Combining Referring Expression Generation and Surface Realization: A Corpus-Based Investigation of Architectures

Sina Zarrieß    Jonas Kuhn

Institute for Natural Language Processing (IMS), University of Stuttgart (Germany)
Generating Coherent Text
Example: realizing a transitive relation and its referential arguments

Abstract Input:

Linguistic Output:

natural:
Two brothers are on trial because of an attack on a young man.

incoherent:
The masked robbers are on trial because two brothers attacked their victim.
Observed corpus data

... If you have been robbed, there are a lot of things ...
... Initial reports indicated that the robbery had happened inside ...
... decision to send a teenager who violently robbed and attacked an elderly woman ...
Observed corpus data

... If you have been robbed implicit_perp, there are a lot of things ...

... Initial reports indicated that the robbery implicit_perp implicit_victim had happened inside ...

... decision to send a teenager who violently robbed and attacked an elderly woman ...

Annotation

... rob

PERP VICTIM ...
Motivation for Combined Generation Task

Corpus-based Generation: Learning Choices From Data

**Observed corpus data**

... If you have been robbed implicit_perp, there are a lot of things...

... Initial reports indicated that the robbery implicit_perp implicit_victim had happened inside...

... decision to send a teenager who violently robbed and attacked an elderly woman ...

**Annotation**

```
... 
rob
```

**Modeling**

\[ v = \text{pronoun} + \text{passive} + p = \text{implicit} \]

\[ v = \text{implicit} + \text{nominalized} + p = \text{implicit} \]

\[ v = \text{indefinite} + \text{active} + p = \text{rel-pron} \]
Predicting Choices

Unseen Input

... because attack... because of an attack on the victim... because attacked they him... because has been attacked the victim by them... 

Ranking v=pronoun+nominalization+p=implicit v=pronoun+active+p=pronoun v=definite+passive+p=pronoun

Natural Output... because of an attack on the victim...
Predicting Choices

Candidates

Unseen Input

... because of an attack on implicit_perp him ...

... because attacked they him ...

... because has been attacked the victim by them ...
**Predicting Choices**

**Unseen Input**
- ... because of an attack on them.
- because attacked them
- because has been attacked by them.

**Candidates**

- ... because of
- an attack on
- implicit_perp
- him
- they
- attacked
- him
- the victim
- by them

**Ranking**

- \( v = \text{pronoun} + \text{nominalization} + p = \text{implicit} \)
- \( v = \text{pronoun} + \text{active} + p = \text{pronoun} \)
- \( v = \text{definite} + \text{passive} + p = \text{pronoun} \)
Motivation for Combined Generation Task
Corpus-based Generation Set-up
Experiments

Predicting Choices

Candidates

Unseen Input

because

attack

PERP VICTIM

Ranking

\[ v = \text{pronoun} + \text{nominalization} + p = \text{implicit} \]

\[ v = \text{pronoun} + \text{active} + p = \text{pronoun} \]

\[ v = \text{definite} + \text{passive} + p = \text{pronoun} \]

Natural Output

... because of an attack on the victim ...
Motivation for Combined Generation Task

Corpus-based Generation Set-up

- A New Data Set
- Generation Components

Experiments

- Evaluation of Standard Pipelines
- Addressing Error Propagation
General Background

- Corpus-based generation for empirical studies of linguistic variation (Langkilde and Knight, 1998; Ratnaparkhi, 2000; Ringger et al., 2004; Filippova and Strube, 2007; Cahill and Riester, 2009)

- Two well-studied paradigms:
  - **Surface realization (SR):** predict word order/syntactic realization on existing treebanks, shared task (Belz et al., 2011)
  - **Referring expression generation (REG):** predict pronominalization a.o. on texts with entity annotations, GREC shared tasks (Belz and Kow, 2010)
General Background

- Corpus-based generation for empirical studies of linguistic variation (Langkilde and Knight, 1998; Ratnaparkhi, 2000; Ringger et al., 2004; Filippova and Strube, 2007; Cahill and Riester, 2009)

- Two well-studied paradigms:
  - **Surface realization (SR):** predict word order/syntactic realization on existing treebanks, shared task (Belz et al., 2011)
  - **Referring expression generation (REG):** predict pronominalization a.o. on texts with entity annotations, GREC shared tasks (Belz and Kow, 2010)

But: SR and REG are often treated as isolated tasks
Choices Depend on Each Other

Standard set-up: surface realization is determined by referring expressions

- Likely: *attacked*

- Likely: *an attack on, was attacked*
Choices Depend on Each Other

Standard set-up: surface realization is determined by referring expressions.

- **Likely:** *attacked*
- **Likely:** *an attack on, was attacked*

If original REs and arguments are given, they reflect syntactic choices and their contexts to a large extent. (Zarrieß, Cahill and Kuhn, 2011;2012)
Combining Choices

- We propose a task that combines surface realization and referring expression generation.

- Challenge: Modeling interaction between choices.
Research Questions

- Standard approach for complex, multi-level generation problems
  - **Pipeline:** (Reiter and Dale, 1997)
    document planning → sentence-planning → surface realization
- Well-known problems
  - Error propagation
  - **Generation Gap:** Discourse-level decisions can depend on sentence-level decisions (Meteer, 1991)
- Place of REG in the pipeline has been a notorious problem
  (Cahill et al., 1999; Mellish et al., 2000)
Goal

- Corpus-based framework for investigating architectures and interactions between choices
- Integrate existing methods for annotation and data set creation (GREC and SR shared tasks)
  add more types of implicit referents
Generation Approach

- Combined SR and REG annotations
- Generator is based on three trainable modules
  - SYN: Deep → shallow dependencies
  - REG: insert RE candidates
  - LIN: linearize dependency tree
- Experimental parameters
  - order of modules
  - treatment of implicit referents
  - models of discourse context
A Data Set for Combined REG and SR

- Language: German
- Size: 200 texts, 2030 sentences
- Text type: newspaper articles about *robberies*
- Referents: frequent mentions of *victim, perpetrator*
- Syntax: unrestricted newspaper-style constructions, free word order

Available on my homepage: www.ims.uni-stuttgart.de/~zarriesa
Combining Annotations for REG and SR

Junge Familie auf dem Heimweg ausgeraubt
Young family robbed on the way home
Die Polizei sucht nach zwei Männern im Alter von 25 Jahren.
The Police looks for two 25-year-old men.
Sie sollen am Montag gegen 20 Uhr eine junge Familie überfallen haben
They are said to have attacked a young family on Monday around 20 o'clock.
Combining Annotations for REG and SR

**Manual RE annotation**

Junge Familie auf dem Heimweg ausgeraubt

Young family robbed on the way home

Die Polizei sucht nach zwei Männern im Alter von 25 Jahren.

The Police looks for two 25-year-old men.

Sie sollen am Montag gegen 20 Uhr eine junge Familie überfallen haben

They are said to have attacked a young family on Monday around 20 o’clock.

**Shallow dependency annotation**

\[
\begin{array}{c}
\text{robbed} \\
\text{subj} \\
\text{young family} \\
\hline
\text{look} \\
\text{subj} \\
\text{police} \\
\hline
\text{for} \\
\text{obj} \\
\text{two men} \\
\hline
\text{have} \\
\text{subj} \\
\text{they} \\
\hline
\text{aux} \\
\text{attack} \\
\text{obj} \\
\hline
\text{a young family}
\end{array}
\]

- Bohnet (2010)
  - dependency parser
- remove surface order of nodes
Combining Annotations for REG and SR

**Manual RE annotation**

*Junge Familie* auf dem *Heimweg* ausgeraubt

*Young family* robbed on the way home

Die Polizei sucht nach zwei Männern im Alter von 25 Jahren.

*The Police looks for two 25-year-old men.*

Sie sollen am Montag gegen 20 Uhr eine junge Familie überfallen haben

*They are said to have attacked a young family on Monday around 20 o’clock.*

**Shallow dependency annotation**

- Bohnet (2010) dependency parser
- remove surface order of nodes

- Bohnet (2010) dependency parser
- remove surface order of nodes
Combining Annotations for REG and SR

Manual RE annotation

Junge Familie auf dem Heimweg ausgeraubt
Young family robbed on the way home
Die Polizei sucht nach zwei Männern im Alter von 25 Jahren.
The Police looks for two 25-year-old men.
Sie sollen am Montag gegen 20 Uhr eine junge Familie überfallen haben
They are said to have attacked a young family on Monday around 20 o’clock.

Shallow dependencies with RE slots

robbed
subj
VICTIM
psubj
PERP
look
subj
police
for
obj
PERP
have
subj
PERP
aux
attack
|
obj
PERP
VICTIM

RE candidates

PERP: “two men”
| “they” | empty
VICTIM: “young family” | “they” | “a young family” | empty
Shallow dependencies with RE slots

- rule-based transformations for passives, nominalizations
- remove auxiliaries, map syntactic functions
Shallow dependencies with RE slots

- rule-based transformations for passives, nominalizations
- remove auxiliaries, map syntactic functions

Deep dependencies with RE slots
Motivation for Combined Generation Task

Corpus-based Generation Set-up

Experiments

A New Data Set

Generation Components

Shallow dependencies with RE slots

- robbed
  - subj: VICTIM
  - psubj: PERP
- look
  - subj: police
  - for: obj: PERP
- haben
  - subj: PERP
  - aux: attack
  - obj: PERP

- shallow verb candidates
- deep → shallow alignments

- rule-based transformations for passives, nominalizations
  - remove auxiliaries, map syntactic functions

Deep dependencies with RE slots

- rob
  - theme: VICTIM
  - agent: PERP
- search
  - agent: Polizei
  - theme: PERP
- attack
  - agent: PERP
  - theme: VICTIM

Zarrieß, Kuhn

Combining REG and SR: Architectures
**Generation Modules**

**REG**
(based on SVMrank, Joachims 2006)

rank RE candidates for a given slot in tree
(→ set of all original REs for a referent/text)

- `label=subj + RE=PRON head=attack + RE=PRON`
- `label=subj + RE=DEF head=attack + RE=DEF`
- `label=subj + RE=IMPL head=attack + RE=IMPL`

**SYN**
(based on SVMrank, Joachims 2006)

rank shallow candidates for a verb node in deep tree
(→ set of verb transformations/training set)

- `agent=victim + V=ACT theme=perp + V=ACT`
- `agent=victim + V=NOM theme=perp + V=NOM`
- `agent=victim + V=PASS theme=perp + V=PASS`

**LIN**
Bohnet et al. (2012), trained on German TIGER corpus
Ordering Modules: Pipelines

- deep tree + RE slots
  - REG:
  - ↓
  - deep tree + surface REs
    - SYN:
    - ↓
    - shallow tree + surface REs
      - LIN
      - ↓
      - shallow tree + surface REs
  - SYN:
  - ↓
  - deep tree + surface REs
    - REG:
    - ↓
    - shallow tree + surface REs
      - LIN
      - ↓
      - shallow tree + surface REs

Zarrieß, Kuhn
Combining REG and SR: Architectures
Ordering Modules: Pipelines

**deep tree + RE slots**

**REG:**
- label=agent + RE=PRONOUN
- label=agent + RE=DEFINITE

**deep tree + surface REs**

**SYN:**

**shallow tree + RE slots**

**REG:**
- label=subj + RE=PRONOUN
- label=subj + RE=DEFINITE

**shallow tree + surface REs**

**LIN**
Ordering Modules: Pipelines

- **deep tree + RE slots**
  - **REG:**
  - **SYN:**
    - `AGENT=VICTIM+shallow=finite`
    - `AGENT=VICTIM+shallow=nominal`

- **deep tree + surface REs**
  - **SYN:**
    - `AGENT=PRONOUN+shallow=finite`
    - `AGENT=PRONOUN+shallow=nominal`

- **shallow tree + RE slots**
  - **REG:**
    - **LIN**

- **shallow tree + surface REs**
First Results for Standard Pipelines

- 10-fold cross-validation on 200 texts
- sentences are lemmatized
- SYN/RE accuracy:
  proportion of exactly matching shallow/RE subtrees
- Baseline: rule-based heuristics for REG and SYN

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REG ... SYN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYN ... REG</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LIN on original shallow</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
First Results for Standard Pipelines

- Pipelines perform very similarly

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LIN on original shallow</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
First Results for Standard Pipelines

- SYN suffers from REG errors

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REG → SYN</td>
<td>54.65</td>
<td>57.09</td>
<td></td>
</tr>
<tr>
<td>SYN → REG</td>
<td>54.28</td>
<td>59.14</td>
<td></td>
</tr>
</tbody>
</table>

LIN on original shallow
First Results for Standard Pipelines

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Baseline</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
</tr>
<tr>
<td>LIN on original shallow</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

REG suffers from SYN errors
First Results for Standard Pipelines

- REG and SYN introduce considerable uncertainty

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>42.38</td>
<td>35.66</td>
<td>33.3</td>
</tr>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
</tr>
<tr>
<td>LIN on original shallow</td>
<td>79.17</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>
How can we address the effect of error propagation?

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
</tr>
</tbody>
</table>
Parallel system: Combine independent predictions of SYN and REG

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
</tr>
<tr>
<td>Parallel</td>
<td></td>
<td>59.14</td>
<td>54.61</td>
</tr>
</tbody>
</table>

Zarrieß, Kuhn - Combining REG and SR: Architectures
More Results

- Parallel system improves only marginally over pipelines

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
</tr>
<tr>
<td>Parallel</td>
<td>54.78</td>
<td>59.14</td>
<td>54.61</td>
</tr>
</tbody>
</table>
Towards Capturing Interactions: Revision-based Architecture

Intermediate linearization

many known interactions between REG and linearization
Centering Theory (Grosz et al., 1995)
approximates integrated architecture
  e.g. (Robin, 1993)

deep tree + RE slots
SYN
shallow tree + RE slots
LIN:
preliminary surface order of nodes
shallow tree + RE slots + order
REG:
additional positional features
shallow tree + REs + order
LIN

Zarrieß, Kuhn
Combining REG and SR: Architectures
## Final Results

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
</tr>
<tr>
<td>Parallel</td>
<td>54.78</td>
<td>59.14</td>
<td>54.61</td>
</tr>
<tr>
<td>Revision</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Final Results

- Revision-based system leads to clear overall improvements
- No error propagation from SYN to REG if shallow trees are linearized

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
</tr>
<tr>
<td>Parallel</td>
<td>54.78</td>
<td>59.14</td>
<td>54.61</td>
</tr>
<tr>
<td>Revision</td>
<td>56.31</td>
<td>59.14</td>
<td>58.81</td>
</tr>
</tbody>
</table>
Final Results

- Is the effect only due to higher lexical overlap because of better RE accuracy?

- **BLEU\textsubscript{r}**: replace REs by placeholder in gold and predicted sentence, BLEU score factoring out lexical RE overlap

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
<th>BLEU\textsubscript{r}</th>
</tr>
</thead>
<tbody>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
<td></td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
<td></td>
</tr>
<tr>
<td>Parallel</td>
<td>54.78</td>
<td>59.14</td>
<td>54.61</td>
<td></td>
</tr>
<tr>
<td>Revision</td>
<td>56.31</td>
<td>59.14</td>
<td>58.81</td>
<td></td>
</tr>
</tbody>
</table>
Final Results

- Improvements in $\text{BLEU}_r$ indicate generally better linearization/sentence quality

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>SYN Accuracy</th>
<th>RE Accuracy</th>
<th>$\text{BLEU}_r$</th>
</tr>
</thead>
<tbody>
<tr>
<td>REG ... SYN</td>
<td>54.65</td>
<td>57.09</td>
<td>54.61</td>
<td></td>
</tr>
<tr>
<td>SYN ... REG</td>
<td>54.28</td>
<td>59.14</td>
<td>52.21</td>
<td></td>
</tr>
<tr>
<td>Parallel</td>
<td>54.78</td>
<td>59.14</td>
<td>54.61</td>
<td>60.05</td>
</tr>
<tr>
<td>Revision</td>
<td>56.31</td>
<td>59.14</td>
<td>58.81</td>
<td>61.30</td>
</tr>
</tbody>
</table>
Conclusions

- Existing annotation standards for REG and SR can be easily integrated
- Data-driven study of generation architectures
- First results point to shortcomings of standard pipeline set-up
- Systems need to capture interactions (e.g. via revision)