EMERGENCE OF INTELLIGENT MACHINES: CHALLENGES AND OPPORTUNITIES

Non-Human Artificial Intelligence

Bart Selman
Cornell University
After a distinguished history of “overpromising,” AI is finally making real progress.

Positive trajectory started in the late 90s:

1997  IBM’s Deep Blue defeats Kasparov
2005  Stanley --- self-driving car (controlled environment)
2011  IBM’s Watson wins Jeopardy! (question answering)
2012  Speech recognition via “deep learning” (Geoff Hinton)
2014  Computer vision is starting to work (deep learning)
2015  Microsoft demos real-time translation (speech to speech)
2016  Google’s AlphaGo defeats Lee Sedol
Reasons for Change

--- series of events

--- main one: *machine perception* is starting to work (finally!)

**systems are starting to “hear” and “see”**

after “only” 50+ yrs or research…

--- dramatic change: lots of AI techniques (reasoning, search, reinforcement learning, planning, decision theoretic methods) were developed assuming perceptual inputs were “somehow” provided to the system. But, e.g., robots could not really see or hear anything…

(e.g. 2005 Stanley car drove around “blind”, Thrun)

Now, we can use output from a perceptual system and leverage a broad range of existing AI techniques.

Our systems are finally becoming “grounded in (our) world.”

Already: super-human face recognition (Facebook)

super-human traffic sign recognition (Nvidia)
Computer vision / Image Processing ca. 2005

(a) Left image: 384x288, 15 labels
(original image)

(b) Ground truth
(human labeled)

Processed image ca. 2005
DEEP LEARNING FOR SELF-DRIVING CARS

(Nvidia 2016; Mobileye)

Statistical model (neural net) trained on >1M images;
Models with > 500K parameters
Requires GPU power
Real-time tracking of environment (360 degrees/ 50+m) and decision making.
Factors in accelerated progress, cont.

--- deep learning / deep neural nets

success is evidence in support of the “hardware hypothesis”
(Moravec) (*)

*core neural net ideas from mid 1980s*

needed: several orders of magnitude increase
in computational power and data

(aside: this advance was not anticipated/predicted *at all*;
many AI/ML researchers had moved away from neural nets…)

+ BIG DATA!
2035/40 cellphone = human brain

Current:
Nvidia: tesla personal supercomputer
1000 cores
4 teraflop
Progress, cont.

--- crowd-sourced human data --- machines need to understand our conceptualization of the world. E.g. vision for self driving cars trained on 100,000+ miles of labeled road data.

--- engineering teams (e.g. IBM’s Watson)
  strong commercial interests
  at a scale never seen before in our field

--- Investments in AI systems are being scaled-up by an order of magnitude (to billions).
Google, Facebook, Baidu, IBM, Microsoft, Tesla etc. ($1B+)
+ military ($19B proposed)

An AI arms race
Next Phase

Further integration of existing techniques --- perception, (deep) learning, inference, planning --- will be a game changer for AI systems.

AlphaGo:
Deep Learning
+ Reasoning
(Google/Deepmind 2016)
What We Can’t Do Yet

--- Need deeper semantics of natural language
--- Commonsense reasoning

Example:

“The large ball crashed right through the table because it was made of styrofoam.”

What was made of Styrofoam? The large ball or the table?

(Oren Etzioni, Allen AI Institute)

Also, is commonsense needed to deal with unforeseen circumstances?
(i.e., not in the training data)
Cause under investigation

Two main possibilities:

(1) Failure vision system: did not see truck because of bright sun.

(2) (more intriguing) “Do-not-break-too-often” system, was too daring.

(Why needed?)

Need for AI Safety research: Combines planning, decision theory, and ethics.
AI focus: **Human** intelligence because that’s the intelligence we know…

**Cognition:** Perception, learning, reasoning, planning, and knowledge.

Deep learning is changing what we thought we could do, at least in perception and learning (with enough data).
Artificial Intelligence

Separate development --- “non-human”: Reasoning and planning. Similar qualitative and quantitative advances but “under the radar.”

Part of the world of software verification, program synthesis, and automating science and mathematical discovery.

Developments proceed without attempts to mimic human intelligence or even human intelligence capabilities.

Truly machine-focused (digital): e.g., “verify this software procedure” or “synthesize procedure” --- can use billions of inference steps --- or “synthesize an optimal plan with 1,000 steps.” (Near-optimal: 10,000+ steps.)
Consider a sequence of 1s and -1s, e.g.:

\[
\begin{array}{cccccccc}
-1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & \ldots \\
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & \ldots \\
2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & \ldots
\end{array}
\]

and look at the sum of sequences and subsequences:

\[
\begin{align*}
-1 + 1 &= 0 \\
-1 + 1 + 1 &= 1 \\
-1 + 1 + 1 + -1 &= 0 \\
-1 + 1 + 1 + -1 + 1 &= 1 \\
-1 + 1 + 1 + -1 + 1 + 1 &= 2 \\
\end{align*}
\]

and “skip by 1”

\[
\begin{align*}
1 + -1 &= 0 \\
1 + -1 + 1 &= 1 \\
1 + -1 + 1 + 1 &= 2 \\
\end{align*}
\]

and “skip by 2”

\[
\begin{align*}
1 + 1 &= 2 \\
1 + 1 + -1 &= 1 \\
\end{align*}
\]

e etc.

We now know (2015): there exists a sequence of 1160 +1s and -1s such that sums of all subsequences never < -2 or > +2.
1160 elements all sub-sums stay between -2 and +2

40 x 29 pattern
So, we now know (2015): there exists a sequence of $1160$ +1s and -1s such that sums of all subsequences never $< -2$ or $> +2$.

Result was obtained with a general reasoning program (a Boolean Satisfiability or SAT solver). Surprisingly, the approach far outperformed specialized search methods written for the problem, including ones based on other known types of sequences. (A PolyMath project started in January 2010.)
Aside: A Taste of Problem Size

Consider a real world Boolean Satisfiability (SAT) problem, from software & hardware verification.

The instance bmc-ibm-6.cnf, IBM LSU 1997:

```
1 7 0
1 6 0
1 5 0
1 -4 0
1 3 0
1 2 0
1 -8 0
9 15 0
9 14 0
9 13 0
9 -12 0
9 11 0
9 10 0
9 -16 0
-17 23 0
-17 22 0
```

“1” for variable x_1, “2” for x_2, etc.

x_1, x_2, x_3, ... our Boolean variables (set to True or False)

((not x_1) or x_7)
((not x_1) or x_6)

etc.

Question: Can we satisfy all statements?

Set x_1 to False ??

SAT problem lies at the core of computer science
Prototypical NP-complete problem (from P vs. NP)
I.e., \((x_{177} \text{ or } x_{169} \text{ or } x_{161} \text{ or } x_{153} \ldots \text{ or } x_{33} \text{ or } x_{25} \text{ or } x_{17} \text{ or } x_{9} \text{ or } x_{1} \text{ or } \neg x_{185})\)

clauses / constraints are getting more interesting...

Note \(x_1\) ...
4000 pages later:

10236 -10050 0
10236 -10051 0
10236 -10235 0
10008 10009 10010 10011 10012 10013 10014
10015 10016 10017 10018 10019 10020 10021
10022 10023 10024 10025 10026 10027 10028
10029 10030 10031 10032 10033 10034 10035
10036 10037 10086 10087 10088 10089 10090
10091 10092 10093 10094 10095 10096 10097
10098 10099 10100 10101 10102 10103 10104
10105 10106 10107 10108 -55 -54 53 -52 -51 50
10047 10048 10049 10050 10051 10235 -10236 0
10237 -10008 0
10237 -10009 0
10237 -10010 0

...
Finally, 15,000 pages later:

Search space of truth assignments:

\[ 2^{50000} \approx 3.160699437 \cdot 10^{15051} \]

Current reasoning engines can solve this instance in a few seconds! (no satisfying assignment exists + proof)
Consider a sequence of 1s and -1s, e.g.:

\[-1, 1, 1, -1, 1, 1, -1, 1, -1 \ldots\]

\[
\begin{array}{cccccccc}
  1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & \ldots \\
  2 & 4 & 6 & 8 & \ldots \\
  3 & 6 & 9 & \ldots \\
\end{array}
\]

and look at the sum of sequences and subsequences:

\[-1 + 1 = 0\]
\[-1 + 1 + 1 = 1\]
\[-1 + 1 + 1 + -1 = 0\]
\[-1 + 1 + 1 + -1 + 1 = 1\]
\[-1 + 1 + 1 + -1 + 1 + 1 = 2\]
\[-1 + 1 + 1 + -1 + 1 + 1 + -1 = 1\]
\[-1 + 1 + 1 + -1 + 1 + 1 + -1 + 1 = 2\]
\[-1 + 1 + 1 + -1 + 1 + 1 + -1 + 1 + -1 = 1\]

and “skip by 1”

\[1 + -1 = 0\]
\[1 + -1 + 1 = 1\]
\[1 + -1 + 1 + 1 = 2\]

and “skip by 2”

\[1 + 1 = 2\]
\[1 + 1 + -1 = 1\]

etc.

etc.
Back to sequences of +1/-1s

Encoding has variables for the sequence $X_1, X_2, \ldots, X_N$

(we interpret True for $+1$ and False for $-1$)

but also e.g.

Proposition: “sum_of_first_2_terms_of_skip_by_2_subseq_=2”

(for any given setting of $X_1 \ldots X_N$ this is either True or False)

and statements of the form:

IF ((sum_of_first_2_terms_of_skip_by_2_subseq_=2 == True)
    AND (X_9 == False))

THEN

(sum_of_first_3_terms_of_skip_by_2_subseq_=1 == True)

Encoding: 37,418 variables and 161,460 clauses / constraints.

Sequence found in about 1 hour (MacBook Air).

Perhaps SAT solver was “lucky” in finding the sequence?
Another example logical constraint:

IF \((\text{sum\_of\_first\_2\_terms\_of\_skip\_by\_2\_subseq\_}=\_2 \quad == \quad \text{True})\) THEN

\((\text{sum\_of\_first\_3\_terms\_of\_skip\_by\_2\_subseq\_}=\_2 \quad == \quad \text{False})\)

Why??

Also:

IF \((\text{sum\_of\_first\_2\_terms\_of\_skip\_by\_2\_subseq\_}=\_2 \quad == \quad \text{True})\) THEN

\((\text{X\_9} \quad == \quad \text{False})\)

Why??

We’ll have thousands of these kinds of small logical statements to capture the problem.

Automatically generated in a fraction of a second.
For 1160 +1/-1’s problem:

Encoding: 37,418 variables and 161,460 clauses / constraints.
Sequence found in about 1 hour (MacBook Air).

Perhaps SAT solver was “lucky” in finding the sequence?
1160 elements
all sub-sums stay between -2 and +2

40 x 29 pattern
But, remarkably, each sequence of 1161 or longer leads to a +3 (or -3) somewhere. (Erdos discrepancy conjecture)

Encoding: 37,462 variables and 161,644 clauses / constraints.

Proof of non-existence of discrepancy 2 sequence found in about 10 hour (MacBook Air).

Proof: 13 gigabytes and independently verified (50 line proof checking program). Proof is around a billion small inference steps.

Machine understands and can verify result easily (milliseconds); Humans: probably never. Still, we can be certain of the result because of the verifier.
Observations

1) Result different from earlier “computer math” results, such as the proof of the 4 color theorem, because here we don’t need to trust the theorem prover. Final proof (“certificate”) can be checked easily by anyone.

2) It’s *not* a brute force search. Earlier SAT solvers cannot find the proof. Specialized programs cannot find the proof. Brute force proof is of order $2^{1161} = 3.13 \times 10^{349}$. Current solver finds complete proof with “only” around $1.2 \times 10^{10}$ steps. Clever learning and reasoning enables a factor $10^{339}$ reduction in proof size.

3) In part inspired by discrepancy 2 result, Terence Tao proved just a few months ago the general Erdos conjecture (for any discrepancy). Deep and subtle math.

4) But, does not fully supersedes the 1161 result for the discrepancy 2. Future math may build further on these types of computational results. (I.e. true, verifiable facts but not human accessible.)
Other examples

**AlphaGo:**

- Core engine
- Monte Carlo Tree Search (UCT, 2006)
- Final boost: deep learning and reinforcement learning.
  
  Search part *and insights* will likely remain beyond human understanding.

**Planning:** We can synthesize optimal plan sequences of 1000+ steps.

Changes the notion of a “program”

A planning-enabled robot will synthesize its plans on-the-fly given its current abilities. Quite different from current pre-programmed industrial robots.
Computational Complexity Hierarchy

EXP-complete: games like Go, …

PSPACE-complete: QBF, planning, chess (bounded), …

#P-complete/hard: #SAT, sampling, probabilistic inference, …

NP-complete: SAT, propositional reasoning, scheduling, graph coloring, puzzles, …

P-complete: circuit-value, …

In P: sorting, shortest path

What are the consequences for human understanding of machine intelligence?