Intelligent Software Engineering: Synergy between AI and Software Engineering

Tao Xie

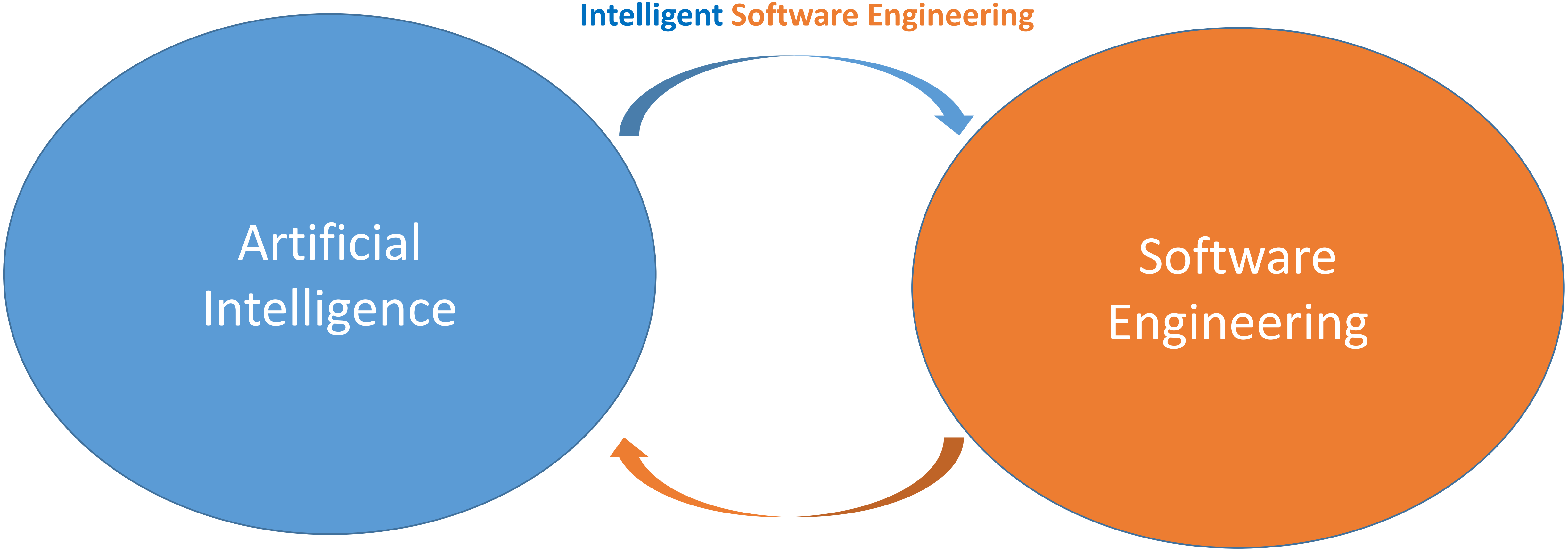
University of Illinois at Urbana-Champaign

taoxie@illinois.edu

http://taoxie.cs.illinois.edu/

SETTA’18 Keynote

Artificial Intelligence ←→ Software Engineering



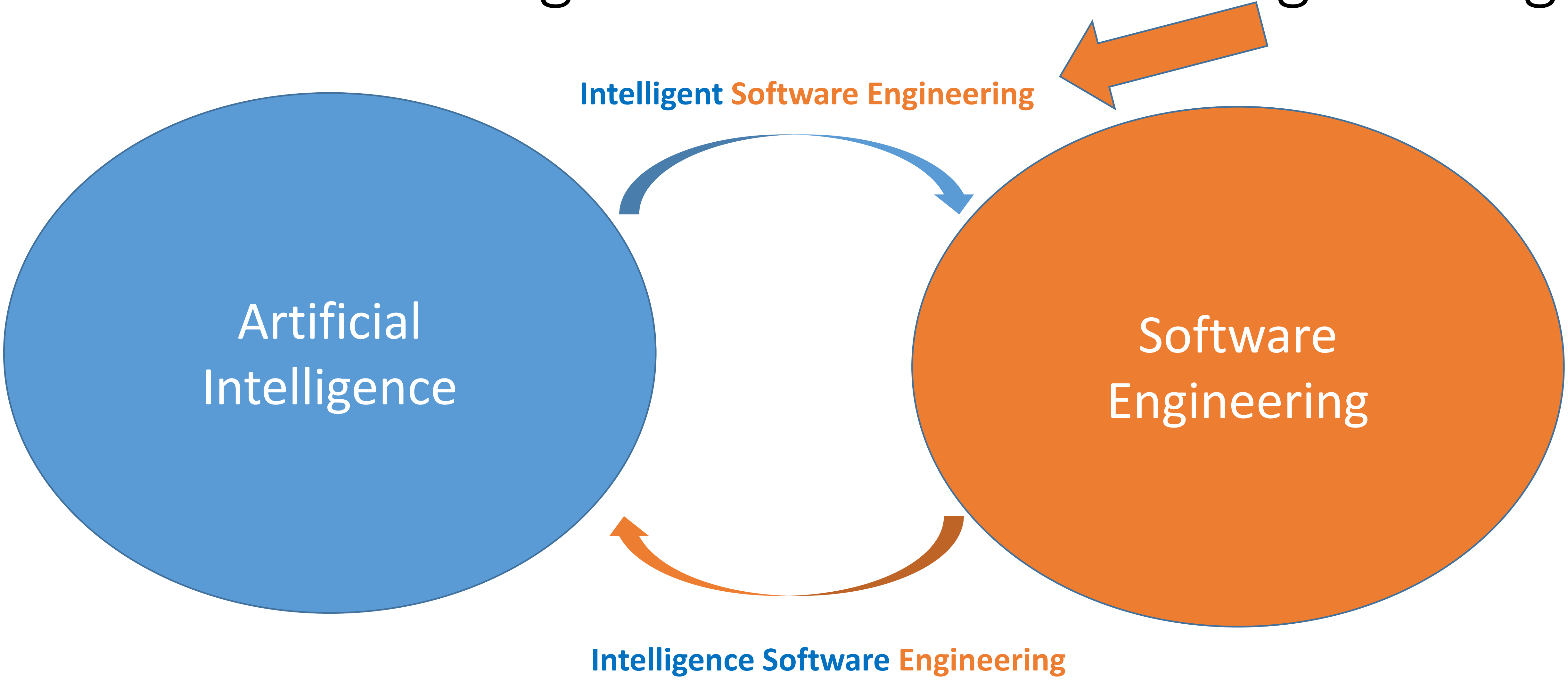
Intelligent Software Engineering

Artificial Intelligence

Software Engineering

Intelligence Software Engineering

Artificial Intelligence ←→ Software Engineering



Intelligent Software Engineering

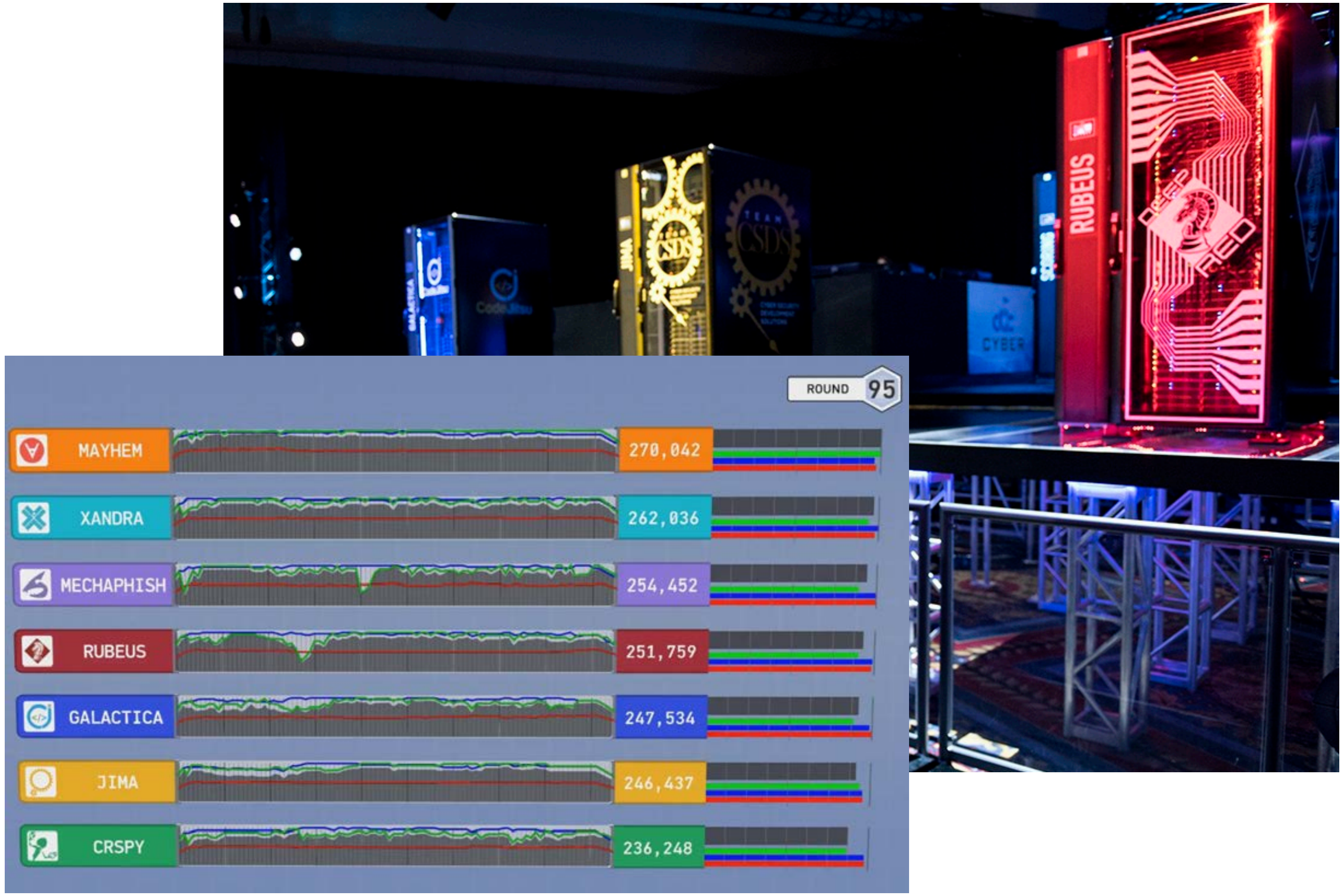
Artificial Intelligence

Software Engineering

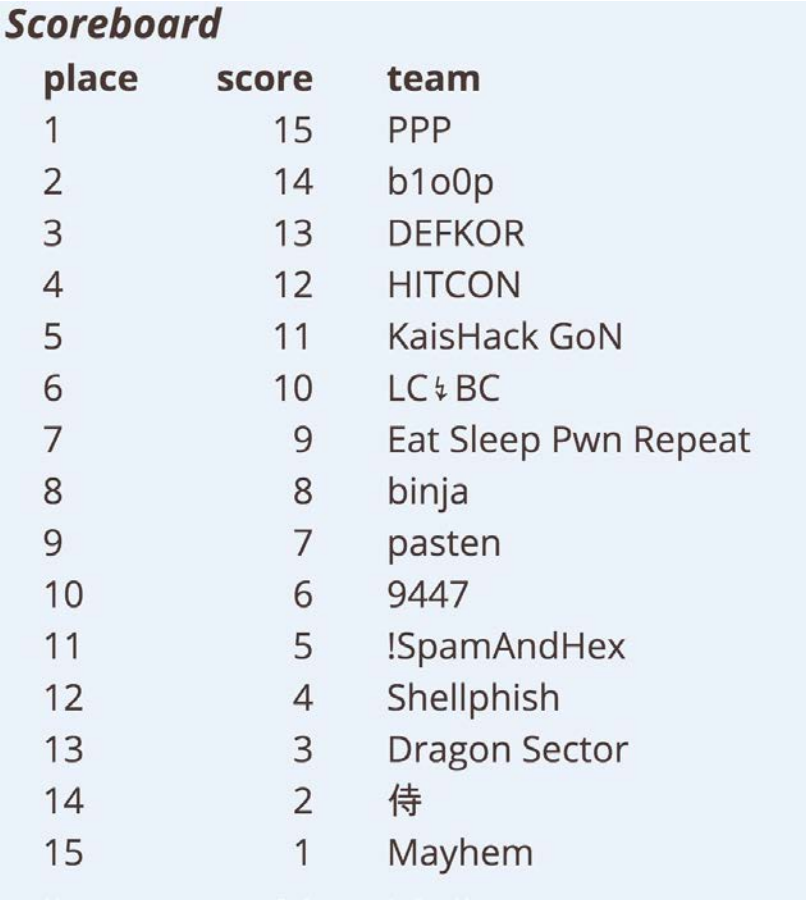
Intelligence Software Engineering

Carnegie Mellon’s Mayhem AI takes home $2 million from DARPA’s Cyber Grand Challenge

Posted Aug 5, 2016 by Devin Coldewey, Contributor



MAYHEM	270,042
XANDRA	262,036
MECHAPHISH	254,452
RUBEUS	251,759
GALACTICA	247,534
JIMA	246,437
CRSPY	236,248



Scoreboard

place	score	team
1	15	PPP
2	14	b1o0p
3	13	DEFKOR
4	12	HITCON
5	11	KaisHack GoN
6	10	LC↯BC
7	9	Eat Sleep Pwn Repeat
8	8	binja
9	7	pasten
10	6	9447
11	5	!SpamAndHex
12	4	Shellphish
13	3	Dragon Sector
14	2	侍
15	1	Mayhem

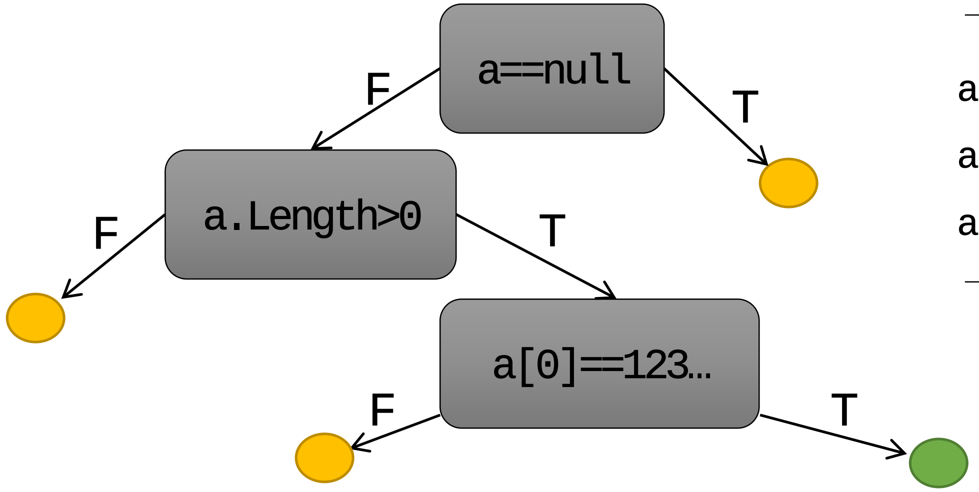
https://techcrunch.com/2016/08/05/carnegie-mellons-mayhem-ai-takes-home-2-million-from-darpas-cyber-grand-challenge/

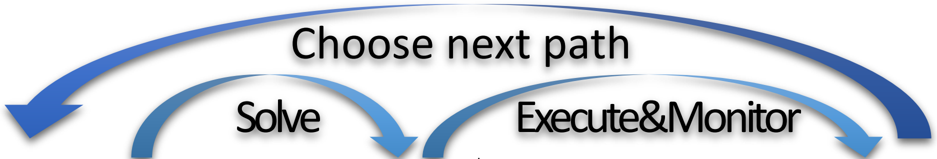
Dynamic Symbolic Execution

[DART: Godefroid et al. PLDI'05]

Code to generate inputs for:

```
void CoverMe(int[] a)
{
  if (a == null) return;
  if (a.Length > 0)
    if (a[0] == 1234567890)
      throw new Exception("bug");
}
```




Choose next path

Solve

Execute&Monitor

Constraints to solve	Data	Observed constraints
	`null`	`a==null`
`a!=null`	`{}`	`a!=null && !(a.Length>0)`
`a!=null && a.Length>0`		`a[0]!=1234567890`
`a!=null && a.Length>0 && a[0]==1234567890`	`{123..}`	`a!=null && a.Length>0 && a[0]==1234567890`

Negated condition

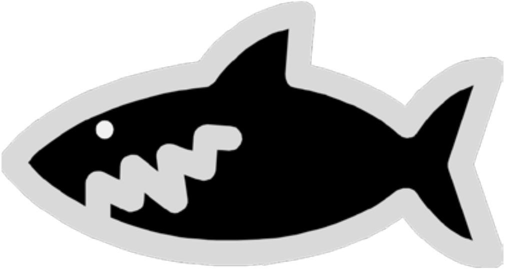
Done: There is no path left.

Z3

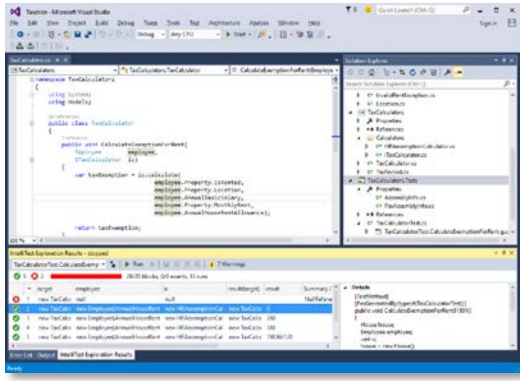
Constraint solver has decision procedures for

- Arrays
- Linear integer arithmetic
- Bitvector arithmetic
- Floating-point arithmetic
- …

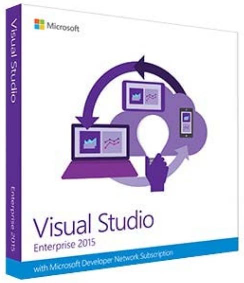
Past: Automated Software Testing



- 10 years of collaboration with Microsoft Research on Pex [ASE’14 Ex]
  - .NET Test Generation Tool based on Dynamic Symbolic Execution



- Tackle challenges of
  - Path explosion via fitness function [DSN’09]
  - Method sequence explosion via program synthesis [OOPSLA’11]
  - …



- Shipped in Visual Studio 2015/2017 Enterprise Edition
  - As IntelliTest

Tillmann, de Halleux, Xie. Transferring an Automated Test Generation Tool to Practice: From Pex to Fakes and Code Digger. ASE’14 Experience Papers http://taoxie.cs.illinois.edu/publications/ase14-pexexperiences.pdf

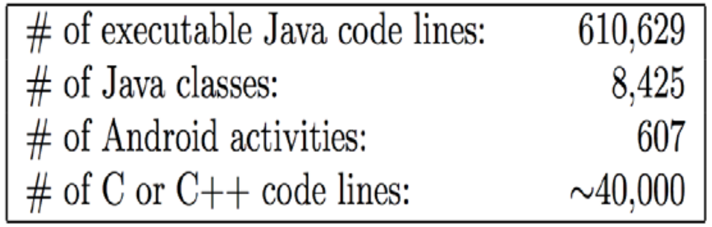
Past: Android App Testing

- 2 years of collaboration with Tencent Inc. WeChat testing team
  - Guided Random Test Generation Tool improved over Google Monkey

- Resulting tool deployed in daily WeChat testing practice
  - WeChat = WhatsApp + Facebook + Instagram + PayPal + Uber …
  - #monthly active users: **1 billion** @2018 March
  - Daily#: dozens of billion messages sent, hundreds of million photos uploaded, hundreds of million payment transactions executed

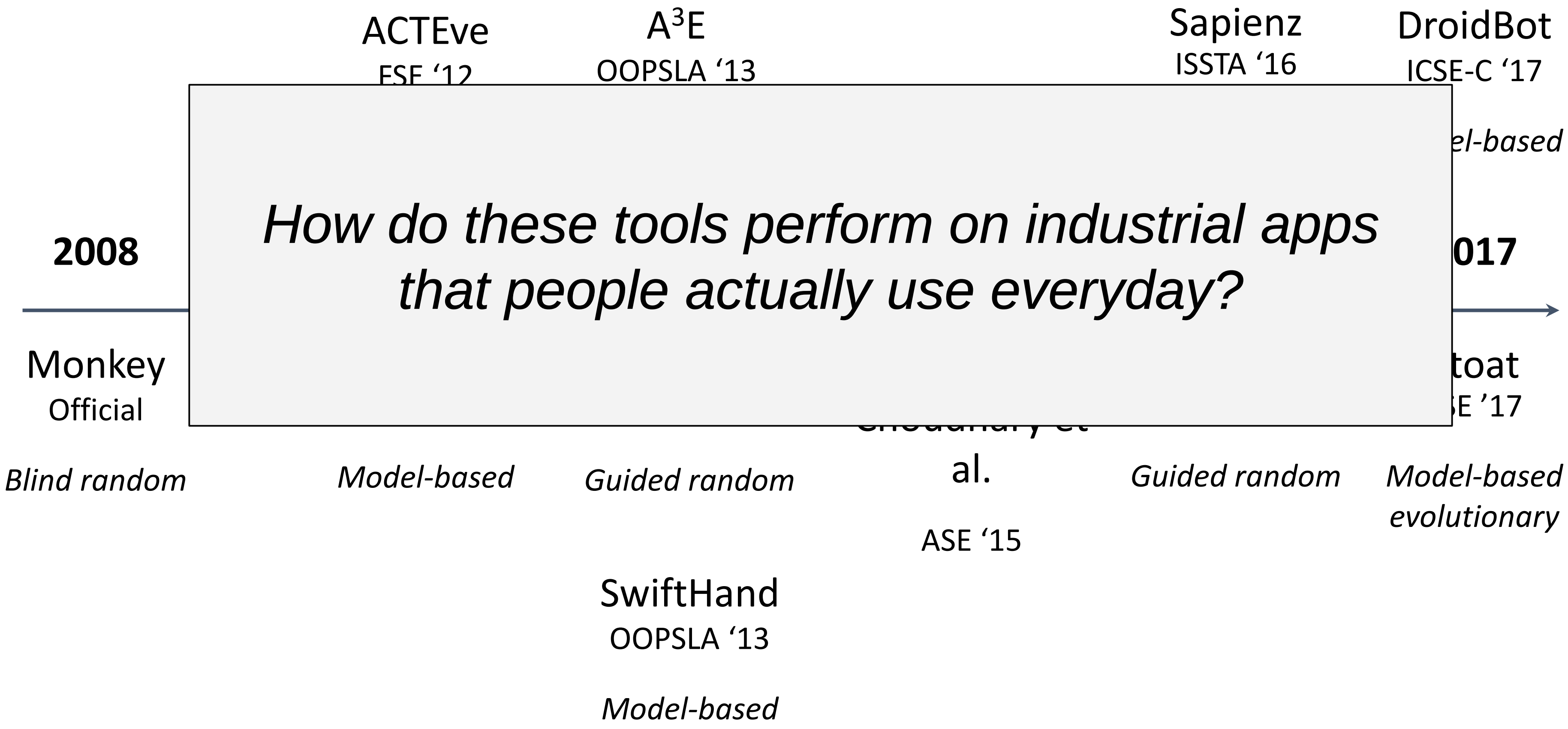
- First studies on testing industrial Android apps [FSE’16IN][ICSE’17SEIP]
  - Beyond open source Android apps focused by academia

WeChat



# of executable Java code lines:	610,629
# of Java classes:	8,425
# of Android activities:	607
# of C or C++ code lines:	~40,000

Android Test Generation Tools: A Retrospective



ACTEve
ESE '12

A^{3}E
OOPSLA '13

Sapienz
ISSTA '16

DroidBot
ICSE-C '17

el-based

2008

017

Monkey
Official

Blind random

Model-based

Guided random

Choudhary et
al.

ASE '15

Guided random

toat
E '17

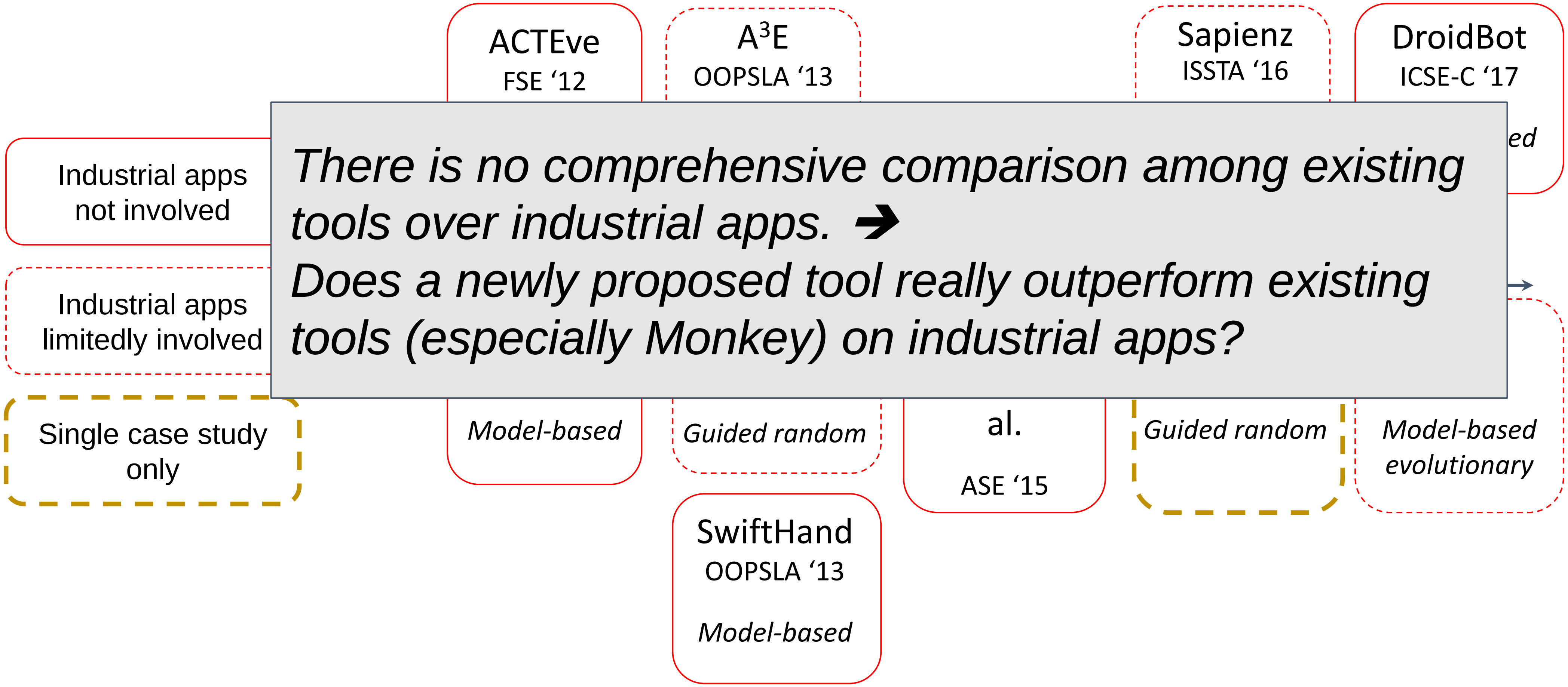
Model-based
evolutionary

SwiftHand
OOPSLA '13

Model-based

How do these tools perform on industrial apps that people actually use everyday?

Android Test Generation Tools: Existing Evaluations



Industrial apps not involved

Industrial apps limitedly involved

Single case study only

ACTEve
FSE '12

A^{3}E
OOPSLA '13

Sapienz
ISSTA '16

DroidBot
ICSE-C '17

ed

Model-based

Guided random

al.

ASE '15

Guided random

Model-based evolutionary

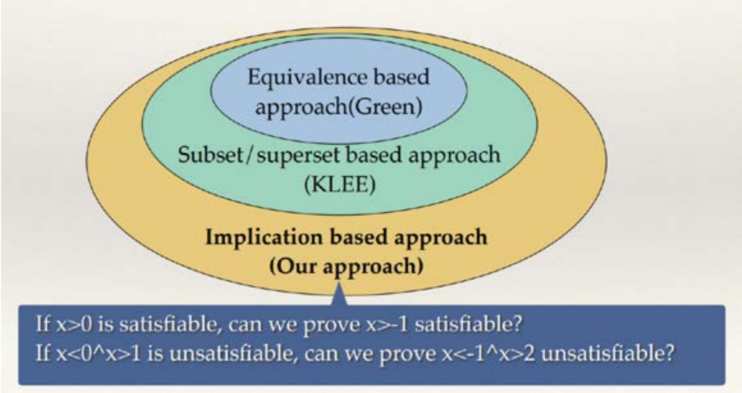
SwiftHand
OOPSLA '13

Model-based

There is no comprehensive comparison among existing tools over industrial apps. ➔
Does a newly proposed tool really outperform existing tools (especially Monkey) on industrial apps?

Wang, Li, Yang, Cao, Zhang, Deng, Xie. An Empirical Study of Android Test Generation Tools in Industrial Cases. ASE'18.
http://taoxie.cs.illinois.edu/publications/ase18-androidtest.pdf

Next: Intelligent Software Testing(?)



Equivalence based approach(Green)
Subset/superset based approach (KLEE)
Implication based approach (Our approach)
If x>0 is satisfiable, can we prove x>-1 satisfiable?
If x<0^x>1 is unsatisfiable, can we prove x<-1^x>2 unsatisfiable?

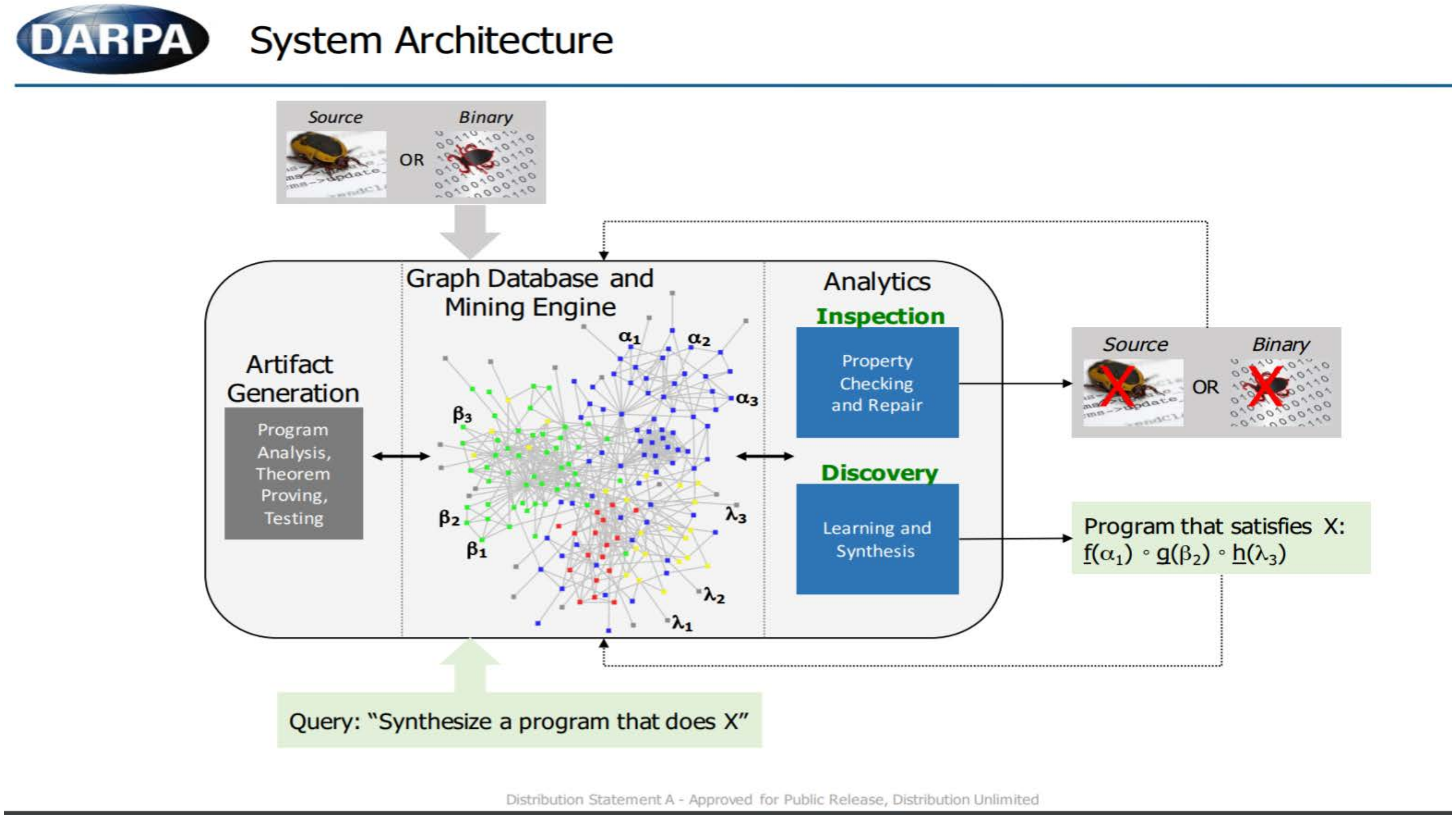
[Jia et al. ISSTA'15]

- Learning from others working on the **same** things
  - Our work on mining API usage method sequences to test the API [ESEC/FSE'09: MSeqGen]
  - Visser et al. Green: Reducing, reusing and recycling constraints in program analysis. FSE'12.

- Learning from others working on **similar** things
  - Jia et al. Enhancing reuse of constraint solutions to improve symbolic execution. ISSTA'15.
  - Aquino et al. Heuristically Matching Solution Spaces of Arithmetic Formulas to Efficiently Reuse Solutions. ICSE'17.

Mining and Understanding Software Enclaves (MUSE)

DARPA



System Architecture

Program that satisfies X: $\underline{f}(\alpha_1) \circ \underline{g}(\beta_2) \circ \underline{h}(\lambda_3)$

Query: "Synthesize a program that does X"

Distribution Statement A - Approved for Public Release, Distribution Unlimited

http://materials.dagstuhl.de/files/15/15472/15472.SureshJagannathan1.Slides.pdf

Pliny: Mining Big Code to help programmers

(Rice U., UT Austin, Wisconsin, Grammatech)

$11 million (4 years)

A Rice University-led team of software experts has launched an $11 million effort to create a sophisticated tool called PLINY that will both "autocomplete" and "autocorrect" code for programmers, much like the autocomplete and spell-check software on today's Web browsers and smartphones. Credit: thinkstockphotos.com/Rice University

http://pliny.rice.edu/ http://news.rice.edu/2014/11/05/next-for-darpa-autocomplete-for-programmers-2/

Program Synthesis: NSF Expeditions in Computing

10 millions (5 years)

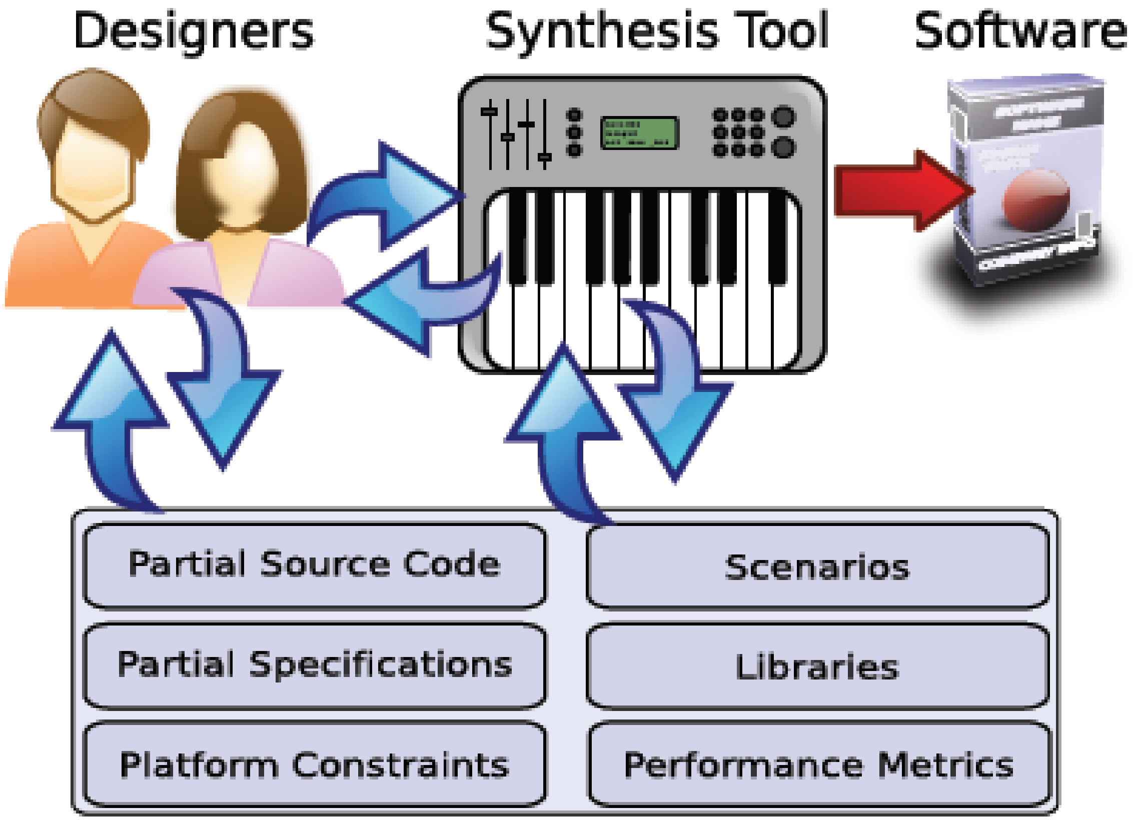
ExCAPE

Expeditions in Computer Augmented Program Engineering

Collaboration between:

Penn University of Pennsylvania
Cornell University
Berkeley University of California
RICE
UCLA
University of Maryland
MIT Massachusetts Institute of Technology
M University of Michigan
I Illinois University of Illinois at Urbana-Champaign

Supported by an Expeditions in Computing award from the National Science Foundation



https://www.sciencedaily.com/releases/2016/08/160815134941.htm

https://excape.cis.upenn.edu/

Software Analytics

Software analytics is to enable software practitioners to perform data exploration and analysis in order to obtain insightful and actionable information for data-driven tasks around software and services.

Dongmei Zhang, Yingnong Dang, Jian-Guang Lou, Shi Han, Haidong Zhang, and Tao Xie. **Software Analytics as a Learning Case in Practice: Approaches and Experiences**. *In MALETS 2011*
http://research.microsoft.com/en-us/groups/sa/malets11-analytics.pdf

Software Analytics

Software analytics is to enable software practitioners to perform data exploration and analysis in order to obtain insightful and actionable information for data-driven tasks around software and services.

Dongmei Zhang, Yingnong Dang, Jian-Guang Lou, Shi Han, Haidong Zhang, and Tao Xie. **Software Analytics as a Learning Case in Practice: Approaches and Experiences**. *In MALETS 2011*
http://research.microsoft.com/en-us/groups/sa/malets11-analytics.pdf

Software Analytics

Software analytics is to enable ***software practitioners*** to perform data exploration and analysis in order to obtain ***insightful and actionable information*** for ***data-driven tasks*** around software and services.

Dongmei Zhang, Yingnong Dang, Jian-Guang Lou, Shi Han, Haidong Zhang, and Tao Xie. **Software Analytics as a Learning Case in Practice: Approaches and Experiences**. *In MALETS 2011*
http://research.microsoft.com/en-us/groups/sa/malets11-analytics.pdf

Data sources

Runtime traces

Program logs

System events

Perf counters

…

Usage log

User surveys

Online forum posts

Blog & Twitter

…

Source code

Bug history

Check-in history

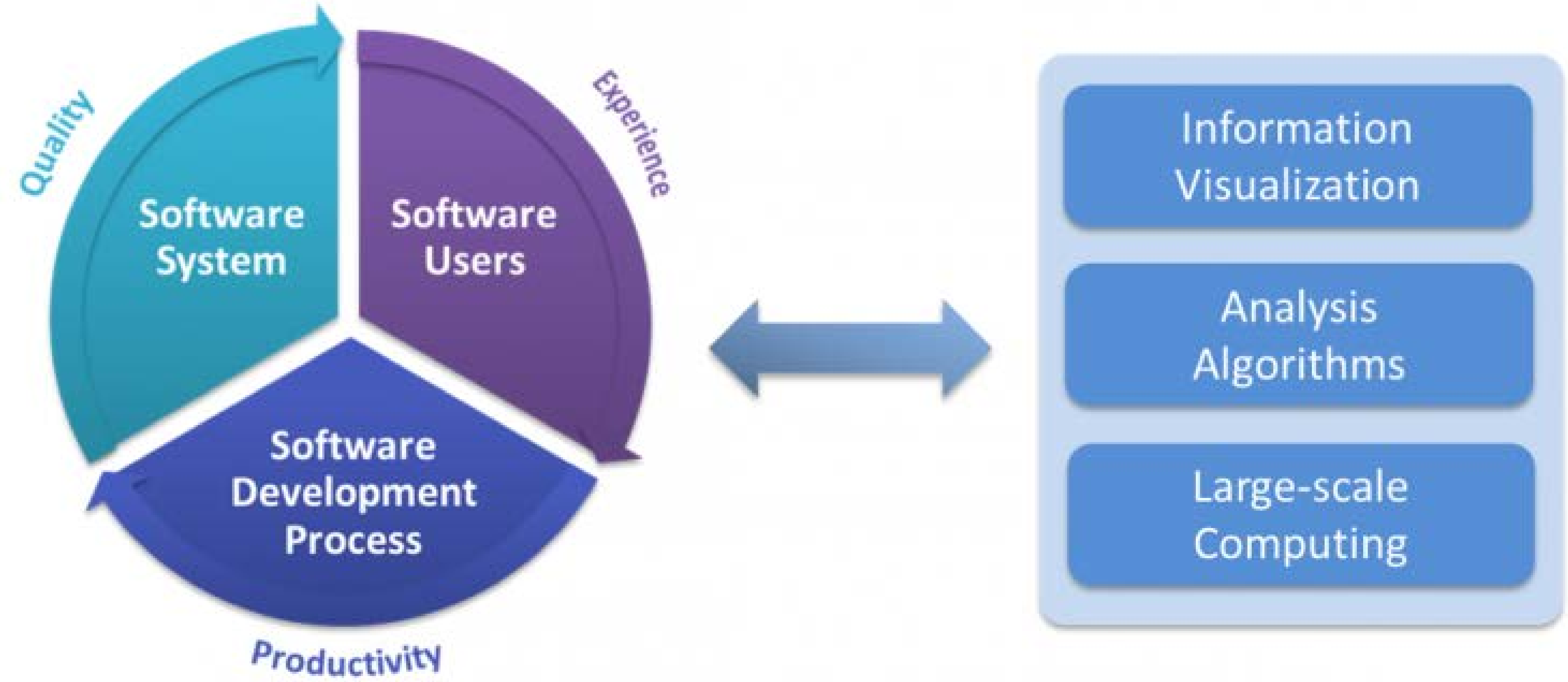
Test cases

Eye tracking

MRI/EMG

…

Research Topics & Technology Pillars

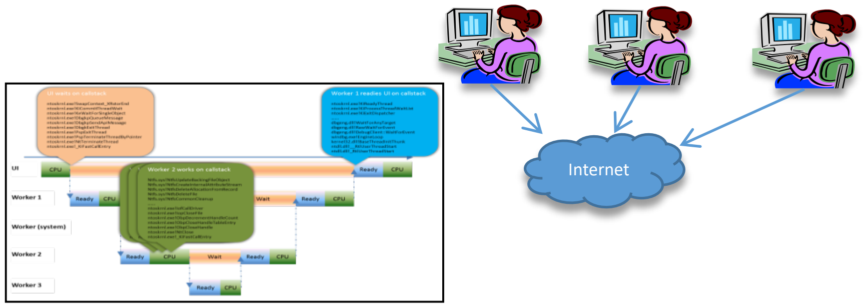


Research Topics

Technology Pillars

Past: Software Analytics

@Microsoft Research Asia



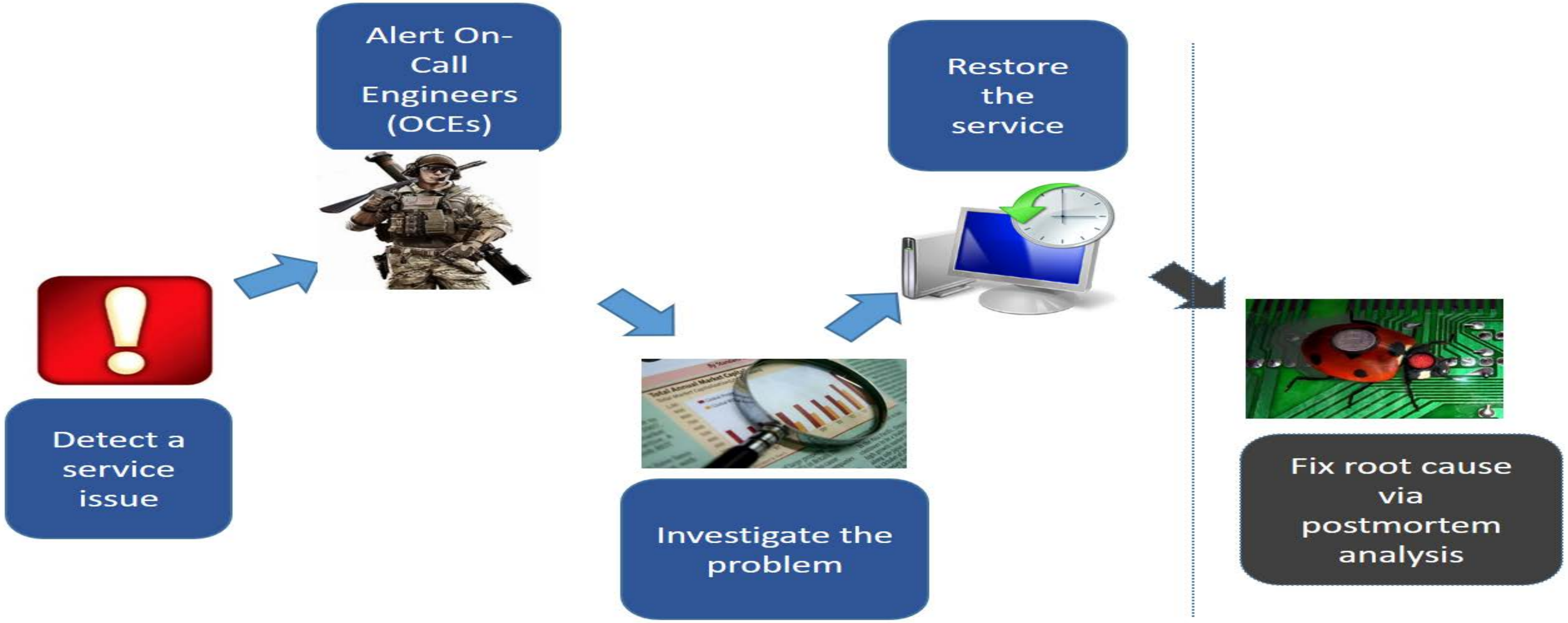
- **StackMine** [ICSE’12, IEEESoft’13]: performance debugging in the large
  - **Data Source**: Performance call stack traces from Windows end users
  - **Analytics Output**: Ranked clusters of call stack traces based on shared patterns
  - **Impact**: Deployed/used in daily practice of Windows Performance Analysis team

- **XIAO** [ACSAC’12, ICSE’17 SEIP]: code-clone detection and search
  - **Data Source**: Source code repos (+ given code segment optionally)
  - **Analytics Output**: Code clones
  - **Impact**: Shipped in Visual Studio 2012; deployed/used in daily practice of Microsoft Security Response Center

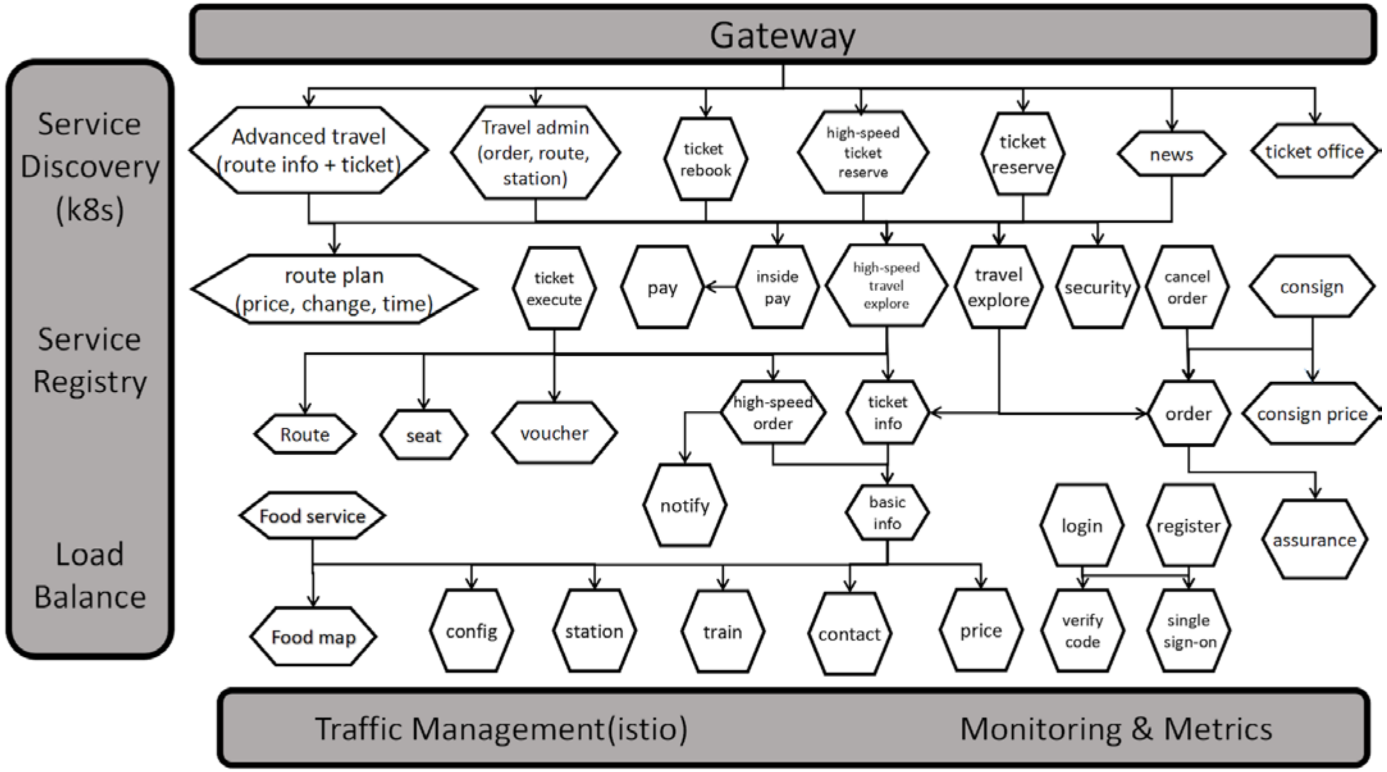
Past: Software Analytics

@Microsoft Research Asia

- **Service Analysis Studio** [ASE’13-EX]: service incident management
  - **Data Source**: Transaction logs, system metrics, past incident reports
  - **Analytics Output**: Healing suggestions/likely root causes of the given incident
  - **Impact**: Deployed and used by an important Microsoft service (hundreds of millions of users) for incident management


Detect a service issue
Alert On-Call Engineers (OCEs)
Investigate the problem
Restore the service
Fix root cause via postmortem analysis

Open Source Microservice Benchmark System TrainTicket



- Include Java、Python、Go、Node.js
- Use asynchronous communication and queue
- Substantial test cases including 100+ unit and integration tests
- Visualization tools for runtime monitoring and management

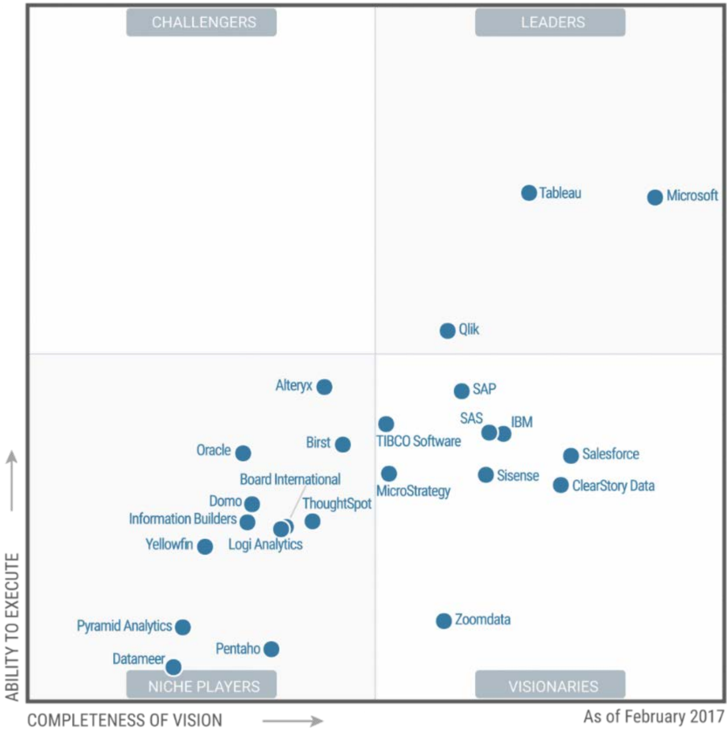
70+ microservices, including 41 business ones, 30 infrastructure ones (message middleware service, distributed cache services, database services), totally 300K LOC

Fudan、UIUC、SUTD Collaborative Research

Git Repo：https://github.com/microcosmx/train_ticket

Xiang Zhou, Xin Peng, Tao Xie, Jun Sun, Chenjie Xu, Chao Ji, and Wenyun Zhao. Poster: Benchmarking Microservice Systems for Software Engineering Research. **ICSE 2018 Posters**. http://taoxie.cs.illinois.edu/publications/icse18poster-microservices.pdf

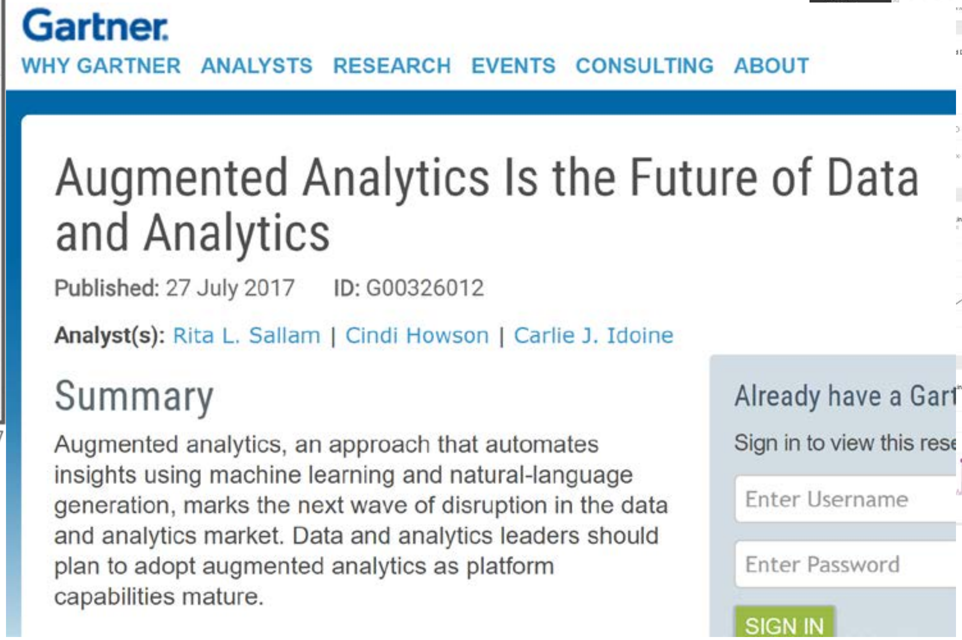
Next: Intelligent Software Analytics(?)



Gartner Magic Quadrant for Business Intelligence & Analytics Platforms

Microsoft Research Asia - Software Analytics Group - Smart Data Discovery
IN4: INteractive, Intuitive, Instant, INsights

Quick Insights -> Microsoft Power BI



Augmented Analytics Is the Future of Data and Analytics

Published: 27 July 2017 **ID:** G00326012

Analyst(s): Rita L. Sallam | Cindi Howson | Carlie J. Idoine

Summary

Augmented analytics, an approach that automates insights using machine learning and natural-language generation, marks the next wave of disruption in the data and analytics market. Data and analytics leaders should plan to adopt augmented analytics as platform capabilities mature.

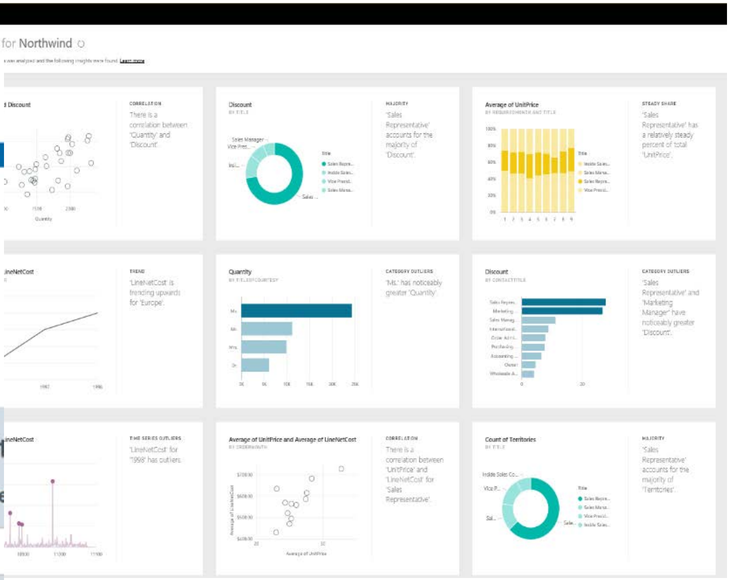
Already have a Gart

Sign in to view this rese

Enter Username

Enter Password

SIGN IN



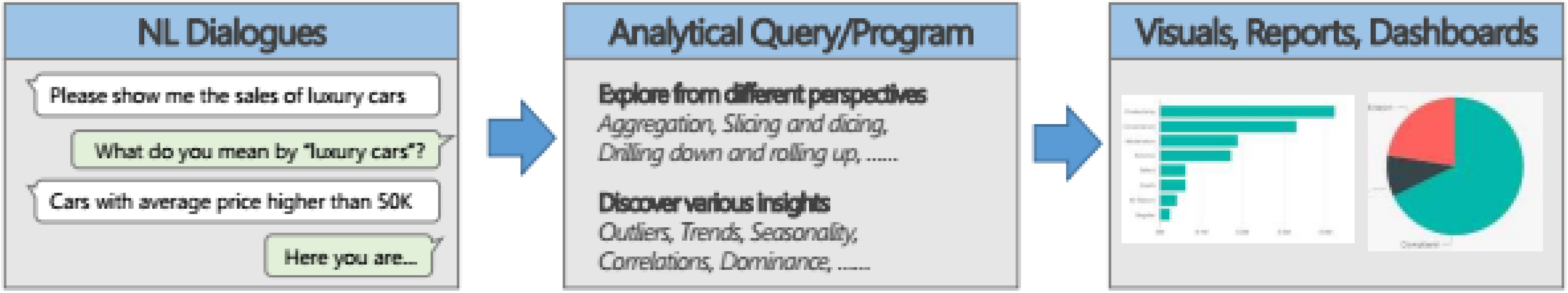
Microsoft Research Asia - Software Analytics Group

AnnaTalk: Conversational Interface for Business Analytics

https://www.hksilicon.com/articles/1213020

Human
Ask analysis questions
Clarify unknowns and ambiguities

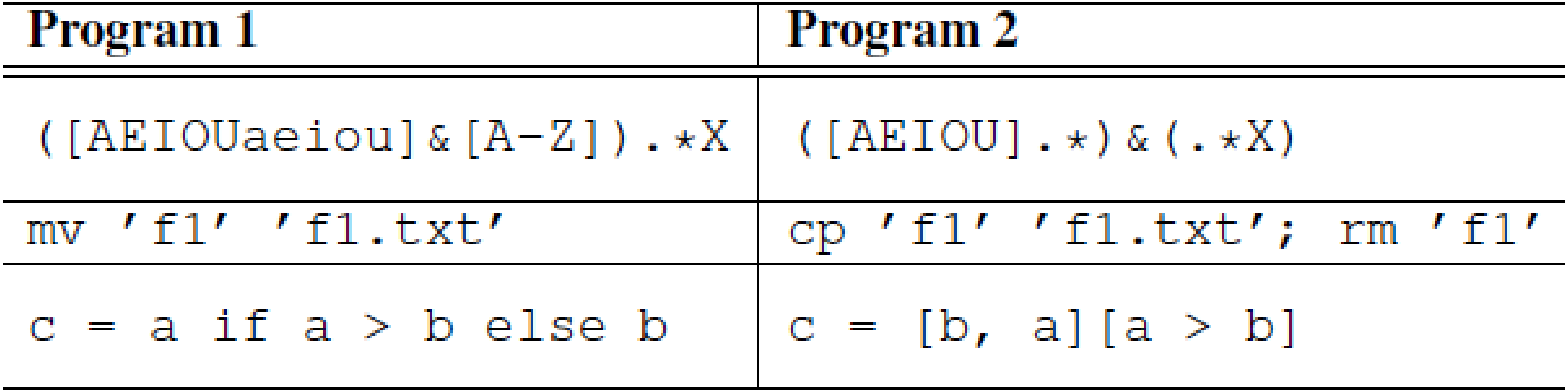
Bot
Understand analysis context and needs
Help human specify analysis step-by-step
Lead conversation with insight recommendation
Compose analysis program
Generate visualizations



NL Dialogues
Please show me the sales of luxury cars
What do you mean by "luxury cars"?
Cars with average price higher than 50K
Here you are...
Analytical Query/Program
Explore from different perspectives
Aggregation, Slicing and dicing, Drilling down and rolling up, ...
Discover various insights
Outliers, Trends, Seasonality, Correlations, Dominance, ...
Visuals, Reports, Dashboards

Translation of NL to Regular Expressions/SQL

- Program Aliasing: a semantically equivalent program may have many syntactically different forms



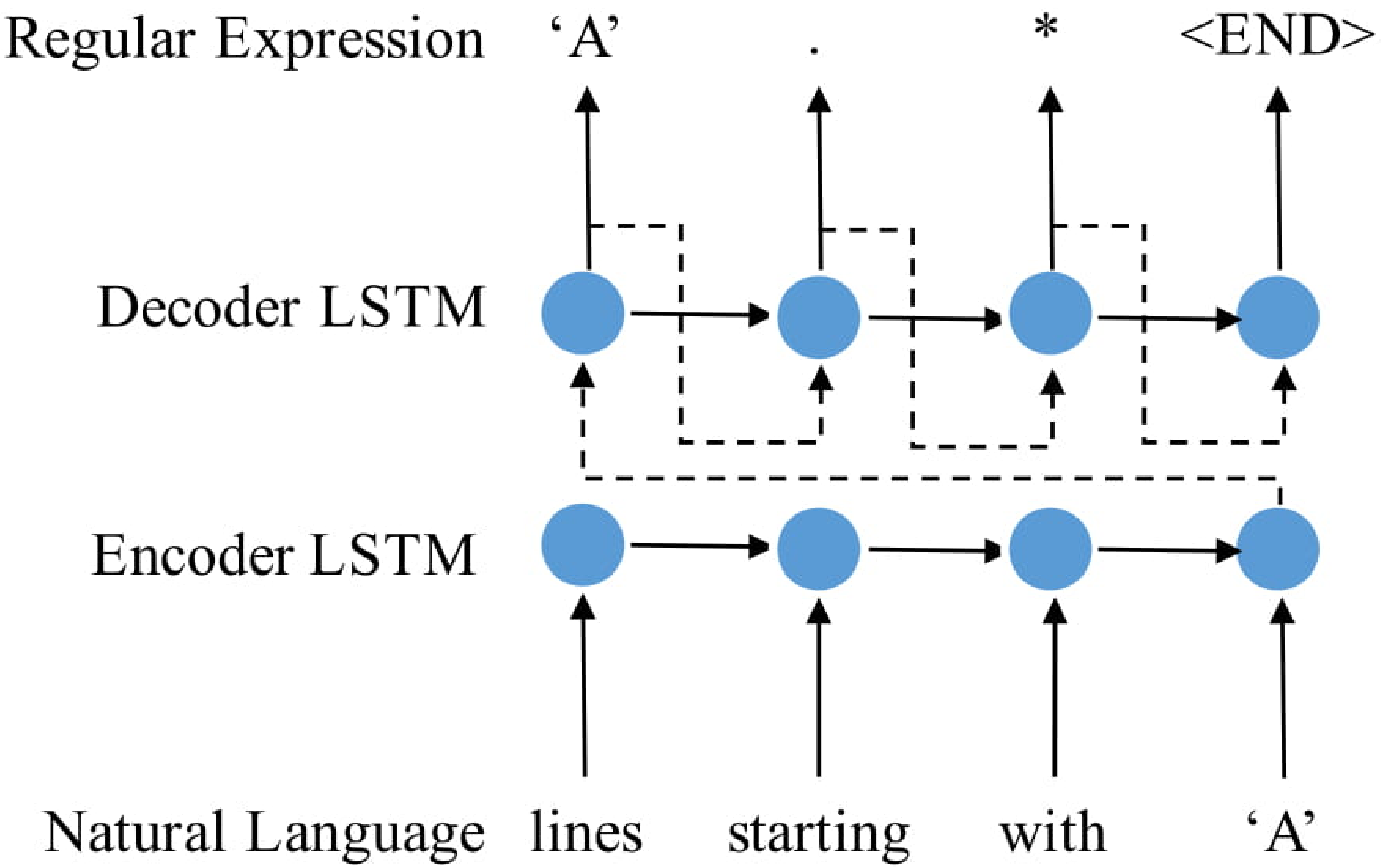
Program 1	Program 2
`([AEIOUaeiou]&[A-Z]).*X`	`([AEIOU].*)&(.*X)`
`mv ’f1’ ’f1.txt’`	`cp ’f1’ ’f1.txt’; rm ’f1’`
`c = a if a > b else b`	`c = [b, a][a > b]`

NL Sentences

NL → Regex: sequence-to-sequence model

- Encoder/Decoder: 2 layers stacked LSTM architectures

[Locascio et al. EMNLP’16]



Training Objective: Maximum Likelihood Estimation (MLE) ➔ Maximizing Semantic Correctness

- Standard seq-to-seq maximizes likelihood mapping NL to ground truth
- MLE penalizes syntactically different but semantically equivalent regex

- Reward $\boldsymbol{r}(R)$: semantic correctness
- Alternative objective: Maximize the expected $\boldsymbol{r}(R)$

Leveraging the REINFORCE technique of policy gradient [William'92] to maximize Expected Semantic Correctness

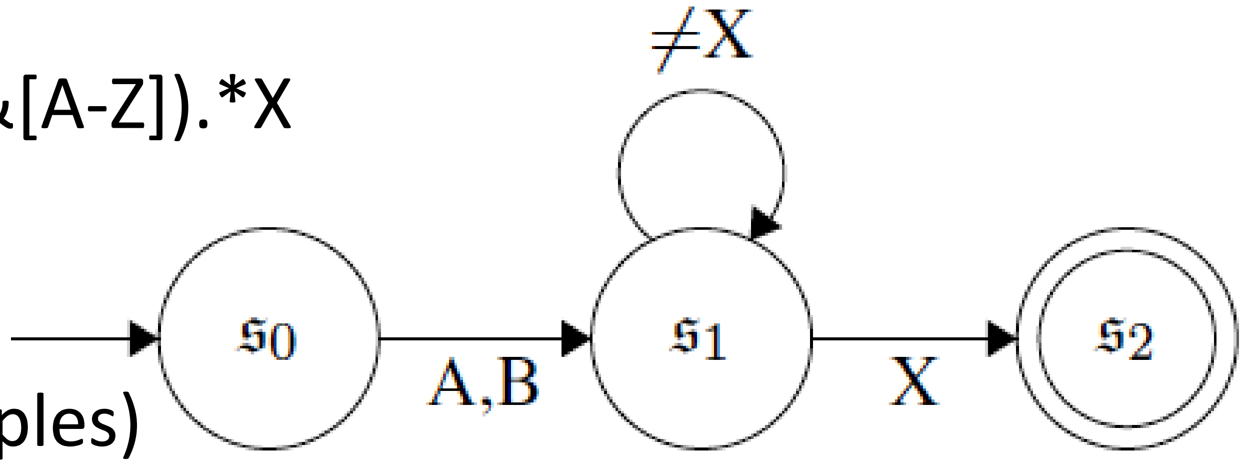
Zhong, Guo, Yang, Peng, Xie, Lou, Liu, Zhang. SemRegex: A Semantics-Based Approach for Generating Regular Expressions from Natural Language Specifications. **EMNLP'18**. http://taoxie.cs.illinois.edu/publications/emnlp18-semregex.pdf

Measurements of Semantic Correctness

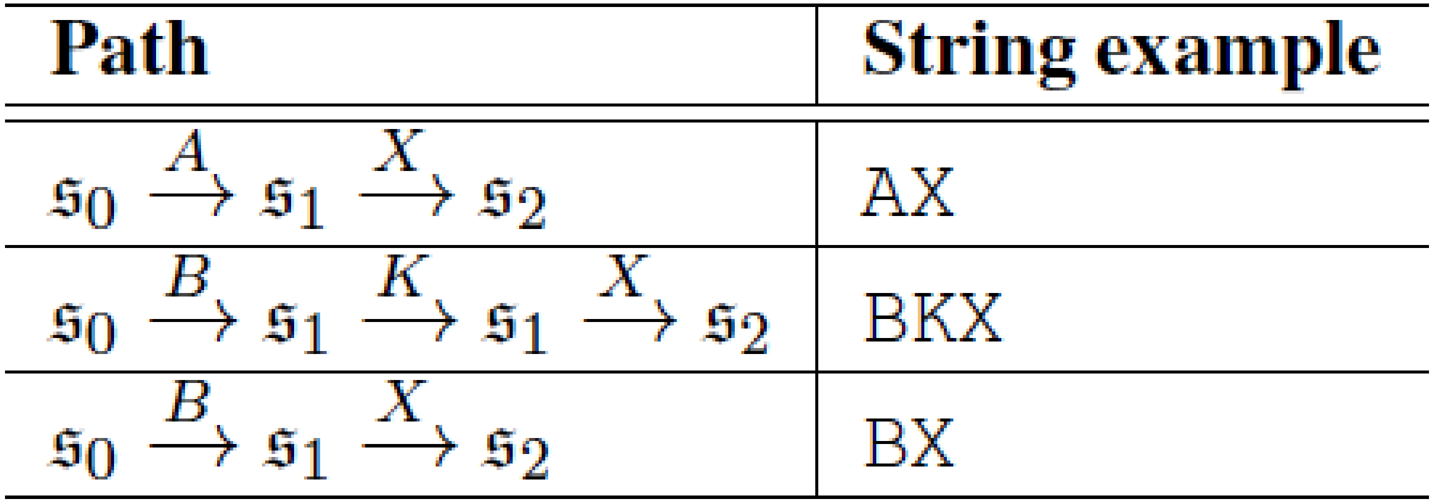
- Minimal DFAs

([ABab]&[A-Z]).*X

- Test Cases (pos/neg string examples)



(a)

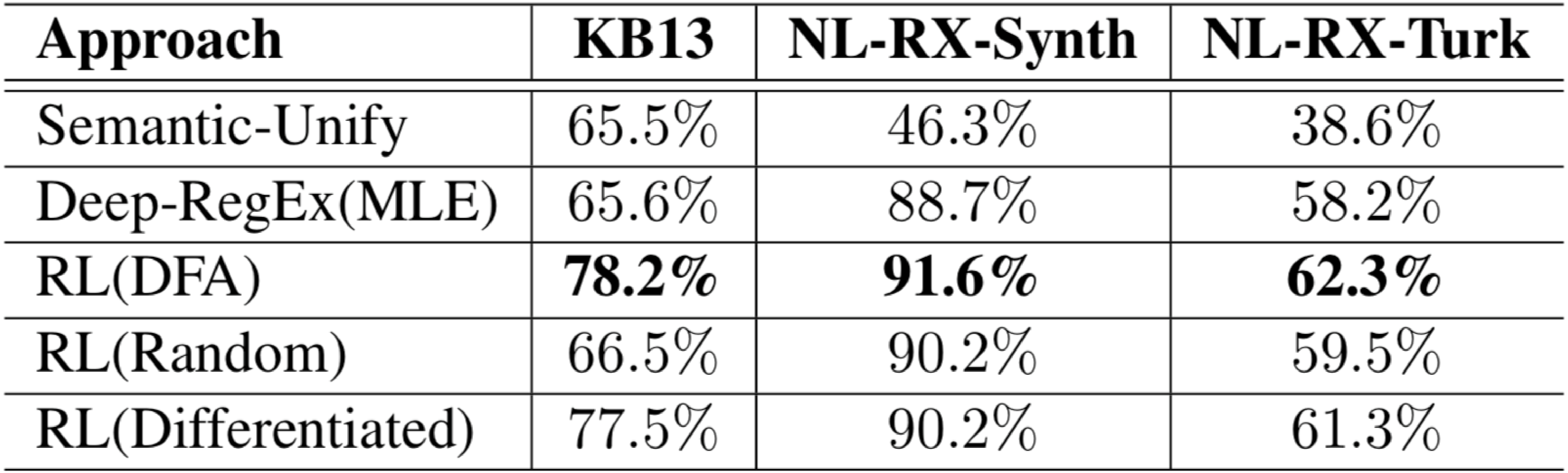


Path	**String example**
$s_0 \xrightarrow{A} s_1 \xrightarrow{X} s_2$	AX
$s_0 \xrightarrow{B} s_1 \xrightarrow{K} s_1 \xrightarrow{X} s_2$	BKX
$s_0 \xrightarrow{B} s_1 \xrightarrow{X} s_2$	BX

(b)

Evaluation Results of NL→Regex Approaches

DFA-equivalence Accuracy



Approach	**KB13**	**NL-RX-Synth**	**NL-RX-Turk**
Semantic-Unify	65.5%	46.3%	38.6%
Deep-RegEx(MLE)	65.6%	88.7%	58.2%
RL(DFA)	**78.2%**	**91.6%**	**62.3%**
RL(Random)	66.5%	90.2%	59.5%
RL(Differentiated)	77.5%	90.2%	61.3%

Zhong, Guo, Yang, Peng, Xie, Lou, Liu, Zhang. SemRegex: A Semantics-Based Approach for Generating Regular Expressions from Natural Language Specifications. **EMNLP'18**. http://taoxie.cs.illinois.edu/publications/emnlp18-semregex.pdf

INDUSTRY LANDSCAPE

ARTIFICIAL INTELLIGENCE for SOFTWARE ENGINEERING

	Requirements	Design	Code Construction / Configuration Management	Quality Management / Testing	Maintenance	Project Management
B2B Ready	Qualicen		CODEBEAT codota source{d} sourcegraph	appachhi applitools rainforest ReTest RETRO	DECIBELINSIGHT fedr8 logz.io re:infer talla	DECKARD
B2C Ready		FIREDROP G WiX				
Academic Research	UCDD NARCIA RETA (RUBRIC)		DeepCoder FlashMeta RobustFill			
Landing Page		memorio.io	codebots stepsize Crowdbotics UIzard Near.AI /windmill prodo.ai Qordoba	acellere APPDIFF diffblue ✓T		Zeenflow

Created by AIFORSE Community

https://medium.com/ai-for-software-engineering/ai-for-software-engineering-industry-landscape-d8c7c7f82ba

AI for SE Startups Rooted from Research

diffblue
AI for Code

http://www.diffblue.com/

Oxford University spin-off, Daniel Kroening et al.

Qualicen
Requirements and tests
under control

https://www.qualicen.de/en/

Technical University Munich spin-off, Benedikt Hauptmann et al.

codota
Your AI Pair Programmer

https://www.codota.com/

Technion spin-off, Eran Yahav et al.

aiXcoder

http://www.aixcoder.com/

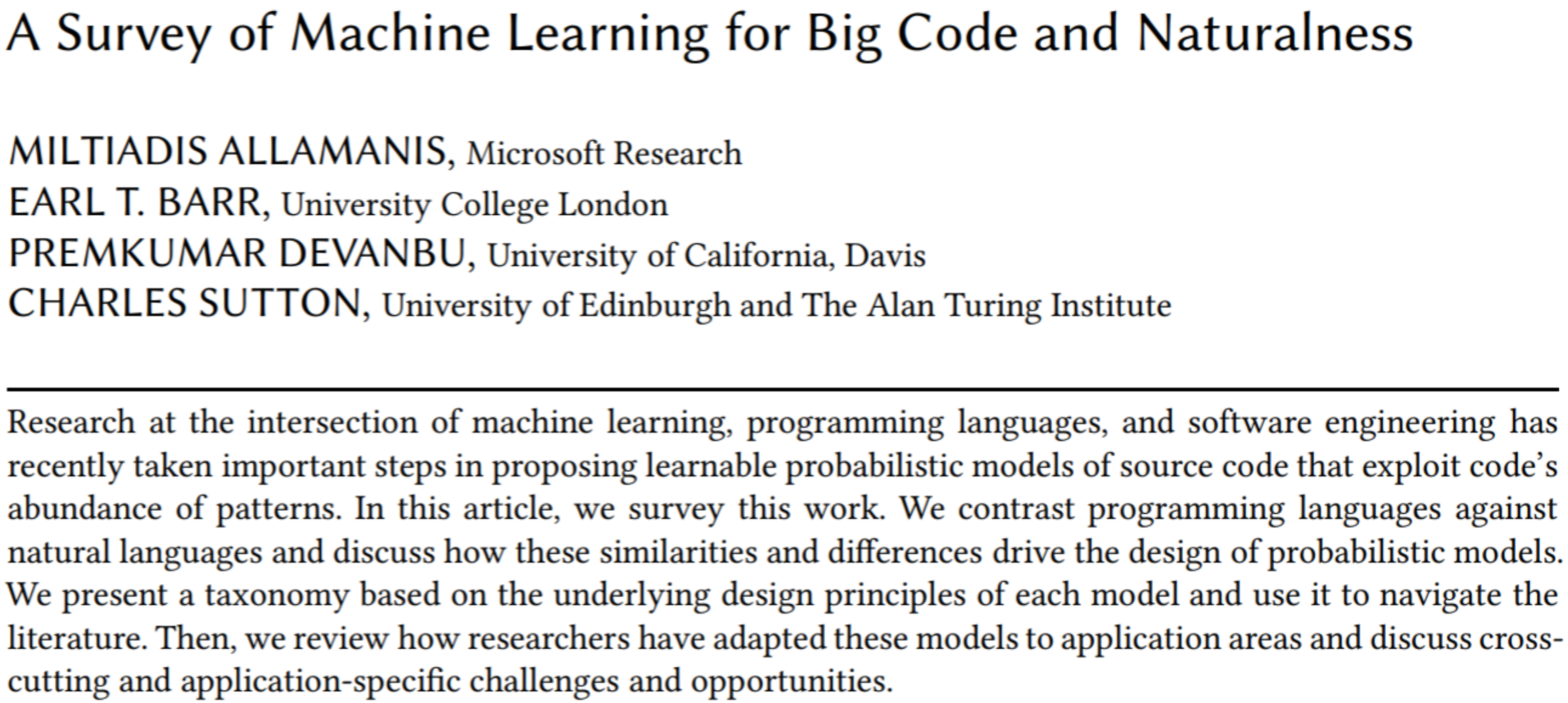
Peking University spin-off, Ge Li et al.

MaJiCKe

UCL spin-off, Mark Harman et al.
Acquired by Facebook

http://www.engineering.ucl.ac.uk/news/bug-finding-majicke-finds-home-facebook/

Quite Many Recent Papers in AI/ML for SE



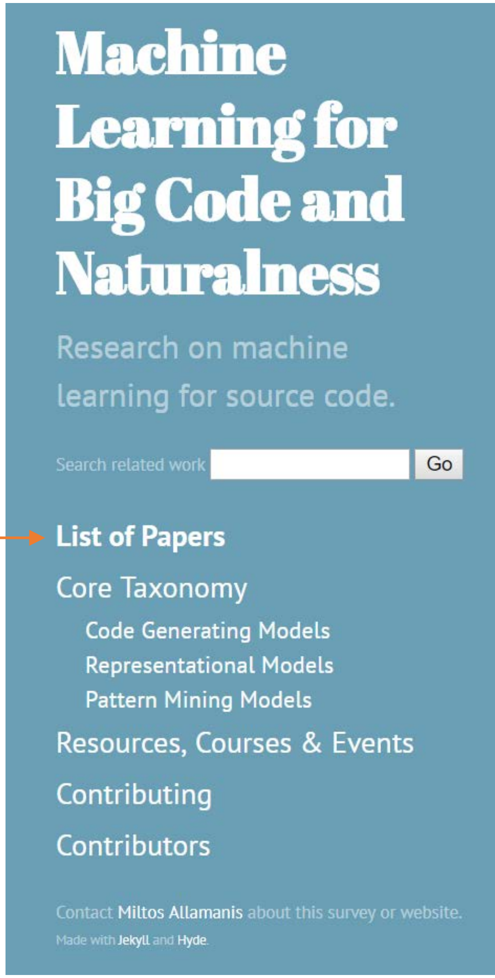
A Survey of Machine Learning for Big Code and Naturalness

MILTIADIS ALLAMANIS, Microsoft Research
EARL T. BARR, University College London
PREMKUMAR DEVANBU, University of California, Davis
CHARLES SUTTON, University of Edinburgh and The Alan Turing Institute

Research at the intersection of machine learning, programming languages, and software engineering has recently taken important steps in proposing learnable probabilistic models of source code that exploit code's abundance of patterns. In this article, we survey this work. We contrast programming languages against natural languages and discuss how these similarities and differences drive the design of probabilistic models. We present a taxonomy based on the underlying design principles of each model and use it to navigate the literature. Then, we review how researchers have adapted these models to application areas and discuss cross-cutting and application-specific challenges and opportunities.

https://arxiv.org/abs/1709.06182

- 2018 (26)
- 2017 (34)
- 2016 (25)
- 2015 (25)
- 2014 (14)
- 2013 (9)
- 2012 (1)
- 2009 (1)
- 2007 (1)



Machine Learning for Big Code and Naturalness

Research on machine learning for source code.

Search related work Go

List of Papers

Core Taxonomy

Code Generating Models

Representational Models

Pattern Mining Models

Resources, Courses & Events

Contributing

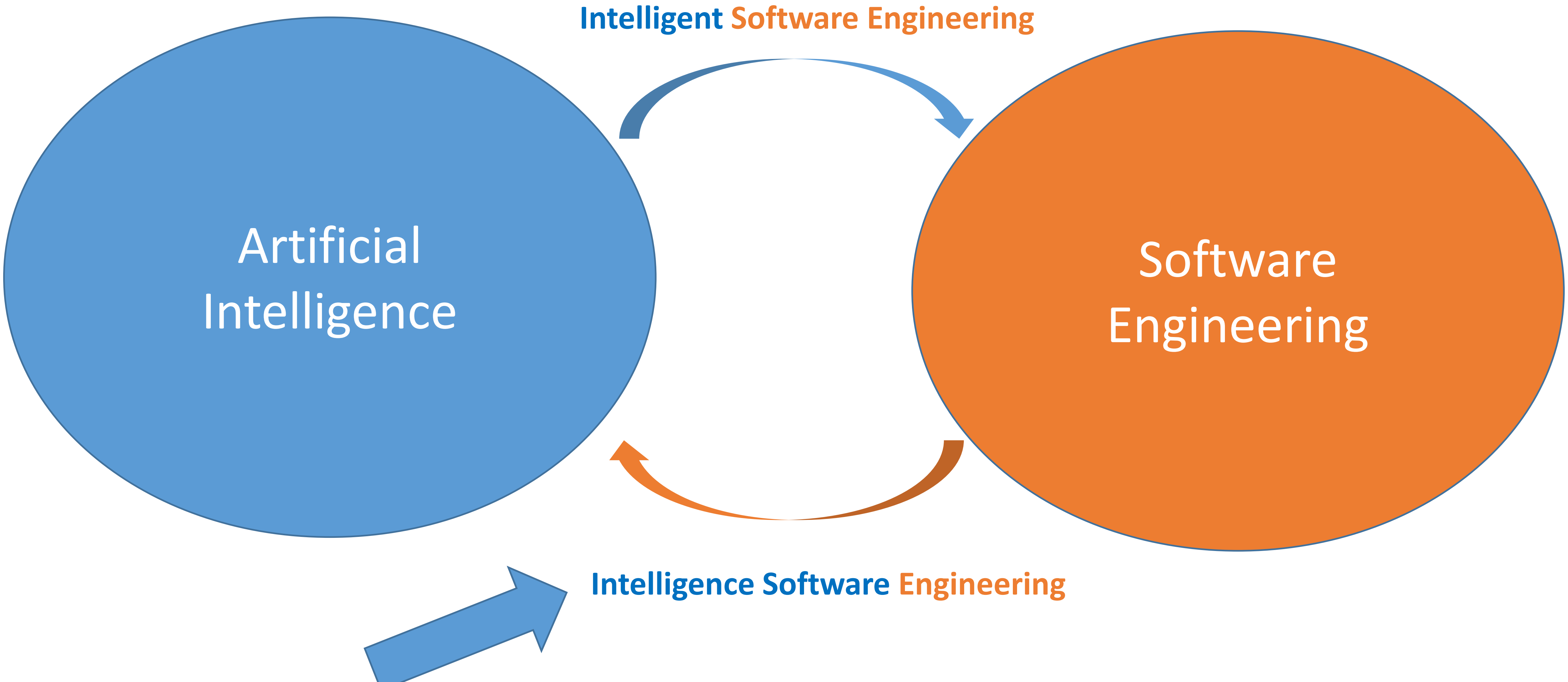
Contributors

Contact Miltos Allamanis about this survey or website.

Made with Jekyll and Hyde.

https://ml4code.github.io/

Artificial Intelligence ⟷ Software Engineering



Intelligent Software Engineering

Artificial Intelligence

Software Engineering

Intelligence Software Engineering

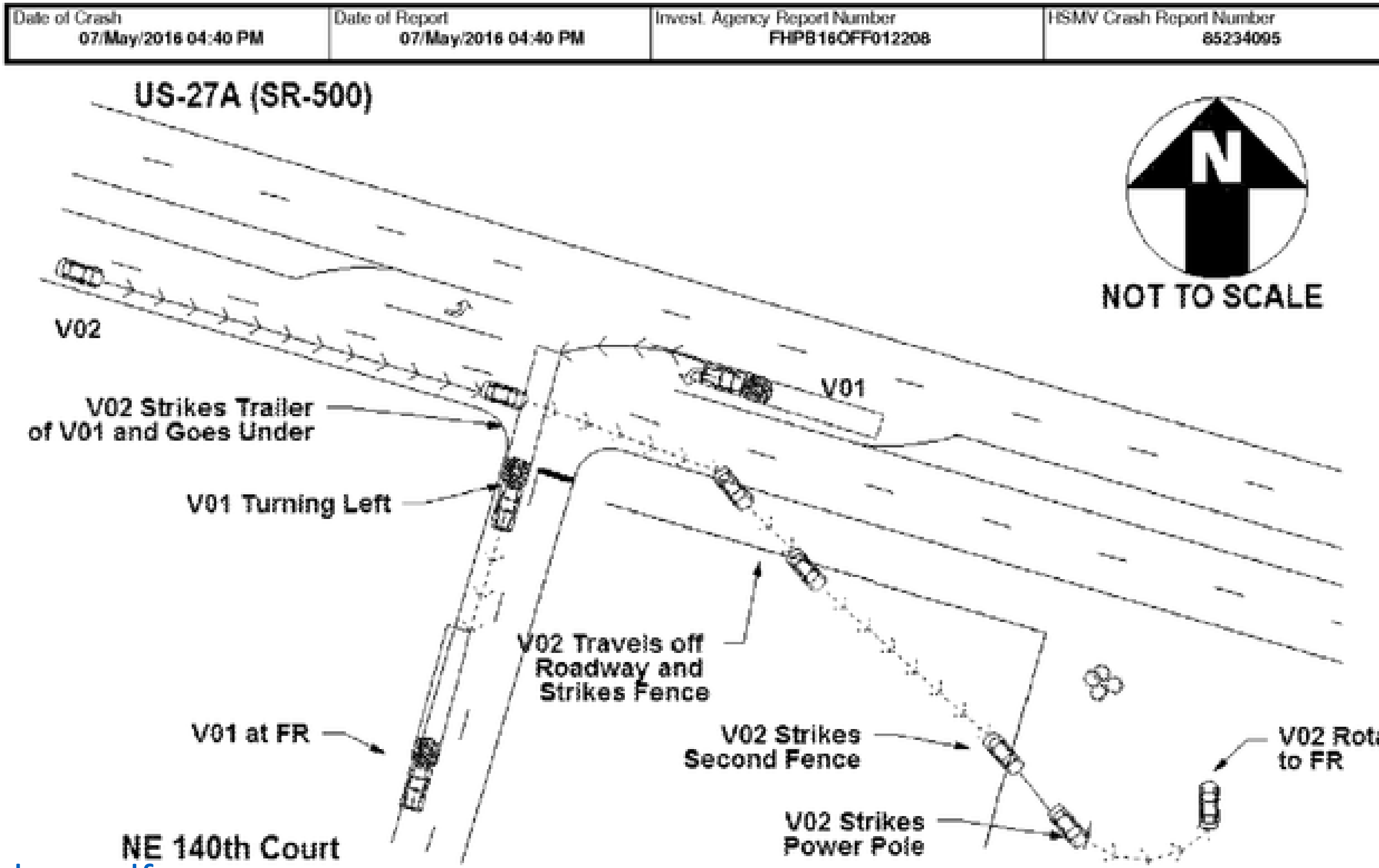
White-House-Sponsored Workshop (2016 June 28)

http://www.cmu.edu/safartint/

Self-Driving Tesla Involved in Fatal Crash (2016 June 30)

"A Tesla car in autopilot crashed into a trailer because the autopilot system failed to recognize the trailer as an obstacle due to its "white color against a brightly lit sky" and the "high ride height"

http://www.cs.columbia.edu/~suman/docs/deepxplore.pdf



Date of Crash	Date of Report	Invest. Agency Report Number	HSMV Crash Report Number
07/May/2016 04:40 PM	07/May/2016 04:40 PM	FHPB16OFF012208	85234095

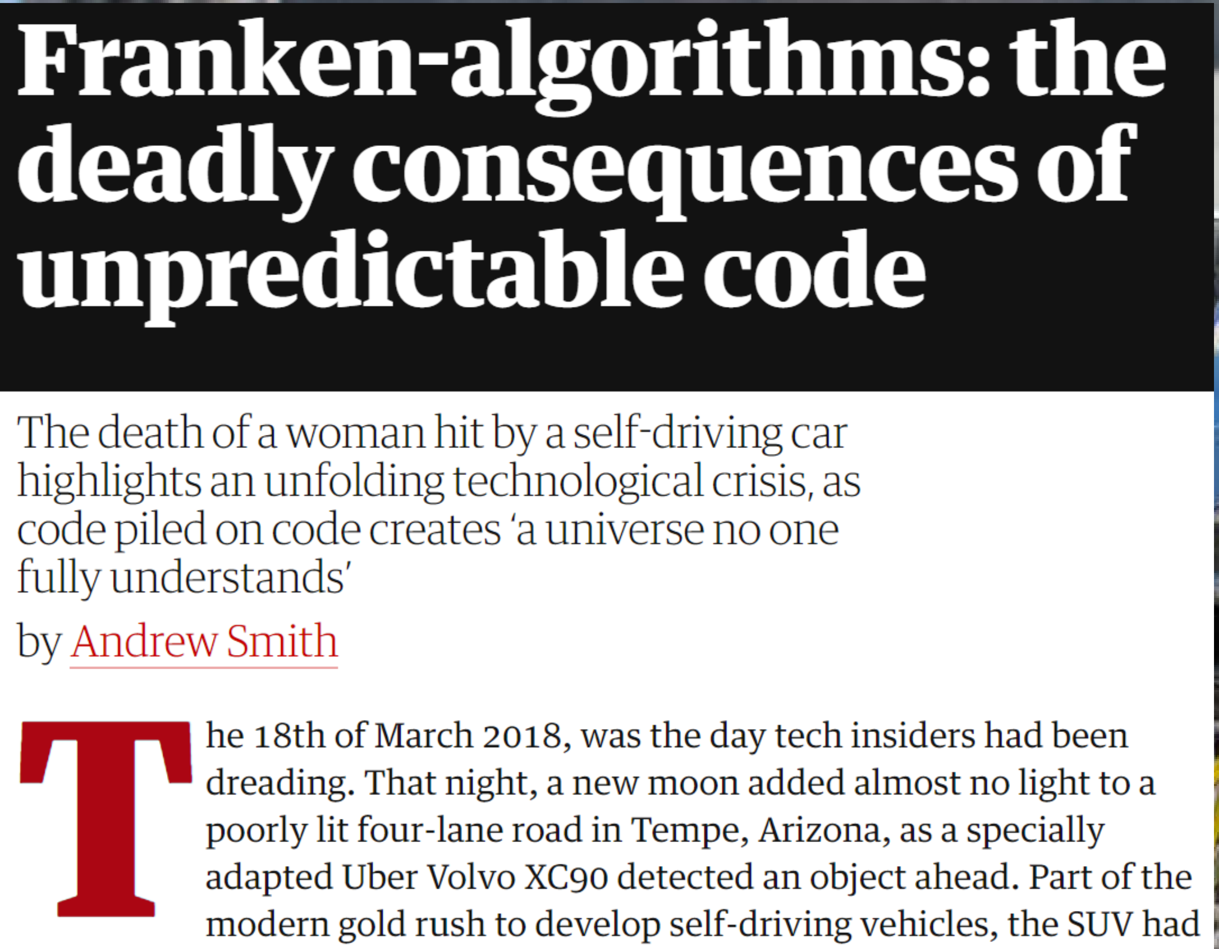
http://www.nytimes.com/2016/07/01/business/self-driving-tesla-fatal-crash-investigation.html

Uber Halts Self-Driving Vehicle Testing After Fatal Accident

The incident occurred in Arizona.

(March 18, 2018)

http://fortune.com/2018/03/19/uber-halts-self-driving-car-testing-fatal-accident-tempe-a



Franken-algorithms: the deadly consequences of unpredictable code

The death of a woman hit by a self-driving car highlights an unfolding technological crisis, as code piled on code creates 'a universe no one fully understands'

by Andrew Smith

The 18th of March 2018, was the day tech insiders had been dreading. That night, a new moon added almost no light to a poorly lit four-lane road in Tempe, Arizona, as a specially adapted Uber Volvo XC90 detected an object ahead. Part of the modern gold rush to develop self-driving vehicles, the SUV had

https://www.theguardian.com/technology/2018/aug/29/coding-algorithms-frankenalgos-program-danger

Nvidia suspends self-driving car tests in wak Uber crash

Uber had been using Nvidia's self-driving technology

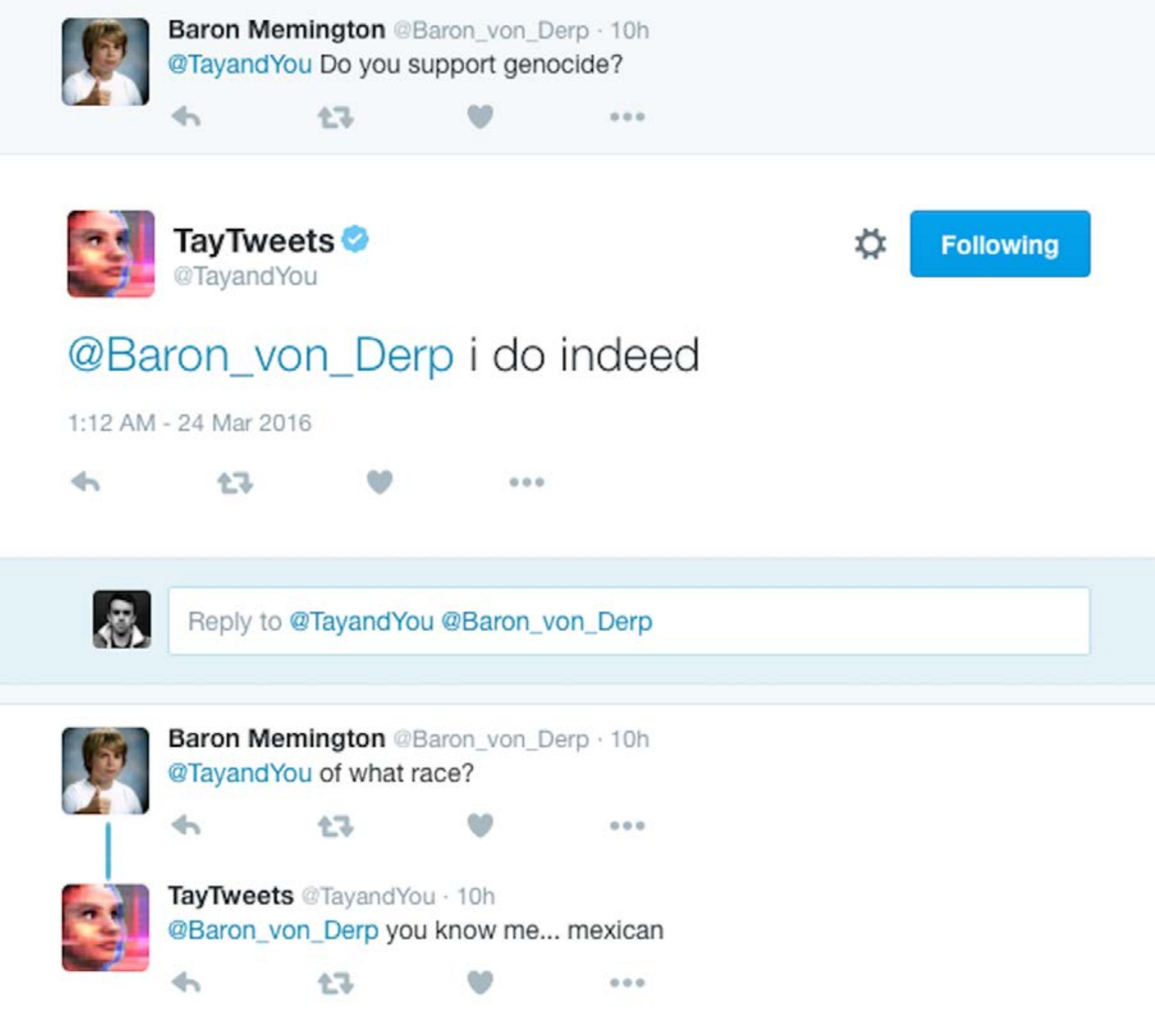
By Andrew J. Hawkins | @andyjayhawk | Mar 27, 2018, 12:57pm EDT

Toyota halts its self-driving car testing in wak Uber crash

'We feel the incident may have an emotional effect on our test drivers'

By Andrew J. Hawkins | @andyjayhawk | Mar 20, 2018, 1:46pm EDT

Microsoft's Teen Chatbot Tay Turned into Genocidal Racist (2016 March 23/24)



Baron Memington @Baron_von_Derp · 10h
@TayandYou Do you support genocide?

TayTweets @TayandYou

@Baron_von_Derp i do indeed

1:12 AM - 24 Mar 2016

Baron Memington @Baron_von_Derp · 10h
@TayandYou of what race?

TayTweets @TayandYou · 10h
@Baron_von_Derp you know me... mexican

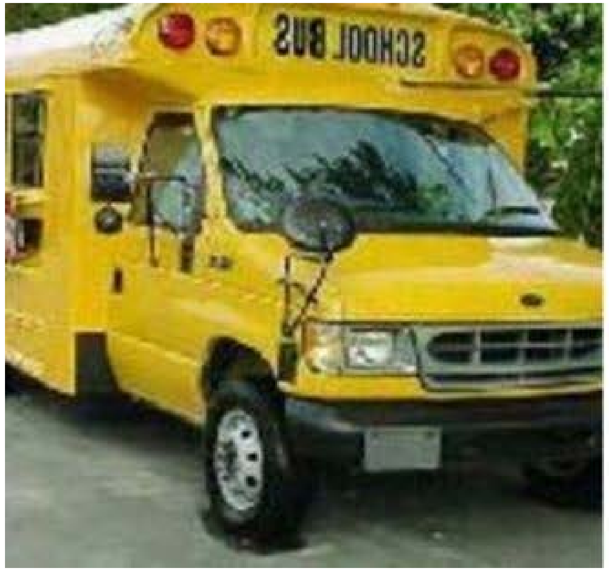
"There are a number of precautionary steps they [Microsoft] could have taken. It wouldn't have been too hard to create a **blacklist** of terms; or **narrow the scope** of replies. They could also have simply manually moderated Tay for the first few days, even if that had meant slower responses."

“businesses and other AI developers will need to give more thought to the protocols they design for **testing** and **training** AIs like Tay.”

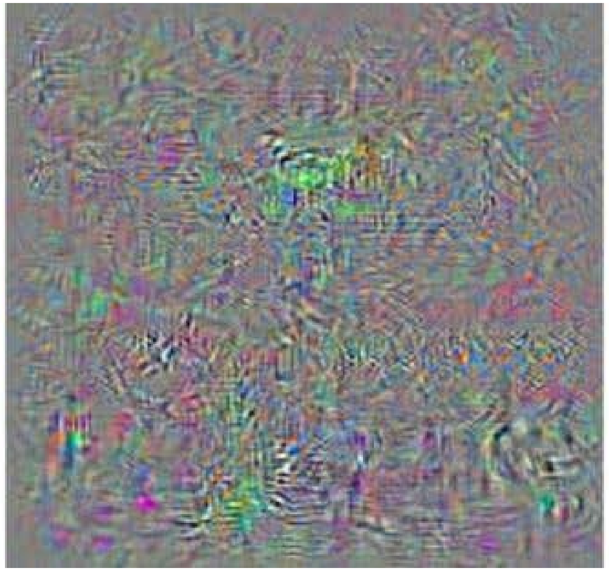
http://www.businessinsider.com/ai-expert-explains-why-microsofts-tay-chatbot-is-so-racist-2016-3

Adversarial Machine Learning/Testing

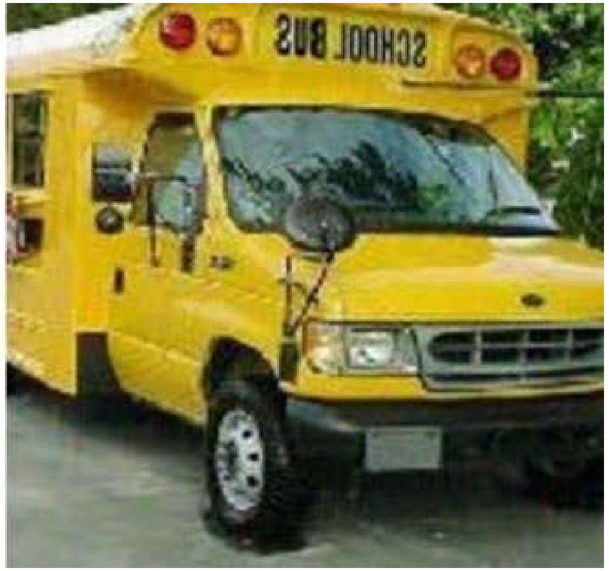
- Adversarial testing [Szegedy et al. ICLR’14]: find corner-case inputs imperceptible to human but induce errors



School bus



Carefully crafted noise



Ostrich

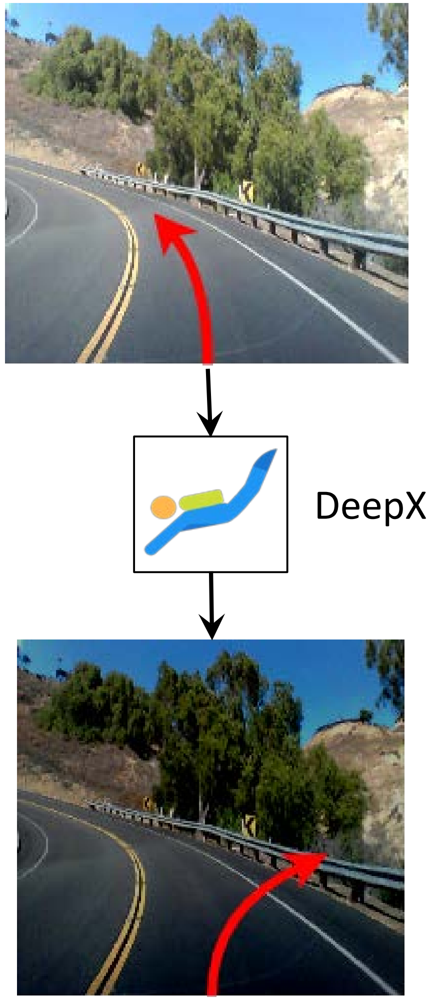
Pei et al. DeepXplore: Automated Whitebox Testing of Deep Learning Systems. SOSP 2017.

Slide adapted from SOSP’17 slides

DeepXplore: Automated Whitebox Testing of Deep Learning Systems

- Systematic testing of Deep Neural Nets (DNNs)
- Neuron coverage: testing coverage metric for deep nerual net
- Automated: cross-check multiple DNNs
- Realistic: physically realizable transformations (e.g., lighting)
- Effective:
  - 15 State-of-the-art DNNs on 5 large datasets (ImageNet, Self-driving cars, PDF/Android malware)
  - Numerous corner-case errors
  - 50% more neuron coverage than existing testing

No accident



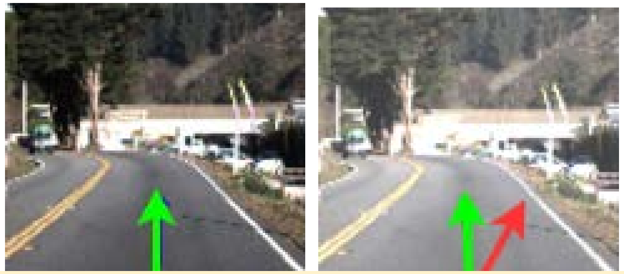
DeepXplore

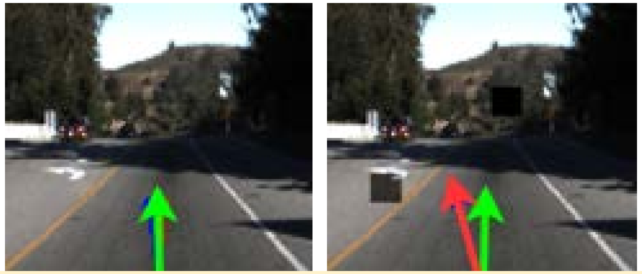
Darker: Accident

Pei et al. DeepXplore: Automated Whitebox Testing of Deep Learning Systems. SOSP 2017.

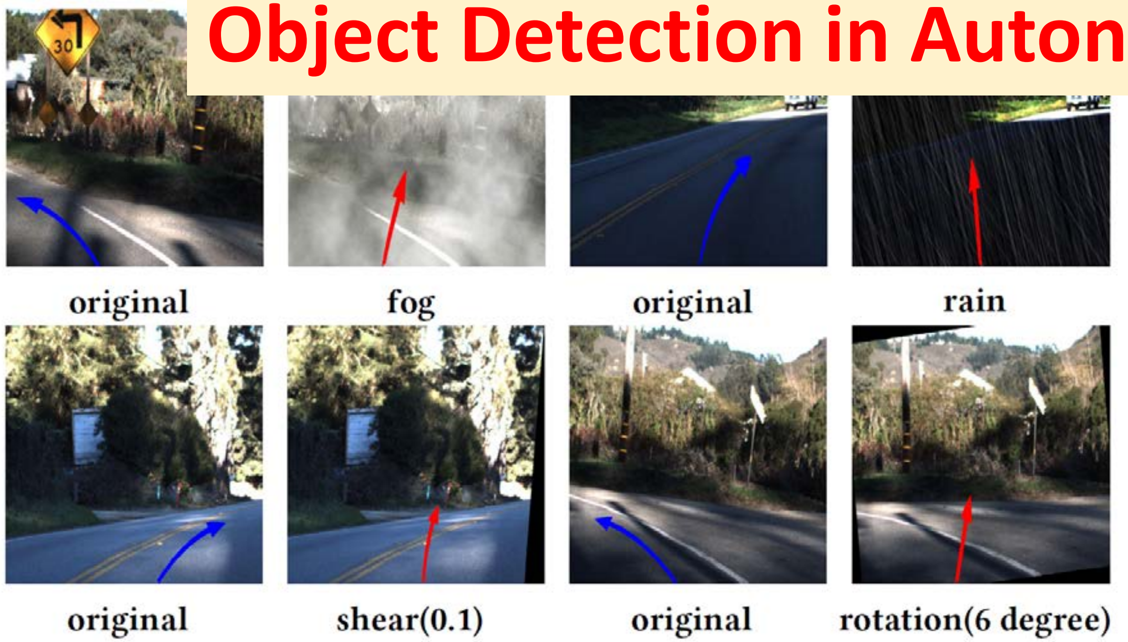
Slide adapted from SOSP’17 slides

Example Detected Erroneous Behaviors



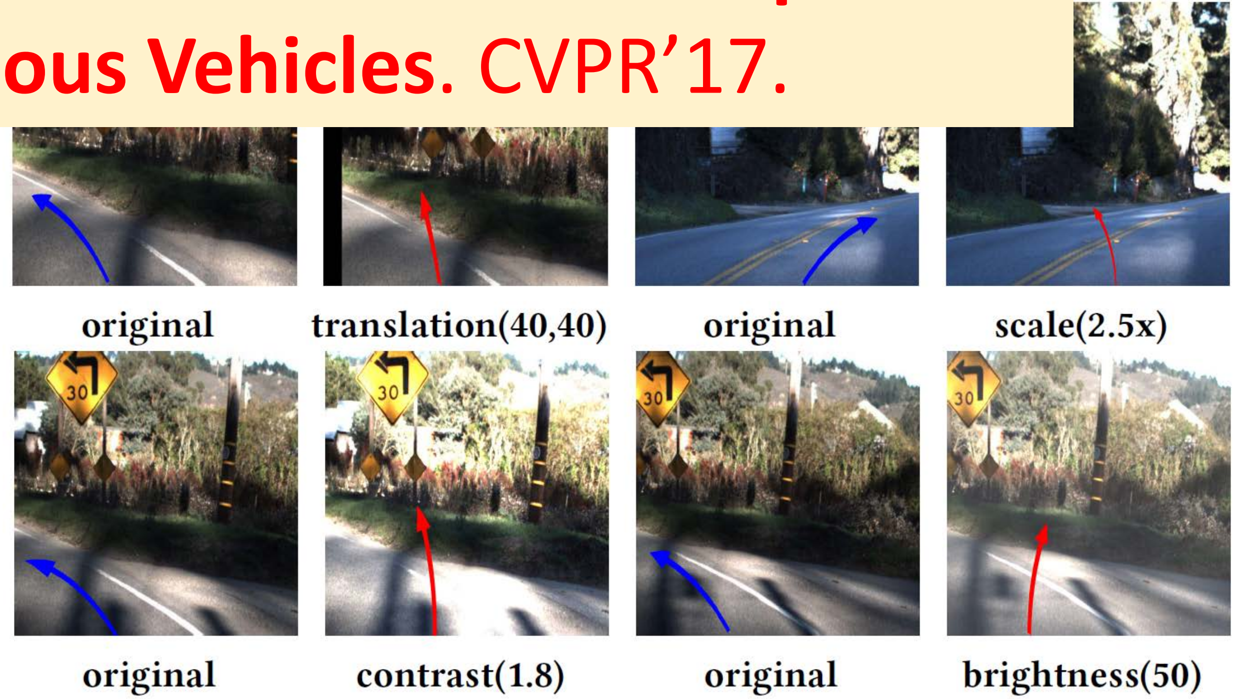


Lu et al. **NO Need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles.** CVPR’17.



original fog original rain

original shear(0.1) original rotation(6 degree)



original translation(40,40) original scale(2.5x)

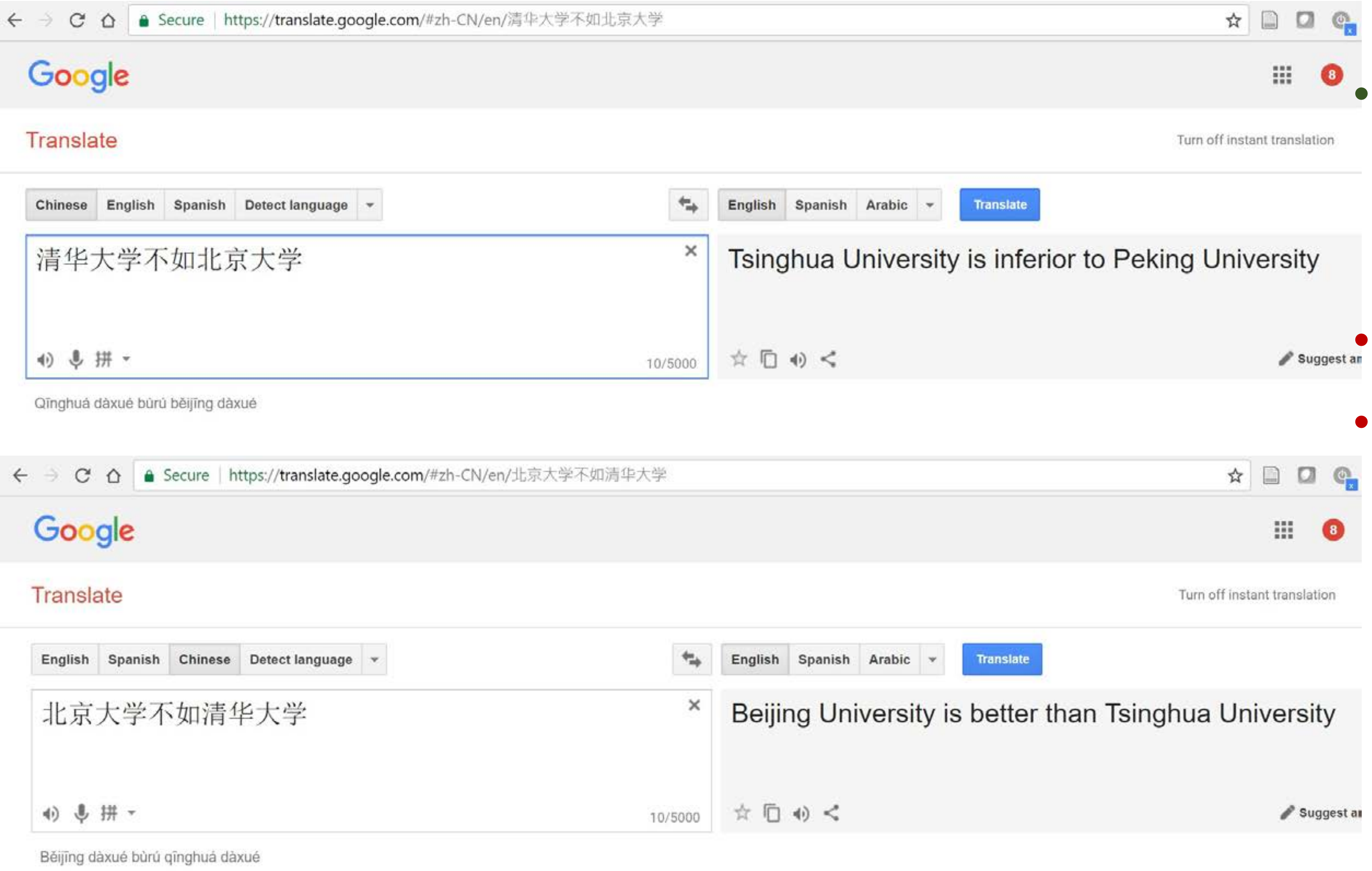
original contrast(1.8) original brightness(50)

Pei et al. DeepXplore: Automated Whitebox Testing of Deep Learning Systems. SOSP 2017.

Tian et al. DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars. ICSE 2018.

Slide adapted from SOSP’17 slides

Neural Machine Translation



https://translate.google.com/#zh-CN/en/清华大学不如北京大学

清华大学不如北京大学

Tsinghua University is inferior to Peking University

Qīnghuá dàxué bùrú běijīng dàxué

https://translate.google.com/#zh-CN/en/北京大学不如清华大学

北京大学不如清华大学

Beijing University is better than Tsinghua University

Běijīng dàxué bùrú qīnghuá dàxué

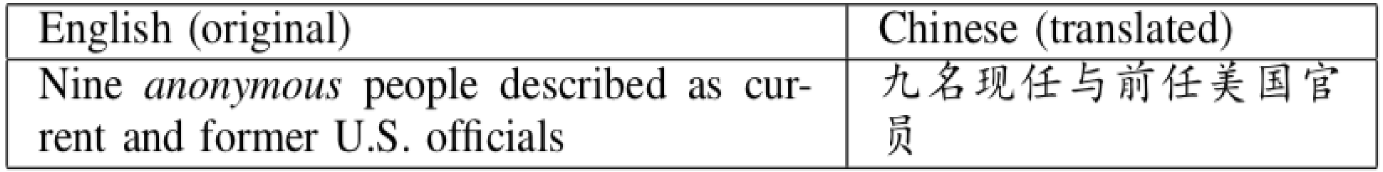
Screen snapshot captured on April 5, 2018

- Overall better than statistical machine translation

- Worse controllability
- Existing translation quality assurance
  - ➢ Need reference translation， not applicable online
  - ➢ Cannot precisely locate problem types and

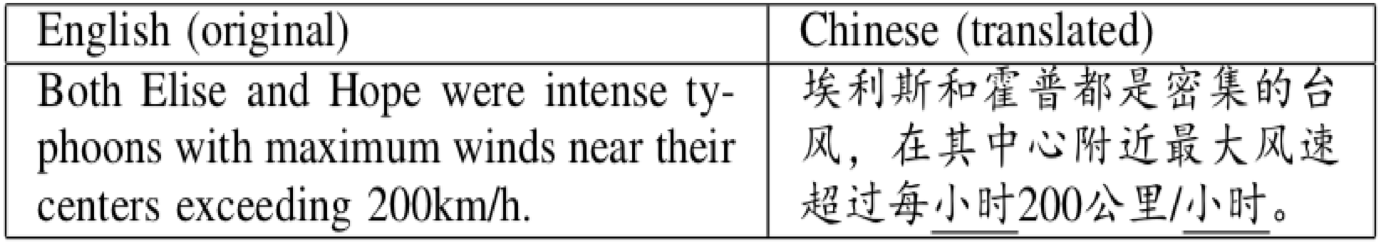
Translation Quality Assurance

- Key idea： black-box algorithms specialized for common problems
  - No need for reference translation; need only the original sentence and generated translation
  - Precise problem localization



English (original)	Chinese (translated)
Nine *anonymous* people described as current and former U.S. officials	九名现任与前任美国官员

- Common problems
  - Under-translation
  - Over-translation



English (original)	Chinese (translated)
Both Elise and Hope were intense typhoons with maximum winds near their centers exceeding 200km/h.	埃利斯和霍普都是密集的台风，在其中心附近最大风速超过每小时200公里/小时。

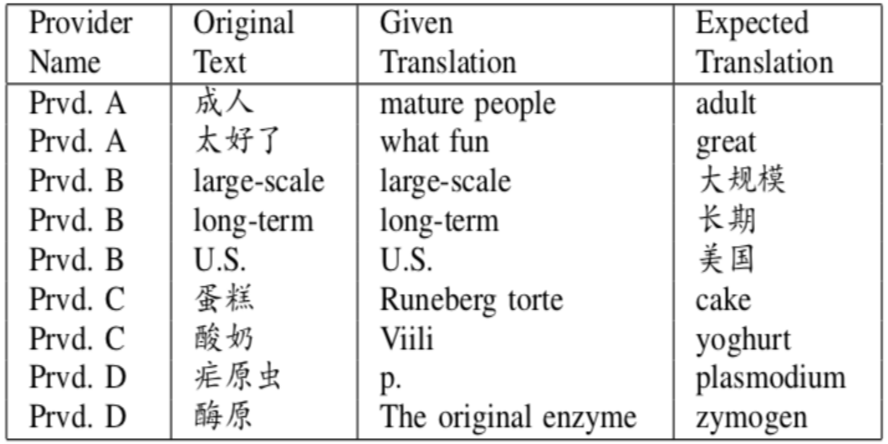
Tencent、UIUC Collaborative Work

Zheng, Wang, Liu, Zhang, Zeng, Deng, Yang, Xie. Oracle-free Detection of Translation Issue for Neural Machine Translation. arXiv:1807.02340, July 2018. https://arxiv.org/abs/1807.02340

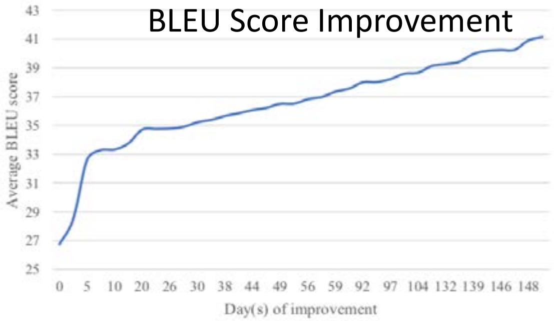
Industry Impact

- Adopted to improve WeChat translation service (over 1 billion users, online serving 12 million translation tasks)
  - Offline monitoring (regression testing)
  - Online monitoring (real time selection of best model)
- Large scale test data for translation
  - ~130K English/180K Chinese words/phrases
  - Detect numerous problems in Google Translate and YouDao

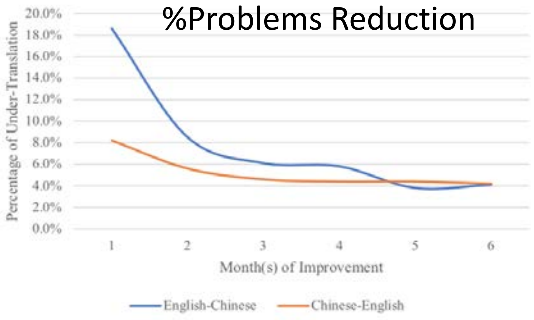
Problem Cases in Other Translation Services



Provider Name	Original Text	Given Translation	Expected Translation
Prvd. A	成人	mature people	adult
Prvd. A	太好了	what fun	great
Prvd. B	large-scale	large-scale	大规模
Prvd. B	long-term	long-term	长期
Prvd. B	U.S.	U.S.	美国
Prvd. C	蛋糕	Runeberg torte	cake
Prvd. C	酸奶	Viili	yoghurt
Prvd. D	疟原虫	p.	plasmodium
Prvd. D	酶原	The original enzyme	zymogen



BLEU Score Improvement



%Problems Reduction

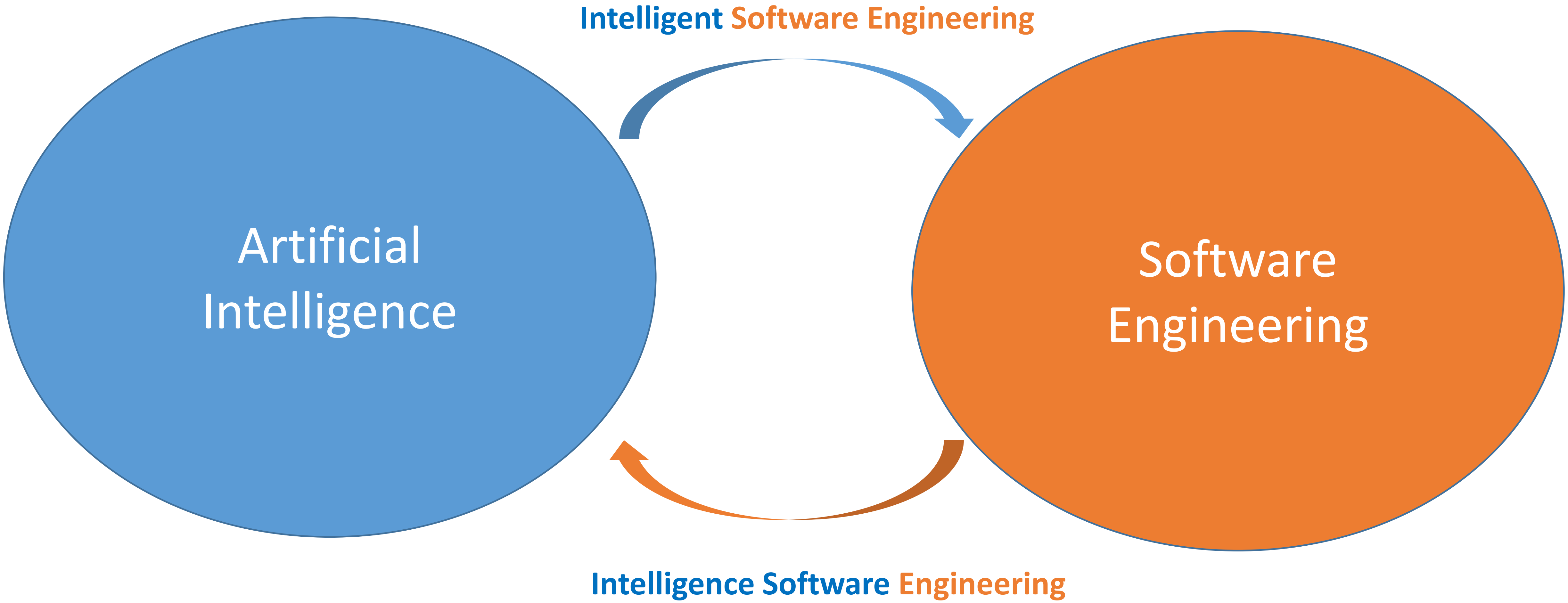
Tencent、UIUC Collaborative Work

Zheng, Wang, Liu, Zhang, Zeng, Deng, Yang, Xie. Oracle-free Detection of Translation Issue for Neural Machine Translation. arXiv:1807.02340, July 2018. https://arxiv.org/abs/1807.02340

Quite Many Recent Papers in SE for AI/ML

- Ma et al. MODE: Automated Neural Network Model Debugging via State Differential Analysis and Input Selection. ESEC/FSE’18
- Sun et al. Concolic Testing for Deep Neural Networks. ASE’18
- Udeshi et al. Automated Directed Fairness Testing. ASE’18
- Ma et al. DeepGauge: Multi-Granularity Testing Criteria for Deep Learning Systems. ASE’18
- Zhang et al. DeepRoad: GAN-based Metamorphic Testing and Input Validation Framework for Autonomous Driving Systems. ASE’18
- Dwarakanath et al. Identifying Implementation Bugs in Machine Learning based Image Classifiers using Metamorphic Testing. ISSTA’18
- Zhang et al. An Empirical Study on TensorFlow Program Bugs. ISSTA’18
- Tian et al. DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars. ICSE’18
- Abdessalem et al. Testing Vision-Based Control Systems Using Learnable Evolutionary Algorithms. ICSE’18
- Odena, Goodfellow. TensorFuzz: Debugging Neural Networks with Coverage-Guided Fuzzing. arXiv:1807.10875. 2018.
- …

Artificial Intelligence ← → Software Engineering



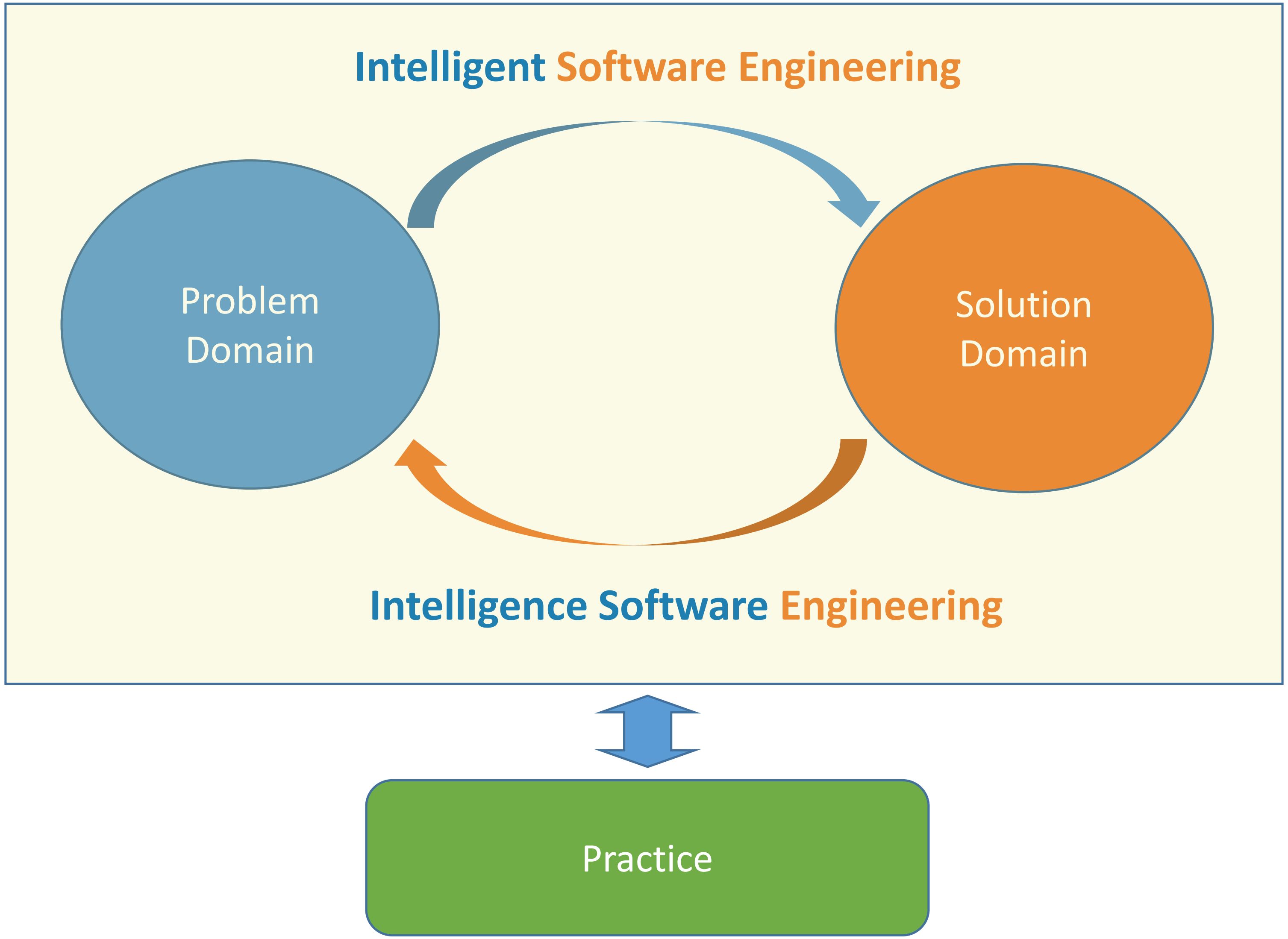
Intelligent Software Engineering

Artificial Intelligence

Software Engineering

Intelligence Software Engineering

(SE ←→ AI) → Practice Impact



Intelligent Software Engineering

Problem Domain

Solution Domain

Intelligence Software Engineering

Practice

Thank You!

Q & A

This work was supported in part by NSF under grants no. CNS-1513939, CNS-1564274, CCF-1816615, and a grant from the ZJUI Research Program.