International Journal of Advanced Culture Technology Vol.10 No.3 360-370 (2022)
DOI https://doi.org/10.17703/IJACT.2022.10.3.360

IJACT 22-9-43

Comparison of Machine Learning Tools for Mobile Application

[1]Yo-Seob Lee

[1]Professor, School of ICT Convergence, Pyeongtaek University
yslee@ptu.ac.kr

Abstract

Demand for machine learning systems continues to grow, and cloud machine learning platforms are widely used to meet this demand. Recently, the performance improvement of the application processor of smartphones has become an opportunity for the machine learning platform to move from the cloud to On-Device AI, and mobile applications equipped with machine learning functions are required. In this paper, machine learning tools for mobile applications are investigated and compared the characteristics of these tools.

Keywords: *Mobile, Machine Learning, Tool, iOS, Android*

1. INTRODUCTION

The demand for machine learning systems continues to grow and has been adopted by a variety of applications.[1] To respond to this demand, cloud machine learning platforms have been widely used.[2] The application processor of recent smartphones is equipped with high-performance modules such as CPU, GPU, and neural engine. Through this application processor, you can easily lift a subject from an image or remove the background to separate the subject, or frame a text containing frame. You can use Live Text for video features such as copy, translate, view, share, etc. by pausing the video in. Live Text quick actions make it much easier to make phone calls from photos, videos and cameras, visit websites, convert calls, translate languages, and more.[3]

In this paper, we collect data on machine learning tools for mobile applications and compare and analyze the characteristics of machine learning tools for each mobile application based on this.

2. MACHINE LEARNING FOR MOBILE APPLICATION

Machine learning for mobile application is a case of performing inference on a model directly in a mobile application, and is used in a similar sense to on-device machine learning. Here, machine learning for mobile applications means a case of performing machine learning in Android and iOS applications.[4]

Machine learning models send input data, such as images, text, or audio, to a server, and instead of processing it on the server, the mobile application processes it. Table 1 shows machine learning tools for mobile applications. iOS ML API, Core ML, and Create ML are tools provided by Apple, and TensorFlow Lite and ML Kit are tools provided by Google. TensorFlow Lite and ML Kit support not only Android but also iOS operating systems. Core ML, Create ML, and TensorFlow Lite support various models, and iOS ML API and ML Kit support machine learning functions in the form of APIs.

Manuscript received: July 31, 2022 / revised: August 31, 2022 / accepted: September 05, 2022
Corresponding Author: yslee@ptu.ac.kr
Tel: +82-31-659-8369, Fax: +82-31-659-8011
School of ICT Convergence, Pyeongtaek University, Korea

Table 1. Machine Learning Tools for Mobile Application

Tool	Owner	Supported System	Type
iOS ML APIs	Apple	iOS	API
Core ML	Apple	iOS	Model
Create ML	Apple	iOS	Model
TensorFlow Lite	Google	iOS, Android	Model
ML Kit	Google	iOS, Android	API

3. MACHINE LEARNING TOOLS FOR MOBILE APPLICATION

3.1 iOS ML APIs

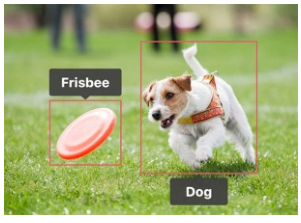
iOS ML APIs provides Vision, Natural Language, Speech, and Sound Analysis frameworks for immediate app integration with support for machine learning capabilities as shown in Table 2.[5] Figure 1 shows the iOS ML Vision APIs. It build features that can process and analyze images and video using computer vision.

(a) Image Classification

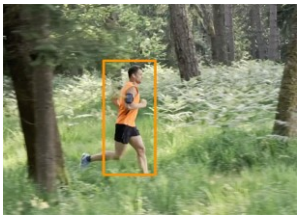

(b) Image Saliency

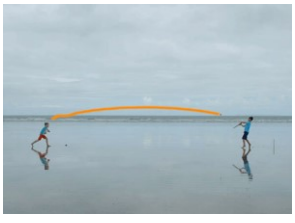

(c) Image Alignment


(d) Image Similarity



(e) Object Detection

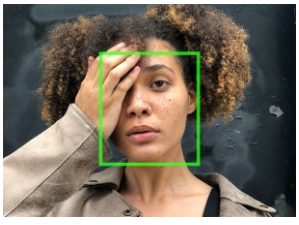

(f) Object Tracking

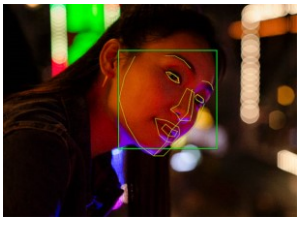

(g) Trajectory Detection

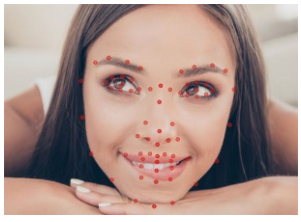

(h) Contour Detection

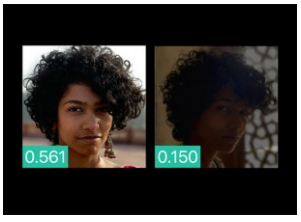

(i) Text Detection

(j) Text Recognition

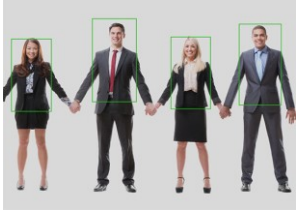

(k) Face Detection


(l) Face Tracking

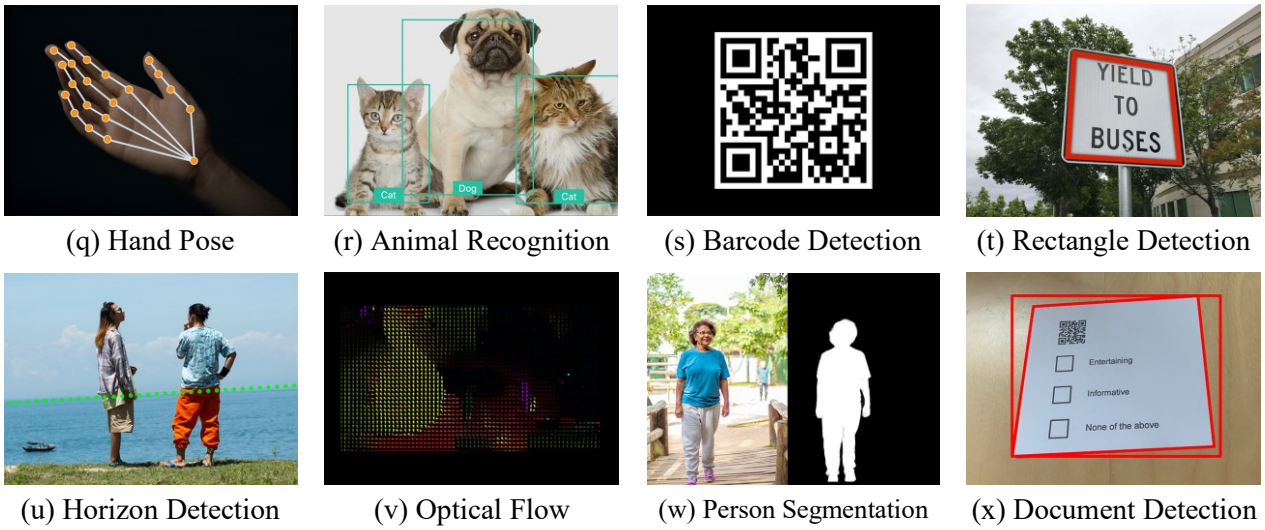

(m) Face Landmarks



(n) Face Capture Quality


(o) Human Body Detection

(p) Body Pose



(q) Hand Pose (r) Animal Recognition (s) Barcode Detection (t) Rectangle Detection

(u) Horizon Detection (v) Optical Flow (w) Person Segmentation (x) Document Detection

Figure 1. iOS ML Vision APIs

Vision framework build features that can process and analyze images and video using computer vision. Natural Language framework analyze natural language text and deduce its language-specific metadata. Speech framework take advantage of speech recognition and saliency features for a variety of languages. Sound Analysis framework analyze audio and recognize it as a particular type, such as laughter or applause.

Table 2. APIs of iOS ML APIs

Category	API	Description
Vision	Image Classification	Automatically identify the content in images.
	Image Saliency	Quantify and visualize the key part of an image or where in the image people are likely to look.
	Image Alignment	Analyze and manage the alignment of images.
	Image Similarity	Generate a feature print to compute distance between images.
	Object Detection	Find and label objects in images.
	Object Tracking	Track moving objects in video.
	Trajectory Detection	Detect the trajectory of objects in motion in video.
	Contour Detection	Trace the edges of objects and features in images and video.
	Text Detection	Detect regions of visible text in images.
	Text Recognition	Find, recognize, and extract text from images.
	Face Detection	Detect human faces in images.
	Face Tracking	Track faces from a camera feed in real time.
	Face Landmarks	Find facial features in images by detecting landmarks on faces.
	Face Capture Quality	Compare face capture quality in a set of images.
	Human Body Detection	Find regions that contain human bodies in images.
	Body Pose	Detect landmarks on people in images and video.
	Hand Pose	Detect landmarks on human hands in images and video.

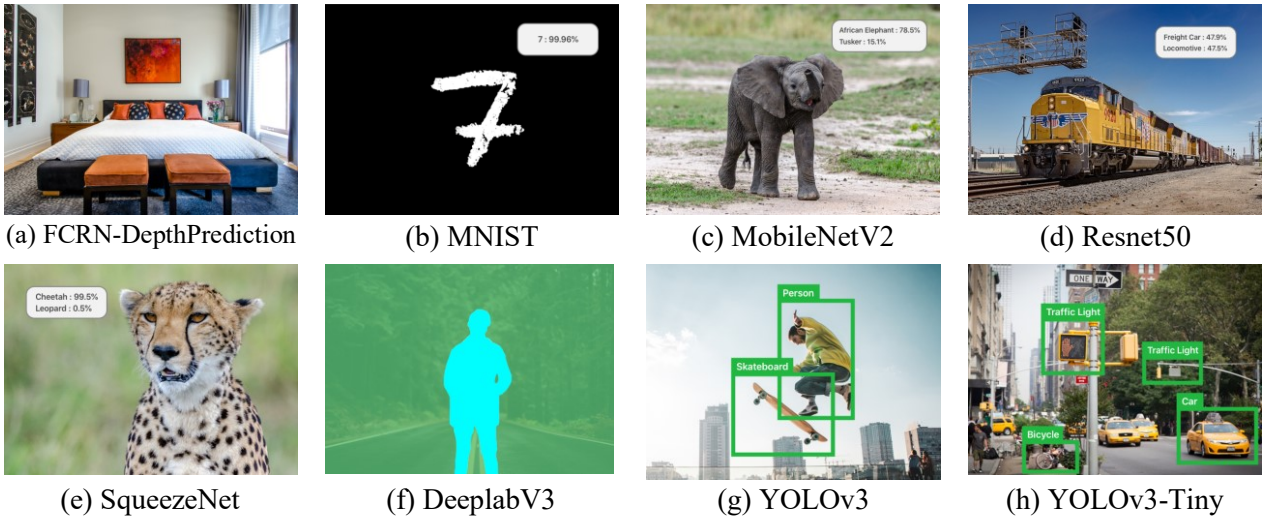
	Animal Recognition	Find cats and dogs in images.
	Barcode Detection	Detect and analyze barcodes in images.
	Rectangle Detection	Find rectangular regions in images.
	Horizon Detection	Determine the horizon angle in images.
	Optical Flow	Analyze the pattern of motion of objects between consecutive video frames.
	Person Segmentation	Produce a matte image for a person in an image.
	Document Detection	Detect rectangular regions in images that contain text.
Natural Language	Tokenization	Enumerate the words in text strings.
	Language Identification	Recognize the language of bodies of text.
	Named Entity Recognition	Use a linguistic tagger to name entities in a string.
	Part of Speech Tagging	Classify nouns, verbs, adjectives, and other parts of speech in a string.
	Word Embedding	Get a vector representation for any word and find similarity between two words or nearest neighbors for a word.
	Sentence Embedding	Get a vector representation for any string and find similarity between two strings.
	Sentiment Analysis	Score text as positive, negative, or neutral based on the sentiment.
Speech	Speech Recognition	Recognize and analyze speech in audio and get back data like transcripts.
Sound Analysis	Sound Classification	Analyze sounds in audio using the built-in sound classifier or a custom Core ML sound classification model.

3.2 Core ML

Core ML provides fast performance on Apple devices with easy integration of machine learning models into your applications.[6]

Core ML models run strictly on the user's device and do not require a network connection, keeping your app responsive and keeping the user's data private. TensorFlow or PyTorch library models can be converted to Core ML much easier than before using the Core ML Converter. Models bundled with your app can be updated with user data on the device, allowing models to remain relevant to user behavior without compromising privacy.

Core ML is tightly integrated with Xcode to provide performance reports, profile with instruments, live preview, model deployment, and model encryption. It generates model performance reports measured on connected devices without having to write any code. Review a summary of load and prediction times along with a breakdown of compute unit usage. Profile with instruments profile your app to view Core ML API calls and associated models using the Core ML instrument. Find out where and when Core ML dispatches work to the hardware and gain more visibility with the Metal and new Neural Engine instruments. Live preview preview your model’s behavior on sample data files or live from your device’s camera and microphones, all right in Xcode. With Core ML Model Deployment, you can easily distribute models to your app using CloudKit. Xcode supports model encryption, enabling additional security for your machine learning models. Figure 2 shows the Core ML Image Models.



(a) FCRN-DepthPrediction (b) MNIST (c) MobileNetV2 (d) Resnet50

(e) SqueezeNet (f) DeeplabV3 (g) YOLOv3 (h) YOLOv3-Tiny

Figure 2. Core ML Models

Core ML model has models of image and text categories as shown in Table 3, and can be integrated into Xcode project, and various versions of models can be selected and optimized for size and architecture.

Table 3. Model of CoreML

Category	Model Type	Model Name	Description
Image	Depth Estimation	CRN-DepthPrediction	Predict the depth from a single image.
	Drawing Classification	MNIST	Classify a single handwritten digit (supports digits 0-9).
	Drawing Classification	UpdatableDrawingClassifier	Drawing classifier that learns to recognize new drawings based on a K-Nearest Neighbors model (KNN).
	Image Classification	MobileNetV2	The MobileNetv2 architecture trained to classify the dominant object in a camera frame or image.
	Image Classification	Resnet50	A Residual Neural Network that will classify the dominant object in a camera frame or image.
	Image Classification	SqueezeNet	A small Deep Neural Network architecture that classifies the dominant object in a camera frame or image.
	Image Segmentation	eeplabV3	Segment the pixels of a camera frame or image into a predefined set of classes.
	Object Detection	YOLOv3, YOLOv3-Tiny	Locate and classify 80 different types of objects present in a camera frame or image.
	Pose Estimation	PoseNet	Estimates up to 17 joint positions for each person in an image.
Text	Question Answering	BERT-SQuAD	Find answers to questions about paragraphs of text.

3.3 Create ML

The Create ML app lets you quickly build and train Core ML models right on your Mac, no code required. Model building is easy with an easy-to-use app interface and models that can be used for training.[7]

Create ML has features such as multimodel training, training control, eGPU training support, on-device training, model previews, and visual evaluation. Multimodel training trains multiple models using different datasets, all in a single project. Training controls pause, save, resume, and extend your training process. eGPU training support uses an external graphics processing unit with your Mac for even better model training performance. On-device training train models blazingly fast right on your Mac while taking advantage of CPU and GPU. Model previews preview your model performance using Continuity with your iPhone camera and microphone on your Mac, or drop in sample data. Visual evaluation interactively learns how your model performs on test data from your evaluation set.

Create ML provides a variety of model types as shown in Table 4, you can train by selecting the model type in the app and adding data and parameters.

Table 4. Model Type of Create ML

Category	Model Type
Image	Image classification, Object detection, Hand pose classification, Style transfer
Video	Action classification, Hand action classification, Style transfer
Motion	Activity classification
Sound	Sound classification
Text	Text classification, Word tagging
Tabular	Tabular classification, Tabular regression, Recommendation

3.4 TensorFlow Lite

TensorFlow Lite is a set of tools that enables on-device machine learning by helping developers run their models on mobile, embedded, and edge devices.[8]

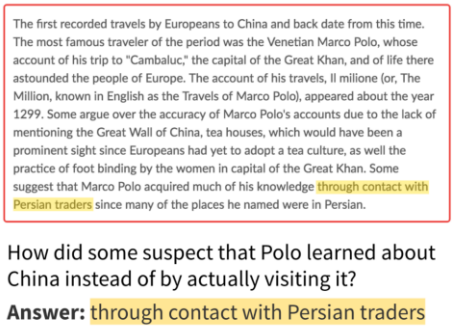
TensorFlow Lite works with model selection, model conversion, Deploy, and Optimization steps. The model selection is pick a new model or retrain an existing one. The model conversion converts a TensorFlow model into a compressed flat buffer with the TensorFlow Lite Converter. Deploy takes the compressed .tflite file and load it into a mobile or embedded device. Optimization quantize by converting 32-bit floats to more efficient 8-bit integers or run on GPU.

TensorFlow Lite uses TensorFlow models converted to a smaller and more efficient machine learning model format. With TensorFlow Lite, you can use pre-trained models, modify existing models, or build your own TensorFlow models and then convert them to the TensorFlow Lite format.

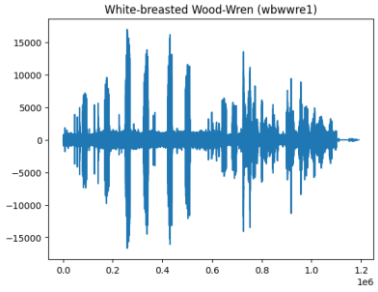
Table 5. Model Type of TensorFlow Lite

Category	Model Type
Image	Image classification, Object detection
Sound	Speech synthesis, Audio embedding
Text	Text classification, Text embedding

(a) Object Detection



(b) BERT Question Answer



(c) Audio Classification

Figure 3. TensorFlow Lite Models

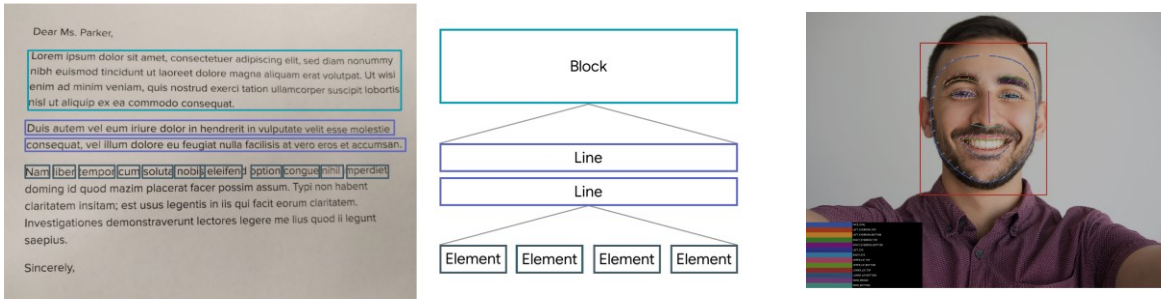
3.5 ML Kit

Table 6. APIs of ML Kit

Category	API	Description
Vision	Image Classification	Automatically identify the content in images.
	Object Detection	Find and label objects in images.
	Object Tracking	Track moving objects in video.
	Text Recognition	Find, recognize, and extract text from images.
	Face Detection	Detect human faces in images.
	Body Pose	Detect landmarks on people in images and video.
	Barcode Detection	Detect and analyze barcodes in images.
	Selfie Segmentation	Separate the background from users within a scene and focus on what matters.
	Digital Ink Recognition	Recognizes handwritten text and handdrawn shapes on a digital surface, such as a touch screen.
Natural Language	Language Identification	Recognize the language of bodies of text.
	Named Entity Recognition	Use a linguistic tagger to name entities in a string.
	Translate Text	Translate text between 58 languages, entirely on device.
	Smart Reply	Generate reply suggestions in text conversations.

ML Kit is a mobile SDK that brings Google's on-device machine learning expertise to Android and iOS apps.[10] ML Kit provides Vision APIs and Natural Language APIs as shown in Table 6. Vision APIs are video and image analysis APIs to label images and detect barcodes, text, faces, and objects. Natural Language APIs are natural language processing APIs to identify and translate between 58 languages and provide reply suggestions.

Vision APIs are composed of Image Classification, Object Detection, Object Tracking, Text Recognition, Face Detection, Body Pose, Barcode Detection, Selfie Segmentation, and Digital Ink Recognition. Natural Language APIs consist of Language Identification, Named Entity Recognition, Translate Text, and Smart Reply.



(a) Text Recognition (b) Face Detection

(c) Selfie Segmentation

Figure 4. ML Kit APIs

4. COMPARISON OF MACHINE LEARNING TOOLS FOR MOBILE APPLICATION

This section compares different machine learning tools such as iOS ML API[4], Core ML[5], Create ML[6], TensorFlow Lite[7] and ML Kit[9].

Table 7 shows the Model Converter of Core ML and TensorFlow Lite. While Core ML Converter converts third-party training libraries to Core ML models, TensorFlow Lite Model Maker uses transfer learning to reduce the amount of training data required and shorten the training time.[9]

Table 7. Model Converter of Machine Learning Tools for Mobile Application

Tool	Model Converter	Description
Core ML	Core ML Converter	Convert models from third-party training libraries into Core ML.
TensorFlow Lite	TensorFlow Lite Model Maker	The TensorFlow Lite Model Maker library simplifies the process of training a TensorFlow Lite model using custom dataset.

Table 8 shows the model types of Core ML, Create ML, and TensorFlow Lite. As most tools support Image Classification and Object Detection, it can be seen that mobile applications utilize these functions a lot. It can be seen that Core ML supports image models, Create ML supports image and audio models, and TensorFlow Lite supports image, audio, and text models evenly.

Table 8. Model Type of Machine Learning Tools for Mobile Application

Model	Core ML	Create ML	TensorFlow Lite
Depth Estimation	O		
Drawing Classification	O		
Image Classification	O	O	O
Object Detection	O	O	O
Pose Estimation	O	O	
Hand Pose/Action Classification		O	
Style transfer		O	
Sound classification		O	
Speech synthesis			O
Audio embedding			O
Text classification		O	O
Text embedding/ Word tagging			O
Question Answering	O		
Tabular classification/regression		O	

Table 9 shows the On-device API Categories of iOS ML API and ML Kit. iOS ML API supports Vision, Natural Language, Speech, and Sound Analysis, whereas ML Kit supports Vision and Natural Language. It can be seen that most mobile applications utilize Vision and Natural Language functions.

Table 9. On-device API Categories of Machine Learning Tools for Mobile Application

API	iOS ML APIs	ML Kit
Vision	O	O
Natural Language	O	O
Speech	O	
Sound Analysis	O	

Table 10 shows the Vision APIs Categories of iOS ML API and ML Kit. t can be seen that most tools in the Vision category support Image Classification, Object Detection, Object Tracking, Text Recognition, Face Detection, Pose Detection, Barcode Scanning, and Selfie Segmentation functions. It can be seen that these functions are currently commonly used functions in mobile applications. What is special is that ML Kit supports the Digital Ink Recognition function, which is thought to be a function provided to utilize the touch screen in mobile devices.

Table 11 shows the NL, Speech, Sound Analysis APIs Categories of iOS ML API and ML Kit. Both iOS ML API and ML Kit support Language Identification and Entity Extraction functions, and it can be seen that there is a high demand for functions that utilize the objects of each country's language and sentence in natural language processing. iOS ML API supports Natural Language, Speech, and Sound Analysis categories, but ML Kit supports only Natural Language category, indicating that there is still a lot of demand for natural language processing.

Table 10. Vision APIs of Machine Learning Tools for Mobile Application

API	iOS ML APIs	ML Kit
Image Classification	O	O
Image Saliency	O	
Image Alignment	O	
Image Similarity	O	
Object Detection /Image Labeling	O	O
Object Tracking	O	O
Trajectory Detection	O	
Contour Detection	O	
Text Detection	O	
Text Recognition	O	O
Face Detection	O	O
Face Tracking	O	
Face Landmarks	O	
Face Capture Quality	O	
Human Body Detection	O	
Body Pose / Pose Detection	O	O
Hand Pose	O	
Animal Recognition	O	
Barcode Detection / Barcode Scanning	O	O
Rectangle Detection	O	
Horizon Detection	O	
Optical Flow	O	
Person Segmentation / Selfie Segmentation	O	O
Document Detection	O	
Digital Ink Recognition		O

Table 11. NL, Speech, Sound Analysis APIs of Machine Learning Tools for Mobile Application

Category	API	iOS ML APIs	ML Kit
Natural Language	Tokenization	O	
	Language Identification	O	O
	Named Entity Recognition / Entity Extraction	O	O
	Part of Speech Tagging	O	
	Word Embedding	O	
	Sentence Embedding	O	
	Sentiment Analysis	O	
	Translate Text		O
	Smart Reply		O
Speech	Speech Recognition	O	
Sound Analysis	Sound Classification	O	

After examining the machine learning tools for mobile applications and comparing the characteristics of these tools, the proposal for improvement of the machine learning tools for the next mobile application should have the following characteristics.

- It should provide a model conversion that can convert the pre-trained models implemented in various formats into a format usable by mobile applications for use.
- It should support multiple models, such as image, audio, and text models.
- It should support various on-device API categories such as Vision, Natural Language, Speech and Sound Analysis.

5. CONCLUSION

In this paper, we investigated machine learning tools for mobile applications such as iOS ML APIs, Core ML, Create ML, TensorFlow Lite, and ML Kit.

Machine learning tools for mobile applications can convert machine learning models made with other libraries into models for mobile applications through model converter or model maker, or improve learning speed. This allows many existing machine learning models to be used in mobile applications.

Most of the machine learning tools for mobile applications support APIs in various fields such as Vision, Natural Language, Speech, and Sound Analysis, making it easy to use machine learning functions in mobile applications.

We believe that review of machine learning tools for mobile applications can help developers and planners who want to integrate machine learning capabilities into mobile applications easily and quickly.

REFERENCES

[1] Y. Lee, “Analysis on trends of machine learning-as-a-service,” Vol. 6, No. 4, International Journal of Advanced Culture Technology, 2018.
[2] Y. Lee, "Analysis of Automatic Machine Learning Solution Trends of Startups," Vol.8, No.2, International Journal of Advanced Culture Technology, 2020.
[3] iOS 16 Preview, https://www.apple.com/ios/ios-16-preview/.
[4] H. Song, "Comparison of On-Device AI Software Tools", Vol.10 No.2, International Journal of Advanced Culture Technology, 2022.
[5] iOS ML APIs, *https://developer.apple.com/machine-learning/api/*.
[6] Core ML, *https://developer.apple.com/machine-learning/core-ml/*.
[7] Create ML, *https://developer.apple.com/machine-learning/create-ml/*.
[8] TensorFlow Lite, *https://www.tensorflow.org/lite*.
[9] TensorFlow Lite Model Maker, *https://www.tensorflow.org/lite/models/modify/model_maker*.
[10] ML Kit, *https://developers.google.com/ml-kit*.