ONNX Pre-processing WG

Monthly meeting - Nov 10, 2021
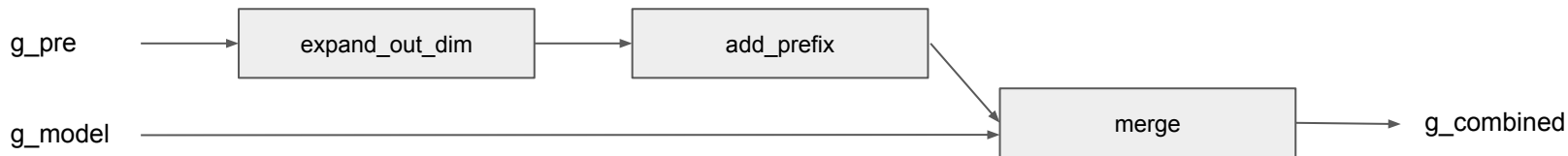
Agenda

- Merging models - Compose Utils
- Control flow operator for batch processing
- Date/String operators for Kaggle preprocessing coverage

Merging models - Compose utils

```python
preprocessing_model = onnx.load('rn50-preprocessing.onnx')
network_model = onnx.load('resnet50v2.onnx')

g_pre = preprocessing_model.graph
g_model = network_model.graph

# Inserts unsqueeze at the end of the preprocessing graph,
# the combined model will take a single sample as an input.
g_pre = onnx.compose.expand_out_dim(g_pre, 0)

# Adds prefix to all relevant names so that they don't collide when merging
g_pre = onnx.compose.add_prefix(g_pre, 'preprocessing/')

# Combine two graphs into a single one. Specific pairs of output->input are specified
g_combined = onnx.compose.merge(g_pre, g_model, io_map=[("preprocessing/xnorm_chw_expanded", "data")])

# Now we are ready to create the model, check it and save it
combined_model = onnx.helper.make_model(g_combined,
                                        opset_imports=[op],
                                        producer_name=network_model.producer_name,
                                        ir_version=network_model.ir_version)
```

Control flow operator for batch processing

- Applies a subgraph to every sample in a sequence
- Produces a single tensor, with all the samples in the sequence preprocessed and concatenated.
- Looking for contributors

Date/String operators for Kaggle preprocessing coverage