Introduction to Artificial Intelligence and its Application in Software Quality

QUEST—May 2018, San Antonio, TX

Dr. Richard Bechtold

Abridge Technology: www.abridge-tech.com
Topics

1. Definitions and Basic Concepts
2. Categories of Machine Learning and Artificial Intelligence
3. AI/ML and its Impact on System Testing
5. Future Directions and Systems
Definitions & Basic Concepts

Any Sufficiently Advanced Technology is Indistinguishable From Magic

Arthur C. Clark
“Hazards of Prophecy”, 1973
Definitions & Basic Concepts

- Standard or Conventional Algorithms:
  - Deterministic
  - Given the same initial state, when given a set of identical inputs you get identical results

- Machine Learning:
  - Non-deterministic
  - Initial state evolves (or ‘learns’)
  - Same inputs at a later time may yield significantly different—if not polar opposite—outputs
Definitions & Basic Concepts

- Sub-specialty examples
  - Computational Statistics: calculating predictive likelihoods of future datapoints as a function of historical data, trends, etc.
  - Supervised Learning: Data is labeled, classified, etc.
  - Unsupervised learning: No classification of data
Definitions & Basic Concepts

- Many of the “most impressive” AI systems are based upon [“Why Self-Taught Artificial Intelligence Has Trouble With the Real World”; Sokol, Joshua; Quanta Magazine; February 21, 2018]
  - Perfect Goals
  - Perfect Clarity
  - Rule Consistency
- 3/26/2016 Microsoft released a Twitter chatbot, “Tay” to “engage people through playful and causal conversation”
  - It quickly learned to maximize engagement by spewing misogynistic, racist, and provocative insults
  - Microsoft pulled Tay offline within 24 hours
Definitions & Basic Concepts

- General inconsistency in industry regarding terms and meanings, however...
  - Machine Learning: Emphasis on the input side of the system
  - Artificial Intelligence: Emphasis on the output side of the system
- “Weak AI”: Focused on a specific, narrow task
- “Strong AI”: Capable of flexibility with situational recognition and problem solving
Definitions & Basic Concepts

- Artificial Intelligence and Machine Learning (AI/ML) is generally distinct from conventional computing (CC) even when CC involves:
  - Massive amounts of data are involved
  - Massive variety exists in data formats
  - Unacceptable processing times are being collapsed by orders of magnitudes
Definitions & Basic Concepts

- AI/ML approaches can be separated into several primary categories where each category of systems tends to be
  - Quite successful at the type of challenge it was designed for
  - Subject to particular weaknesses
- When evaluating quality, testing efforts need to evaluate the most likely characteristic weaknesses of that AI/ML category
Topics

1. Definitions and Basic Concepts
2. Categories of Machine Learning and Artificial Intelligence
3. AI/ML and its Impact on System Testing
5. Future Directions and Systems
Categories of ML/AI

Each category is characterized by how that type of system ‘learns’

A. Rule-Based
B. Exploration-Based
C. Training-Based
D. Temporal Context-Based
E. Hybrid-Based
A. Rule-Based

- The system ‘learns’ as a function of the knowledge humans add to it logically
- Example
  - Expert Systems
- Characterized by an environment that ‘fires rules’ as a function of satisfied pre-conditions
Overview: Expert Systems

- Distinguishing algorithmic characteristic is a huge set of conditional rules (e.g., “If X then Y”)
- Compound logic is often acceptable: If (A and (B or C)) then (D and E)
  - Less total rules
  - Rapidly increasing complexity
- Code doesn’t run from ‘top to bottom’; instead when pre-conditions are satisfied, the consequences occurs
- Subject to ‘race conditions’ (when multiple rules are simultaneously satisfied)
Overview: Expert Systems

- Characteristic weaknesses of rule-based systems include:
  - Circular logic (and infinite loops)
  - Conflicting assertion sequences (yielding inconsistent results)
  - “Live Locks” where, due to system state, part of the ruleset becomes permanent unavailable
B. Exploration-Based

- The system ‘learns’ by mimicking natural selection
- Example
  - Genetic Algorithms
- Do not need or use training data
- Generate potential solutions randomly, then cycle through ‘generations’ of increasing better solutions
Overview: Genetic Algorithms

- Distinguishing algorithmic characteristics
  - Objective functions
  - Cross-over
  - Selection
  - Mutation
  - Initial Population
  - Successive Generations
Example: Genetic Algorithm

- Pass-key of length 8: 1842 6716
- Fitness function: How closely does a candidate match the Pass-key?
  - 1 point for each correct number in a correct location
- Population per generation: 200
- Number of generations to explore: 100
- Crossover: Evenly (4 digits from each parent)
- Mutation: 1 digit per 1,000 candidates
Example 1: Genetic Algorithms

Pass-key: 1842 6716

<table>
<thead>
<tr>
<th>Parent-1</th>
<th>Parent-2</th>
<th>Child-1</th>
<th>Pt</th>
<th>Child-2</th>
<th>Pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>7346 8375</td>
<td>6732 6861</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example 1: Genetic Algorithms

Pass-key: 1842 6716

<table>
<thead>
<tr>
<th>Parent-1</th>
<th>Parent-2</th>
<th>Child-1</th>
<th>Pt</th>
<th>Child-2</th>
<th>Pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>7346 8375</td>
<td>6732 6861</td>
<td>7346 6861</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example 1: Genetic Algorithms

Pass-key: 1842 6716

<table>
<thead>
<tr>
<th>Parent-1</th>
<th>Parent-2</th>
<th>Child-1</th>
<th>Pt</th>
<th>Child-2</th>
<th>Pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>7346 8375</td>
<td>6732 6861</td>
<td>7346 6861</td>
<td></td>
<td>6732 8375</td>
<td></td>
</tr>
</tbody>
</table>
Example 1: Genetic Algorithms

Pass-key: 1842 6716

<table>
<thead>
<tr>
<th>Parent-1</th>
<th>Parent-2</th>
<th>Child-1</th>
<th>Pt</th>
<th>Child-2</th>
<th>Pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>7346 8375</td>
<td>6732 6861</td>
<td>7346 6861</td>
<td>2</td>
<td>6732 8375</td>
<td>1</td>
</tr>
</tbody>
</table>
Example 1: Genetic Algorithms

Pass-key: 1842 6716

<table>
<thead>
<tr>
<th>Parent-1</th>
<th>Parent-2</th>
<th>Child-1</th>
<th>Pt</th>
<th>Child-2</th>
<th>Pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>7346 8375</td>
<td>6732 6861</td>
<td>7346 6861</td>
<td>2</td>
<td>6732 8375</td>
<td>1</td>
</tr>
<tr>
<td>7232 4641</td>
<td>7584 2540</td>
<td>7232 2540</td>
<td>1</td>
<td>7584 4641</td>
<td>0</td>
</tr>
</tbody>
</table>
Example 1: Genetic Algorithms

Pass-key: 1842 6716

<table>
<thead>
<tr>
<th>Parent-1</th>
<th>Parent-2</th>
<th>Child-1</th>
<th>Pt</th>
<th>Child-2</th>
<th>Pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>7346 8375</td>
<td>6732 6861</td>
<td>7346 6861</td>
<td>2</td>
<td>6732 8375</td>
<td>1</td>
</tr>
<tr>
<td>7232 4641</td>
<td>7584 2540</td>
<td>7232 2540</td>
<td>1</td>
<td>7584 4641</td>
<td>0</td>
</tr>
<tr>
<td>8734 6738</td>
<td>6384 0982</td>
<td>8734 0982</td>
<td>0</td>
<td>6384 6738</td>
<td>2</td>
</tr>
</tbody>
</table>
Example 1: Genetic Algorithms

Pass-key: 1842 6716

<table>
<thead>
<tr>
<th>Parent-1</th>
<th>Parent-2</th>
<th>Child-1</th>
<th>Pt</th>
<th>Child-2</th>
<th>Pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>7346 8375</td>
<td>6732 6861</td>
<td>7346 6861</td>
<td>2</td>
<td>6732 8375</td>
<td>1</td>
</tr>
<tr>
<td>7232 4641</td>
<td>7584 2540</td>
<td>7232 2540</td>
<td>1</td>
<td>7584 4641</td>
<td>0</td>
</tr>
<tr>
<td>8734 6738</td>
<td>6384 0982</td>
<td>8734 0982</td>
<td>0</td>
<td>6384 6738</td>
<td>2</td>
</tr>
</tbody>
</table>
Example 1: Genetic Algorithms

Pass-key: 1842 6716

<table>
<thead>
<tr>
<th>Parent-1</th>
<th>Parent-2</th>
<th>Child-1</th>
<th>Pt</th>
<th>Child-2</th>
<th>Pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>7346 8375</td>
<td>6732 6861</td>
<td>7346 6861</td>
<td>2</td>
<td>6732 8375</td>
<td>1</td>
</tr>
<tr>
<td>7232 4641</td>
<td>7584 2540</td>
<td>7232 2540</td>
<td>1</td>
<td>7584 4641</td>
<td>0</td>
</tr>
<tr>
<td>8734 6738</td>
<td>6384 0982</td>
<td>8734 0982</td>
<td>0</td>
<td>6384 6738</td>
<td>2</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>7346 6861</td>
<td>6384 6738</td>
<td>7346 6738</td>
<td>3</td>
<td>6384 6861</td>
<td>1</td>
</tr>
<tr>
<td>Child-1, Row 1</td>
<td>Child-2, Row 3</td>
<td>7346 6738</td>
<td>3</td>
<td>6384 6861</td>
<td>1</td>
</tr>
</tbody>
</table>
Example 2: Genetic Algorithms

- 'Genetic' format: Each bit location represents yes/no for a trading technique
  - Bit 1: 30 day average return
  - Bit 2: 90 day average increases above 30 day
  - Bit 3: Greatest increase last 5 days
  - Bit 4: Volatility < 10% over last 3 months
  - Bit 5: Stop loss at x%
  - Etc...

- Run 100,000 candidates through 100,000 generations against 10 years of stock market index fund daily results
Summary: Genetic Algorithms

- Generally use a ‘hill climber’ algorithm
  - “If the new answer is better than the prior one, then progress is in the right direction”

- Issue: Highly subject to finding ‘local optima’
C. Training-Based

- The system ‘learns’ as a function of (typically, or ideally) extensive training data matching input concepts (or data) to correct output actions (or data)
- Fairly popular approach to implementing voice-recognition
- Example
  - Neural Nets
Overview: Neural Nets

- Distinguishing algorithmic characteristics
  - Training data
  - Input layer
  - Output layer
  - Any number of intermediate layers (including none)
- Ability to ‘learn’ is highly correlated to the quality of the training data
Example: Neural Net
Example: Neural Net
Example: Neural Net
Example: Neural Net
Example: Neural Net
Example: Neural Net

![Neural Net Diagram]
Example: Neural Net
Example: Neural Net

Abridge Technology
www.abridge-tech.com

rbechttold@abridge-tech.com

Slide #: 37
Example: Neural Net
Example: Neural Net
Example: Neural Net
Overview: Neural Nets

- Incredible variety exists in NN structures
  - Number of layers
  - Number of nodes per layer
  - Variance (or not) of nodes per layer
    - Self-adjusting?
    - Specified minimums and/or maximums?
  - “Back Propagation” where wrong answers result in weakening or removal of connections
Summary: Neural Nets

- These are some of the “spookiest” algorithms and data structures
- After training, they will yield ‘answers’ but have to ability to explain why
- We can look at the answers and (in all but the most trivial cases) we can’t explain why either
- So, what are their weaknesses, and how can we investigate or test for them? (Topic 4)
D. Temporal Context-Based

- The system ‘learns’ by
  - Monitoring behaviors or events
  - Predicting future behaviors or events
  - Adjusting subsequent predications as a function of the success (or lack thereof) of prior predictions

- Example
  - Context-Sensitive predictive ‘look ahead’ for
    - Driving directions
    - Completing search queries
  - Digital Assistants (Alexa, Cortana, Siri)
Overview: Context-Sensitive

- Distinguishing algorithmic characteristics
  - Potentially massive supporting datasets
  - Learns by detecting patterns, *in context*
  - Predictive algorithms yield different suggestions as a function of recent activity
    - After recently searching math functions, you type “bar” and the computer offers: “bar graph”
    - After recently searching hotels, you type “bar” and the computer offers “bars and restaurants”
    - Destination suggestions from your GPS as a function of current location, time of day, and the types of establishments at recent stops (e.g., residential, office, business, etc.)
E. Hybrid-Based

- The system ‘learns’ in spite of repeated exposure to false or deliberately misleading information
- Involves any combination of AI/ML strategies and techniques, e.g.,
  - Genetic algorithms used to find ‘best’ answers
  - Results of the genetic systems used as training data for neural nets
  - Neural nets working in competition, and learning from each other (including the other net’s weaknesses)
  - Expert rules for handling ‘outlier’ or extreme situations
E. Hybrid-Based

- Hybrid systems are likely necessary to reduce category-specific known weaknesses.
- Non-hybrids usually result in “Weak AI” which can exceed human performance, but only in a very specific type of task (Chess, Go, Backgammon, Poker, etc.).
- General trend is toward using increasing combinations of techniques, data structures, and algorithms.
Topics

1. Definitions and Basic Concepts
2. Categories of Machine Learning and Artificial Intelligence
3. *AI/ML and its Impact on System Testing*
5. Future Directions and Systems
3. AI/ML and its Impact on System Testing

- Conventional systems will continue to see significant advances in the use of conventional techniques for test automation.
- Some conventional systems may be advertised as ‘AI’ due to:
  - Massively increasing datasets
  - Significant increases in the conventional use of historical data (e.g., statistical analysis and predictive modeling)
3. AI/ML and its Impact on System Testing

Key distinctions

- Conventional Systems
  - Deterministic
  - Same set of inputs yields the same set of outputs

- Artificial Intelligence / Machine Learning
  - Non-deterministic
  - Output may/will change as a function of system ‘learning’
  - If the same inputs (at a later time) yield the same outputs, *something might be wrong*
3. AI/ML and its Impact on System Testing

- Advances in machine response time with conventional techniques will have far greater impact on testing than advances in artificial intelligence.
- You don’t need AI to determine:
  - Where the greatest number of defects tend to occur in your system.
  - Which defects have the greatest adverse impact on end-user experience.
  - Which types of defects the system (or subsystem) is most subject to.
  - Which types of tests need to be run on which sub-systems.
3. AI/ML and its Impact on System Testing

- Areas where AI/ML will be making contributions to testing
  - Using an interactive *verbal* set of instructions or interactive dialogue versus test scripts
  - Enhanced "user experience" labs where AI-based *visual* systems are watching you
    - As you run through a sequence of test activities
    - As a designated representative or actual end-user
- Better detection of 'soft' defects where there are no errors, per se, but there is a very poor or frustrating end-user (or tester) experience
Topics

1. Definitions and Basic Concepts
2. Categories of Machine Learning and Artificial Intelligence
3. AI/ML and its Impact on System Testing
4. *Cognitive Testing of AI/ML Systems*
5. Future Directions and Systems

- A key concept in AI/ML systems is to move the system or algorithmic design closer to the ‘human design’
- A rather obvious flaw to this approach is that humans are subject to more than a few cognitive weaknesses
- Hence, a major *(THE major?)* focus for the testing of AI/ML systems are investigations into cognitive system weaknesses

Cognitive Flaw #01: Excessive impact of “first impressions”

- Investigate this by starting with an initial pattern of inputs in one direction and then sharply changing the style of inputs
  - Active voice versus passive voice
  - Rapid versus slow speech pattern

- Cognitive Flaw #02: Excessive sensitivity to being wrong
  - Investigate this by
    - Giving the system heavily negative ‘ratings’ or responses
    - Quickly and repeatedly abandoning and changing a given line of enquiry

- Cognitive Flaw #03: Excessive confidence in sparse information
  - Investigate this by using progressively less training data
  - How appropriate are the responses?
  - How apparently “confident” is the machine in those responses?

- Cognitive Flaw #04: System tendency to tell or show you whatever it thinks you ‘want’ to hear or see
  - Excessively tailors its response to you as opposed to providing you a more accurate response
  - Investigate this by providing rather extreme opposing feedback in closely related areas

- Cognitive Flaw #05: Tendency to jump to conclusions
  - System may excessively rate the correctness of results due to inadequate breadth/depth of searches
  - Investigate by pursuing very resource-intensive searches or tasks, especially when the system is already under heavy load

- Cognitive Flaw #06: Excessive pursuit of perfection
  - Test by looking for increasingly precise answers
  - Test by making increasingly specific

Cognitive Flaw 07: Inability to appropriately respond to “black swan” events [Roman poet Juvenal; popularized by Nassim Nicholas Taleb, 2001 “Fooled by Randomness”]

- Disproportionate impact of some extreme low-probability events
- Non-computability of probabilistic consequences of highly rare/unusual events
- Investigate by examining situations where, though rare, the consequences may be catastrophic to a company or organization (but acceptable to a machine)
Topics

1. Definitions and Basic Concepts
2. Categories of Machine Learning and Artificial Intelligence
3. AI/ML and its Impact on System Testing
5. Future Directions and Systems
Future Directions and Systems

- Expect shifting and conflicting definitions of artificial intelligence and machine learning
  - One person’s “magic” is another person’s set of algorithms, data-structures, and protocols
- Is it AI if it can give us answers that
  - (a) work
  - (b) but we can’t explain?
Future Directions and Systems

- Consider: City-wide traffic management based entirely on system optimization rules.
- Technically achievable with conventional hardware, logic, and data-structures.
- However, at any given moment it would be impossible to explain the current status of traffic lights.
Future Directions and Systems

- Significant issues remain to be addressed
  - AI/ML systems are the ultimate manifestation of discrimination
    - They can/will detect essentially anything
    - They can/will remember essentially everything
  - When and where do you force a system to either not detect, not remember, or not consider?
Future Directions and Systems

During the last year there has been increasing work in the area of system learning other than pattern recognition and correlation discovery.

Researchers are encoding:

- “Curiosity” so systems are biased toward exploring areas where they have the least data.
- “Surprise” so systems explore variations on events that did not occur as predicted/expected.
- “Failure-driven” learning: If you hit a baseball and it went foul instead of being a home run, then it would have been perfect if you were trying to hit the foul. [TechExplore, 3/2/2018, Nancy Owano]
Summary

1. Definitions and Basic Concepts
2. Examples of Different Categories of Machine Learning and Artificial Intelligence
3. AI/ML and its Impact on System Testing
5. Future Directions and Systems
Concluding Remarks

“We Need to Approach AI Risks Like We Do Natural Disasters”; Gangu, Prashanth; Harvard Business Review; February 7, 2018

- “Risks posed by intelligent devices will soon surpass the magnitude of those associated with natural disasters.”

- Defect detection and management in AI/ML systems will become increasingly similar to crisis or disaster management
Concluding Remarks

Recommendations

- Consider shifting the focus of your expertise toward quality investigations and evaluations of AI/ML systems

- *Develop at least a foundation understanding in cognitive and behavioral sciences*
  - Especially investigate “cognitive bias”

- “Quality Investigator” may be a more appropriate title when evaluating the quality of artificial intelligence and machine learning systems
Contact Information

Dr. Richard Bechtold
President; Senior Consultant
Abridge Technology; Ashburn, VA
(USA) 703.729.6085

rbechtold@abridge-tech.com
www.abridge-tech.com
Biographical Highlights

Dr. Bechtold is the President of Abridge Technology, a Virginia-based company he founded in 1996. Abridge Technology is a CMMI Institute Partner and is authorized to provide the Institute’s official training and SCAMPI appraisal services. Dr. Bechtold, an Institute-Certified Lead Appraiser, and Certified Instructor, provides consulting, training, and support services in the areas of project management, process improvement, process engineering, measurement, and risk management. Dr. Bechtold has assisted government and industry with implementing the Software CMM since 1992, Acquisition CMM since 1996, and the CMMI since 2000. Dr. Bechtold’s expertise spans organizations of all types and sizes, from multi-billion dollar companies and agencies to organizations with less than 10 personnel.