Verbs as Predicates: Towards Inference in a Discourse

Zuzana Nevěřilová, Marek Grác

NLP Centre, Faculty of Informatics,
Masaryk University, Botanická 68a, 602 00 Brno, Czech Republic

2. prosince 2011
RASLAN 2011, Karlova Studánka

Logic & Language: Is there a relationship?

"at one extreme, logic is considered unnatural and irrelevant; at the opposite extreme, language is incurably vague and should be replaced by logic"

[Sowa, 2007]

Logic & Language: A broader definition

- logic = any precise notation for expressing statements that can be judged true or false
- inference = a truth-preserving transformation: when applied to a true statement, the result is guaranteed to be true

Logic & Language: A broader definition

- logic = any precise notation for expressing statements that can be judged true or false
- inference = a truth-preserving transformation: when applied to a true statement, the result is guaranteed to be true

Verbs as Predicates

- "the verb is a hook upon which the rest of a sentence hangs" [Schuler, 2005]
- syntactic valencies refer to syntactic properties (preposition, case)
- semantic properties refer to semantic roles: "A semantic role is the underlying relationship that a participant has with the main verb in a clause" [Loos et al., 2004]

Verbs as Predicates

- "the verb is a hook upon which the rest of a sentence hangs" [Schuler, 2005]
- syntactic valencies refer to syntactic properties (preposition, case)
- semantic properties refer to semantic roles: "A semantic role is the underlying relationship that a participant has with the main verb in a clause" [Loos et al., 2004]

Verbs as Predicates

- "the verb is a hook upon which the rest of a sentence hangs" [Schuler, 2005]
- syntactic valencies refer to syntactic properties (preposition, case)
- semantic properties refer to semantic roles: "A semantic role is the underlying relationship that a participant has with the main verb in a clause" [Loos et al., 2004]

Inference Rule

$$I = (Vinput, Voutput, n, \mathcal{S}, t)$$

- input verb, e.g. *dochutit* (*to flavour*)
- output verb, e.g. *chutnat* (*taste like*)
- gramatical polarity, e.g. positive→positive
- syntactic transformation
- inference type, e.g. effect

Inference Rule

$$I = (V input, V output, n, \mathcal{S}, t)$$

- input verb, e.g. *dochutit* (*to flavour*)
- output verb, e.g. *chutnat* (*taste like*)
- gramatical polarity, e.g. positive→positive
- syntactic transformation
- inference type, e.g. effect

Inference Rule

$I = (V input, V output, n, \mathcal{S}, t)$

- input verb, e.g. *dochutit* (*to flavour*)
- output verb, e.g. *chutnat* (*taste like*)
- gramatical polarity, e.g. positive$\rightarrow$positive
- syntactic transformation
- inference type, e.g. effect

Inference Rule

$$I = (V input, V output, n, \mathcal{S}, t)$$

- input verb, e.g. *dochutit* (*to flavour*)
- output verb, e.g. *chutnat* (*taste like*)
- gramatical polarity, e.g. positive→positive
- syntactic transformation
- inference type, e.g. effect

Inference Rule

$$I = (V input, V output, n, \mathcal{S}, t)$$

- input verb, e.g. *dochutit* (*to flavour*)
- output verb, e.g. *chutnat* (*taste like*)
- gramatical polarity, e.g. positive→positive
- syntactic transformation
- inference type, e.g. effect

Syntactic Transformation

for each NP or PP:

preposition p_i + case c_i $\rightarrow$ preposition p_j + case c_j

p_i and p_j can be ϵ

Syntactic Transformation

for each NP or PP:

preposition p_i + case c_i $\rightarrow$ preposition p_j + case c_j

p_i and p_j can be ϵ

dochutit (to flavour) maso (the meat) česnekem (with garlic)

$\rightarrow$

chutnat (to taste) maso (the meat) po česneku (like garlic)

Syntactic Transformation

for each NP or PP:

preposition p_i + case c_i $\rightarrow$ preposition p_j + case c_j

p_i and p_j can be ϵ

dochutit (to flavour) maso (the meat) česnekem (with garlic)

$\rightarrow$

chutnat (to taste) maso (the meat) po česneku (like garlic)

- ϵ + accusative $\rightarrow$ ϵ + nominative
- ϵ + instrumental $\rightarrow$ po + locative

Syntactic Transformation

for each NP or PP:

preposition p_i + case c_i $\rightarrow$ preposition p_j + case c_j

p_i and p_j can be ϵ

dochutit (to flavour) maso (the meat) česnekem (with garlic)

$\rightarrow$

chutnat (to taste) maso (the meat) po česneku (like garlic)

- ϵ + accusative $\rightarrow$ ϵ + nominative
- ϵ + instrumental $\rightarrow$ po + locative

Example

```
<title>'dochutit' has effect 'chutnat'</title>
<verbalex:inference type="effect" verb="dochutit">
  <verbalex:ruleset id="taste_like" inferred_verb="chutnat"
   negation="False">
    <verbalex:rule case="c4" prep="" inferred_case="c1"
     inferred_prep=""/>
    <verbalex:rule case="c7" prep="" inferred_case="c6"
     inferred_prep="po"/>
  </verbalex:ruleset>
</verbalex:inference>
```

Preprocessing

- sentence detection (SET)
- NP, PP, VP detection (SET)
- NP, PP, VP confirmation (manual)
- dependency detection (manual)

Preprocessing

- sentence detection (SET)
- NP, PP, VP detection (SET)
- NP, PP, VP confirmation (manual)
- dependency detection (manual)

Preprocessing

- sentence detection (SET)
- NP, PP, VP detection (SET)
- NP, PP, VP confirmation (manual)
- dependency detection (manual)

Preprocessing

- sentence detection (SET)
- NP, PP, VP detection (SET)
- NP, PP, VP confirmation (manual)
- dependency detection (manual)

The Algorithm

For the VP in sentence $Vinput \in Sinput$ find all inference rules that contain $Vinput$ as a input verb phrase.

For each $I = (Vinput, Voutput, n, \mathcal{S}, t)$:

1. find all dependents $D_1, \ldots, D_n$ of $Vinput$ in $Sinput$.
2. transform $Vinput$ to $Voutput$ (using majka [Šmerk, 2009]).
3. transform all possible dependents according to their corresponding rule $S = (SPinput_i, SPoutput_i)$ (using majka [Šmerk, 2009]).
4. generate a sentence $Soutput$ from $Voutput$ and transformed dependents.

The Algorithm

For the VP in sentence $Vinput \in Sinput$ find all inference rules that contain $Vinput$ as a input verb phrase.

For each $I = (Vinput, Voutput, n, \mathcal{S}, t)$:

1. find all dependents $D_1, \ldots, D_n$ of $Vinput$ in $Sinput$.
2. transform $Vinput$ to $Voutput$ (using majka [Šmerk, 2009]).
3. transform all possible dependents according to their corresponding rule $S = (SPinput_i, SPoutput_i)$ (using majka [Šmerk, 2009]).
4. generate a sentence $Soutput$ from $Voutput$ and transformed dependents.

The Algorithm

For the VP in sentence $Vinput \in Sinput$ find all inference rules that contain $Vinput$ as a input verb phrase.

For each $I = (Vinput, Voutput, n, \mathcal{S}, t)$:

1. find all dependents $D_1, \ldots, D_n$ of $Vinput$ in $Sinput$.
2. transform $Vinput$ to $Voutput$ (using majka [Šmerk, 2009]).
3. transform all possible dependents according to their corresponding rule $S = (SPinput_i, SPoutput_i)$ (using majka [Šmerk, 2009]).
4. generate a sentence $Soutput$ from $Voutput$ and transformed dependents.

The Algorithm

For the VP in sentence $Vinput \in Sinput$ find all inference rules that contain $Vinput$ as a input verb phrase.

For each $I = (Vinput, Voutput, n, \mathcal{S}, t)$:

1. find all dependents $D_1, \ldots, D_n$ of $Vinput$ in $Sinput$.
2. transform $Vinput$ to $Voutput$ (using majka [Šmerk, 2009]).
3. transform all possible dependents according to their corresponding rule $S = (SPinput_i, SPoutput_i)$ (using majka [Šmerk, 2009]).
4. generate a sentence $Soutput$ from $Voutput$ and transformed dependents.

Example Outputs

S input	*t*	*S output*
Nahrubo nastrouháme všechny sýry	precondition	na všechny sýry vezmeme struhadlo
velmi prudce opečeme z obou stran	equals	z obou stran orestujeme
najemno nasekáme zelenou papriku	precondition	na zelenou papriku vezmeme nůž
Očistíme ryby	equals	ryby zbavíme nečistot
Broskve oloupeme	equals	broskve zbavíme slupky
na orestovanou cibuli dáme žampiony	precondition	žampiony máme
podáváme s vinnou pěnou – šodó	equals	s vinnou pěnou servírujeme

Evaluation

cooking recipes corpus with 37 thousand tokens

Evaluation

cooking recipes corpus with 37 thousand tokens

174 verbs occuring in cooking recipes

Evaluation

cooking recipes corpus with 37 thousand tokens

174 verbs occuring in cooking recipes

232 inference rules

Evaluation

cooking recipes corpus with 37 thousand tokens

174 verbs occuring in cooking recipes

232 inference rules

- test on 2 400 sentences
- 822 VPs detected and confirmed correct
- 253 new sentences generated
- 135 sentences syntactically correct
- 118 sentences semantically correct

Evaluation

cooking recipes corpus with 37 thousand tokens

174 verbs occuring in cooking recipes

232 inference rules

- test on 2 400 sentences
- 822 VPs detected and confirmed correct
- 253 new sentences generated
- 135 sentences syntactically correct
- 118 sentences semantically correct

Evaluation

cooking recipes corpus with 37 thousand tokens

174 verbs occuring in cooking recipes

232 inference rules

- test on 2 400 sentences
- 822 VPs detected and confirmed correct
- 253 new sentences generated
- 135 sentences syntactically correct
- 118 sentences semantically correct

Evaluation

cooking recipes corpus with 37 thousand tokens

174 verbs occuring in cooking recipes

232 inference rules

- test on 2 400 sentences
- 822 VPs detected and confirmed correct
- 253 new sentences generated
- 135 sentences syntactically correct
- 118 sentences semantically correct

Evaluation

cooking recipes corpus with 37 thousand tokens

174 verbs occuring in cooking recipes

232 inference rules

- test on 2 400 sentences
- 822 VPs detected and confirmed correct
- 253 new sentences generated
- 135 sentences syntactically correct
- 118 sentences semantically correct

Conclusion: the problem lies in syntactic analysis

- unknown words
- parsing of coordinations
- NP/PP/VP boundaries detection
- parsing errors (e.g. interchanging nominative and accusative)

Conclusion: the problem lies in syntactic analysis

- unknown words
- parsing of coordinations
- NP/PP/VP boundaries detection
- parsing errors (e.g. interchanging nominative and accusative)

Conclusion: the problem lies in syntactic analysis

- unknown words
- parsing of coordinations
- NP/PP/VP boundaries detection
- parsing errors (e.g. interchanging nominative and accusative)

Conclusion: the problem lies in syntactic analysis

- unknown words
- parsing of coordinations
- NP/PP/VP boundaries detection
- parsing errors (e.g. interchanging nominative and accusative)

Conclusion: how to improve syntactic analysis?

domain-specific preprocessing

Conclusion: how to improve syntactic analysis?

domain-specific preprocessing

- named entities (e.g. crème fraîche)
- conjunction expansion (e.g. chop onion, stir and fry) → chop onion, stir onion and fry onion
- improbable forms (e.g. participles)
- most probable forms (e.g. 1st person plural, imperative)
- common errors as named entities (e.g. creme fraiche)

Conclusion: how to improve syntactic analysis?

domain-specific preprocessing

- named entities (e.g. crème fraîche)
- conjunction expansion (e.g. chop onion, stir and fry) $\rightarrow$ chop onion, stir onion and fry onion
- improbable forms (e.g. participles)
- most probable forms (e.g. 1st person plural, imperative)
- common errors as named entities (e.g. creme fraiche)

Conclusion: how to improve syntactic analysis?

domain-specific preprocessing

- named entities (e.g. crème fraîche)
- conjunction expansion (e.g. chop onion, stir and fry) → chop onion, stir onion and fry onion
- improbable forms (e.g. participles)
- most probable forms (e.g. 1st person plural, imperative)
- common errors as named entities (e.g. creme fraiche)

Conclusion: how to improve syntactic analysis?

domain-specific preprocessing

- named entities (e.g. crème fraîche)
- conjunction expansion (e.g. chop onion, stir and fry) → chop onion, stir onion and fry onion
- improbable forms (e.g. participles)
- most probable forms (e.g. 1st person plural, imperative)
- common errors as named entities (e.g. creme fraiche)

Conclusion: how to improve syntactic analysis?

domain-specific preprocessing

- named entities (e.g. crème fraîche)
- conjunction expansion (e.g. chop onion, stir and fry) → chop onion, stir onion and fry onion
- improbable forms (e.g. participles)
- most probable forms (e.g. 1st person plural, imperative)
- common errors as named entities (e.g. creme fraiche)

Conclusion: how to improve syntactic analysis?

in general:

- multiple domains
- several small preprocessing apps for each domain

Conclusion: how to improve syntactic analysis?

in general:

- multiple domains
- several small preprocessing apps for each domain

Future Work

- more rules
- objects introduced by rules
- modificators (adverbs)

Future Work

- more rules
- objects introduced by rules
- modificators (adverbs)

Future Work

- more rules
- objects introduced by rules
- modificators (adverbs)


Loos, E. E., Anderson, S., Dwight H., Day, J., Jordan, P. C., and Wingate, J. D. (2004).
Glossary of linguistic terms.
http://www.sil.org/linguistics/GlossaryOfLinguisticTerms/.


Schuler, K. K. (2005).
VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon.
PhD thesis, Faculties of the University of Pennsylvania.


Sowa, J. (2007).
Fads and fallacies about logic.
IEEE Intelligent Systems, page 84–87.


Šmerk, P. (2009).
Fast morphological analysis of czech.
In *Proceedings of the Raslan Workshop 2009*. Masarykova univerzita.