Transparent information extraction from natural language

Gábor Recski

TU Wien

gabor.recski@tuwien.ac.at

CSH seminar
06/22/2022
Information extraction (IE)

text → IE → structured data
IE example: Text classification

[9]
Citing high fuel prices, [ORG United Airlines] said [TIME Friday] it has increased fares by [MONEY $6] per round trip on flights to some cities also served by lower-cost carriers. [ORG American Airlines], a unit of [ORG AMR Corp.], immediately matched the move, spokesman [PER Tim Wagner] said. [ORG United], a unit of [ORG UAL Corp.], said the increase took effect [TIME Thursday] and applies to most routes where it competes against discount carriers, such as [LOC Chicago] to [LOC Dallas] and [LOC Denver] to [LOC San Francisco].

[4, Ch. 22]
IE example: Relation extraction

Billy Mays, the bearded, boisterous pitchman who, as the undisputed king of TV yell and sell, became an unlikely pop culture icon, died at his home in Tampa, Fla, on Sunday.

Pandit worked at the brokerage Morgan Stanley for about 11 years until 2005, when he and some Morgan Stanley colleagues quit and later founded the hedge fund Old Lane Partners.

He received an undergraduate degree from Morgan State University in 1950 and applied for admission to graduate school at the University of Maryland in College Park.
IE example: Slot filling

I would like to arrive Taipei on November 2\textsuperscript{nd}.

ticket booking system

Slot

\{ Destination: Taipei \\
   time of arrival: November 2\textsuperscript{nd} \}
IE approaches: machine learning

Process:

▶ annotate training data
▶ train model
▶ if solution is inadequate, blame the data and/or the user
▶ repeat (until funding runs out)

Why are ML models inadequate?

▶ learns patterns, not tasks
▶ learns artefacts/unintended bias [1, 3, 10, 13, 14, 7, 11]
▶ not configurable
▶ limited/false explainability [2, 11]
▶ black box (even for developers!)
IE approaches: machine learning

Process:

- annotate training data
- train model
- if solution is inadequate, blame the data and/or the user
- repeat (until funding runs out)

Why are ML models inadequate?

- learns patterns, not tasks
- learns artefacts/unintended bias [1, 3, 10, 13, 14, 7, 11]
- not configurable
- limited/false explainability [2, 11]
- black box (even for developers!)
IE approaches: machine learning

Process:
- annotate training data
- train model
- if solution is inadequate, blame the data and/or the user
- repeat (until funding runs out)

Why are ML models inadequate?
IE approaches: machine learning

Process:

▶ annotate training data
▶ train model
▶ if solution is inadequate, blame the data and/or the user
▶ repeat (until funding runs out)

Why are ML models inadequate?

▶ learns patterns, not tasks
▶ learns artefacts/unintended bias [1, 3, 10, 13, 14, 7, 11]
▶ not configurable
▶ limited/false explainability [2, 11]
▶ black box (even for developers!)
IE approaches: rule-based systems

- explainable, interpretable
- flexible, customizable
- reliable
- auditable

Why isn't everyone using them?

- they are, actually.
  - expensive to build and maintain/reconfigure
  - requires both domain and tech expertise
  - need to model variability of natural language for each task
IE approaches: rule-based systems

- explainable, interpretable
- flexible, customizable
- reliable
- auditable

Why isn’t everyone using them?
IE approaches: rule-based systems

▶ explainable, interpretable
▶ flexible, customizable
▶ reliable
▶ auditable

Why isn’t everyone using them?
▶ they are, actually...
▶ expensive to build and maintain/reconfigure
▶ requires both domain and tech expertise
▶ need to model variability of natural language for each task
Our approach: semantics-based IE

Step 1: map text to a generic intermediate (semantic) representation

- task-independent
- domain-independent
- syntax-independent
- language-independent*

Step 2: map semantic representation to task-specific structures

- as specific as necessary
- intuitive model of the task’s semantics
People involved

Kinga Gémes  Eszter Iklódi  Ádám Kovács  Gábor Recski

https://nlp.ec.tuwien.ac.at/people/
Our approach: semantics-based IE

Task 1 output  Task 2 output  Task 3 output

Task-specific components

Task 1 rules  Task 2 rules  Task 3 rules

Generic components

Semantic parsing and inference

Universal Dependency (UD) parsing

Task 1 input  Task 2 input  Task 3 input
The BRISE use-case
The BRISE use-case

Zoning plans (Flächenwidmungs- und Bebauungspläne)
The BRISE use-case

Textual regulations (Textliche Bestimmungen)


4.2. Auf den mit BB2 bezeichneten Grundflächen sind die zur Errichtung gelangenden Dächer entsprechend dem Stand der Technik als begrünte Flachdächer auszuführen.


4.6. Auf den mit BB7 bezeichneten Grundflächen ist in der Höhenlage der anschließenden Verkehrsfläche eine öffentliche Durchfahrt mit einer lichten Höhe von 3,0 m freizuhalten und zu dulden.
The BRISE rule extraction task

*Flachdächer bis zu einer Dachneigung von fünf Grad sind entsprechend dem Stand der technischen Wissenschaften zu begrünen.*

*Flat roofs with a pitch not exceeding 5 degrees must be greened using state of the art technologies.*

```
{"modality": "obligation",
 "attributes": [
  {"type": "content",
   "name": "BegrüenungDach",
   "value": true},

  {"type": "condition",
   "name": "Dachart",
   "value": "Flachdach"},

  {"type": "condition",
   "name": "DachneigungMax",
   "value": "5 Grad"}]
```
Semantics-based IE

Task1 output  Task2 output  Task3 output

Task-specific components

Task1 rules  Task2 rules  Task3 rules

Generic components

Semantic parsing and inference

Universal Dependency (UD) parsing

Task1 input  Task2 input  Task3 input
Extraction via graph patterns

Flachdächer bis zu einer Dachneigung von fünf Grad sind ... zu begrünen.
Flachdächer bis zu einer Dachneigung von fünf Grad sind ... zu begrünen.
Extraction via graph patterns

Flachdächer bis zu einer Dachneigung von fünf Grad sind ... zu begrünen.
Flachdächer bis zu einer Dachneigung von fünf Grad sind ... zu begrünen.
Die Gebäudehöhe darf höchstens 7,5 m betragen.
Die Gebäudehöhe darf höchstens 7,5 m betragen.
Die Gebäudehöhe darf höchstens 7,5 m betragen.
But how can we build such rules efficiently? And without much technical background?
pure Python library for building rule-based text classifiers over graphs

interactive UI for human-in-the-loop (HITL) learning of graph patterns

supports multiple linguistic graph formalisms (UD, AMR, stanza)

provides a REST-API to use extracted features for inference in production mode

open-source (MIT license), installable via pip
Rule chooser and modifier

First, choose class you want to use to build rules

GebaeudeHoeheMax

You can modify any rule you want to

Remember, we use the PENMAN notation to describe a rule. You can find more information about the rules in the README of our repository.

<table>
<thead>
<tr>
<th>rules</th>
<th>negated_rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>[u_123 / Gebaeudehoehe]</td>
<td></td>
</tr>
</tbody>
</table>

Graph viewer and evaluator

Browse graphs:

Choose from the rules

(u_123 / Gebaeudehoehe)

Result of using all the rules: Precision: 0.488, Recall: 0.824, Fscore: 0.613

The rule's result: Precision: 0.488, Recall: 0.824, Fscore: 0.613, True positives: 42, False positives: 44
HITL learning

Rule chooser and modifier

First, choose class you want to use to build rules

GebaeudeHoehemax

You can modify any rule you want to

Remember, we use the PENMAN notation to describe a rule. You can find more information about the rules in the README of our repository.

<table>
<thead>
<tr>
<th>rules</th>
<th>negated_rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>(u_123 / Gebaeudehoche)</td>
<td></td>
</tr>
<tr>
<td>(u_1 / .:* :1 (u_123 / Gebaeudehoche))</td>
<td></td>
</tr>
</tbody>
</table>

Graph viewer and evaluator

Browse graphs:

Choose from the rules

(u_1 / .:* :1 (u_123 / Gebaeudehoche))

Result of using all the rules: Precision: 0.808, Recall: 0.824, Fscore: 0.816

The rule's result: Precision: 0.808, Recall: 0.824, Fscore: 0.816, True positives: 42, False positives: 10
HITL learning

You can modify any rule you want to.

Remember, we use the PENMAN notation to describe a rule. You can find more information about the rules in the README of our repository.

Rules:

- (u_1 / *1 u_123 / Gebaeudehoehe)

After you modified any rule, click on the save updates button to save your changes.

Tick the box next to the rules you want to evaluate and then, click on the evaluate button. If you don’t tick any rule, the evaluator will evaluate all the rules.

Delete or refine selected rules (you can only refine underspecified rules that contain a regexp)

- delete
- refine

Result of using all the rules:
- Precision: 0.808
- Recall: 0.824
- Fscore: 0.816

The rule’s result:
- Precision: 0.808
- Recall: 0.824
- Fscore: 0.816
- True positives: 42
- False positives: 10

Select the graphs you want to view

- True Positive graphs

Tick the box next to the graphs you want to see. The rule that applied will be highlighted in the graph.

The penman format of the graph will also shown, you can copy any of the part directly from the penman format if you want to add a new rule.

<table>
<thead>
<tr>
<th>id</th>
<th>sentence</th>
</tr>
</thead>
</table>

---
HITL learning

<table>
<thead>
<tr>
<th>rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>(u_123 / Gebaeudehoehe)</td>
</tr>
<tr>
<td>(u_1 / .*:1 (u_123 / Gebaeudehoehe))</td>
</tr>
<tr>
<td>(u_1 / betragen</td>
</tr>
</tbody>
</table>

The rule's result: Precision: 0.977, Recall: 0.824, Fscore: 0.894, True positives: 42, False positives: 1
HITL learning

<table>
<thead>
<tr>
<th>rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐  (u_1 / betragen</td>
</tr>
<tr>
<td>☐  (u_1 / betragen</td>
</tr>
</tbody>
</table>

The rule's result: Precision: **1.000**, Recall: **0.412**, Fscore: **0.583**, True positives: **21**, False positives: **0**
HITL learning

...
HITL learning

Rule chooser and modifier

First, choose class you want to use to build rules

GebäudeHocheMax

You can modify any rule you want to

Remember, we use the PENMAN notation to describe a rule. You can find more information about the rules in the README of our repository.

Rules

- \( u_{-482} / \text{betragen} \cdot (u_{96} / m : 0 (u_{-744} / : u_{0} (u_{127} / \text{höchstens}): 1 (u_{123} / \text{Gebäudehoche})) \)
- \( u_{150} / \text{betragen} \cdot 1 (u_{123} / \text{Gebäudehoche}) : 0 (u_{127} / \text{höchstens}) \)
- \( u_{-889} / \text{überraschen} : 2 (u_{369} / \text{Hochohe}) : 1 (u_{987} / \text{Bauten} : 0 (u_{411} / \text{Gebäude})) \)

After you modified any rule, click on the save updates button to save your changes.

Tick the box next to the rules you want to evaluate and then, click on the evaluate button. If you don’t tick any rule, the evaluator will evaluate all the rules.

Delete or Refine selected rules (you can only refine underspecified rules that contain a scope)

- none
- delete
- refine

Graph viewer and evaluator

Choose from the rules

\( \{ u_{-482} / \text{betragen} : 2 (u_{96} / m : 0 (u_{-744} / : u_{0} (u_{127} / \text{höchstens})): 1 (u_{123} / \text{Gebäudehoche}) \} \)

Result of using all the rules: Precision: 0.975, Recall: 0.765, Fscore: 0.857

The rule's result: Precision: 1.000, Recall: 0.196, Fscore: 0.328, True positives: 10, False positives: 0
POTATO workflow

Dataset → POTATO backend
  - Generate graphs and subgraphs
  - Train and rank features
  - Evaluate features

Load graphs → POTATO UI
  - Manual rule modification
  - HITL supervised learning

Evaluation → Is it good enough?
  - YES → Inference mode
  - NO → Get automatic rule suggestions

Annotated Data? → Advanced mode
  - NO → Yes
  - YES → HITL supervised learning
Thank you!

Questions?

Code:
- https://github.com/recski/tuw-nlp
- https://github.com/adaamko/POTATO

Reading:
- On building semantic graphs [12]
References I


References II


## Mapping from syntax to semantics

<table>
<thead>
<tr>
<th>Dependency</th>
<th>Edge</th>
</tr>
</thead>
<tbody>
<tr>
<td>advcl, advmod, amod, ...</td>
<td>$w_1 \overset{0}{\rightarrow} w_2$</td>
</tr>
<tr>
<td>nsubj, csubj, ...</td>
<td>$w_1 \overset{1}{\Leftrightarrow} 0 \overset{0}{\rightarrow} w_2$</td>
</tr>
<tr>
<td>obj, ccomp, ...</td>
<td>$w_1 \overset{2}{\rightarrow} w_2$</td>
</tr>
<tr>
<td>nmod:poss</td>
<td>$w_2 \overset{1}{\leftarrow} HAS \overset{2}{\rightarrow} w_1$</td>
</tr>
<tr>
<td>*:tmod</td>
<td>$w_1 \overset{1}{\leftarrow} AT \overset{2}{\rightarrow} w_2$</td>
</tr>
</tbody>
</table>

... 

<table>
<thead>
<tr>
<th>Edge</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_1 \overset{obl}{\rightarrow} w_2 \overset{case}{\rightarrow} w_3$</td>
</tr>
<tr>
<td>$w_1 \overset{1}{\leftarrow} w_3 \overset{2}{\rightarrow} w_2$</td>
</tr>
<tr>
<td>$w_1 \overset{acl:relcl}{\rightarrow} w_2 \overset{nsubj}{\rightarrow} w_3$</td>
</tr>
<tr>
<td>$w_1 \overset{0}{\rightarrow} w_2$</td>
</tr>
</tbody>
</table>