Open Information Extraction

Matthias Huck and Alexander Fraser

Center for Information and Language Processing
LMU Munich

23 January 2019
Introduction

Reminder: *We know how to:*
- Recognize named entities in natural language text
- Extract binary relations between named entities

We have seen an application scenario:
- Relations can be stored in a knowledge base
- And be used in question answering or spoken dialogue systems

But so far, there are limitations, most notably:
- We have dealt with narrow domains (such as geographical location, food, plant seed development)
- The sets of entity types and relations were closed and manually defined

Open IE aims at:
- Not being limited to any single domain
- Not being limited to pre-defined entity types and relations
Outline

1. Open IE: Motivation & Task Definition
2. Open Relation Extraction: TEXTRUNNER & REVERB
3. Open Relation Extraction: OLLIE
4. Open Relation Extraction: STANFORD OPENIE
5. Discussion: Further Challenges
6. Conclusion
Open IE:
Motivation & Task Definition
Example Queries:
What kills bacteria?
Who built the Pyramids?
What did Thomas Edison invent?
What contains antioxidants?

Typed Example Queries:
What countries are located in Africa?
What actors starred in which films?
What is the symbol of which country?
What foods are grown in which countries?
What drug ingredients has the FDA approved?

[openie.allenai.org query, 16 Jan. 2017]
Open IE: Motivation (2)

62 answers from 584 sentences (cached)

all deceased person (7) monarch (5) location (3) ethnicity (3) misc.

Egyptians (132)

Ancient Egypt (123)

aliens (44)

the people (38)

slaves (29)
Open IE: Motivation (3)

41 answers from 275 sentences (cached)

Kenya (31)
Ghana (28)
Nigeria (16)
Egypt (15)
Morocco (11)
Algeria (10)

[openie.allenai.org query, 16 Jan. 2017]
Open IE: Motivation (4)

1 answers from 27 sentences (cached)

Elvis Presley (27)

[openie.allenai.org query, 16 Jan. 2017]
Open IE: Motivation (5)

7 answers from 93 sentences (cached)

- Bill Gates (64)
- Paul Allen (18)
- the man (3)
- Allen (2)
- Microsoft Bill Gates (2)

[openie.allenai.org query, 16 Jan. 2017]
Open IE: Motivation (6)

Argument 1: Scots
Argument 2: 
Relation: eat

2 answers from 4 sentences (cached)

Haggis (2)
Potato (2)

[openie.allenai.org query, 16 Jan. 2017]
6 answers from 55 sentences (cached)

sold his soul to (35)
makes a pact with (11)
makes a deal with (3)
strikes a deal with (2)
had sold his soul to (2)
asks (2)
Open IE: Motivation (8)

6 answers from 35 sentences (cached)

- Tent (20)
- Dormitory (7)
- classrooms (2)
- car (2)
- thatched-roof bandas (2)
- class (2)

[openie.allenai.org query, 16 Jan. 2017]
We want to find any relation that is expressed in large data.

- Cannot resort to specialized domain knowledge
- Cannot think of all possible relation types beforehand
- Should rather not force all possible arguments into a rigid set of entity types

We want to scale to billions of documents that are heterogeneous wrt. domains, quality, credibility.

- Which relations are correct?
- Which are uninformative or incoherent?
- Which are redundant?
Open Relation Extraction: TextRunner & ReVerb
Open Relation Extraction: Example

Hudson was born in Hampstead, which is a suburb of London.

(Hudson, was born in, Hampstead)
(Hampstead, is a suburb of, London)

[Fader et al.. Identifying Relations for Open Information Extraction. Proc. of EMNLP, Edinburgh, Scotland, UK, July 2011.]
Open Relation Extraction: Basic Approach

Learn a general model of how (arbitrary) relations are expressed in a particular language.

- Neither relation names nor argument types known in advance
- Bootstrap with heuristics or distant supervision
- Train a (sequence) classifier (often with unlexicalized features)

<table>
<thead>
<tr>
<th>Rel. Freq.</th>
<th>Category</th>
<th>Simplified Lexico-Syntactic Pattern</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>37.8</td>
<td>Verb</td>
<td>$E_1$ Verb $E_2$</td>
<td>X established Y</td>
</tr>
<tr>
<td>22.8</td>
<td>Noun+Prep</td>
<td>$E_1$ NP Prep $E_2$</td>
<td>X settlement with Y</td>
</tr>
<tr>
<td>16.0</td>
<td>Verb+Prep</td>
<td>$E_1$ Verb Prep $E_2$</td>
<td>X moved to Y</td>
</tr>
<tr>
<td>9.4</td>
<td>Infinitive</td>
<td>$E_1$ to Verb $E_2$</td>
<td>X plans to acquire Y</td>
</tr>
<tr>
<td>5.2</td>
<td>Modifier</td>
<td>$E_1$ Verb $E_2$ Noun</td>
<td>X is Y winner</td>
</tr>
<tr>
<td>1.8</td>
<td>Coordinate$_n$</td>
<td>$E_1$ (and</td>
<td>,</td>
</tr>
<tr>
<td>1.0</td>
<td>Coordinate$_v$</td>
<td>$E_1$ (and</td>
<td>,) $E_2$ Verb</td>
</tr>
<tr>
<td>0.8</td>
<td>Appositive</td>
<td>$E_1$ NP (:</td>
<td>,)? $E_2$</td>
</tr>
</tbody>
</table>

[Banko and Etzioni. The Tradeoffs Between Open and Traditional Relation Extraction. Proc. of the ACL, Columbus, OH, USA, June 2008.]
Open Relation Extraction: “Three-Step Method”

1. **Label**: Sentences are automatically labeled with extractions using heuristics or distant supervision.

2. **Learn**: A relation phrase extractor is learned, e.g. using a sequence-labeling graphical model (CRF).

3. **Extract**: The system takes a sentence as input, identifies a candidate pair of NP arguments \((\text{arg1}, \text{arg2})\) from the sentence, and then uses the learned extractor to label each word between the two arguments as part of the relation phrase or not.

Open Relation Extraction as Sequence Labeling

Kafka, a writer born in Prague, wrote "The Metamorphosis."
Uninformative Relations

Problem 1: The sequence classifier may come up with an uninformative relation name.

Faust made a deal with the devil.

(Faust, made, deal)
(Faust, made deal with, devil)

<table>
<thead>
<tr>
<th>Uninformative</th>
<th>Completion</th>
</tr>
</thead>
<tbody>
<tr>
<td>is</td>
<td>is an album by, is the author of, is a city in</td>
</tr>
<tr>
<td>has</td>
<td>has a population of, has a Ph.D. in</td>
</tr>
<tr>
<td>made</td>
<td>made a deal with, made a promise to</td>
</tr>
<tr>
<td>took</td>
<td>took place in, took control over, took advantage of</td>
</tr>
<tr>
<td>gave</td>
<td>gave birth to, gave a talk at, gave new meaning to</td>
</tr>
<tr>
<td>got</td>
<td>got tickets to, got a deal on, got funding from</td>
</tr>
</tbody>
</table>

[Fader et al.. Identifying Relations for Open Information Extraction. Proc. of EMNLP, Edinburgh, Scotland, UK, July 2011.]
Incoherent Relations

Problem 2: The sequence classifier may come up with an incoherent relation name.

<table>
<thead>
<tr>
<th>Sentence</th>
<th>Incoherent Relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>The guide contains dead links and omits sites.</td>
<td>contains omits</td>
</tr>
<tr>
<td>The Mark 14 was central to the torpedo scandal of the fleet.</td>
<td>was central torpedo</td>
</tr>
<tr>
<td>They recalled that Nungesser began his career as a precinct leader.</td>
<td>recalled began</td>
</tr>
</tbody>
</table>

[Fader et al.. Identifying Relations for Open Information Extraction. Proc. of EMNLP, Edinburgh, Scotland, UK, July 2011.]
POS-based Constraints to Avoid Incoherence & Uninformativeness

Extendicare agreed to buy Arbor Health Care for about US $432 million in cash and assumed debt.

(Arbor Health Care, for assumed, debt)

- POS-based regular expressions help avoid extraction of uninformative or incoherent relation phrases
- Manually written; e.g. the relation phrase must match:

<table>
<thead>
<tr>
<th>V</th>
<th>V P</th>
<th>V W* P</th>
</tr>
</thead>
<tbody>
<tr>
<td>V = verb particle? adv?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>W = (noun</td>
<td>adj</td>
<td>adv</td>
</tr>
<tr>
<td>P = (prep</td>
<td>particle</td>
<td>inf. marker)</td>
</tr>
</tbody>
</table>

- Choose longest possible match
- Require the relation phrase to appear between its arguments

Overspecific Relations & How to Avoid Them

Problem 3: Some relations are specific to an argument pair, or have only a few possible instances.

The Obama administration is offering only modest greenhouse gas reduction targets at the conference.

(Obama administration, is offering only modest greenhouse gas reduction targets at, conference)

- **Intuition**: a valid relation phrase should take many distinct arguments in a large corpus
- **Lexical constraint**: relation phrases are required to match at least $k$ distinct argument pairs in the data (e.g., $k = 20$)

[Fader et al.. Identifying Relations for Open Information Extraction. Proc. of EMNLP, Edinburgh, Scotland, UK, July 2011.]
Relation Phrase Normalization

Shakespeare (has written | wrote | was writing) Hamlet.

Allow for minor variations in relation phrases.

- Remove inflection
- Remove auxiliary verbs, adjectives, adverbs

[Fader et al.. Identifying Relations for Open Information Extraction. Proc. of EMNLP, Edinburgh, Scotland, UK, July 2011.]
Confidence Function

- Train classifier to assign a confidence score to each extraction
- Trade recall for precision by tuning a confidence threshold

<table>
<thead>
<tr>
<th>Weight</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.16</td>
<td>(x, r, y) covers all words in s</td>
</tr>
<tr>
<td>0.50</td>
<td>The last preposition in r is <em>for</em></td>
</tr>
<tr>
<td>0.49</td>
<td>The last preposition in r is <em>on</em></td>
</tr>
<tr>
<td>0.46</td>
<td>The last preposition in r is <em>of</em></td>
</tr>
<tr>
<td>0.43</td>
<td>len(s) ≤ 10 words</td>
</tr>
<tr>
<td>0.43</td>
<td>There is a <em>WH</em>-word to the left of r</td>
</tr>
<tr>
<td>0.42</td>
<td>r matches VW* P</td>
</tr>
<tr>
<td>0.39</td>
<td>The last preposition in r is <em>to</em></td>
</tr>
<tr>
<td>0.25</td>
<td>The last preposition in r is <em>in</em></td>
</tr>
<tr>
<td>0.23</td>
<td>10 words &lt; len(s) ≤ 20 words</td>
</tr>
<tr>
<td>0.21</td>
<td>s begins with x</td>
</tr>
<tr>
<td>0.16</td>
<td>y is a proper noun</td>
</tr>
<tr>
<td>0.01</td>
<td>x is a proper noun</td>
</tr>
<tr>
<td>-0.30</td>
<td>There is an NP to the left of x in s</td>
</tr>
<tr>
<td>-0.43</td>
<td>20 words &lt; len(s)</td>
</tr>
<tr>
<td>-0.61</td>
<td>r matches V</td>
</tr>
<tr>
<td>-0.65</td>
<td>There is a preposition to the left of x in s</td>
</tr>
<tr>
<td>-0.81</td>
<td>There is an NP to the right of y in s</td>
</tr>
<tr>
<td>-0.93</td>
<td>Coord. conjunction to the left of r in s</td>
</tr>
</tbody>
</table>

Open Relation Extraction: Ollie
OLLIE (Open Language Learning for Information Extraction)

Bootstrapping with high precision seed tuples from existing system (REVERB, cf. previous part)

[Mausam et al.. Open Language Learning for Information Extraction. Proc. of EMNLP, Jeju Island, Korea, July 2012.]
Employing Dependency Parses

I learned that the 2012 Sasquatch music festival is scheduled for May 25th until May 28th.

(the 2012 Sasquatch Music Festival, is scheduled for, May 25th)
Open Pattern Templates

Open pattern templates encode the ways in which a relation may be expressed in a sentence.

- Based on a dependency parse path
- with lexical constraint
- and POS constraint

<table>
<thead>
<tr>
<th>Extraction Template</th>
<th>Open Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. (arg1; be {rel} {prep}; arg2)</td>
<td>{arg1} ↑nsubjpass↑ {rel:postag=VBN} ↓{prep} ↓{arg2}</td>
</tr>
<tr>
<td>2. (arg1; {rel}; arg2)</td>
<td>{arg1} ↑nsubj ↑{rel:postag=VBD} ↓dobj ↓{arg2}</td>
</tr>
<tr>
<td>3. (arg1; be {rel} by; arg2)</td>
<td>{arg1} ↑nsubjpass↑ {rel:postag=VBN} ↓agent ↓{arg2}</td>
</tr>
<tr>
<td>4. (arg1; be {rel} of; arg2)</td>
<td>{rel:postag=NN;type=Person} ↑nn ↑{arg1} ↓nn ↓{arg2}</td>
</tr>
<tr>
<td>5. (arg1; be {rel} {prep}; arg2)</td>
<td>{arg1} ↑nsubjpass↑ {slot:postag=VBN;lex ∈ announce</td>
</tr>
</tbody>
</table>

[Mausam et al.. Open Language Learning for Information Extraction. Proc. of EMNLP, Jeju Island, Korea, July 2012.]
Previously (in REV), we required the relation phrase to appear between its arguments:

Elvis married Priscilla.

Open pattern templates may help with:

Elvis and Priscilla are married.

Other systems are designed to have verb-mediated relation phrases:

Bill Gates founded Microsoft.

OLLIE can deal with noun-mediated relations:

Bill Gates is founder of Microsoft.

Many relationships are most naturally expressed via noun phrases:

is capital of, is president of, is professor at, ... 

Dependency parse is useful; parsers not deemed too deemed too slow any more.

[Mausam et al.. Open Language Learning for Information Extraction. Proc. of EMNLP, Jeju Island, Korea, July 2012.]
OLLIE: Evaluation

Figure 5: Comparison of different Open IE systems.

We find that OLLIE achieves substantially larger area under the curve (73% of ReVerb) than other Open IE systems. OLLIE also misses very few extractions returned by ReVerb, mostly due to parser errors. As expected, WOE (with comparable precisions around 0.66) misses extractions found up to 146 times as many extractions than ReVerb and 4.8 times more than OLLIE. We perform further analysis to understand the reasons behind the high yield from OLLIE’s better performance compared to the two systems.

Open Relation Extraction: Stanford OpenIE
Heavily based on dependency parses.

1. Each dependency-parsed sentence is first split into a set of entailed clauses
2. Clauses are then maximally shortened, producing a set of entailed shorter sentence fragments
3. The fragments are segmented into relation triples, and output by the system

Born in a small town, she took the midnight train going anywhere.

**Stanford OpenIE: Illustration**

- She took the midnight train going anywhere
- She took the midnight train
- Born in a small town
- She Born in small town
- She Born in town
- (She; born in; small town)
- (She; born in; town)

[Angeli et al., Leveraging Linguistic Structure For Open Domain Information Extraction. Proc. of the ACL, Beijing, China, July 2015.]
Clause Splitting as a Classification Task

- Inspect the dependency structure
- Decide whether to split on a dependency arc
- Classifier using a set of dependency-based features
- Distant supervision for training: sequence which recovers a known relation is correct

<table>
<thead>
<tr>
<th>Feature Class</th>
<th>Feature Templates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge taken</td>
<td>{l, short_name(l)}</td>
</tr>
<tr>
<td>Last edge taken</td>
<td>{incoming_edge(p)}</td>
</tr>
<tr>
<td>Neighbors of parent</td>
<td>{nbr(p), (p, nbr(p))}</td>
</tr>
<tr>
<td>Grandchild edges</td>
<td>{out_edge(c), (e, out_edge(c))}</td>
</tr>
<tr>
<td>Grandchild count</td>
<td>{count (nbr(e_child)), (e, count (nbr(e_child)))}</td>
</tr>
<tr>
<td>Has subject/object</td>
<td>(\forall e \in {e, e_child} \forall l \in {subj, obj} 1(l \in nbr(e)))</td>
</tr>
<tr>
<td>POS tag signature</td>
<td>{pos(p), pos(c), (pos(p), pos(c))}</td>
</tr>
<tr>
<td>Features at root</td>
<td>{1(p = root), POS(p)}</td>
</tr>
</tbody>
</table>

## Atomic Patterns over Short Entailed Sentences

<table>
<thead>
<tr>
<th>Input</th>
<th>Extraction</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>cats play with yarn</em></td>
<td><em>(cats; play with; yarn)</em></td>
</tr>
<tr>
<td><em>fish like to swim</em></td>
<td><em>(fish; like to; swim)</em></td>
</tr>
<tr>
<td><em>cats have tails</em></td>
<td><em>(cats; have; tails)</em></td>
</tr>
<tr>
<td><em>cats are cute</em></td>
<td><em>(cats; are; cute)</em></td>
</tr>
<tr>
<td><em>Tom and Jerry are fighting</em></td>
<td><em>(Tom; fighting; Jerry)</em></td>
</tr>
<tr>
<td><em>There are cats with tails</em></td>
<td><em>(cats; have; tails)</em></td>
</tr>
</tbody>
</table>

### Verb-mediated:

<table>
<thead>
<tr>
<th>Input</th>
<th>Extraction</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Durin, son of Thorin</em></td>
<td><em>(Durin; is son of; Thorin)</em></td>
</tr>
<tr>
<td><em>Thorin’s son, Durin</em></td>
<td><em>(Thorin; ’s son; Durin)</em></td>
</tr>
<tr>
<td><em>IBM CEO Rometty</em></td>
<td><em>(Rometty; is CEO of; IBM)</em></td>
</tr>
<tr>
<td><em>President Obama</em></td>
<td><em>(Obama; is; President)</em></td>
</tr>
<tr>
<td><em>Fischer of Austria</em></td>
<td><em>(Fischer; is of; Austria)</em></td>
</tr>
<tr>
<td><em>IBM’s research group</em></td>
<td><em>(IBM; ’s; research group)</em></td>
</tr>
<tr>
<td><em>US president Obama</em></td>
<td><em>(Obama; president of; US)</em></td>
</tr>
<tr>
<td><em>Our president, Obama,</em></td>
<td><em>(Our president; be; Obama)</em></td>
</tr>
</tbody>
</table>

### Noun-mediated:

[Angeli et al.. Leveraging Linguistic Structure For Open Domain Information Extraction. Proc. of the ACL, Beijing, China, July 2015.]
Validating Deletions with Natural Logic

Scopes of operators *all, no, many, ...*

- All rabbits eat fresh vegetables yields (rabbits, eat, vegetables)
- All young rabbits drink milk does not yield (rabbits, drink, milk)

Non-subsective adjectives

- A *fake gun* is not a gun

Prepositional attachment

- Alice played baseball on Sunday entails Alice played on Sunday
- Obama signed the bill on Sunday should not entail Obama signed on Sunday

[Angeli et al.. Leveraging Linguistic Structure For Open Domain Information Extraction. Proc. of the ACL, Beijing, China, July 2015.]
## STANFORD OPENIE: Example Extractions

*Born in Honolulu, Hawaii, Obama is a US Citizen.*

<table>
<thead>
<tr>
<th>Our System</th>
<th>Ollie</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Obama; is; US citizen)</td>
<td>(Obama; is; a US citizen)</td>
</tr>
<tr>
<td>(Obama; born in; Honolulu, Hawaii)</td>
<td>(Obama; be born in; Honolulu)</td>
</tr>
<tr>
<td>(Obama; is; citizen of; US)</td>
<td>(Honolulu; be born in; Hawaii)</td>
</tr>
</tbody>
</table>

*Friends give true praise.*

*Enemies give fake praise.*

<table>
<thead>
<tr>
<th>Our System</th>
<th>Ollie</th>
</tr>
</thead>
<tbody>
<tr>
<td>(friends; give; true praise)</td>
<td>(friends; give; true praise)</td>
</tr>
<tr>
<td>(friends; give; praise)</td>
<td>(friends; give; praise)</td>
</tr>
<tr>
<td>(enemies; give; fake praise)</td>
<td>(enemies; give; fake praise)</td>
</tr>
</tbody>
</table>

*Heinz Fischer of Austria visits the US*

<table>
<thead>
<tr>
<th>Our System</th>
<th>Ollie</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Heinz Fischer; visits; US)</td>
<td>(Heinz Fischer of Austria; visits; the US)</td>
</tr>
</tbody>
</table>

Our System (without nominals) achieves both higher precision and extends its coverage further than Ollie. Furthermore, the PR curve for Our System is relatively smooth near the tail, suggesting that the system is leveraging features from higher-confidence extractions.}

We plot a precision/recall curve of our extractions in Figure 4 in order to get an informal sense of our approach's performance. For comparison, Ollie extracted 12 274 319 open IE triples, covering 1 180 770 relation types. While our system extracted 4 079 198 triples, covering 2 873 239 relation types, it is reasonable that the errors made in the clause splitter are the likely cause of these errors. Systematic errors in the clause splitter manifest across a range of sentences in the text, it is reasonable that the errors made in the clause splitter are the likely cause of these errors. Systematic errors in the clause splitter manifest across a range of sentences in the text, it is reasonable that the errors made in the clause splitter are the likely cause of these errors. Systematic errors in the clause splitter manifest across a range of sentences in the text, it is reasonable that the errors made in the clause splitter are the likely cause of these errors. Systematic errors in the clause splitter manifest across a range of sentences in the text, it is reasonable that the errors made in the clause splitter are the likely cause of these errors. Systematic errors in the clause splitter manifest across a range of sentences in the text, it is reasonable that the errors made in the clause splitter are the likely cause of these errors.
DISCUSSION:
FURTHER CHALLENGES
Synonym Resolution

The same entity may be referred to by a variety of names.

- *Michael Jackson; Jacko; The King of Pop; …*

The same fact may be expressed in a variety of ways.

- *IBM built Watson*
  *IBM created Watson*
  *IBM invented Watson*
  …

- *Dookie is a record by Green Day*
  *Dookie is an album by Green Day*
  …

RESOLVER identifies synonymous relations and objects


Disambiguation

The same string may refer to different entities (especially across different domains).

- *Watson*, the founder of IBM; *Watson*, the computer system
- *mouse*, the animal; *mouse*, the input device
- *1984*, the year; *1984*, the book
- *Paris*, France; *Paris*, Texas

Vagaries of Natural Language

- pronoun resolution
- metaphor
- anaphora
- complex or ungrammatical sentences
- irony, sarcasm
- ...

Incorrect Information

Nowadays referred to as “fake news”.

- *Elvis killed JFK*

Rate the reliability of an extracted relation.

- The relation extractor may have made an error: cf. the previously discussed confidence function
- Occurrence frequencies over the whole corpus can give an indication
- Credibility of the source of a document

YAGO-NAGA ranks facts $f$ via:

$$
\text{confidence}(f) = \max \left\{ \text{accuracy}(f, s) \times \text{trust}(s) \mid s \in \text{witnesses}(f) \right\}
$$


Temporal and Spatial Aspects

Time.

• Plato has not met with Tsipras

Space.

• An elephant does not fit into a coffee mug
• Trees don’t travel
• Somebody who pays in GBP is probably located in Britain
• Plato has never seen a kangaroo
Fact Consistency Checks

Avoid contradictory facts within the knowledge base.

- *Elvis died in 460 AD* cannot refer to *Elvis Presley* if we already know that Elvis Presley was born in 1935.
- born(X,Y) ∧ died(X,Z) ⇒ Y < Z
- appears(A,P,B) ∧ R(A,B) ⇒ expresses(P,R)
  appears(A,P,B) ∧ expresses(P,R) ⇒ R(A,B)
- means("Elvis",Elvis_Presley, 0.8)
  means("Elvis",Elvis_Costello, 0.2)

Implemented in the SOFIE IE system, which aims to extend the YAGO knowledge base

CONCLUSION
Summary: Open IE

- Discovering relations without a closed set of pre-defined relation types
- Open-domain
- Learning from the whole Web
- Distant supervision / bootstrapping to get started
- Attention to detail required to avoid pitfalls
- The system should benefit from the sheer size of the data
- It should learn more by itself when being run perpetually, and become more reliable
The End!

Thank you for your attention

Matthias Huck
mhuck@cis.lmu.de

27 answers from 129 sentences (cached)

feel free to ask (25)
here are (17)
have (16)
here are my answers to (7)
please feel free to ask (6)
Give your answer to (5)
go ahead with (5)