**Source: Samsung Electronics Co., Ltd.**

**Title: [FS\_AI4Media] Update on use cases**

**Agenda Item: 9.8**

**Document for: Agreement**

1. Introduction

This contribution provides an update to the text on use cases currently in the Permanent Document (S4-220500).

1. Revised Text

TR 22.874 [1] has identified a set of use cases for AI/ML with the following key operations:

* AI/ML operation splitting between AI/ML endpoints;
* AI/ML model/data distribution and sharing over 5G system;
* Distributed/Federated Learning over 5G system.

These operations have been identified as they require exchange of ML and media data over 5G, and in some cases may have some requirements on the QoS for proper operation.

Of interest to SA4 are the media-based use cases, which are identified in this clause.

2.1 Object Recognition in Image and Video

In this set of use cases, images and video streams are processed to identify and recognize objects and extract some metadata, such as bounding boxes, object labels, movement counters, etc.

The specific scenarios that are considered are the following:

* Delivery of trained ML model(s) for object recognition to the UE in 5GS, including the selection of models for different tasks or environments. This scenario involves the key operation of AI/ML model/data distribution.
* Split inference of trained ML model(s) for object recognition between multiple endpoints, typically between the network and UE. Split points may depend on various factors including UE capabilities, network conditions, and model characteristics. Model characteristics include split inference with a task-specific model head running on the UE for object recognition. For example, in one UE, the task is to recognize pedestrians, whereas in another it is to recognize traffic signs. The core of the network model as well as the input image/video are the same, but the tasks (and their required task-specific models) in the UEs are different. This scenario involves both AI/ML operation splitting, and AI/ML model/data distribution.
* Distributed online training of image and video recognition models based on input from different UEs. Depending on the configuration of the ML training framework, different data may need to be delivered between the UEs and the network. Typically a shared model in the network is calibrated continuously based on the training results from all UEs. This scenario involves all the three key operations related to AI/ML model distribution, splitting, and distributed/federated learning.

2.2 Video Quality Enhancement in Streaming

In this use case, the sender and receiver apply parts of an autoencoder DNN model to enhance the quality of a video stream. This is depicted in the following diagram:



The sender is typically represented by various media functions in the network, which processes the high-fidelity video using the down-scaling part of a pre-trained DNN model to generate a metadata stream that is streamed together with a lower fidelity encoding of the video. The receiver (UE) runs an inference algorithm (e.g. the up-scaling part of DNN model) on the received metadata and video stream to produce a high-quality video for rendering.

The main scenario in this use case is about streaming intermediate model output data from the network for processing on the UE, involving AI/ML data distribution and operation splitting.

2.3 Crowd-Sourcing media Capture

A set of users attending a live concert and capturing the event on their UEs, use a shared (or a set of shared) DNN model(s) to process and improve their respective captured video and/or audio. Audio and video data may be captured in a noisy environment or an environment with poor lighting conditions. Multiple tasks may then be performed on the processed video and/or audio for media content analysis, e.g. to extract lyrics, annotate the video, improve audio and video quality, translate language, anonymize a face, etcThis use case involves the distribution of multiple models to a large number of UEs in a short period of time. The UEs are heterogenous, running with different types of operating systems (e.g., Android or iOS), supporting different AI/ML engines/frameworks or having different GPU/CPU/NPU and RAM capabilities available for running the AI/ML service on the UE. This will require the distribution of a huge amount of various AI/ML models adapted to the different device capabilities. Depending on each user’s UE, instead of requesting the download of the DNN models for device inference, it may also request to upload the media data for network inference.

In many scenarios, the size of AI/ML model may be quite large (200-500 Mbytes) while the network resources may be limited. As an example, a VGGNet model which handles object detection has a size of 536 MBytes. The distribution time for this model may range from 1 second at 4.2 Gbps to 100 seconds at 42 Mbps. In addition, moving or changing the environment (localization, energy, processing unit, memory, etc.) may require AI/ML model updates, where the

The AI/ML application may also optimize the end-to-end latency (e.g., to achieve latency below 1s) or the expected accuracy level of the inference result (e.g., to achieve image recognition precision of 99%) by modifying the model. The desired latency and/or accuracy level can therefore impact the size of the AI/ML model to be distributed.

The distribution of the AI/ML models for a large number of UEs at the same time may require the 5GS to serve the models from different endpoints (e.g., cloud, edge, or other UEs), and may use several or different communication links (e.g. unicast, multicast or broadcast).

The main scenario here is to improve the media capture of each UE by using an up-to-date model adapted to the context event.

Another scenario may be the sharing of the input media from multiple sources for network inference, as well as the selection of suitable DNN models according to the UE and/or task. Depending on the requirements of each UE, the scenario may involve all three key operations related to AI/ML model distribution, splitting, and distributed/federated learning.

2.4 NLP on Speech

This set of use cases covers a wide range of speech processing use cases, e.g. to perform automatic speech recognition, voice translation, voice commands, speech synthesis, etc.

The AI/ML models for NLP are improved with distributed/federated training using multiple UEs. As more users make use of the service, the quality and accuracy of the models improves. The results of the local training of the models by the UEs are shared with the network.

The main scenario here is about UE downloading a partially trained model identified with its training state for local training, and then sharing the results with the network for distributed/federated learning.

1. Proposal

We propose to update the text in the PD with the revised text in section 2 of this contribution.