Introduction to Shallow Parsing

Hoojung Chung
OUR Aim

Developing a practical shallow parser (for Bio-medical text)
  • Need to make a fast and accurate module

Study existing shallow parsing approaches
  • Based on Hand-crafted patterns
  • Based on Machine Learning techniques

Investigate related tasks, such as
  • Text chunking
  • Clause identification
Contents

Text Chunking
  • Definition
  • CoNLL 2000 Shared Task
  • History

Clause Identification
  • Definition
  • CoNLL 2001 Shared Task
  • History

Introduction to JMLR Special Issues on Shallow Parsing
  • Introduction
  • Overview of JMLR Special Issue papers
  • Conclusion
Text Chunking

- Dividing a text in syntactically correlated parts of words

  [NP He ] [VP reckons ] [NP the current account deficit ]
  [VP will narrow ] [PP to ] [NP only $1.8 billion ]
  [PP in ] [NP September ] .

- An intermediate step towards full parsing
- **Shared task of CoNLL–2000** (Computational Natural Language Learning)
  - Training and test data is available
  - Goal: make machine learn chunking test data
Text Chunking: CoNLL Data

He PRP B-NP
reckons VBZ B-VP
the DT B-NP
current JJ I-NP
account NN I-NP
deficit NN I-NP
will MD B-VP
narrow VB I-VP
to TO B-PP
only RB B-NP
# # I-NP
1.8 CD I-NP
billion CD I-NP
in IN B-PP
September NNP B-NP
. . O
## Text Chunking: CoNLL Result

<table>
<thead>
<tr>
<th></th>
<th>precision</th>
<th>recall</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ZDJ01]</td>
<td>94.29%</td>
<td>94.01%</td>
<td>94.13 (*)</td>
</tr>
<tr>
<td>[KM01]</td>
<td>93.89%</td>
<td>93.92%</td>
<td>93.91 (*)</td>
</tr>
<tr>
<td>[KM00]</td>
<td>93.45%</td>
<td>93.51%</td>
<td>93.48</td>
</tr>
<tr>
<td>[Hal00]</td>
<td>93.13%</td>
<td>93.51%</td>
<td>93.32</td>
</tr>
<tr>
<td>[TKS00]</td>
<td>94.04%</td>
<td>91.00%</td>
<td>92.50</td>
</tr>
<tr>
<td>[ZST00]</td>
<td>91.99%</td>
<td>92.25%</td>
<td>92.12</td>
</tr>
<tr>
<td>[Dej00]</td>
<td>91.87%</td>
<td>92.31%</td>
<td>92.09</td>
</tr>
<tr>
<td>[Koe00]</td>
<td>92.08%</td>
<td>91.86%</td>
<td>91.97</td>
</tr>
<tr>
<td>[Osb00]</td>
<td>91.65%</td>
<td>92.23%</td>
<td>91.94</td>
</tr>
<tr>
<td>[VB00]</td>
<td>91.05%</td>
<td>92.03%</td>
<td>91.54</td>
</tr>
<tr>
<td>[PMP00]</td>
<td>90.63%</td>
<td>89.65%</td>
<td>90.14</td>
</tr>
<tr>
<td>[Joh00]</td>
<td>86.24%</td>
<td>88.25%</td>
<td>87.23</td>
</tr>
<tr>
<td>[VD00]</td>
<td>88.82%</td>
<td>82.91%</td>
<td>85.76</td>
</tr>
<tr>
<td>baseline</td>
<td>72.58%</td>
<td>82.14%</td>
<td>77.07</td>
</tr>
</tbody>
</table>
# Text Chunking: History

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1991</td>
<td>[S. Abney] proposed to approach parsing by starting with <strong>finding correlated chunks of words</strong>.</td>
</tr>
<tr>
<td>1995</td>
<td>[L. Ramshaw and M. Marcus] have approached chunking by <strong>using a machine learning method</strong>. Classified every non-NP chunk as VP chunk.</td>
</tr>
<tr>
<td>2000</td>
<td>CoNLL–2000 shared task</td>
</tr>
<tr>
<td>2002</td>
<td>JMLR Special Issues (on Shallow Parsing)</td>
</tr>
</tbody>
</table>
Clause Identification: about

Clause
• Word sequences which contain a subject and a predicate
• Example
  (S The deregulation of railroads and trucking companies
   (SBAR that
     (S began in 1980)
   ) enabled
   (S shippers to bargain for transportation)
  )

CoNLL–2001 Shared task
• Training and test data is available
• Consists of 3 parts
  — Identifying clause start positions,
  — Recognizing clause end positions
  — Building complete clauses
Clause Identification: CoNLL Data

The DT B-NP S/X/{S*
deregulation NN I-NP X/X/*
of IN B-PP X/X/*
railroads NNS B-NP X/X/*
and CC O X/X/*
trucking NN B-NP X/X/*
companies NNS I-NP X/X/*
that WDT B-NP S/X/{S*
began VBD B-VP S/X/{S*
in IN B-PP X/X/*
1980 CD B-NP X/E/*S)S}
enabled VBD B-VP X/X/*
shippers NNS B-NP S/X/{S*
to TO B-VP X/X/*
bargain VB I-VP X/X/*
for IN B-PP X/X/*
transportation NN B-NP X/E/*S)
. . O X/E/*S)
## Clause Identification: CoNLL Result

<table>
<thead>
<tr>
<th>test</th>
<th>precision</th>
<th>recall</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>[CMPR02]</td>
<td>90.18%</td>
<td>72.59%</td>
<td>80.44</td>
</tr>
<tr>
<td>[CM01]</td>
<td>84.82%</td>
<td>73.28%</td>
<td>78.63</td>
</tr>
<tr>
<td>[MP01]</td>
<td>70.89%</td>
<td>65.57%</td>
<td>68.12</td>
</tr>
<tr>
<td>[TKS01]</td>
<td>76.91%</td>
<td>60.61%</td>
<td>67.79</td>
</tr>
<tr>
<td>[PG01]</td>
<td>73.75%</td>
<td>60.00%</td>
<td>66.17</td>
</tr>
<tr>
<td>[Dej01]</td>
<td>72.56%</td>
<td>54.55%</td>
<td>62.77</td>
</tr>
<tr>
<td>[Ham01]</td>
<td>55.81%</td>
<td>45.99%</td>
<td>50.42</td>
</tr>
</tbody>
</table>

| baseline | 98.44%    | 31.48% | 47.71 |
# Clause Identification: History

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
</table>
| 1990 | [S. Abney] used a clause filter as a part of his CASS parser  
- one part for recognizing basic clauses  
- one part for repairing difficult cases  
(clauses without subjects and clauses with additional VPs). |
| 1996 | [Eje96] showed that a parser can benefit from automatically identified clause boundaries in discourse |
| 2001 | CoNLL Shared Task |
JMLR Special Issue on Shallow Parsing
Not all NLP application requires a complete syntactic analysis

- Full parser often provides more information than needed (or less)

For instances,

- Information Retrieval:
  - finding simple NPs and VPs is enough
- Information Extraction, Summary Generation, Question Answering:
  - specific syntactico-semantic relations (agent, object, location, time, etc) is necessary
  - rather than elaborate configurational syntactic analyses
What is Shallow Parsing?

Shallow (or partial) Parsing...

- is the task of recovering only a limited amount of syntactic information from NL sentences
- has proved to be a useful technology for written and spoken language domains

Application Domain of Shallow Parsing

- Speech-to-Speech translation system
  - Used to add robustness (Verbmobil project: Wahlster, 2000)
- Question Answering on the WWW
  - Used to efficiently process large ill-formed doc.s (Buchhholz and Daelemans 2001, Srihari and Li, 1999)
- Text-mining Application
  - Used to biology text mining (Sekimizu et al., 1998)

Used to reduce the search space for full-blown ‘deep’ parser (Collins, 96)
What is Shallow Parsing?

Introduced in [Abney 1991]

- Argued the relevance of shallow parsing from the point of view of psycholinguistic evidence and practical applications
- Used hand-crafted cascaded FST (Finite State Transducers) – Abney 1996

We will cover this paper next week
Typical Shallow Parser Architecture

Sentences (Words & their context) → Part-of-Speech Tagging → Morpho-syntactic class of the word (noun, verb, …)

Chunking

Words that can be grouped as chunks (NP, VP, …)

Relation Finding

Relations they have with the main verb (subj, obj, loc, …)

I am a boy → [NP-SUBJ I/NNP] [VP am/VB] [NP-OBJ a/DT boy/NN]
How to Shallow-parse?

Use Rule Sets

- Because shallow parsers have to deal with entire NL, they need **thousands of rules**
  - Determiner are good predictors of NP
- Rule sets tend to be largely ‘soft’
  - … fatalities on non-interstate roads were **about the same**

Therefore, building shallow parser is a labor-intensive task

Unsurprisingly, shallow parsers are usually automatically built, using **Machine Learning** techniques
Machine Learning Approach for SP

Inspired by [Ramshaw and Marcus, 1995]

- Formulates NP-chunking as a tagging task
- Use I (for word inside an NP),
  O (for outside of an NP),
  B (for between the end of one and the start of another NP) tags

[ Some/B bankers/I ] [ are/O reporting/O ] [ more/I inquiries/I than/I usual/I ] [ about/B CDs/I ] [ since/B Friday/I ]

The method can be easily extended

- to tag other types of chunks
  — Skut and Brants 1998
- to find relations
  — Buchholz et al., 1999
Machine Learning Approach for SP

Applying ML to shallow parsing is not straightforward, because

• The amount of data to be processed will push batch systems to the limit
  ➔ need to scale

• Labeled training material is frequently noisy and exists in small quantity
  ➔ need to deal with overfitting

• Real world sentences tend to be long
  ➔ Learners which do not operate in (near) linear time are simply unfit for the task

Therefore, shallow parsing is a challenging domain for ML research
Overview of JMLR Special Issue papers

Tjong Kim Sang
• Memory based shallow parsing
  – Base NP identification, arbitrary base phrase recognition, clause detection, NP parsing, full parsing
• MBL + weighted majority voting, stacking

Molina and Pla
• HMM based shallow parsing
  – Chunking and Clause identification
• Not use ensemble learning methods, but acquire comparable results
Overview of JMLR Special Issue papers

Zhang et al.
- Text chunking based on generalized version of the Winnow algorithm
- Used very large set of features
- Achieved the best result for a non-ensemble classifier in the CoNLL–2000 shared task

Megyesi
- Retrain 3 POS taggers for shallow parsing Swedish
- Ignoring lexical information improved performance of the system (Because of the char. of Swedish? Unclear)
Overview of JMLR Special Issue papers

Dejean

- Present top-down rule induction system for learning linguistic structures
  - Deal with noisy data
  - Distinguish linguistically motivated exceptions from noise
- Increase efficiency by using prior knowledge
- Apply to CoNLL–2000 task (chunking)
Overview of JMLR Special Issue papers

Osborne

• Consider noisy and non-stationary training material
  – Use various type of artificially noisy material for experiment

• Draw various conclusion
  – Shallow parsers are robust (large quantities of noise will impair performance)
  – For better performance for other domain, annotate more for target distribution and use additional training material from other distribution
  – Best performers in the literature are not always the best at dealing with noise
  – Ensemble learning is not always a sure-fire strategy
Conclusion

Feature Selection is important
• Important consideration for ML of shallow parsers
• Some learner needs to carefully select(and weight) features, while others can cope with many irrelevant features

Ensemble Learning is recent trend
• Train several classifiers and combine their results
• (weighted) voting and stacked classifier
• Not guaranteed to produce the best result [Osborne]
Conclusion

The majority use probabilistic method
- Exception: MBL

All parsers assumed labeled input
- Exception: Zhang uses other knowledge sources, in addition to the training set

Shallow parsers are noise-tolerant
- Only massive quantities of noise will significantly undermine performance

Some parsers use generative model, some use discriminative model