



BUILD SOLUTIONS TO  
UNLOCK THE POTENTIAL OF  
HETEROGENEOUS COMPUTING

**LEAP** powered by Intel® oneAPI AI Analytics Toolkit

Problem Statement : **Open Innovation in Education**

Team Name : C5ailabs

Team Members : Rohit Sroch, Sujith R Kumar, Shubham Jain, Mohan K Rachumallu

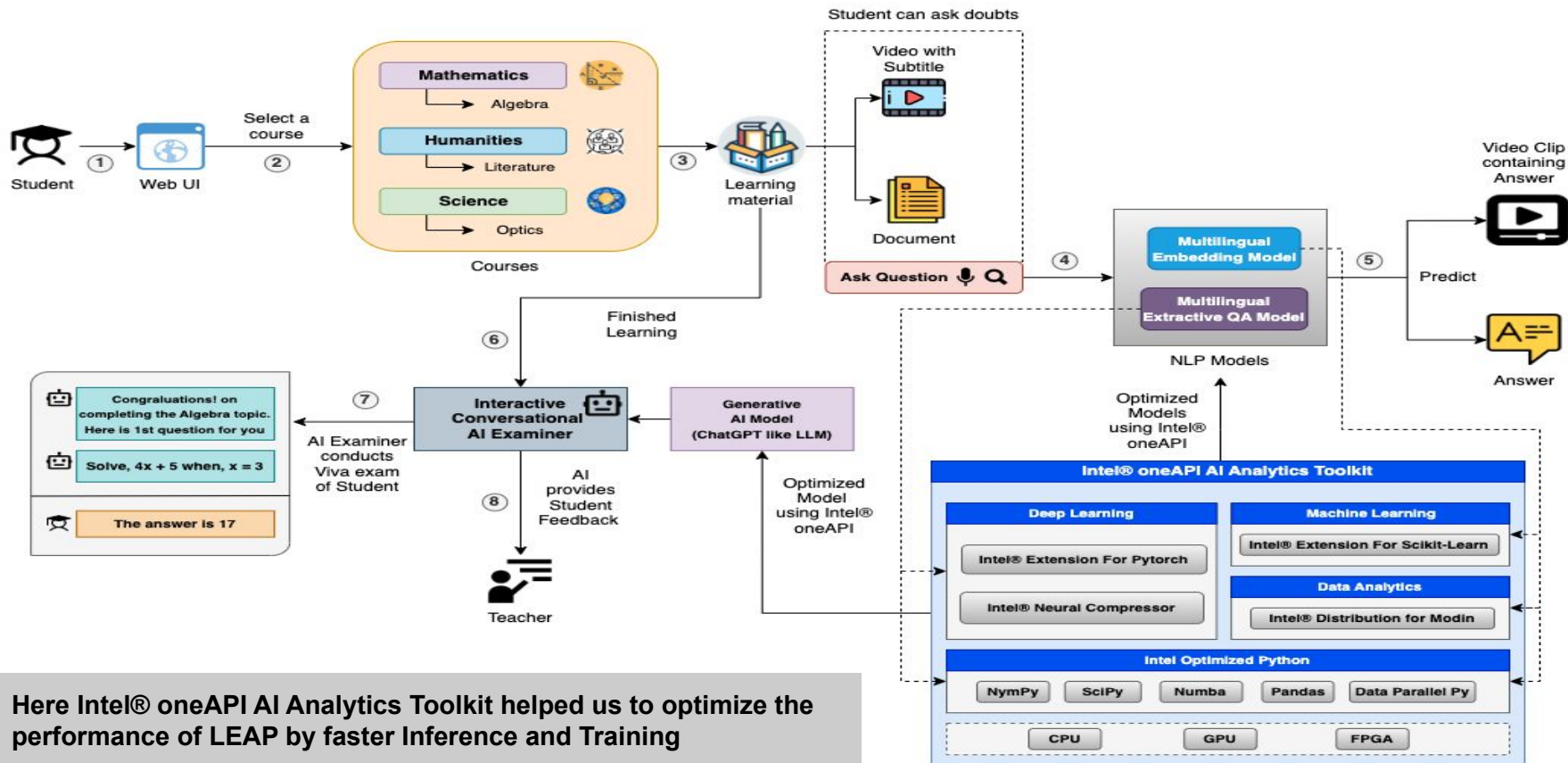
- MOOCs (Massive Open Online Courses) have surged in popularity in recent years, particularly during the COVID-19 pandemic. These online courses are typically free or low-cost, making education more accessible worldwide.
- Online learning is crucial for students even post-pandemic due to its flexibility, accessibility, and quality. But still, the learning experience for students is not optimal, as in case of doubts they need to repeatedly go through videos and documents or ask in the forum which may not be effective because of the following **challenges**:
  - Resolving doubts can be a time-consuming process.
  - It can be challenging to sift through pile of lengthy videos or documents to find relevant information.
  - Teachers or instructors may not be available around the clock to offer guidance
- To mitigate the above challenges, we propose **LEAP (Learning Enhancement and Assistance Platform)**, which is an AI-powered platform designed to enhance student learning outcomes and provide equitable access to quality education. The platform comprises two main features that aim to improve the overall learning experience of the student:
  - ❑ **Ask Question/Doubt:** This allows the students to ask real-time questions around provided reading material, which includes videos and documents, and get back answers along with the exact timestamp in the video clip containing the answer (so that students don't have to always scroll through). Also, It supports asking multilingual question, ensuring that language barriers do not hinder a student's learning process.
  - ❑ **Interactive Conversational AI Examiner:** This allows the students to evaluate their knowledge about the learned topic through an AI examiner conducting viva after each learning session. The AI examiner starts by asking question and always tries to motivate and provide hints to the student to arrive at correct answer, enhancing student engagement and motivation.

Our proposed **LEAP (Learning Enhancement and Assistance Platform)** is an AI-powered platform designed to enhance student learning outcomes and provide equitable access to quality education. The platform comprises two main features that aim to improve the overall learning experience of the student:

- ❑ **Ask Question/Doubt:** This allows the students to ask real-time questions around provided reading material, which includes videos and documents, and get back answers along with the exact timestamp in the video clip containing the answer (so that students don't have to always scroll through). Also, It supports asking multilingual question, ensuring that language barriers do not hinder a student's learning process.
- ❑ **Interactive Conversational AI Examiner:** This allows the students to evaluate their knowledge about the learned topic through an AI examiner conducting viva after each learning session. The AI examiner starts by asking question and always tries to motivate and provide hints to the student to arrive at correct answer, enhancing student engagement and motivation.

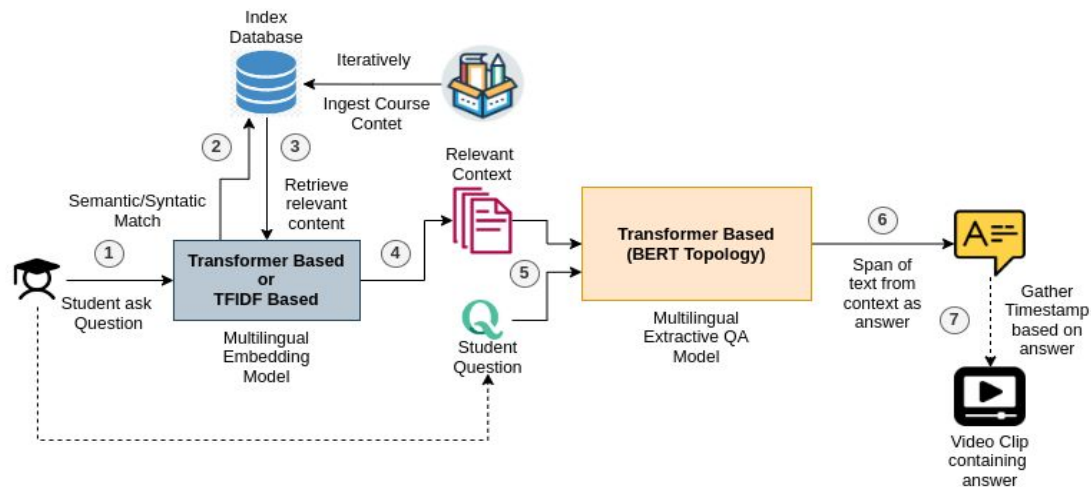
Apart from above two main feature, following are a few more features

- ❑ **Feedback from AI Examiner:** Based on the performance of the student in the viva exam, the AI examiner can provide feedback of the student to the teacher which helps the teacher to pay attention to the weak points of student and prioritize accordingly.
- ❑ **Highly Reliable:** LEAP provides its answer by extracting the span of text from context as answer and can be trained specifically on the edu course content, ensuring that its answers are accurate and reliable.

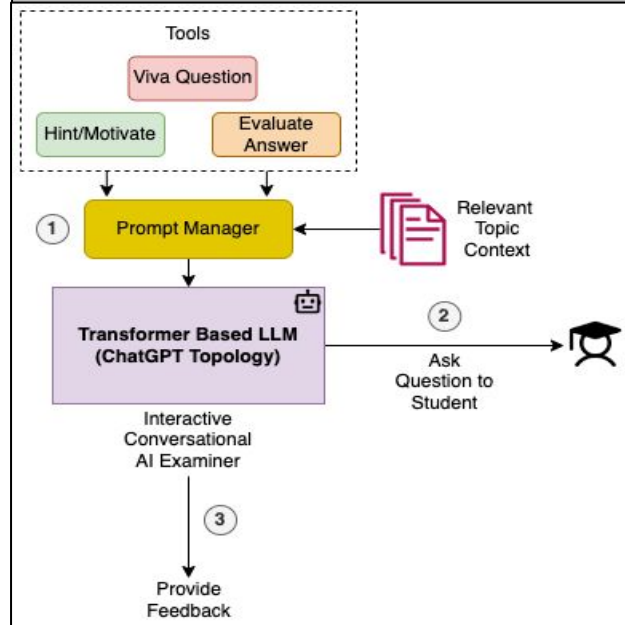


Here Intel® oneAPI AI Analytics Toolkit helped us to optimize the performance of LEAP by faster Inference and Training

## Ask Question/Doubt Model

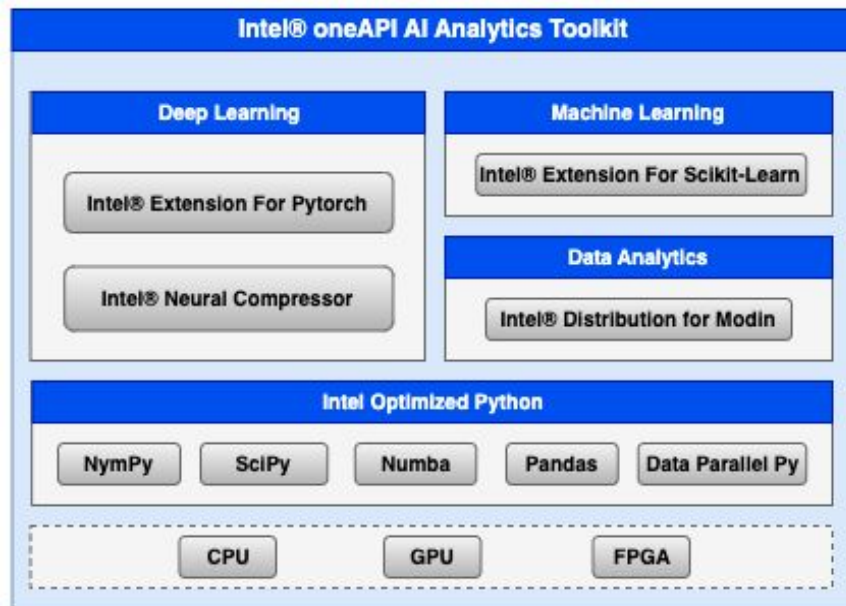


## Interactive Conversational AI Examiner Model



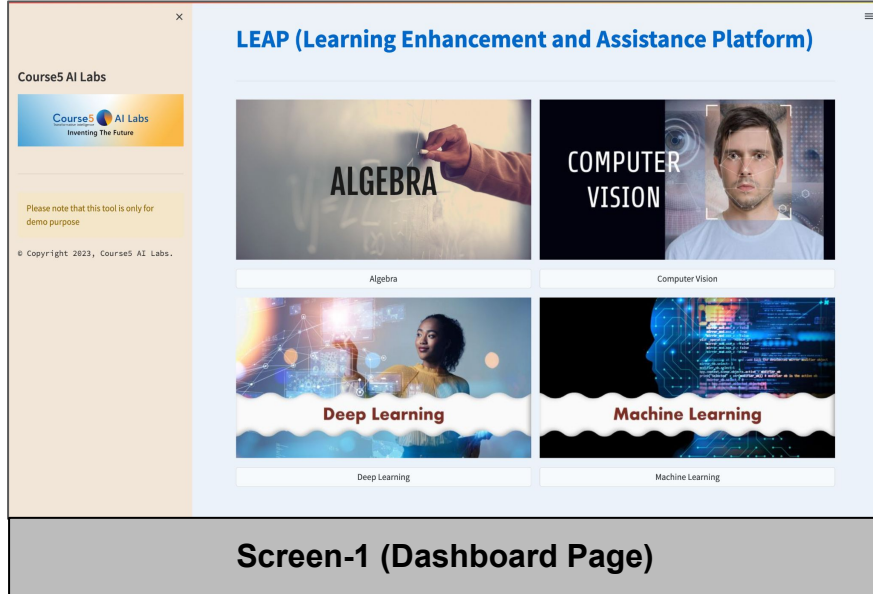
# Core components of oneAPI/SYCL used in the project

- Intel® Extension for Pytorch: Used for our Multilingual Extractive QA model Training/Inference optimization.
- Intel® Neural Compressor: Used for Multilingual Extractive QA model and Generative AI model Inference optimization.
- Intel® Extension for Scikit-Learn: Used for Multilingual Embedding model Training/Inference optimization.
- Intel® distribution for Modin: Used for basic initial data analysis.
- Intel® optimized Python: Used for data pre-processing, reading etc.




Mainly we have showed benchmark results with Intel® Extension for Pytorch, Intel® Neural Compressor and Intel® Extension for Scikit-Learn

Link: <https://www.youtube.com/watch?v=QoVWsOSlwvI>



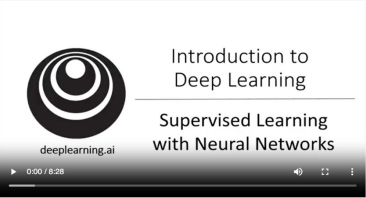
Course5 AI Labs

  
Inventing The Future

Please note that this tool is only for demo purpose

© Copyright 2023, Course5 AI Labs.


## LEAP (Learning Enhancement and Assistance Platform)



Video Transcript:


00:00:33.320 — 00:00:34.890  
There's been a lot of hype about neural networks. And perhaps some of that hype is justified, given how well they're working. But it turns out that so far, almost all the economic value created by neural networks has been through one type of machine learning, called supervised learning. Let's see what that means, and let's go over some examples. In supervised learning, you have some input  $x$ , and you want to learn a function mapping to some output  $y$ . So for example, just now we saw the housing price prediction application where

Ask Doubt:




Screen-3 (Ask Question/Doubt Page)

Course5 AI Labs

  
Inventing The Future

Please note that this tool is only for demo purpose

© Copyright 2023, Course5 AI Labs.



what is ReLU


ReLU function which stands for rectified linear units.

Get More Info

### Supervised Learning

Input(x)	Output (y)	Application
Home features	Price	Real Estate
Ad, user info	Click on ad? (0/1)	Online Advertising
Image	Object (1.....1000)	Photo tagging

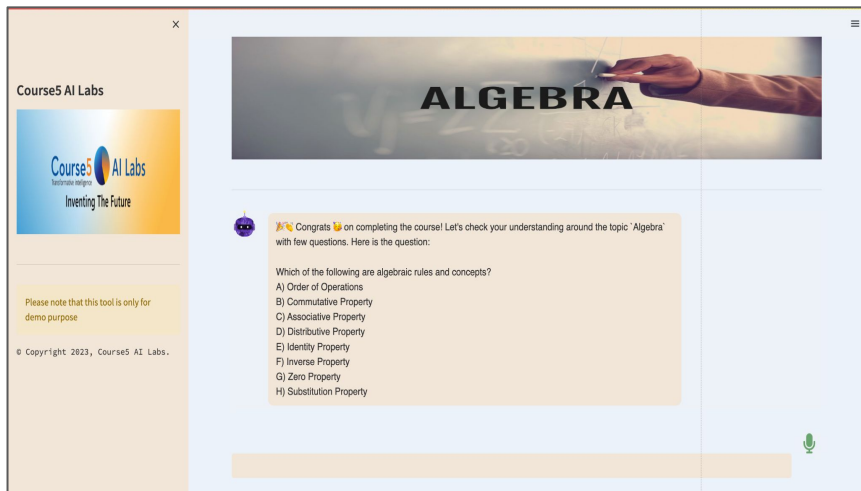
1:50 / 8:28



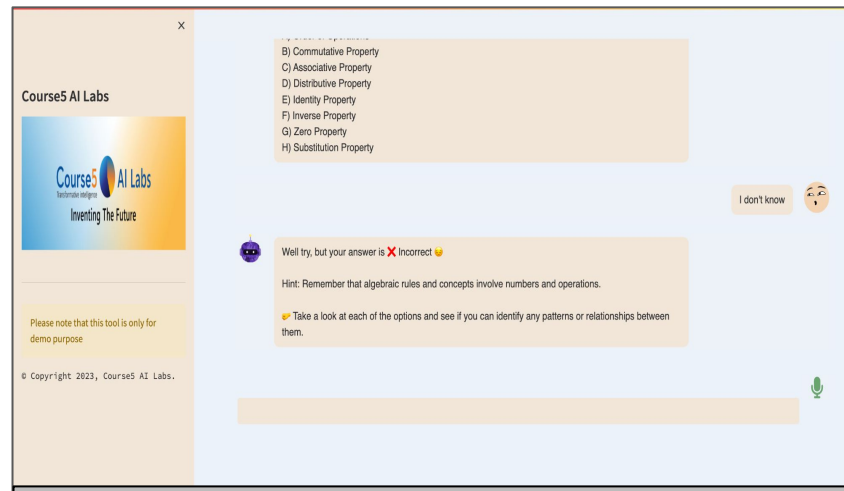
takes a max of zero, and then outputs the estimated price. And by the way in the neural network literature, you see this function a lot. This function which goes to zero sometimes and then it takes of as a straight line. This function is called a ReLU function which stands for rectified linear units. So ReLU. And rectify just means taking a max of 0 which is why you get a function shape like this. You don't need to worry about ReLU units for now but it's just something you see again later in this course.

Screen-4 (Ask Question/Doubt Page)





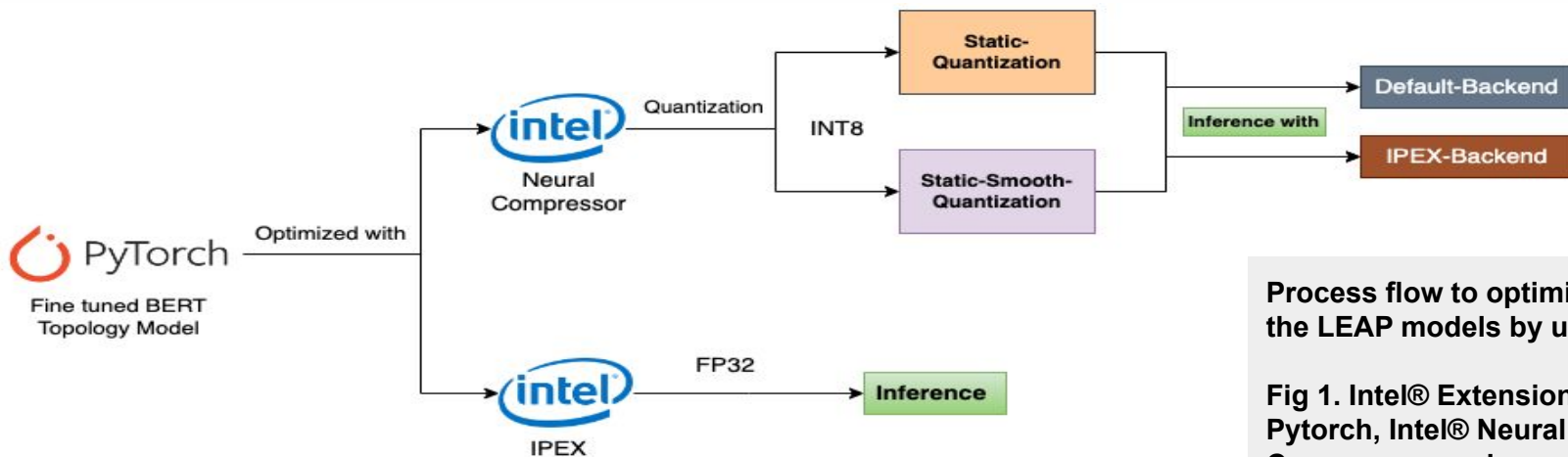
**Screen-5 (Interactive Conversational AI Examiner Asks Question to student)**



**Screen-6 (Interactive Conversational AI Examiner provides hints and motivates a student in case of a wrong answer)**

GitHub Link (Codes should be public and available after  
hackathon also)

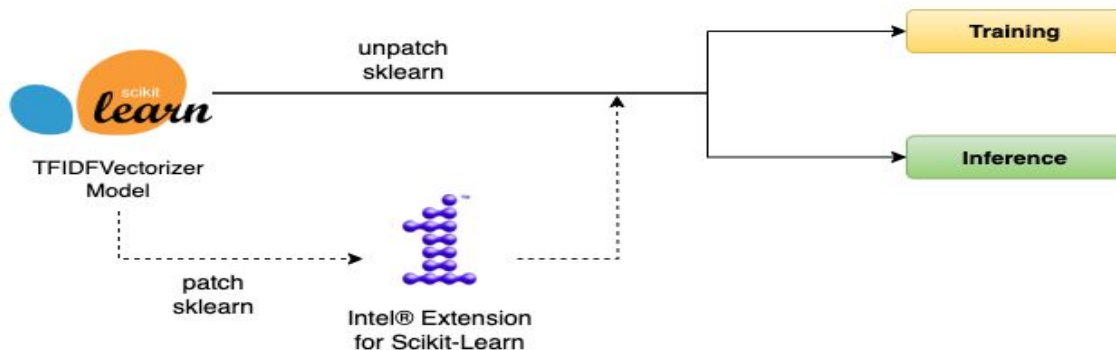
<https://github.com/rohitc5/intel-oneAPI>



Process flow to optimize  
the LEAP models by using

Fig 1. Intel® Extension for  
Pytorch, Intel® Neural  
Compressor and

Fig 2. Intel® Extension for  
Scikit-Learn



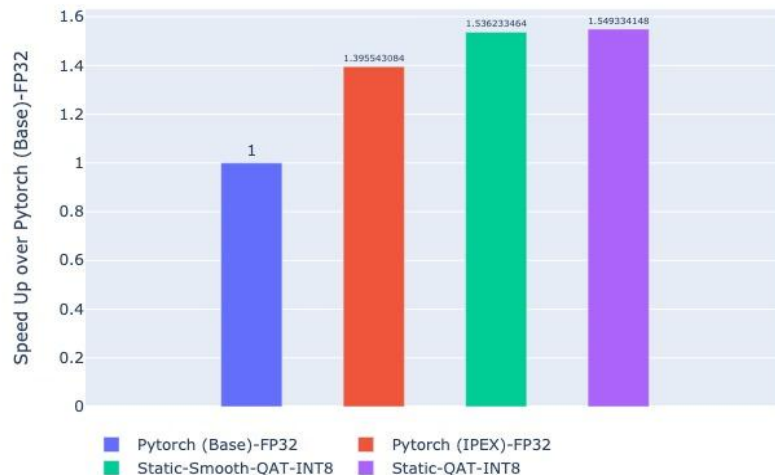
Extractive QA Model Latency Comparison  
(\* using backend as **IPEX** for INT8 models)Extractive QA Model Speed Up Comparison  
(\* using backend as **IPEX** for INT8 models)

Fig: Latency/Speed-Up Benchmark result for our **Extractive Question Answering Model (Multilingual)** on Intel® Dev Cloud machine (Intel Xeon Processor (Skylake, IBRS) - 10v CPUs 16GB RAM) with optimization using IPEX-FP32 and Static INT8-Quantization using Intel® Neural Compressor. Please Note that we use backend as **IPEX** for INT8 models here to get further benefit.

Extractive QA Model Throughput Comparison  
(\* using backend as **IPEX** for INT8 models)



Extractive QA Model F1 Score (SQuAD-v1) Comparison  
(\* using backend as **IPEX** for INT8 models)

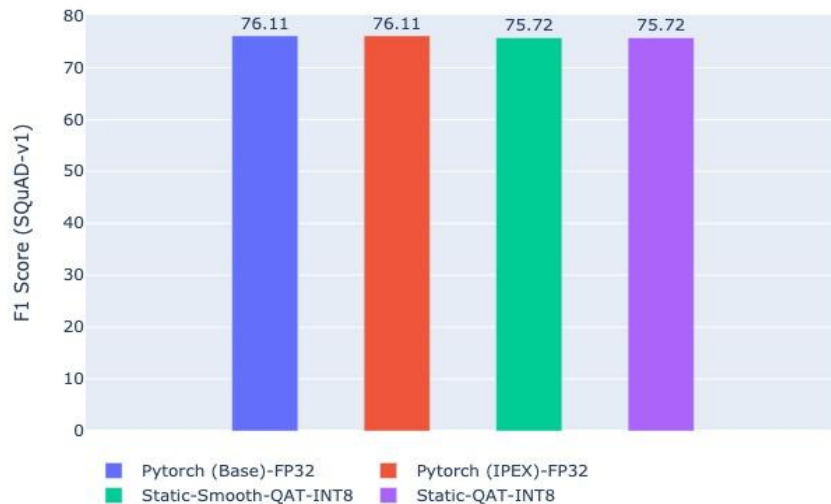


Fig: Throughput/F1 Score Benchmark result for our **Extractive Question Answering Model (Multilingual)** on Intel® Dev Cloud machine (Intel Xeon Processor (Skylake, IBRS) - 10v CPUs 16GB RAM) with optimization using IPEX-FP32 and Static INT8-Quantization using Intel® Neural Compressor. Please Note that, we use backend as **IPEX** for INT8 models here to get further benefit. Also, the model (<https://huggingface.co/ai4bharat/indic-bert>) was fine-tuned on SQuAD-v1 dataset.

Extractive QA Model Latency Comparison  
(\* using backend as **Default** for INT8 models)



Extractive QA Model Speed Up Comparison  
(\* using backend as **Default** for INT8 models)

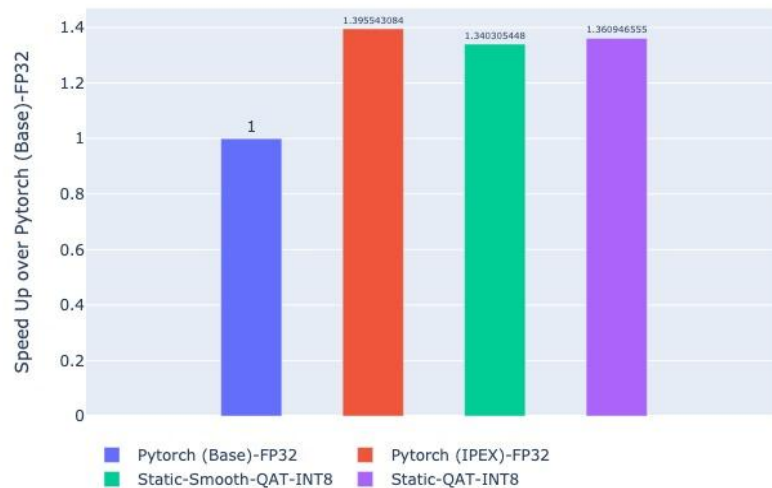
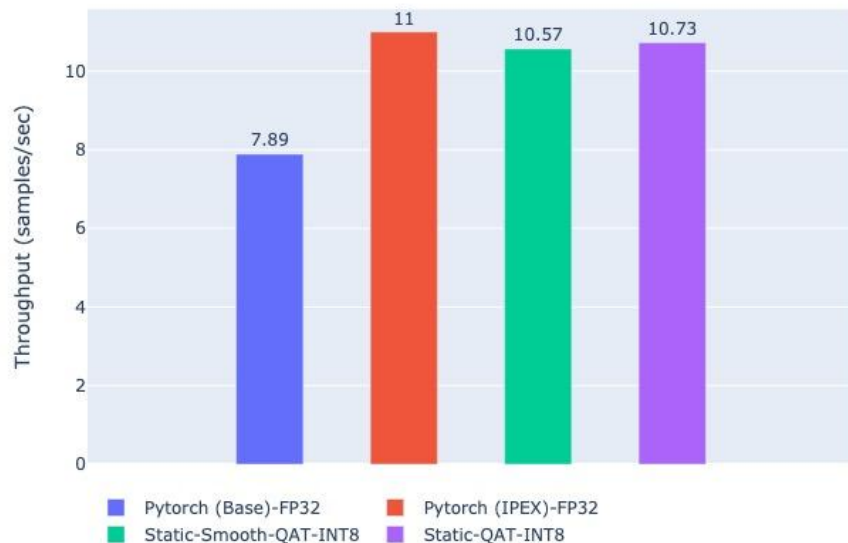


Fig: Latency/Speed-Up Benchmark result for our **Extractive Question Answering Model (Multilingual)** on Intel® Dev Cloud machine (Intel Xeon Processor (Skylake, IBRS) - 10v CPUs 16GB RAM) with optimization using IPEX-FP32 and Static INT8-Quantization using Intel® Neural Compressor. Please Note that we use backend as **Default** for INT8 models here to get further benefit.

# intel<sup>®</sup> Extractive QA Model (BERT Topology) Throughput/F1 Score Comparison with IPEX and Intel<sup>®</sup> Neural Compressor

Extractive QA Model Throughput Comparison  
(\* using backend as **Default** for INT8 models)

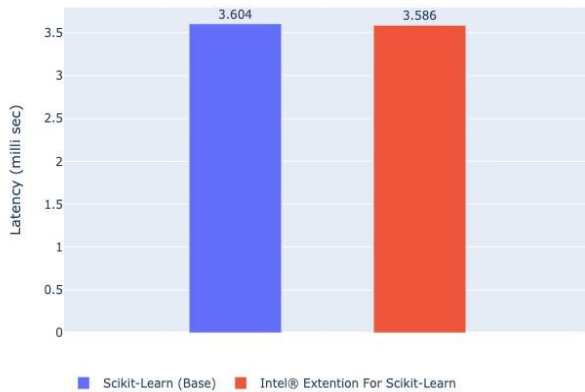


Extractive QA Model F1 Score (SQuAD-v1) Comparison  
(\* using backend as **Default** for INT8 models)

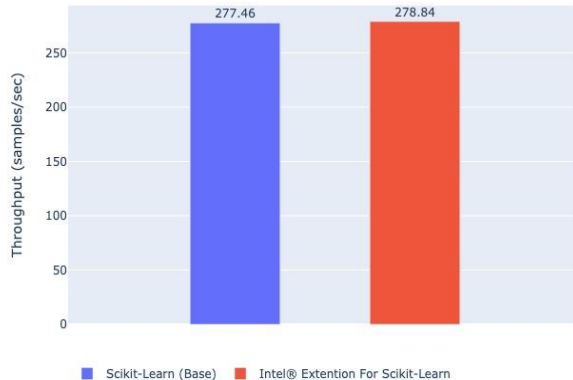


Fig: Throughput/F1 Score Benchmark result for **our Extractive Question Answering Model (Multilingual)** on Intel<sup>®</sup> Dev Cloud machine (Intel Xeon Processor (Skylake, IBRS) - 10v CPUs 16GB RAM) with optimization using IPEX-FP32 and Static INT8-Quantization using Intel<sup>®</sup> Neural Compressor. Please Note that we use backend as **Default** for INT8 models here to get further benefit. Please Note that, the model (<https://huggingface.co/ai4bharat/indic-bert>) was fine-tuned on SQuAD-v1 dataset.

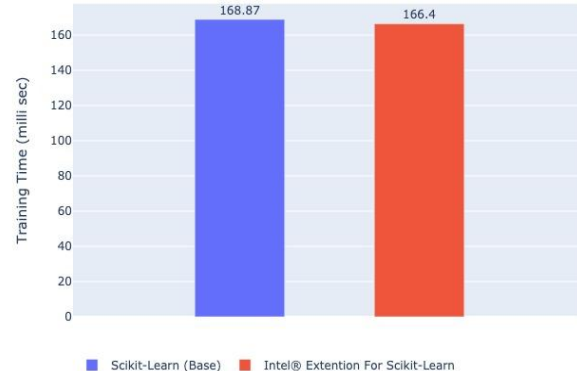
TFIDF Embedding Latency Comparison



TFIDF Embedding Throughput Comparison



TFIDF Embedding Training Time Comparison



*Fig: Benchmark results for **TFIDFVectorizer** Embedding model during training and inference on Intel® Dev Cloud machine (Intel Xeon Processor (Skylake, IBRS) - 10v CPUs 16GB RAM). Please Note that we don't see much of a difference may be because we used a tiny dataset.*



**1**  
**oneAPI**  
**<HACK>ATHON**

**THANK YOU**