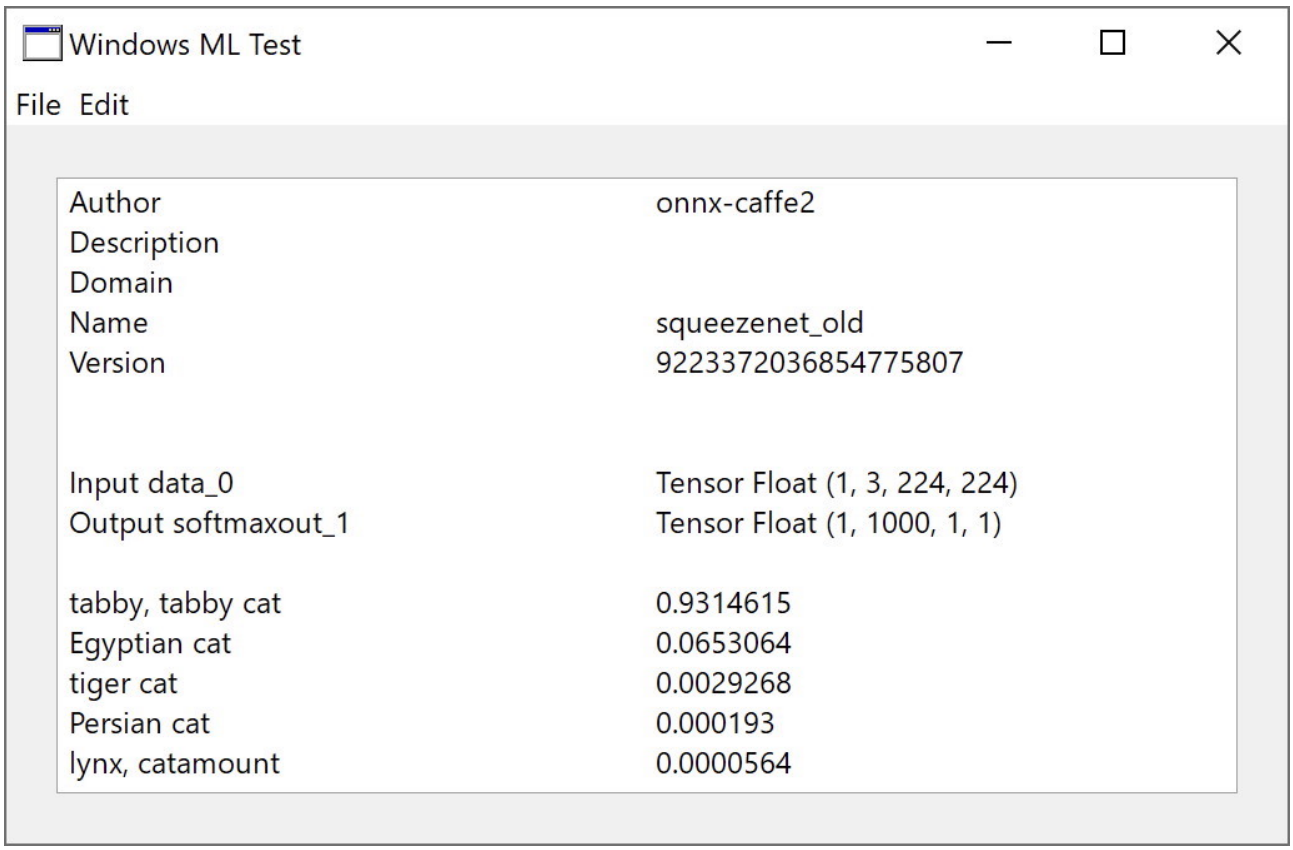
Playing with Machine Learning on Windows

We have started to look on the Windows Learning APIs to implement some functions for our plugins. Since 2017 we have CoreML functions for macOS. FileMaker 19 now ships similar functions built-in, but we think the plugin can still do more. As we are now using newer Visual Studio 2019, we can finally also check the Windows Learning functions:

You can use ONNX Models with the classes, so check the Microsoft website on how to get models. This mainly points to the ONNX Model Zoo, which has some interesting models available.

We started by porting the desktop SqueezeNetObjectDetection example from the Windows-Machine-Learning repository. You may want to download the SqueezeNet.onnx file from models folder and the kitten_224.png file from the media folder.



For **FileMaker** we embrace JSON and use it to pass values for the new WindowsML functions. Use WindowsML.Open to load the model and query all information about it with WindowsML.Description function. Use bind functions like WindowsML.BindImageFile to assign input image, run

the model with WindowsML.Evaluate function and then you get a result as JSON. You may use our JSON functions to work on the result and show it to the user.

A difference between the macOS/iOS implementation by Apple and the one by Microsoft is the missing of labels for the latter. For Windows you get a Labels.txt file with the list of what index in the result points to what label they mean. Our example code will show how to handle this. Those functions and new classes are coming for next pre-release versions in October 2020. We may get a good start set and may add more as needed later. Especially as we learn what other models may need as input and output features.