AutoML Platform for Sensor Data

Tina Shyuan, Product Manager
July 25, 2019
Enabling Intelligence at the Edge

Billions of sensors are collecting data on every device imaginable. Qeexo aims to equip every sensor with machine learning to discover knowledge, make predictions, and generate actionable insights at the microcontroller level.

“We need something that works on cheap microcontrollers, that uses very little energy, that relies on compute not radio, and that can turn all our wasted sensor data into something useful.”

– Pete Warden, Lead of the TensorFlow Mobile/Embedded team
Company

Sept 2012
Spun out of CMU

April 2015
Commercial launch with Huawei

July 2018
Commercial launch with OPPO

Jan 2019
EarSense CES Innovation Awards Honoree

Feb 2019
EarSense finalist of MWC GLOMO Awards

Q2 2019
Sold 200+ million units

June 2019
EarSense finalist of SensorsExpo Best of Sensors Awards

H2 2019
Launch Qeexo AutoML

Expertise: machine learning, signal processing, sensor fusion, embedded computing
Trend: Machine Learning is Moving to the Edge

Running ML at the Edge…

- Improves privacy & availability
- Reduces latency, bandwidth & power usage
- Enables smarter, more agile applications

Industry leaders going “lightweight”

- Google’s TensorFlow Lite
- NVIDIA’s Jetson Nano (ARM A57)
- Qualcomm’s Snapdragon AI
- Intel’s Neural Compute Stick 2 (“VPU”)
- ARM, NXP, ST Micro, QuickLogic, etc.

Edge AI Systems (hardware) Forecast

CAGR 10.6%  16.6%  *Source: IDC
Case Study: Power Consumption on a Smartphone

<table>
<thead>
<tr>
<th>Component</th>
<th>Power Consumption (milliwatts)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Cell Radio</td>
<td>800</td>
</tr>
<tr>
<td>Display</td>
<td>400</td>
</tr>
<tr>
<td>GPS</td>
<td>176</td>
</tr>
<tr>
<td>Gyroscope</td>
<td>130</td>
</tr>
<tr>
<td>Microphone</td>
<td>101</td>
</tr>
<tr>
<td>Bluetooth</td>
<td>100</td>
</tr>
<tr>
<td>Accelerometer</td>
<td>21</td>
</tr>
<tr>
<td>Microcontroller</td>
<td>&lt; 1</td>
</tr>
</tbody>
</table>

5G powering the Internet of Things… is it really feasible?
- High cost (infrastructure & bandwidth)
- High power consumption

Better solution: on-device computation (only send the results, only when necessary)

Source: https://petewarden.com/2015/10/08/smartphone-energy-consumption
Problem: Machine Learning at the Edge is Complex

1. Requires a rare breed of “rocket scientists” (ML engineers)
   Existing machine learning tools and platforms require deep technical knowledge to achieve commercial performance. There is a shortage of machine learning experts who can effectively bridge the gap from the lab to the real world.

2. Labor-intensive and time-consuming
   Due to inefficient tools and the nature of machine learning for Edge devices, even experts take a long time to optimize machine learning models that need to function in the field.

3. Under-utilized data = lost opportunities
   Machine learning at the Edge is subject to highly-constrained environments. Sensor data common at the Edge is often unintuitive and requires signal processing expertise (unlike, e.g. image data). Both result in under-utilized sensor data.
Solution: Qeexo AutoML Platform for Sensor Data

- Intuitive one-click machine learning platform for everyone
- Automates labor-intensive tasks
- Optimizes lightweight machine learning package for Edge devices (e.g. classification on MCUs)
AutoML: Genesis

Exponential increase in # of variants
- Each variant is a slightly different machine learning problem
- How are we ever going to be able to scale?

AUTOMATE!
Qeexo AutoML Platform for Sensor Data

Web Front-end

- Define Project e.g. Classification
- Select Sensors and Target Hardware (e.g. XDK)
- Collect / Upload Data

Qeexo AutoML Back-end

- Fully-Automated, Behind-the-Scenes Machine Learning
  - Data Preprocessing
  - Feature Extraction & Selection
  - Model Selection
  - Hyperparameter Optimization
  - Model Validation
  - Model Conversion to C
  - Compile Target-Specific ML Package

On-Demand Visualization & Performance Reporting (directly from Web App)
Qeexo AutoML Web Interface

Step 1: Enter Project Details
- Project Name: Please enter your project name.

Step 2: Select Target Hardware & Sensors
- Select Target Hardware:
- Select Sensors:
  - Accelerometer
  - Thermometer
  - Camera
  - Ultrasonic
  - Gyroscope
  - Humidity Sensor
  - Touchscreen
  - Radar
  - Magnetometer
  - Pressure Sensor
  - Microphone
  - Geophone

Secure Authentication
Data Visualization
Sensor Selection
On-Demand Reporting
Supported Sensors

**Motion Sensors:**
accelerometer, gyroscope & magnetometer, radar, etc.

**Acoustic Sensors:**
microphone, ultrasonic, geophone, vibrometers

**Environmental Sensors:**
temperature, humidity, air pressure, illumination, IR, etc.

**Touchscreen Sensors:**
capacitive & IR touchscreens

**Image Sensors:**
image & thermal cameras

**Biometric Sensors***:
fingerprint, heart rate, etc.

* Sensors currently under research
### Qeexo AutoML Advantage

<table>
<thead>
<tr>
<th>Feature</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Ingests 1D &amp; 2D data</td>
<td>✓</td>
</tr>
<tr>
<td>Feature Extraction &amp; Selection</td>
<td>✓</td>
</tr>
<tr>
<td>Built-In Quantization</td>
<td>✓</td>
</tr>
<tr>
<td>Choice of ML Models (e.g. CNN, GBM, Random Forest)</td>
<td>✓</td>
</tr>
<tr>
<td>Model Compression</td>
<td>✓</td>
</tr>
<tr>
<td>Tunable Library Generation</td>
<td>✓</td>
</tr>
<tr>
<td>Ease of Integration</td>
<td>✓</td>
</tr>
<tr>
<td>Target Code Generation (Mobile &amp; Edge)</td>
<td>✓</td>
</tr>
<tr>
<td>Out-of-the-Box Data Visualization</td>
<td>✓</td>
</tr>
<tr>
<td>One-Click Performance Evaluation</td>
<td>✓</td>
</tr>
</tbody>
</table>
Qeexo AutoML Model Selection Examples

Case Study #1: Fan
- 2-class classification (normal vs. abnormal)
- Sensors: accelerometer, gyroscope, magnetometer
- Sampling rate: 3330 Hz (accel, gyro); 100 Hz (mag)

<table>
<thead>
<tr>
<th>ML Model</th>
<th>Accuracy</th>
<th>Binary Size</th>
<th>Avg. Latency* (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>99.25%</td>
<td>85 KB</td>
<td>0.0104</td>
</tr>
<tr>
<td>Random Forest</td>
<td>99.42%</td>
<td>133 KB</td>
<td>0.0251</td>
</tr>
<tr>
<td>GBM</td>
<td>98.91%</td>
<td>189 KB</td>
<td>0.0115</td>
</tr>
<tr>
<td>ANN</td>
<td>99.25%</td>
<td>431 KB</td>
<td>0.2206</td>
</tr>
</tbody>
</table>

* Latency calculated on MacBook Pro.

Case Study #2: Activity Tracking
- 4-class classification (rest, walk, run, bike)
- Sensors: accelerometer, gyroscope, magnetometer
- Sampling rate: 13 Hz

<table>
<thead>
<tr>
<th>ML Model</th>
<th>Accuracy</th>
<th>Binary Size</th>
<th>Avg. Latency* (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>93.31%</td>
<td>52 KB</td>
<td>0.0065</td>
</tr>
<tr>
<td>Random Forest</td>
<td>94.82%</td>
<td>&gt; 1024 KB</td>
<td>N/A</td>
</tr>
<tr>
<td>GBM</td>
<td>95.32%</td>
<td>81 KB</td>
<td>0.0207</td>
</tr>
<tr>
<td>ANN</td>
<td>93.99%</td>
<td>227 KB</td>
<td>0.0643</td>
</tr>
</tbody>
</table>

* Latency calculated on MacBook Pro.
Cost Savings

Qeexo AutoML platform generates huge time and cost savings, freeing machine learning engineers from repetitive work and allowing users to quickly iterate through prototypes and solutions.

### Case Study: ML Engineer Time Spent on one Project

<table>
<thead>
<tr>
<th>Task</th>
<th>Manual (hours)</th>
<th>AutoML (hours)</th>
<th>Time Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Visualization</td>
<td>8.00</td>
<td>0.25</td>
<td>97%</td>
</tr>
<tr>
<td>Data Cleaning &amp; Preprocessing</td>
<td>16</td>
<td>0.1</td>
<td>99%</td>
</tr>
<tr>
<td>Building Initial Machine Learning Model</td>
<td>0.25</td>
<td>0.05</td>
<td>80%</td>
</tr>
<tr>
<td>Validating Models (CVs, etc.)</td>
<td>4</td>
<td>4</td>
<td>0%</td>
</tr>
<tr>
<td>Implementation on Target Hardware</td>
<td>80</td>
<td>4</td>
<td>95%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>108.25</strong></td>
<td><strong>8.4</strong></td>
<td><strong>92%</strong></td>
</tr>
</tbody>
</table>
# Mobile Machine Learning Applications Built on AutoML

<table>
<thead>
<tr>
<th>Software</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FingerSense</td>
<td>Input-type classification for touchscreens</td>
</tr>
<tr>
<td>EarSense 2.0</td>
<td>Software-only proximity sensing solution</td>
</tr>
<tr>
<td>TouchTools</td>
<td>Understand user intention from hand pose on screen</td>
</tr>
<tr>
<td>idSense</td>
<td>Software-only anti-spoofing solution</td>
</tr>
</tbody>
</table>
Thanks! Questions?
Mobile ML Applications built on AutoML
FingerSense

Software-only solution that knows how the user is touching the screen

- Uses data from existing sensors
- Applies machine learning to distinguish between finger and knuckle touch inputs
- Provides rapid way to access advanced functionality
- 36.6% of total Huawei users frequently use FingerSense*
- In the market since 2015

*survey results from Shanghai ZT Consulting Co., Ltd
FingerSense

https://www.youtube.com/watch?v=sCE8FYwQjz8
EarSense 2.0

Limitations with hardware proximity sensors

• Must be placed at the top of the smartphone
• Requires bezels/notches/cutouts

Qeexo EarSense

• AI-based software-only alternative to hardware proximity sensors
• Allows device manufacturers to achieve true full-screen, bezel-less designs
• Applies machine learning to distinguish between face/ear/finger
EarSense 2.0

Ultrasonic, capacitive-touch, and motion-sensor-based proximity detection that enables bezel-less, full-screen design

Hover or Touch Event

Ultrasonic Data

Touch Data

IMU Data

Qeexo ML Engine

Compute Touch TYPE: FINGER or EAR

Classification Outputs

Turns Screen On/Off

Screen On

Screen Off
https://www.youtube.com/watch?v=j4WsW7yZ7Cs
TouchTools

AI that understands user intent by looking at the pose of their hands to summon virtual tools

- Intuitively & immediately summon desired tools (e.g. mouse, dial, stylus, eraser, etc.)
- Replaces cumbersome toolbars
- Multi-user support
- Great in classrooms, boardrooms, or in-car systems
TouchTools

https://www.youtube.com/watch?v=MPw-umFzcZo

https://www.youtube.com/watch?v=OlgAlnLC-o0
Fingerprint sensor limitations

- Fake fingerprints can easily be created from inexpensive tools and materials
- Lacks robust anti-spoofing capabilities

Qeexo idSense

- AI-based, software-only solution that enhances security of fingerprint sensors for smartphones, door locks, etc.
- Applies machine learning to distinguish between real finger and other objects
- Uses data from existing sensors, no additional hardware required
idSense

Touch-data-based machine learning solution that differentiates between real human finger and fake finger
idSense

https://www.youtube.com/watch?v=llR1Fpn6BNA
Embedded ML Applications built on AutoML
Qeexo Anomaly Detection/Predictive Maintenance

https://youtu.be/4SCKi1ylLNU
Qeexo Interactive Wall

https://youtu.be/QMmSqrRN5kUA
Qeexo Intelligent Shipping

https://www.youtube.com/watch?v=1zZchmksAYs
Qeexo Smart Countertop

https://youtu.be/ja6-Pnmfa-I