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

**Title: [FS\_AI4Media] Discussion on basic architecture for AI/ML**

**Agenda Item: 9.8**

**Document for: Agreement**

1. Introduction

This contribution provides a brief starting point for discussions on a basic architecture for AI/ML media services.

1. Scenarios

Considering the related use cases as documented in TR 22.874 and also as documented in the latest version of the Permanent Document (S4-220500), we can start from some basic scenarios for consideration of a basic architecture for AI/ML media services.

The basic starting steps are:

1. Delivery of a pre-trained AI/ML model from network to UE, typically at the start of an AI media service, but may also require updates during the service. At the most basic level AI/ML models can be delivered as a file (e.g. TensorFlow SavedModel, PDF5, ONNX file, NNEF file etc.) containing all the necessary information required for the UE to perform on device inference using the delivered model. For split scenario, a (partial) AI model to be used in the UE may be delivered.
2. Split inference of a pre-trained AI/ML model(s) for two scenarios:
	1. Basic scenario with an inference in the network or in the UE.
	2. Split scenario with inferences between the network and the UE, where the intermediate data output from the network inference (resp. UE inference) is transferred to the UE (resp. network) to be used as the input for UE device inference (resp. network inference). Depending on the characteristics of the intermediate data, such as if the intermediate data is media content data, it may be practical to consider 5GMS architectures, procedures and/or protocols for the streaming delivery of such intermediate media data.
3. Architectures

 

Figure 1: simple architecture for AI/ML model delivery with an inference in the UE

Figure 1 shows a possible simple architecture for AI/ML model delivery, as described in step 1 of section 2, with an inference of a pre-trained AI/ML model in the UE, as described in scenario a) of section 2.

In the network:

* An AI AF provides various control functions to the AI Model Session Handler on the UE and/or to the AI Media Application Provider. It may also interact with existing 5G Network Functions, such as a PCF, NEF, Data Collection AF and/or NWDAF.
* An AI Model AS supports AI model data hosting, ingesting AI models from the AI Media Application Provider.
* An external AI Media Application Provider with AI-specific media functionality (e.g. AI model creation, splitting, updating etc.)

In the UE:

* An AI Model Session Handler function that communicates with the AI Media AF in order to establish, control and support an AI model delivery session. Depending on the session(s) required may also perform consumption/QoE related collection and reporting.
* An AI Client function that communicates with the AI Model AS to receive AI model data from the AI Model Handler, including management of the received AI model (such as processing any updates to the AI model), as well as its inference by an AI engine, using the relevant media input data from the AI Media Application (or possibly from other internal UE sources).

A typical procedure for AI/ML model delivery may consist of:

1. AI model service and provisioning
2. AI model data ingestion from AI media application Provider
3. Service announcement and service access information acquisition
4. AI model request from the UE
5. Delivery of AI model data

Depending on the exact service scenario, AI model updates may be necessary during the service, and different AI model data delivery pipelines may be considered for such purposes.



Figure 2: simple architecture for split inference between network and UE

Figure 2 shows a simple architecture for split inferences between the network and the UE with the network AI Media AS including AI engine functionality in order to perform network inference in the split inference, as described in scenario b) of section 2.

For the split inference scenario, additional components are required:

In the network:

* An AI media AS supports AI media (a.k.a intermediate data) hosting, ingesting AI media from the AI Media Application Provider.

In the UE:

* An AI media Session Handler function that communicates with the AI AF in order to establish, control and support an AI media delivery session.

Extra factors should be considered, including those such as:

* Configuration of the split inference between the network and UE
* Resource allocation and management for network inference, including ingestion of network AI model data and media data
* Intermediate data delivery pipelines between the network and UE, in particular considering the use of 5GMS defined pipelines to stream intermediate data that is media content data.
1. Proposal

We propose to include section 3 of this contribution into the Permanent Document.