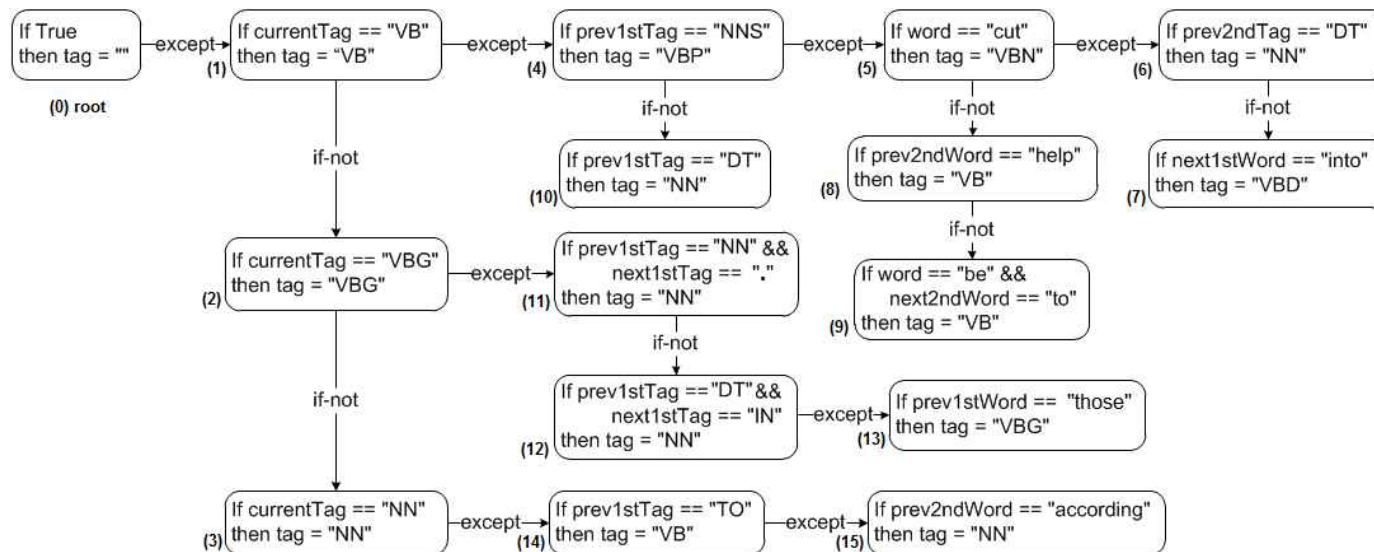


## TRANSFORMATION-BASED TAGGING – BEISPIEL RDRPOSTAGGER

Single Classification Ripple Down Rules (SCRDR): Korrektur von Bestandsregeln („except“, „if-not“);  
Ziel: schnelleres Training und Tagging als in Brills Ansatz („relabeling“)

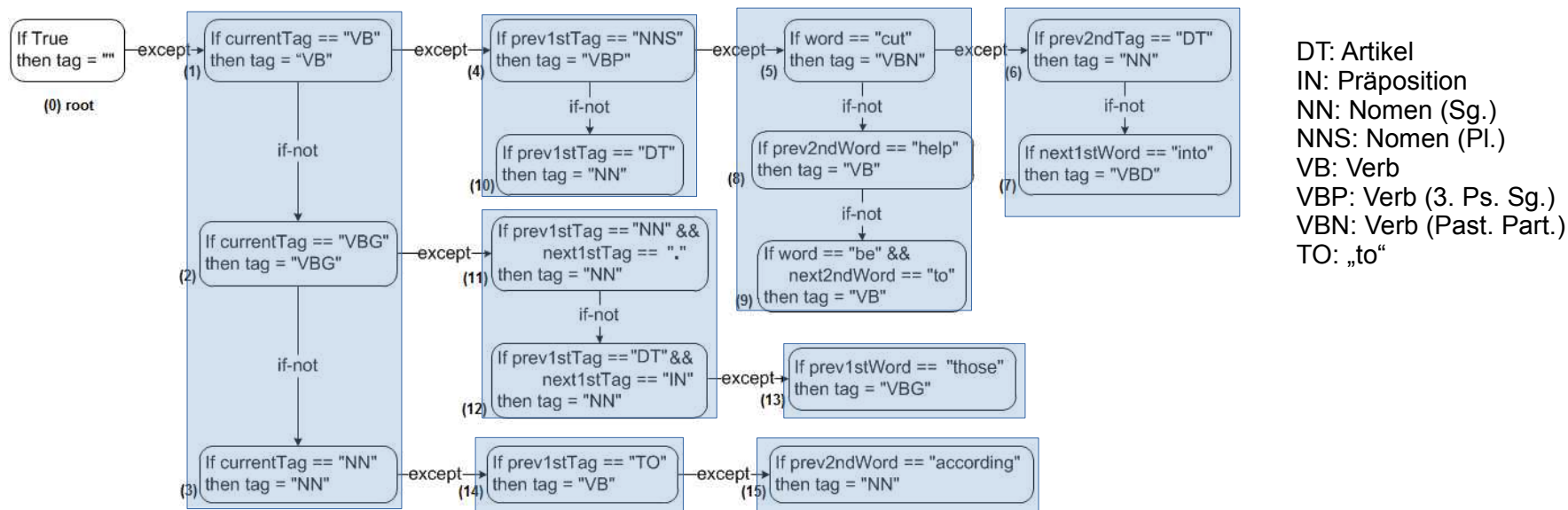


DT: Artikel  
IN: Präposition  
NN: Nomen (Sg.)  
NNS: Nomen (Pl.)  
VB: Verb  
VBP: Verb (3. Ps. Sg.)  
VBN: Verb (Past. Part.)  
TO: „to“

Dat Quoc Nguyen, Dai Quoc Nguyen, Dang Duc Pham, and Son Bao Pham. 2016. A robust transformation-based learning approach using ripple down rules for part-of-speech tagging. *AI Commun.* 29, 3 (2016), 409–422. <https://doi.org/10.3233/AIC-150698>

## TRANSFORMATION-BASED TAGGING – BEISPIEL RDRPOSTAGGER

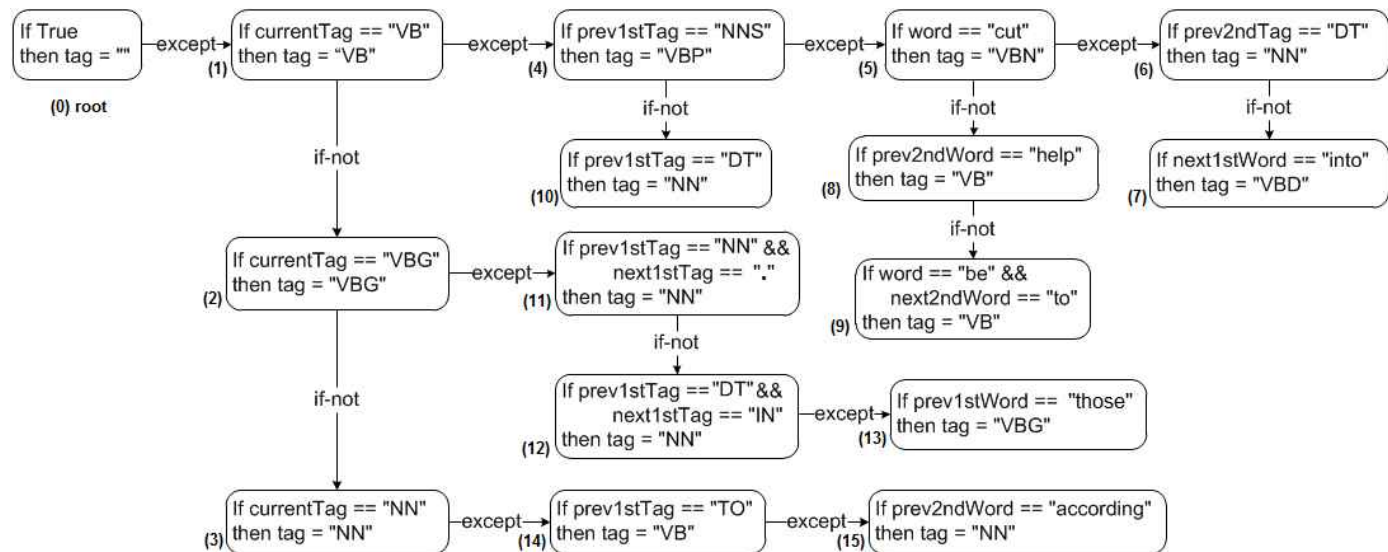
Single Classification Ripple Down Rules (SCRDR): Korrektur von Bestandsregeln („except“, „if-not“);  
Ziel: schnelleres Training und Tagging als in Brills Ansatz („relabeling“)



Dat Quoc Nguyen, Dai Quoc Nguyen, Dang Duc Pham, and Son Bao Pham. 2016. A robust transformation-based learning approach using ripple down rules for part-of-speech tagging. *AI Commun.* 29, 3 (2016), 409–422. <https://doi.org/10.3233/AIC-150698>

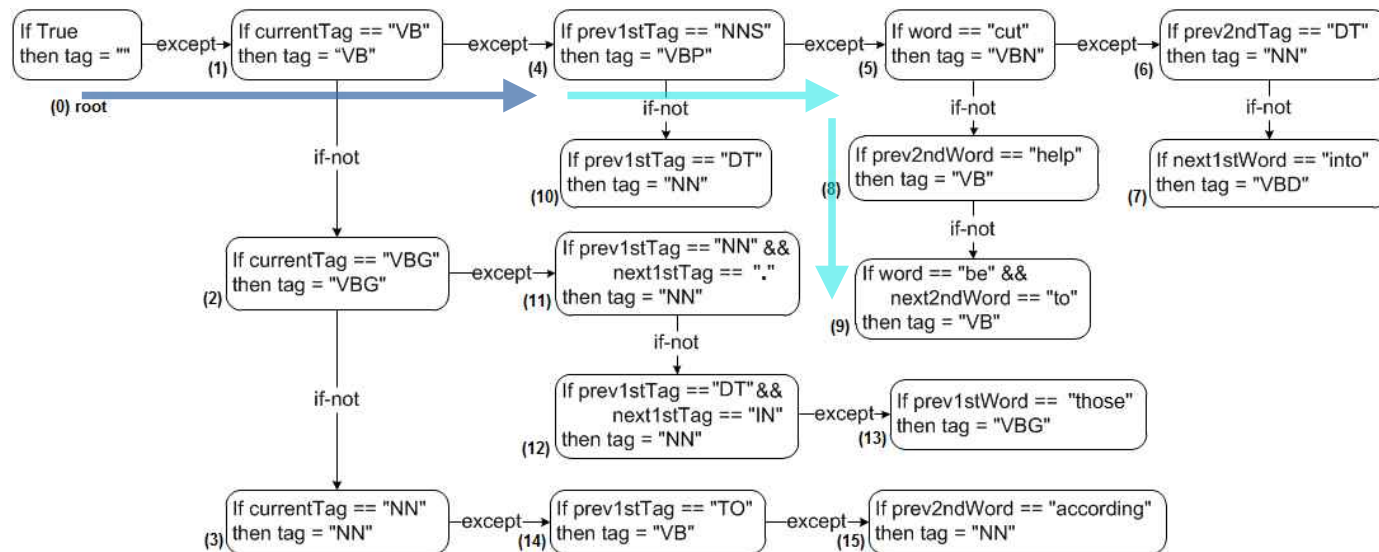
## TRANSFORMATION-BASED TAGGING – BEISPIEL RDRPOSTAGGER

- “as/IN investors/NNS anticipate/VB a/DT recovery/NN”



# TRANSFORMATION-BASED TAGGING – BEISPIEL RDRPOSTAGGER

- “as/IN investors/NNS anticipate/**VBP** a/DT recovery/NN”



## TRANSFORMATION-BASED TAGGING – BEISPIEL RDRPOSTAGGER

- Features für Regeln:
  - Wort (maximaler Abstand = 2), Wort-Bigramme, Wort-Trigramme
  - POS-Tags (maximaler Abstand = 2), POS-Tag-Bigramme
  - Suffix (Buchstaben-n-Gramme) (n=2-5)
  - (Kombinationen)
- Beispiel: “as/IN investors/NNS anticipate/**VBP** a/DT recovery/NN”
- Features für „anticipate/VBP“: ?

## TRANSFORMATION-BASED TAGGING – BEISPIEL RDRPOSTAGGER

- Features für Regeln:
  - Wort (maximaler Abstand = 2), Wort-Bigramme, Wort-Trigramme
  - POS-Tags (maximaler Abstand = 2), POS-Tag-Bigramme
  - Suffix (Buchstaben-n-Gramme) (n=2-5)
  - (Kombinationen)
- Beispiel: “as/IN investors/NNS anticipate/VBP a/DT recovery/NN”
- Features für „anticipate/VBP“:
  - Position 0: „anticipate“, „VBP“, „te“, „ate“, „pate“, „ipate“
  - Position -1: „investors“, „NNS“ // „investors“ „anticipate“ „NNS VBP“
  - „IN NSS“, „NNS VBP“, „investors anticipate“, „anticipate a“, usw.

## TRANSFORMATION-BASED TAGGING – BEISPIEL RDRPOSTAGGER

→ Regel-Templates

Template	Example
#2: <i>if</i> previous1 <sup>st</sup> Word == “ <b>object.previous1<sup>st</sup>Word</b> ” <i>then</i> tag = “ <b>correctTag</b> ”	(13)
#3: <i>if</i> word == “ <b>object.word</b> ” <i>then</i> tag = “ <b>correctTag</b> ”	(5)
#4: <i>if</i> next1 <sup>st</sup> Word == “ <b>object.next1<sup>st</sup>Word</b> ” <i>then</i> tag = “ <b>correctTag</b> ”	(7)
#10: <i>if</i> word == “ <b>object.word</b> ” && next2 <sup>nd</sup> Word == “ <b>object.next2<sup>nd</sup>Word</b> ” <i>then</i> tag = “ <b>correctTag</b> ”	(9)
#15: <i>if</i> previous1 <sup>st</sup> Tag == “ <b>object.previous1<sup>st</sup>Tag</b> ” <i>then</i> tag = “ <b>correctTag</b> ”	(4)
#20: <i>if</i> previous1 <sup>st</sup> Tag == “ <b>object.previous1<sup>st</sup>Tag</b> ” && next1 <sup>st</sup> Tag == “ <b>object.next1<sup>st</sup>Tag</b> ” <i>then</i> tag = “ <b>correctTag</b> ”	(11)

- Aufbau: inkrementelles Hinzufügen von Regeln zum Entscheidungsbaum bei Fehlklassifikation (except – if-not)
- Auswahl der konkreten Regel:
  - Regel darf nicht für bereits korrekt klassifizierte Vorkommen anwendbar sein
  - Score: korrekte - falsch klassifizierte Vorkommen (mit Threshold)

## TRANSFORMATION-BASED TAGGING – BEISPIEL RDRPOSTAGGER

- Falls keine Regel auswählbar: zurück Richtung Wurzel im „except“-Baum

