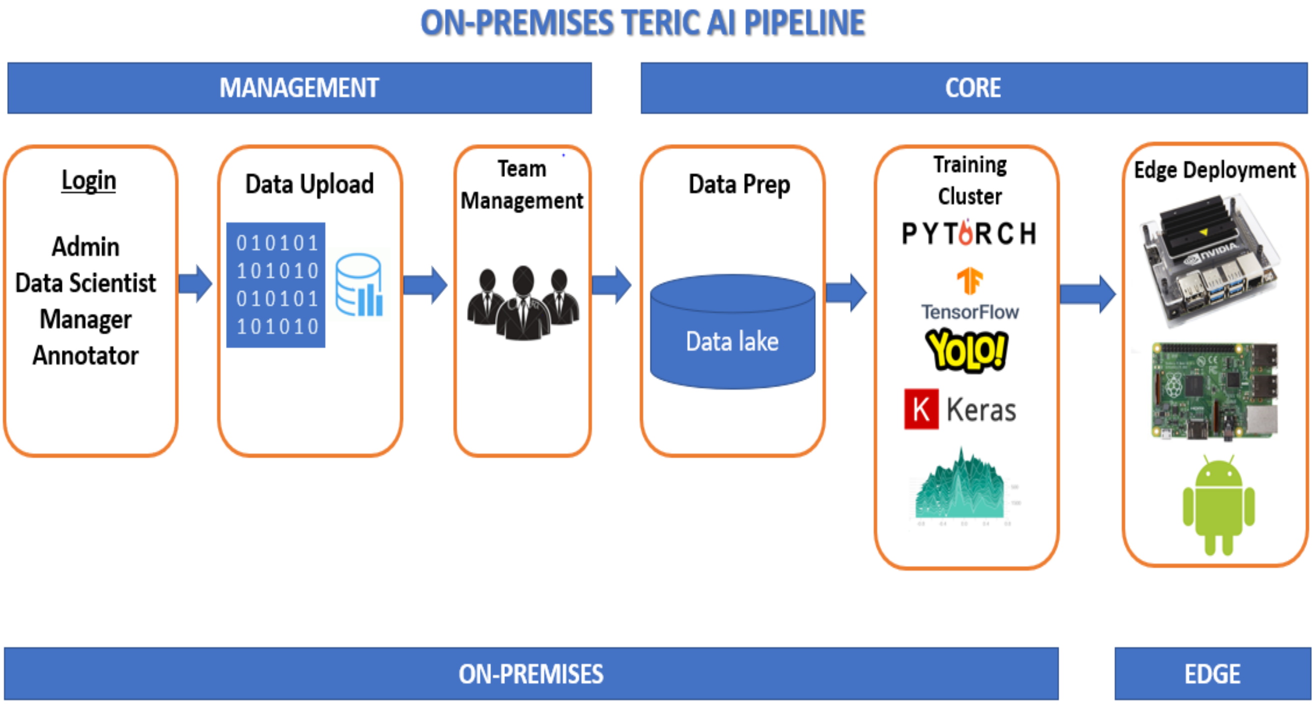
What if a single tool addresses the addition of classes at later stage without re-annotating the whole data?

What if a single tool provides the inter-connection between the tasks eliminating the transferring job of data?

What if a single tool provides the excitement of building an AI model as enthusiastic as a gaming operation does?

Well, the answer is a big **YES!!!**

teric.ai platform provides an end-to-end framework for building any AI model. teric.ai is easy to use, easy to deploy, easy to handle and easy to build any AI model.



The pipeline of teric.ai describes the workflow which specifies each task and its functionality. Let’s travel the journey of teric.ai pipeline and understand the game rules on how to play.

So, let’s start the journey with the first step:

1. Login Page

There are 4 separate accounts designed in teric.ai platform. These are created by the Teric team and credentials are given to the Enterprise.

- ***Admin***: He/she is the prime member in the Enterprise. He can frequently check the status and reports of the tasks and can set deadlines for each task.
- ***Data Scientist***: The data scientist creates an experiment where he/she can upload the data, define classes and type of operation as in: **(i) Classification (ii) Detection (iii) Segmentation**. Then, he can raise a request to the Manager to assign relevant annotators for labelling the data. He/she can then check the annotated data and apply ***Active Learning*** which will be discussed later. He/she has the authority to train the annotated data under any framework he/she wishes. The data scientist can test with different frameworks based on accuracy levels. He/she can then further add classes to the existing dataset and only raise a request for that particular class which isn't labelled. Once, the model file is generated, he/she can select the edge device on which the model is to be deployed and the machine automatically install all dependencies. Then, the data scientist can upload the model file onto edge device deployment section in teric.ai and can run for getting outcomes.
- ***Manager***: The manager allocates the annotation tasks or rather divide annotation task among available and relevant annotators. Since, the manager would know which annotator is the best and fast, he/she divides the annotation task among selective annotators and in different percentages based on their speed of completion.
- ***Annotator***: The annotator gets request from the manager regarding allotted annotation tasks. Based on the description provided by the Data Scientist about the theme of the experiment, the annotator annotates accordingly (classification, detection or segmentation). Then the annotator can annotate the images manually. However, due to active learning, he/she can just annotate minimal amount of data. Once the annotation task is completed, the page automatically redirects the annotator to experiment page displaying "task completed".

2. Image Library

As mentioned in the earlier section, once the images are uploaded, the user can re-use the images and their annotations for different experiment by simply adding extra classes to it.

So, how is this possible?

Well, teric.ai provides an option called **Image Library** where the user can upload all his dataset under different classes and names. The data will be stored in the tool. Once he is done with an experiment with certain classes, and gets another task of different dataset having images in which half of the images are similar to previous experiment and the rest half are completely different. Then, the user can call the previous experiment annotations and images and add to his current experiment by adding new classes. This ***saves*** a lot of ***time*** and ***efforts***.

Image Library Saves

TIME EFFORTS

New Experiments

Similar Data

3. Annotation Interface

Well, this is the backbone as well as most hectic task in any AI model.

But,

teric.ai platform has **automated smart annotation** tool and **Active Learning** (which will be discussed in the later section). This Annotation Interface is most likely to be used by the annotators. According to the allotment of the Data Scientist on whether the annotation type is of Classification or Detection or Segmentation, the annotator can use relevant smart tools for annotation.

- **Classification**

For classification, it is simply naming class of a single image. Multiple images can be selected under single image. Multiple classes on multiple images one-on-one map by just selecting images from the created experiment. The images are saved in the Image Library and their respective annotations along with their images are stored in the experiment.

- **Detection**

In detection, it is finding the location of the object in the image and drawing a boundary box around it. This is completely different from classification operation.

So, how to annotate for any detection-based operation?

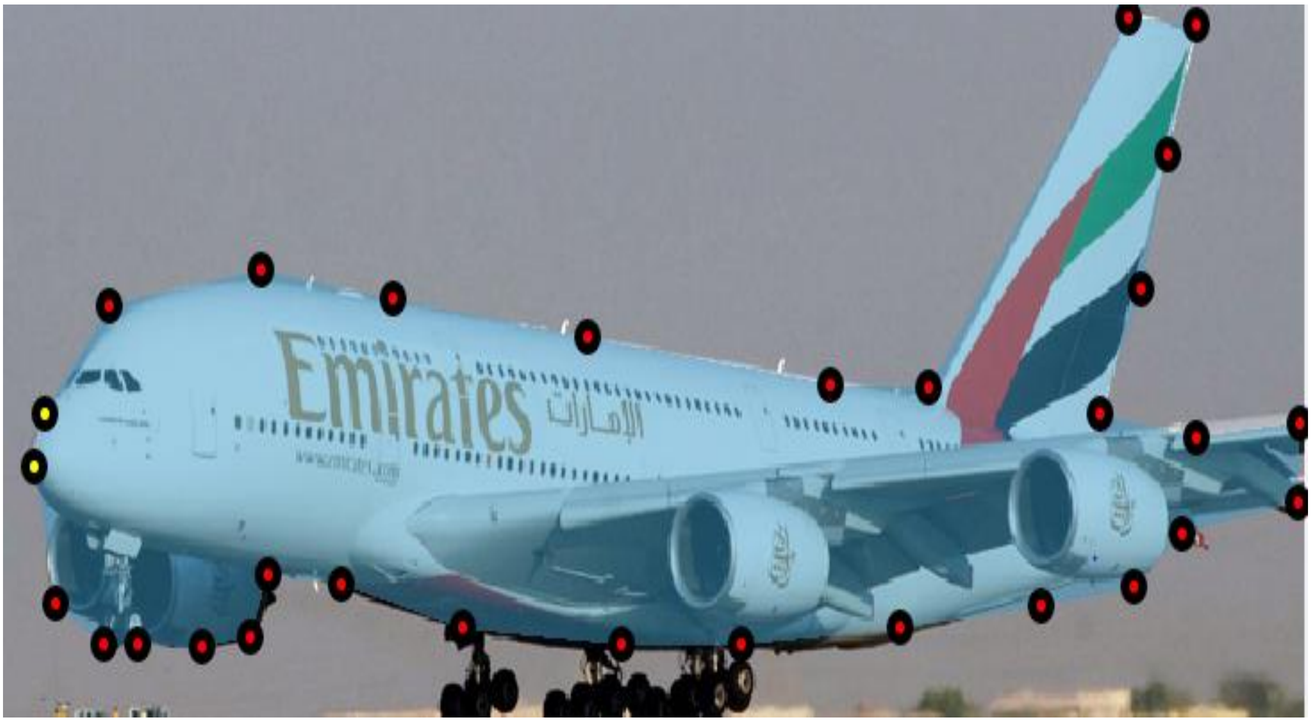
For detection-based operation the annotation is structured by drawing a rectangular box around the object/s. Each object is having a class. So, after drawing each box around the object, the user has to select the class for each box.

Well, that's very hectic task by selecting each class for each boundary box. So, is there any smart way which loosens the efforts and tightens the excitement?

In teric.ai, each class is assigned with a colour. Of course, the user can decide which colour for what class. The user can select the coloured checkbox on which class name appears and draw on objects. This is in fact saves half of the time than using the selection after drawing each box. This doesn't take much time and efforts and even the interface will give a smart experience feel for the user while using it.

- **Segmentation**

In segmentation, it is as simple as colouring the object. Technically, it is applying masks for objects in an image. This operation is the most tedious work among all annotation jobs and most importantly this annotation must be done more carefully. Below is a sample of annotation which is applied to segmentation:



Now, imagine if segmentation operation is to be applied to thousands of images which is typically common for Bio-Medical use cases and even other domains.

Well! Well! Well! teric.ai provides a smart tool called ***"Auto Polygon"*** which automatically applies masks to the object which works on pixel-to-pixel grading. This works on Pre-Trained COCO model. The user just needs to mention class name and the machine automatically applies masks on the objects based on class name.

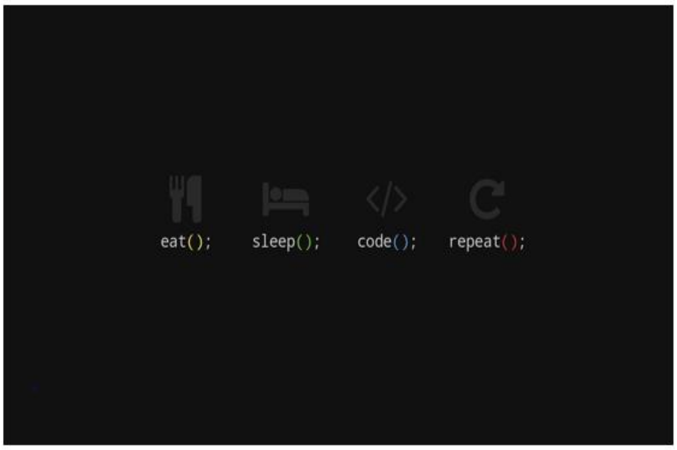
Another cool feature is ***"Smart Polygon"*** where the user has to just drag the mouse on the object, the machine computes pixel-to-pixel level grading and automatically masks the object. This can be done on any custom model.

4. Training Cluster

The Data Scientist looks after the training cluster wherein training of the models takes place. He/she is given wide variety of options to choose their framework on which the training can happen.

Here, comes the exciting part of the tool. The user can train the data with all possible frameworks and choose the best one or even can compare the accuracy graphs. It is as excited as playing a game with multiple themes in multiple ways.

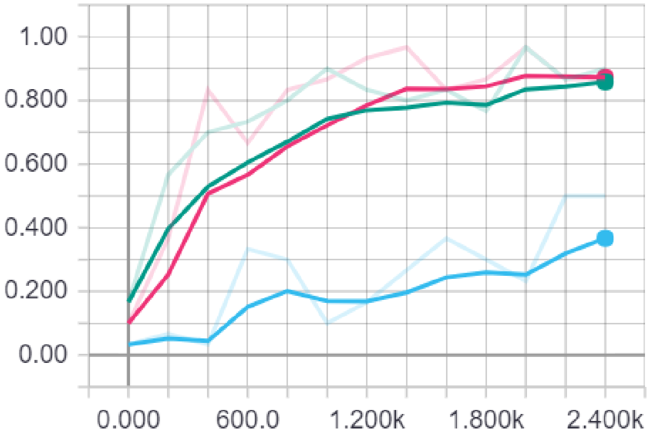
SOUNDS COOL ! RIGHT !!



CODE THE GAME !!

GAME THE CODE !!

Imagine teric.ai creating the actual gaming like experience for developers who can enjoy the excitement of winning the model whose outcome would be highest accuracy than others?



The data scientist can observe the accuracy loss of each framework and choose the best player in the game. Teric.ai platform creates that excitement similar to gaming, in fact more than gaming excitement where the outcome will be thrilling. This creates an excitement while training to see the best player in the game. Well, as training takes hours of time, these all operations queued before training can be completed within couple of minutes and data is all set for training. Hence, the data scientist excitement wouldn't fade away as teric.ai is providing end-to-end solution on a single channel within few hours.

In other words, teric.ai training cluster creates an environment wherein the user is watching a suspense and thrilling movie where he/she will be eagerly thinking

"what will be next??"

"it's really thrilling…right?"

Well, hold on your breath!!!! The next phase is more thrilling and exciting. It is the heart and brain of teric.ai.

Let's experience the thrilling excitement in a couple of seconds……….

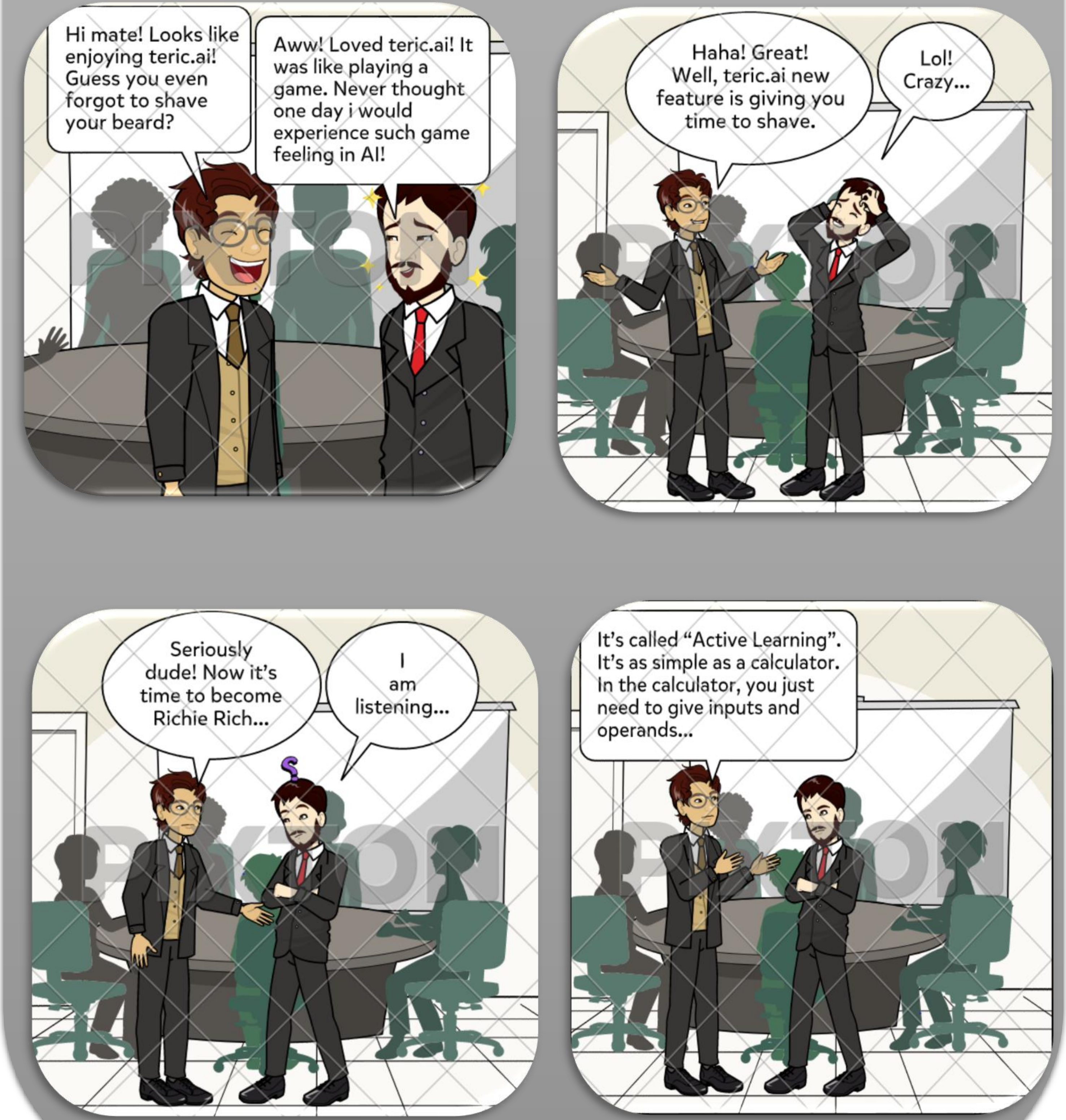
5. Active Learning

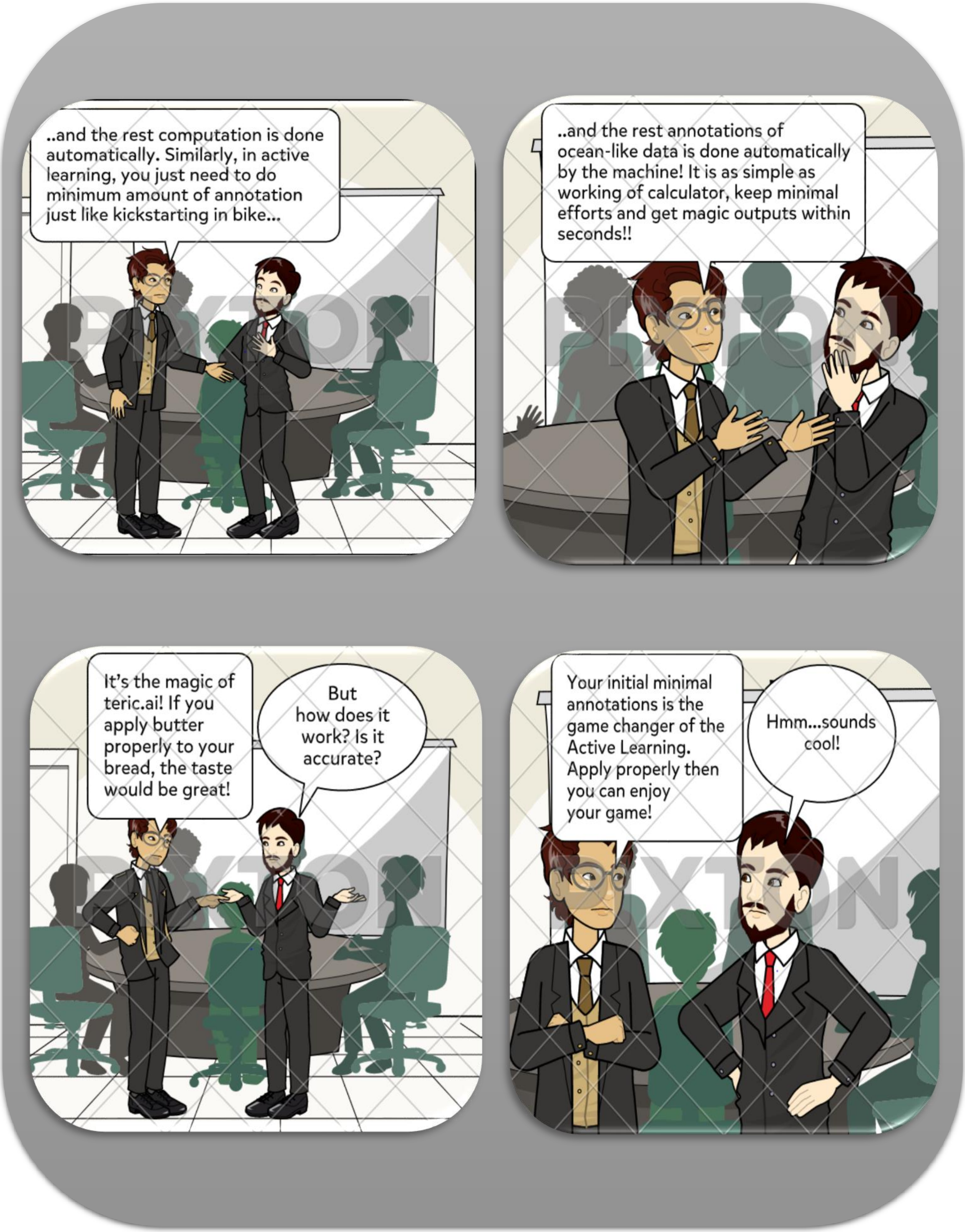
Here comes the most exciting and awaited phase, **Active Learning.**

Well, let's not take much time and unveil the curtain raiser of Active Learning.

We all love Comics right? So, let's see one comic:

Two Data Scientists are having a small conversation at their workplace


Hi mate! Looks like enjoying teric.ai! Guess you even forgot to shave your beard?
Aww! Loved teric.ai! It was like playing a game. Never thought one day i would experience such game feeling in AI!
Haha! Great! Well, teric.ai new feature is giving you time to shave.
Lol! Crazy...
Seriously dude! Now it's time to become Richie Rich...
I am listening...
It's called "Active Learning". It's as simple as a calculator. In the calculator, you just need to give inputs and operands...


..and the rest computation is done automatically. Similarly, in active learning, you just need to do minimum amount of annotation just like kickstarting in bike...
..and the rest annotations of ocean-like data is done automatically by the machine! It is as simple as working of calculator, keep minimal efforts and get magic outputs within seconds!!
It's the magic of teric.ai! If you apply butter properly to your bread, the taste would be great!
But how does it work? Is it accurate?
Your initial minimal annotations is the game changer of the Active Learning. Apply properly then you can enjoy your game!
Hmm...sounds cool!

Yea, ofcourse it works for all the 3 Musketeers!! This saves your time, efforts and yea gives you magic outputs!!
Does it work for all 3 Musketeers?
Dude, teric.ai is turning my life into Richie Rich!!
Yea and even saves your time to shave...!!

So, enjoyed the comic? Active Learning is an innovative automation in building AI models. We may have heard about AI models for automation, but this is the first ever product in bringing automation in making AI models. Hence, teric.ai is poised to bring disruption in AI field especially in automation industry in different domains.

A brief summary on Active Learning is the user can annotate minimum amount of data and run the model, based on that, the machine automatically annotates the rest left out humongous amount of data. A mathematical example is:

*Suppose the user is given 1 lac images, then he/she can annotate a minimal amount of data such as **1000-2000** images and run the model, the rest left out **98,000-99,000** images will get automatically annotated by the machine.*

So, now the coolest and amazing part is completed, but hold on this is not the climax. The Climax is thrilling and amazing. Let's not wait much and move on to the thrilling climax…

6. Deployment

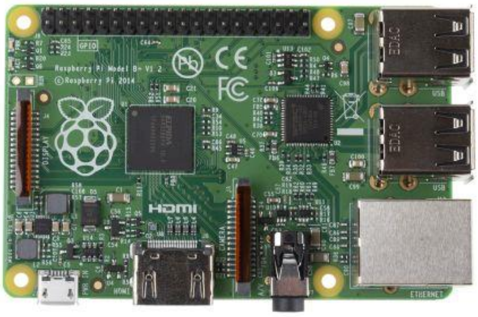
Well, most of the existing solutions lack this feature of easy deployment. We want our life to be easy right? Don't we? And AI is all about making our lives easy and better. So, how could we possibly make a tough climax of the brilliant journey. All brilliant journeys deserve a better climax. Teric.ai is providing a thrilling and amazing climax of **Easy Edge Deployment**. The AI product must be into production and everyone wants it. Most of the production operations are too complicated because the real-time production deals with deploying in real-world. So, imagine an easy edge deployment which can be directly put into real-time production system?

Well, let's be a bit technical in understanding why real-time production using edge devices are complex.

Edge devices are nothing but hardware kits where we deploy them in real-time to get real-time output. In simple words, CCTV camera footage obtained from CCTV, which is connected to an edge device for example, NVIDIA Jetson Nano board. We get CCTV footage which is real-time output from Edge Device which operates/controls it. Well, of course there is a program running to get that output.

Well, teric.ai supports 3 Edge Deployments:

Nvidia Jetson Nano, Raspberry Pi and **Android**

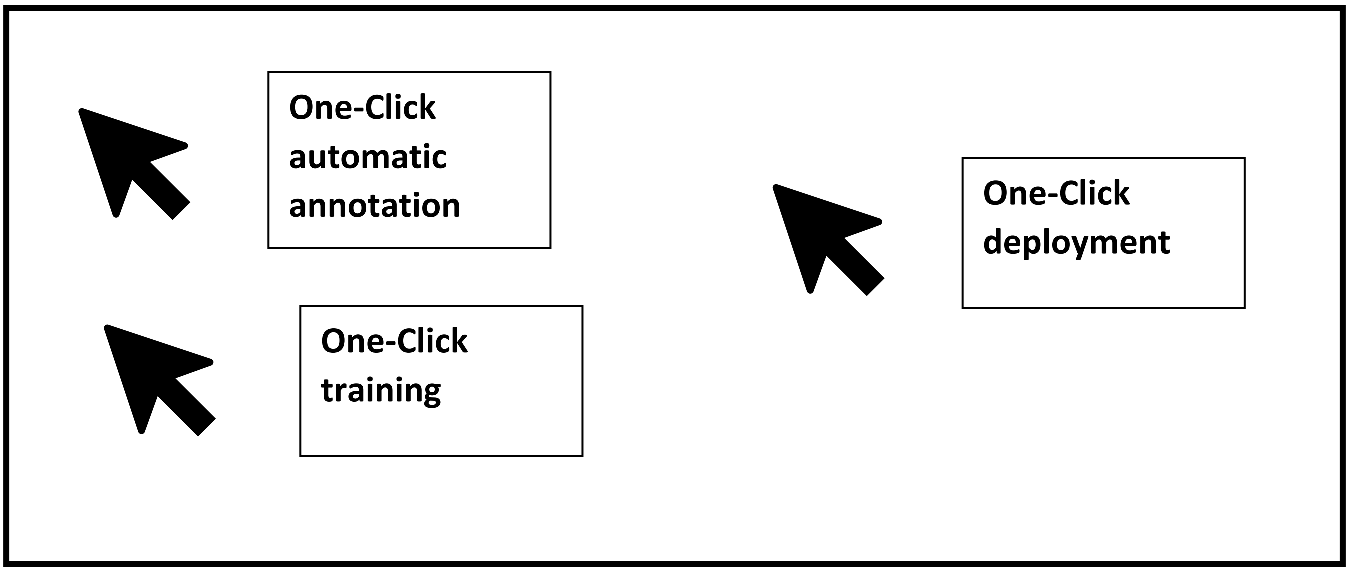


Basically, for deploying any AI model onto any Edge device, the user must install all necessary dependencies in the edge device and if the trained model has to run, then he/she have to create some files which are compatible or executable files which gives a supporting system to the generated trained model. This is a bit hectic work because much of files differ from edge device-to-edge device. This is the real-time production work where the user deploys in real-world. Of course, we know that real-time production is not too easy, it takes lot of time to do so.

What if I say teric.ai provides the user a **One-Click deployment?** Sounds great right?

Teric.ai provides a **One-Click** edge deployment, which makes real-time deployment easier. The user must just select the edge-device he/she wants to deploy the model. Teric.ai takes care of the rest, it will automatically generate all necessary stuff and gives the user **"All set Boss!! Good for deployment!!".** The user has to open teric.ai in his/her edge-device and run the model. The user gets the real-world output from a real-time deployment with *just one-click.*

Well, a brief summary is **"Experience a powerful 3-clicks!"**



One-Click automatic annotation

One-Click training

One-Click deployment

Teric.ai

UPLOAD | BUILD | TRAIN | DEPLOY

Team Tericsoft

info@tericsoft.com