Supervised Sequential Classification Under Budget Constraints

Kirill Trapeznikov and Venkatesh Saligrama
Boston University

May 1st, 2013

Department of Electrical & Computer Engineering
Overview

- Introduce sequential decision problem
  - Myopic approach: relies on current uncertainty to make a decision
    - Consider synthetic examples
    - Why it does not always work
  - Our approach: incorporate future uncertainty in current decision
    - Examine a two stage system
    - Reduce to supervised learning
- Experiment
- Extend to Multiple Stages
- Generalization Results
The Problem: Sequential Decision System

$K$ stage decision system:
- Stage $k$ can use sensor $k$ for a cost $c_k$
- Measurements can be high dimensional
- Order of stages/sensors is fixed
The Problem: Sequential Decision System

\( K \) stage decision system:
- Stage \( k \) can use sensor \( k \) for a cost \( c_k \)
- Measurements can be high dimensional
- Order of stages/sensors is fixed

Decision at each stage:
- classify using current measurements, or
- request (reject to) next sensor
The Problem: Sequential Decision System

$K$ stage decision system:

- Stage $k$ can use sensor $k$ for a cost $c_k$
- Measurements can be high dimensional
- Order of stages/sensors is fixed

Decision at each stage:

- classify using current measurements, or
- request (reject to) next sensor

Goal: Find decisions: $F = \{f^1, f^2, \ldots, f^K\}$
trade-off error rate vs average acquisition cost
Sensors of Increasing Resolutions

classify handwritten digit images

Do we need all sensors for every decision?
Difficult Decision

\[ \begin{array}{c}
? \\
\rightarrow f^1( ) \\
\rightarrow f^2( ) \\
\rightarrow f^3( ) \\
\rightarrow f^4( )
\end{array} \]
Difficult Decision

\[ ? \rightarrow f^1(\text{[image]}) \xrightarrow{\text{reject}} f^2(\ ) \rightarrow f^3(\ ) \rightarrow f^4(\ ) \]
Difficult Decision

? \rightarrow f^1(\text{image}) \rightarrow f^2(\text{image}) \xrightarrow{\text{reject}} f^3(\text{image}) \rightarrow f^4(\text{image})
Difficult Decision

\[ \square \xrightarrow{f^1} f^2 \xrightarrow{f^3} f^4 \]

- High acquisition cost: need full resolution to make a decision
Difficult Decision

high acquisition cost: need full resolution to make a decision
Easy Decision

\[
\begin{array}{c}
? \quad \rightarrow \quad f^1(\quad) \quad \rightarrow \quad f^2(\quad) \quad \rightarrow \quad f^3(\quad) \quad \rightarrow \quad f^4(\quad)
\end{array}
\]
Easy Decision

\[ ? \rightarrow f^1(\text{image}) \xrightarrow{\text{reject}} f^2() \rightarrow f^3() \rightarrow f^4() \]
Easy Decision

\[
\begin{align*}
? & \quad \rightarrow \quad f^1(\text{image}) & \rightarrow & \quad f^2(\text{image}) & \rightarrow & \quad f^3() & \rightarrow & \quad f^4()
\end{align*}
\]

classify

1
Easy Decision

\[ ? \rightarrow f^1(\text{image}) \rightarrow f^2(\text{image}) \rightarrow f^3() \rightarrow f^4() \]

classify

\[ 1 \]

**small acquisition cost:** full resolution is unnecessary
How to reduce sensor cost?

Sensor 1 is cheap, Sensor 2 is expensive

Centralized strategy:
use both sensors
high cost, low error

Non-adaptive strategy:
only use sensor 1
low cost, high error
How to reduce sensor cost?

Sensor 1 is cheap, Sensor 2 is expensive

Centralized strategy:
- use both sensors
- high cost, low error

Non-Adaptive strategy:
- only use sensor 1
- low cost, high error
How to reduce sensor cost?

Sensor 1 is cheap, Sensor 2 is expensive

Centralized strategy:
- use both sensors
- high cost, low error

Non-adaptive strategy:
- only use sensor 1
- low cost, high error
A better strategy: be adaptive

Only request 2nd sensor on difficult examples
How does it compare?

Same error rate as centralized for half the cost

<table>
<thead>
<tr>
<th>Cost</th>
<th>Sample</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2nd sensor</td>
</tr>
<tr>
<td>cost=0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1st sensor</td>
</tr>
<tr>
<td>cost=1</td>
<td></td>
</tr>
</tbody>
</table>

Average Cost / Sample

Centralized
Non-adaptive
Adaptive

Error Rate
0.2
0.1
0.5
1

Average Cost / Sample
Deciding to reject

How to decide if to use the next sensor?

\[ x \rightarrow f_1(x) \rightarrow \text{reject} \rightarrow f_2(x) \]

- cheap/fast sensor
- expensive/slow sensor

\[ \text{Risk of a decision: } \min \left[ \text{current uncertainty}, \alpha \times \text{cost} + \text{future uncertainty} \right] \]

Acquisition cost justify the reduction in uncertainty?
Deciding to reject

How to decide if to use the next sensor?

Risk of a decision:

\[
\min \left[ \text{current uncertainty}, \alpha \times \text{cost} + \text{future uncertainty} \right]
\]

(uncertainty is in correct classification)

Acquisition cost justify the reduction in uncertainty?
Deciding to reject

Risk = \min \left[ \text{current uncertainty, classify}, \alpha \times \text{cost + future uncertainty reject to next stage} \right]

**Difficulty**: sensor output is not known since it has not been acquired
- How to determine future uncertainty?
- **Must base decision on collected measurements!**
Myopic Approach

Not clear how to determine uncertainty of the future:

\[
\min \begin{bmatrix}
\text{current uncertainty,} & \alpha \times \text{cost + future uncertainty} \\
\text{classify} & \text{reject to next stage}
\end{bmatrix}
\]
Myopic Approach

Not clear how to determine uncertainty of the future:

\[
\min \left[ \text{current uncertainty}, \alpha \times \text{cost + future uncertainty} \right]
\]

classify \hspace{2cm} reject to next stage

Ignore the future, and only use current uncertainty to make a decision:

\[
\min \left[ \text{current uncertainty}, \alpha \times \text{cost} \right]
\]

classify \hspace{2cm} reject to next stage
Myopic Approach

Not clear how to determine uncertainty of the future:

$$\min \begin{cases} \text{classify} & \text{current uncertainty}, \\ \text{reject to next stage} & \alpha \times \text{cost} + \text{future uncertainty} \end{cases}$$

Ignore the future, and only use current uncertainty to make a decision:

$$\min \begin{cases} \text{classify} & \text{current uncertainty}, \\ \text{reject to next stage} & \alpha \times \text{cost} \end{cases}$$

Reduces to:

$$\text{decision} = \begin{cases} \text{classify, uncertainly < threshold} \\ \text{reject, uncertainty} \geq \text{threshold} \end{cases}$$
Train a classifier at a stage $h(x)$

**Classifier uncertainty $\approx$ distance to decision boundary (margin)**

- Small distance $\rightarrow$ high uncertainty
- Large distance $\rightarrow$ low uncertainty

Related work: [Liu et al., 2008]
Example 1

Data:

Sensor 1

Sensor 2
1st Stage Classifier: only utilizes Sensor 1
Example 1

2nd Stage Classifier: utilizes Sensors 1 and 2
Example 1

Myopic Reject Classifier

Stage 1 Decision

Stage 2 Decision

Reject

Classify
Example 1

Myopic Reject Classifier

- Requests sensor 2 where sensor 1 is ambiguous
- Current uncertainty seems to be a good criteria to reject

![Graph showing sensor data]

- Sensor 1
- Sensor 2
- \(-6 - 4 - 2 0 2 4 6\)
- \(-6 - 4 - 2 0 2 4 6\)
Example 1: Error vs Budget

sweep threshold to generate different operating points

Good performance: close to optimal, seems to work
Example 2

Sensor 2

Sensor 1
Example 2

1st Stage Classifier: only utilizes Sensor 1
Example 2

2nd Stage Classifier: utilizes Sensors 1 and 2
Example 2

Region 1

separable only with sensor 2
Example 2

Region 2

neither sensor helps

Sensor 2
Sensor 1
Example 2

Myopic Reject Decision

- Sensor 1 uncertainty is equally distributed between regions 1 and 2
- Uniformly rejects in both regions

(sensor 1 uncertainty distribution chart)

(reject to 2nd stage)

(sensor 2 and sensor 1 data points with bars for histogram)

19 / 50
Myopic Reject Decision

- Current uncertainty is equally distributed between regions 1 and 2
- Without future uncertainty cannot tell where sensor 2 is useful
Future Uncertainty is Important

Need to incorporate future uncertainty in the decision

$$\min \ [ \underbrace{\text{current uncertainty}}_{\text{classify}}, \underbrace{\alpha \times \text{cost} + \text{future uncertainty}}_{\text{reject to next stage}} ]$$
Known model: partially observable Markov decision process (POMDP)

- Posterior Model: $P(\text{state } | \text{sensor measurements})$
- Likelihood Model: $P(\text{sensor k } | \text{sensor j})$

Method 1: Learn models and solve POMDP

- hard to learn models,
- cannot solve POMDP in general case

Previous Work:

Method 2: Greedily maximize expected utility of a sensor

- One step look ahead approximation to POMDP, unclear how to choose utility
- Correlation across sensors: hard to learn likelihood (e.g. sensor output = image)

Previous Work: [Kanani and Melville, 2008, Koller and Gao, 2011]
Our Approach

- Avoid estimating probability models
- Directly learn decision at each stage from training data
- Empirical Risk Minimization (ERM):
  incorporates uncertainty of future in the current decision
Two Stage System

X → $f_1(x)$ → reject $f_2(x)$ → classify

cheap/fast sensor

expensive/slow sensor

classify

classify

$X$
Fix classifiers at each stage:

- $h_1(x)$ is standard classifier trained on sensor 1
- $h_2(x)$ is standard classifier trained on sensor 1 & 2
Decompose classification and rejection decisions:

- $g(x)$ is reject / not reject decision

\[
  f_1(x) = \begin{cases} 
  h_1(x), & g(x) = \text{not reject} \\
  \text{reject}, & \text{else} 
  \end{cases}
\]
Risks of Each Stage:

Current: \( R_{cu}(x) = \mathbb{1}[h_1 \text{ misclassifies } x] \)

Future: \( R_{fu}(x) = \mathbb{1}[h_2 \text{ misclassifies } x] + \alpha \times \text{sensor 2 cost} \)
Risks of Each Stage:

Current: \( R_{cu}(x) = \mathbb{I}[h_1 \text{ misclassifies } x] \)

Future: \( R_{fu}(x) = \mathbb{I}[h_2 \text{ misclassifies } x] + \alpha \times \text{sensor 2 cost} \)

Stage 1 reject decision \( g(x) \):

\[
g(x) = \begin{cases} 
\text{classify at 1,} & R_{cu}(x) < R_{fu}(x) \\
\text{reject to 2nd sensor,} & R_{cu}(x) \geq R_{fu}(x)
\end{cases}
\]
Risk Based Approach

Risks of Each Stage:

Current:  \( R_{cu}(x) = \mathbb{1}[h_1 \text{ misclassifies } x] \)

Future:  \( R_{fu}(x) = \mathbb{1}[h_2 \text{ misclassifies } x] + \alpha \times \text{sensor 2 cost} \)

Stage 1 reject decision  \( g(x) \):

\[
g(x) = \begin{cases} 
\text{classify at 1,} & R_{cu}(x) < R_{fu}(x) \\
\text{reject to 2nd sensor,} & R_{cu}(x) \geq R_{fu}(x) 
\end{cases}
\]

Difficulty:  \( R_{cu}, R_{fu} \) require ground truth  \( y \) and  \( R_{fu} \) requires sensor 2
Use training data with full measurement,

$$(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$$
Use training data with full measurement,

$$(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$$

And system risk for a point $x$ and decision $g(x)$

$$R(g, x, y) = \begin{cases} 
R_{cu}(x, y), & g(x) = \text{not reject} \\
R_{fu}(x, y), & g(x) = \text{reject} 
\end{cases}$$
Empirical Risk Minimization

Use training data with full measurement,

\( (x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N) \)

And system risk for a point \( x \) and decision \( g(x) \)

\[
R(g, x, y) = \begin{cases} 
R_{cu}(x, y), & g(x) = \text{not reject} \\
R_{fu}(x, y), & g(x) = \text{reject}
\end{cases}
\]

Minimize empirical risk,

\[
\min_g \mathbb{E}_{x,y} [R(g, x, y)] \approx \min_{g \in G} \frac{1}{N} \sum_{i=1}^{N} R(g, x_i, y_i)
\]
Example 2

reject to 2nd stage

Sensor 1
Sensor 2

Sensor 1
Example 2

Smaller error for the same cost

Ours

Myopic

Error=14.8%

Error=19%
Example 2

Incorporating future uncertainty in current decision improves performance

![Graph](image.png)

Legend:
- red circle: myopic
- blue circle: ours
- black dashed line: optimal
Learning to Reject

How to learn reject decision $g(x)$ (green region)?

**Reduce reject option to learning a binary decision**

Define a weighted supervised learning problem:

- risk difference induces pseudo labels on training data,

  $$\text{pseudo label of } x_i = \begin{cases} \text{reject}, & R_{cu}(x_i) > R_{fu}(x_i) \\ \text{not reject}, & R_{cu}(x_i) \leq R_{fu}(x_i) \end{cases}$$

- importance weights, risk difference = penalty for misclassifying

  $$\text{weight of } x_i = |R_{cu}(x_i) - R_{fu}(x_i)|$$
Learning to Reject

Risks induce pseudo-labels

Learn to reject decision

pseudo label of $x_i = \begin{cases} 
\text{reject}, & R_{cu}(x_i) > R_{fu}(x_i) \\
\text{not reject}, & R_{cu}(x_i) \leq R_{fu}(x_i) 
\end{cases}$

weight of $x_i = |R_{cu}(x_i) - R_{fu}(x_i)|$
Theorem:
Empirical risk minimization simplifies to weighted supervised learning:

$$\arg \min_{g \in G} \frac{1}{N} \sum_{i=1}^{N} R(g, x_i, y_i) = \arg \min_{g \in G} \sum_{i=1}^{N} \mathbb{1}[g(x_i) \neq \text{pseudo label of } x_i] \times \text{weight of } x_i$$
Reduction to supervised learning

\[
\min_{g \in \mathcal{G}} \sum_{i=1}^{N} \mathbb{1}[g(x_i) \neq \text{pseudo label of } x_i] \times \text{weight of } x_i
\]

Can be solved with existing supervised learning tools
- pick a surrogate loss \( \mathcal{L}[z] \geq \mathbb{1}_{[z>0]} \) (e.g. logistic)
- pick a classifier family \( \mathcal{G} \) (e.g. linear)

\[
\min_{g \in \mathcal{G}} \sum_{i=1}^{N} \mathcal{L}[g(x_i) \times \text{pseudo label of } x_i] \times \text{weight of } x_i
\]
**Example**

**Sensors Varying Resolutions**

classify handwritten digit images (mnist)

![Diagram](image)

- **x** handwritten digit image
- **y** ∈ {0, 1, ..., 9} label

<table>
<thead>
<tr>
<th>Stage</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor</td>
<td>4x4</td>
<td>8x8</td>
<td>16x16</td>
<td>32x32</td>
</tr>
<tr>
<td>Cost</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

**Base Learner:** logistic regression with linear classifiers
Example

<table>
<thead>
<tr>
<th>Sensor 1</th>
<th>Sensor 2</th>
<th>Sensor 3</th>
<th>Sensor 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digit 0</td>
<td>Digit 0</td>
<td>Digit 0</td>
<td>Digit 0</td>
</tr>
<tr>
<td>Digit 1</td>
<td>Digit 1</td>
<td>Digit 1</td>
<td>Digit 1</td>
</tr>
<tr>
<td>Digit 8</td>
<td>Digit 8</td>
<td>Digit 8</td>
<td>Digit 8</td>
</tr>
</tbody>
</table>

Sensor selection depends on example
Handwritten Digit Dataset

Same performance as centralized (best) with much lower budget
Measurement $\mathbf{x} = [x_1, \ldots, x_K]$ and true label $y$

Seek decisions at each stage $F = f_1, f_2, \ldots f_k$

$$f_k(\mathbf{x}) = \begin{cases} h_k(\mathbf{x}), & g_k(\mathbf{x}) = \text{not reject} \\ \text{reject}, & \text{else} \end{cases}$$

$h_k$ is a standard classifier trained on sensors $1, \ldots, k$
Stage-wise Decomposition

System Risk:  \[ R(F, x, y) = \text{Loss}(F(x), y) + \alpha \text{Cost}(F, x) \]
Stage-wise Decomposition

**System Risk:** \[ R(F, x, y) = \text{Loss}(F(x), y) + \alpha \text{Cost}(F, x) \]

**Stage-wise recursion:**

\[ R(F, x, y) = R_0(F, x, y) \]

\[ R_k(x, y, f_k) = \begin{cases} \alpha c_{k+1} + R_{k+1}(\cdot), & \text{reject to next stage} \\ 1, & \text{error & not reject} \\ 0, & \text{correct & not reject} \end{cases} \]
Stage-wise Decomposition

\[ R_k(x, y, f_k) = \begin{cases} \alpha c_{k+1} + R_{k+1}(\cdot), & \text{reject to next stage} \\ 1, & \text{error} \& \text{not reject} \\ 0, & \text{correct} \& \text{not reject} \end{cases} \]

**Key Observation:**
 Given the past: \( f_1, \ldots, f_{k-1} \), and the future: \( f_{k+1}, \ldots, f_K \)

- Find current decision, \( f_k \), from single stage risk \( R_k \)
- Equivalent to a two stage problem:

\[
\begin{align*}
R_{cu} &= \mathbb{I}[h_k \text{ misclassifies } x] \\
R_{fu} &= R_{k+1}(x, \ldots)
\end{align*}
\]
Algorithm

For every training example $x_i$:

- $R_{k+1}(x_i, \ldots)$: **cost-to-go**, empirical risk of future stages,
- $\text{state}_k(x_i)$: indicates if example is still active at stage $k$
Algorithm

For every training example $x_i$:
- $R_{k+1}(x_i, \ldots)$: **cost-to-go**, empirical risk of future stages,
- $\text{state}_k(x_i)$: indicates if example is still active at stage $k$

Algorithm: alternatively minimize one stage at a time

For every stage $k$:

1: Learn decision $f_k$:

$$
\min_{f \in F} \sum_{i=1}^{N} \text{state}_k(x_i) \ R_k[f, x_i, y_i, R_{k+1}(\cdot)]
$$

2: Update $\text{state}_j(x_i)$ for future stages $j > k$

3: Update **cost-to-go**($x_i$) for past stages $j < k$

Repeat until convergence
## Other Experiments

Achieve target error rate with fraction of max budget

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stages</th>
<th>Sensors</th>
<th>Target Error</th>
<th>Myopic</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>synthetic</td>
<td>2</td>
<td></td>
<td>.147</td>
<td>52%</td>
<td>28%</td>
</tr>
<tr>
<td>pima</td>
<td>3</td>
<td>weight, age, blood tests</td>
<td>.245</td>
<td>41%</td>
<td>15%</td>
</tr>
<tr>
<td>threat</td>
<td>3</td>
<td>ir, pmmw, ammw</td>
<td>.16</td>
<td>89%</td>
<td>71%</td>
</tr>
<tr>
<td>covertype</td>
<td>3</td>
<td>soils, wild. areas, elev, aspect</td>
<td>.285</td>
<td>79%</td>
<td>40%</td>
</tr>
<tr>
<td>letter</td>
<td>3</td>
<td>pixel counts, moments, edge feat’s</td>
<td>.25</td>
<td>81%</td>
<td>51%</td>
</tr>
<tr>
<td>mnist</td>
<td>4</td>
<td>res. levels</td>
<td>.085</td>
<td>90%</td>
<td>52%</td>
</tr>
<tr>
<td>landsat</td>
<td>4</td>
<td>hyperspectral bands</td>
<td>.17</td>
<td>56%</td>
<td>31%</td>
</tr>
<tr>
<td>mam</td>
<td>2</td>
<td>CAD feat’s, expert rating</td>
<td>.173</td>
<td>65%</td>
<td>25%</td>
</tr>
</tbody>
</table>
Generalization Results

How well does it perform on unseen data, $E_{x,y}[F(x) \neq y]$?
Generalization Results

How well does it perform on unseen data, $E_{x,y} \left[ F(x) \neq y \right]$?

Standard VC dimension test error bound:

For a classifier $F(x)$ in a family $\mathcal{F}$ with VC dimension $= h$, w.p. $1 - \delta$,

$$\text{Test Error} \leq \text{Train Error} + \sqrt{\frac{h \log(\frac{2N}{h} + 1) + \log \frac{4}{\delta}}{N}}$$

Smaller VC dimension $\rightarrow$ better generalization
Generalization Results

**System VCD does not explode!**

**Theorem:** VCD of a $K$ stage sequential decision:

$$\leq O(K \log K) \max_k \{VCD(\mathcal{F}_k)\}$$

$VCD(\mathcal{F}_k)$ is VC dimension of $k$th stage

Complexity grows only as $K \log K$ times the most complex stage
Conclusion

- Introduced sequential decision problem
- Myopic approach: relies on current uncertainty to make a decision
  - Considered synthetic examples
  - Current uncertainty is not always enough
- Our approach: incorporate future uncertainty in current decision
  - Examined a two stage system
  - Reduced to supervised learning
- Experiment
- Extend to Multiple Stages
- Generalization Results
Future Work

Optimization Improvement
- Currently, cyclical local optimization of each stage
- Need a convex formulation of system risk, achieve global optimum and better performance

More general architecture
- Option to skip a sensor if unnecessary
- Arbitrary sensing order, intractable even with full models, need approximations
Read our paper:
K. Trapeznikov, V. Saligrama,
*Supervised Sequential Classification Under Budget Constraints*,
AISTATS, 2013

we are organizing:

**Workshop on Learning with Test Time Budgets**
International Conference on Machine Learning, Atlanta, June 21-22

website: https://sites.google.com/site/budgetedlearning2013/

Thanks for Listening!
Cost-sensitive feature acquisition and classification.
In Pattern Recognition.

Prediction-time active feature-value acquisition for cost-effective customer targeting.
In NIPS.

Breaking boundaries: Active information acquisition across learning and diagnosis.
In NIPS.

Active value.

Tefe: A time-efficient approach to feature extraction.
In ICDM.

Pruning improves heuristic search for cost-sensitive learning.
In ICML.