# Deterministic Incremental Dependency Parsing

Joakim Nivre, 2008

Jacob Louis Hoover

2021.04.16

#### Outline

- 1. Preliminaries
- 1.1 Dependency structures
- 1.2 Incremental dependency parsing
- 2. Stack-based Algorithms
- 2.1 Arc-standard stack-based algorithm
- 2.2 Arc-eager stack-based algorithm
- 3. List-based Algorithms
- 3.1 Non-projective list-based algorithm
- 3.2 Projective list-based algorithm
- 4. Experimental Evaluation

1 Preliminaries

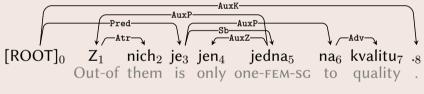
Incremental dependency parsing

#### Dependency structures:

#### **Examples**

■ A projective English dependency tree from the Penn Treebank (converted to dependency parse with Nivre's Penn2Malt).

■ A non-projective Czech dependency tree from the Prague Dependency Treebank.



("Only one of them concerns quality.")

1 Preliminaries

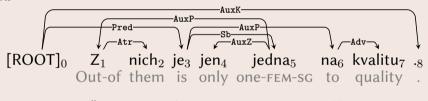
Incremental dependency parsing

#### Dependency structures:

#### **Examples**

■ A projective English dependency tree from the Penn Treebank (converted to dependency parse with Nivre's Penn2Malt).

■ A non-projective Czech dependency tree from the Prague Dependency Treebank.



("Only one of them concerns quality.")

# Formally

 $\blacksquare$  define dependency graph G = (V, A)

1 Preliminaries

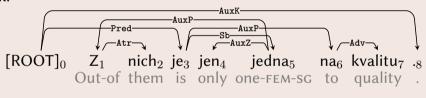
Incremental dependency parsing

#### Dependency structures:

#### **Examples**

■ A projective English dependency tree from the Penn Treebank (converted to dependency parse with Nivre's Penn2Malt).

■ A non-projective Czech dependency tree from the Prague Dependency Treebank.



("Only one of them concerns quality.")

- $\blacksquare$  define dependency graph G = (V, A)
  - $V = \{1, \dots, n\}$  set of nodes

1 Preliminaries

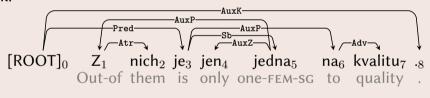
Incremental dependency parsing

#### Dependency structures:

#### **Examples**

■ A projective English dependency tree from the Penn Treebank (converted to dependency parse with Nivre's Penn2Malt).

■ A non-projective Czech dependency tree from the Prague Dependency Treebank.



("Only one of them concerns quality.")

- $\blacksquare$  define dependency graph G = (V, A)
  - $V = \{1, \dots, n\}$  set of nodes
  - $\blacksquare$   $A = V \times L \times V$  set of labeled directed arcs

1 Preliminaries

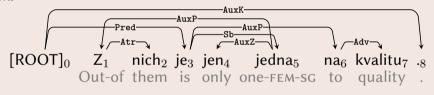
Incremental dependency parsing

#### Dependency structures:

#### **Examples**

■ A projective English dependency tree from the Penn Treebank (converted to dependency parse with Nivre's Penn2Malt).

■ A non-projective Czech dependency tree from the Prague Dependency Treebank.



("Only one of them concerns quality.")

- $\blacksquare$  define dependency graph G = (V, A)
  - $lacksquare V=\{1,\ldots,n\}$  set of nodes
  - $A = V \times L \times V$  set of labeled directed arcs
- define notion of well-formedness

1 Preliminaries

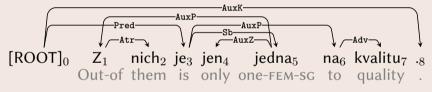
Incremental dependency parsing

#### Dependency structures:

#### **Examples**

■ A projective English dependency tree from the Penn Treebank (converted to dependency parse with Nivre's Penn2Malt).

■ A non-projective Czech dependency tree from the Prague Dependency Treebank.



("Only one of them concerns quality.")

- define dependency graph G = (V, A)
  - $V = \{1, \dots, n\}$  set of nodes
  - $\blacksquare$   $A = V \times L \times V$  set of labeled directed arcs
- define notion of well-formedness
  - by properties Root, SINGLE-HEAD, and ACYCLICITY

1 Preliminaries

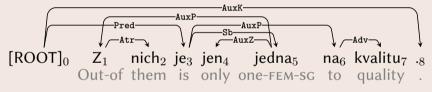
Incremental dependency parsing

#### Dependency structures:

#### **Examples**

■ A projective English dependency tree from the Penn Treebank (converted to dependency parse with Nivre's Penn2Malt).

■ A non-projective Czech dependency tree from the Prague Dependency Treebank.



("Only one of them concerns quality.")

- define dependency graph G = (V, A)
  - $V = \{1, \dots, n\}$  set of nodes
  - $\blacksquare$   $A = V \times L \times V$  set of labeled directed arcs
- define notion of well-formedness
  - by properties Root, Single-Head, and Acyclicity
- define property projectivity

1 Preliminaries

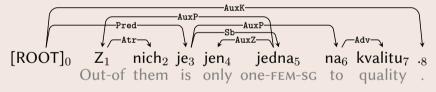
Incremental dependency parsing

#### Dependency structures:

#### **Examples**

■ A projective English dependency tree from the Penn Treebank (converted to dependency parse with Nivre's Penn2Malt).

■ A non-projective Czech dependency tree from the Prague Dependency Treebank.



("Only one of them concerns quality.")

- define dependency graph G = (V, A)
  - $V = \{1, \ldots, n\}$  set of nodes
  - $\blacksquare$   $A = V \times L \times V$  set of labeled directed arcs
- define notion of well-formedness
  - by properties Root, Single-Head, and Acyclicity
- define property projectivity
  - there exists a path from the head of an arc to any node inside the span of the arc

Incremental dependency parsing

# Common formalization incremental dependency parsing algorithms:

■ transition system  $S = (C, T, c_s, C_t)$ 

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - $\blacksquare$  C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - $lackbox{ } C$  a set of configurations, each containing a buffer eta of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare T$  a set of transitions  $t: C \to C$ ,

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - lacksquare C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare T$  a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x=(w_0,w_1,\ldots,w_n)$  to a configuration with  $\beta=[1,\ldots,n]$ ,

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - lacksquare C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare T$  a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x=(w_0,w_1,\ldots,w_n)$  to a configuration with  $\beta=[1,\ldots,n]$ ,
  - lacksquare  $C_{ ext{terminal}} \subseteq C$  a set of terminal configurations.

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - lacksquare C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare T$  a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \dots, n]$ ,
  - $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - lacksquare C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare T$  a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \dots, n]$ ,
  - $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,
- incrementality

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - $lue{C}$  a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare$  T a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \ldots, n]$ ,
  - lacksquare  $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,
- incrementality
  - buffer always decreasing in size, and derivation is over when empty

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - lacksquare C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare T$  a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \dots, n]$ ,
  - $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,
- incrementality
  - buffer always decreasing in size, and derivation is over when empty
  - once an arc is added to A, it is never removed

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - lacksquare C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare$  T a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \dots, n]$ ,
  - $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,
- incrementality
  - buffer always decreasing in size, and derivation is over when empty
  - once an arc is added to A, it is never removed
- correctness of an algorithm (soundness and completeness) for a class of dependency graphs

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare$  T a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \dots, n]$ ,
  - $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,
- incrementality
  - buffer always decreasing in size, and derivation is over when empty
  - once an arc is added to A, it is never removed
- correctness of an algorithm (soundness and completeness) for a class of dependency graphs
- an oracle

1 Preliminaries

Incremental dependency parsing

- transition system  $S = (C, T, c_s, C_t)$ 
  - C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare$  T a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \dots, n]$ ,
  - $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,
- incrementality
  - buffer always decreasing in size, and derivation is over when empty
  - once an arc is added to A, it is never removed
- correctness of an algorithm (soundness and completeness) for a class of dependency graphs
- an oracle
  - for a transition system S, an oracle is a function  $o: C \to T$  specifying which transition to take for any given configuration

Incremental dependency parsing

# Common formalization incremental dependency parsing algorithms:

- transition system  $S = (C, T, c_s, C_t)$ 
  - C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare$  T a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \dots, n]$ ,
  - $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,
- incrementality
  - buffer always decreasing in size, and derivation is over when empty
  - once an arc is added to A, it is never removed
- correctness of an algorithm (soundness and completeness) for a class of dependency graphs
- an oracle
  - for a transition system S, an oracle is a function  $o: C \to T$  specifying which transition to take for any given configuration

# $\mathsf{Parse}(x = (w_0, \dots, w_n))$

- 1:  $c \leftarrow c_s(x)$
- 2: while  $c \notin C_t$  do
- $c \leftarrow [o(c)](c)$
- 4: end while
- 5: return  $G_c$

Incremental dependency parsing

# Common formalization incremental dependency parsing algorithms:

- transition system  $S = (C, T, c_s, C_t)$ 
  - C a set of configurations, each containing a buffer  $\beta$  of (remaining) nodes and a set A of dependency arcs,
  - $\blacksquare$  T a set of transitions  $t: C \to C$ ,
  - $c_{\text{start}}$  an initialization function, mapping a sentence  $x = (w_0, w_1, \dots, w_n)$  to a configuration with  $\beta = [1, \dots, n]$ ,
  - $C_{\text{terminal}} \subseteq C$  a set of terminal configurations.
- transition sequence (derivation): a series of transitions through configurations, starting in a start configuration,
- incrementality
  - buffer always decreasing in size, and derivation is over when empty
  - once an arc is added to A, it is never removed
- correctness of an algorithm (soundness and completeness) for a class of dependency graphs
- an oracle
  - for a transition system S, an oracle is a function  $o: C \to T$  specifying which transition to take for any given configuration

# $\mathsf{Parse}(x = (w_0, \dots, w_n))$

- 1:  $c \leftarrow c_s(x)$
- 2: while  $c \notin C_t$  do
- $c \leftarrow [o(c)](c)$
- 4: end while
- 5: return  $G_c$
- Stack-based Algorithms (for projective structures)
  - arc-standard
  - arc-eager
- List-based Algorithms
  - non-projective
  - projective

Arc-eager stack-based algorithm

#### **Stack-based Algorithms**

#### Definition

A stack-based configuration for a sentence  $x = (w_0, w_1, \dots, w_n)$  is a triple  $c = (\sigma, \beta, A)$ , where

- 1.  $\sigma$  is a stack of tokens  $i \leq k$  (for some  $k \leq n$ )  $\leftarrow$  will represent as a list with head to right
- 2.  $\beta$  is a buffer of tokens j > k,  $\leftarrow$  will represent as a list with head to left
- 3. A is a set of dependency arcs such that  $G=(\{0,1,\ldots,n\},A)$  is a dependency graph for x.

#### Definition

A stack-based transition system is a quadruple  $S = (C, T, c_{\text{start}}, C_{\text{terminal}})$ , where

- 1. C is the set of all stack-based configurations,
- 2.  $c_{\text{start}}(x = (w_0, w_1, \dots, w_n)) = ([0], [1, \dots, n], \varnothing),$
- 3. T is a set of transitions, each of which is a function  $t: C \to C$ ,
- 4.  $C_{\text{terminal}} = \{c \in C \mid c = (\sigma, [], A)\}.$

# Transitions Left-Arc<sub>l</sub> $(\sigma|i,j|\beta,A) \Rightarrow (\sigma,j|\beta,A \cup \{(j,l,i)\})$ Right-Arc<sub>l</sub> $(\sigma|i,j|\beta,A) \Rightarrow (\sigma,i|\beta,A \cup \{(i,l,j)\})$ Shift $(\sigma,i|\beta,A) \Rightarrow (\sigma|i,\beta,A)$ Preconditions Left-Arc<sub>l</sub> $\neg[i=0]$ $\neg\exists k\exists l'[(k,l',i) \in A]$ Right-Arc<sub>l</sub> $\neg\exists k\exists l'[(k,l',j) \in A]$

Figure: Transitions for the arc-standard stack-based parsing algorithm.

"

The arc-standard parser is the closest correspondent to the familiar shift-reduce parser for context-free grammars (Aho, Sethi, and Ullman 1986).

The Left-Arc $_l$  and Right-Arc $_l^s$  transitions correspond to reduce actions, replacing a head-dependent structure with its head, whereas the Shift transition is exactly the same as the shift action.

One peculiarity of the transitions, as defined here, is that the "reduce" transitions apply to one node on the stack and one node in the buffer, rather than two nodes on the stack. The reason for this formulation is to facilitate comparison with the arc-eager parser and to simplify the definition of terminal configurations.

# **Illustration** Arc-standard transition sequence for English example sentence:

	$\sigma$	β	A	
1.	( [0],	$[1,\ldots,9],$	Ø	)

Arc-eager stack-based algorithm

# **Illustration** Arc-standard transition sequence for English example sentence:

			P	21	
1. 2. Shift =	(	[0],	$[1, \ldots, 9], [2, \ldots, 9],$	Ø	)

Arc-eager stack-based algorithm

# **Illustration** Arc-standard transition sequence for English example sentence:

			$\sigma$	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$Shift \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2, \ldots, 9],$	$A_1 = \{(2, NMOD, 1)\}$	)

Arc-eager stack-based algorithm

# **Illustration** Arc-standard transition sequence for English example sentence:

			$\sigma$	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)

# **Illustration** Arc-standard transition sequence for English example sentence:

			$\sigma$	$\beta$	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$Shift \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3, \ldots, 9],$	$A_2 = A_1 \cup \{(3, \text{SBJ}, 2)\}$	)

# **Illustration** Arc-standard transition sequence for English example sentence:

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$Shift \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1=\{(2,\mathtt{NMOD},1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$SHIFT \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_2$	)

Arc-eager stack-based algorithm

# **Illustration** Arc-standard transition sequence for English example sentence:

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$Shift \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Shift \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_2$	)
7.	$Shift \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_2$	)

Arc-eager stack-based algorithm

#### **Illustration** Arc-standard transition sequence for English example sentence:

—NMOD— ∠—SBJ— [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9 β A $\sigma$ 1. [0], $[1, \ldots, 9],$ Ø 2. Shift  $\Longrightarrow$ [0, 1], $[2, \ldots, 9],$ Ø

\_NMOD—

3.  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathtt{NMOD}} \implies$ [0], $A_1 = \{(2, NMOD, 1)\}$ 4. [0, 2], $[3, \ldots, 9],$ Shift  $\Longrightarrow$  $A_1$  $A_2 = A_1 \cup \{(3, SBJ, 2)\}$ 5.  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathsf{SBJ}} \implies$ [0],

6. Shift  $\Longrightarrow$ [0, 3], $[4, \ldots, 9],$  $A_2$ 

7. Shift  $\Longrightarrow$ [0, 3, 4], $[5, \ldots, 9],$  $A_2$ 

 $A_3 = A_2 \cup \{(5, \texttt{NMOD}, 4)\}$  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathsf{NMOD}} \implies$ 8. [0, 3], $[5, \ldots, 9],$ 

Arc-eager stack-based algorithm

# **Illustration** Arc-standard transition sequence for English example sentence:

			$\sigma$	β	A
1.		(	[0],	$[1,\ldots,9],$	Ø )
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø )
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\} $
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$ )
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\} $
6.	$SHIFT \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_2$ )
7.	$SHIFT \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_2$ )
8.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_3 = A_2 \cup \{(5, \texttt{NMOD}, 4)\}  )$
9.	$Shift \implies$	(	[0, 3, 5],	$[6, \ldots, 9],$	$A_3$

# **Illustration** Arc-standard transition sequence for English example sentence:

			$\sigma$	β	$\overline{A}$
1.		(	[0],	$[1, \ldots, 9],$	Ø )
2.	Shift $\Longrightarrow$	(	[0, 1],	$[2,\ldots,9],$	Ø )
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\}$
4.	Shift $\Longrightarrow$	(	[0, 2],	$[3,\ldots,9],$	$A_1$ )
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\} $
6.	Shift $\Longrightarrow$	(	[0, 3],	$[4,\ldots,9],$	$A_2$
7.	Shift $\Longrightarrow$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_2$
8.	$Left\text{-}Arc_{NMOD} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_3 = A_2 \cup \{(5, \texttt{NMOD}, 4)\}  )$
9.	Shift $\Longrightarrow$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_3$
10.	Shift $\Longrightarrow$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_3$

✓—NMOD—— ✓—SBJ— ∠NMOD¬ [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9

			$\sigma$	$\beta$	A
1.		(	[0],	$[1,\ldots,9],$	Ø )
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø )
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\} $
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$ )
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\} $
6.	$SHIFT \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_2$ )
7.	$SHIFT \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_2$ )
8.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_3 = A_2 \cup \{(5, \mathtt{NMOD}, 4)\}  )$
9.	$SHIFT \implies$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_3$
10.	$SHIFT \implies$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_3$
11.	$SHIFT \implies$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_3$

12.

 $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathsf{NMOD}} \implies$ 

Arc-standard stack-based algorithm

#### **Illustration** Arc-standard transition sequence for English example sentence:

-NMOD-\_\_\_\_SBJ-\_ \_NMOD\_ [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9 β A $\sigma$ [0],1.  $[1, \ldots, 9],$ Ø 2. [0, 1], $[2, \ldots, 9],$ Ø Shift  $\Longrightarrow$ 3.  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathtt{NMOD}} \implies$ [0], $A_1 = \{(2, NMOD, 1)\}$ 4. [0, 2], $[3, \ldots, 9],$ Shift  $\Longrightarrow$  $A_1$  $A_2 = A_1 \cup \{(3, SBJ, 2)\}$ 5. [0], $[3, \ldots, 9],$  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathtt{SBJ}} \implies$ 6. [0, 3], $[4, \ldots, 9],$  $A_2$ Shift  $\Longrightarrow$ [0, 3, 4], $[5, \ldots, 9],$ 7. Shift  $\Longrightarrow$  $A_3 = A_2 \cup \{(5, NMOD, 4)\}$ 8. [0, 3], $[5, \ldots, 9],$  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathsf{NMOD}} \implies$ 9. Shift  $\Longrightarrow$ [0, 3, 5], $[6, \ldots, 9],$  $A_3$ 10.  $\mathsf{Shift} \implies$ [0, 3, 5, 6],[7, 8, 9], $A_3$ 11. [0, 3, 5, 6, 7],[8, 9],Shift  $\Longrightarrow$  $A_3$  $A_4 = A_3 \cup \{(8, \texttt{NMOD}, 7)\}$ 

[8, 9],

[0, 3, 5, 6],

Arc-eager stack-based algorithm

[ROO	$[OT]_0$ Economic $_1$ ne	` '	√NMOD¬ 3 little <sub>4</sub> effect <sub>5</sub>	on <sub>6</sub> financial <sub>7</sub> markets <sub>8</sub>	3 •9
		$\sigma$	$oldsymbol{eta}$	A	
1.		([0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	([0,1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	([0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$SHIFT \implies$	([0,2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	([0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$SHIFT \implies$	([0,3],	$[4,\ldots,9],$	$A_2$	)
7.	$SHIFT \implies$	([0,3,4]	$], \qquad [5, \ldots, 9],$	$A_2$	)
8.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	([0,3],	$[5,\ldots,9],$	$A_3 = A_2 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$SHIFT \implies$	([0,3,5]	$], \qquad [6, \ldots, 9],$	$A_3$	)
10.	$SHIFT \implies$	([0,3,5]	$,6], \qquad [7,8,9],$	$A_3$	)
11.	$SHIFT \implies$	([0,3,5]	,6,7], [8,9],	$A_3$	)
12.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	([0,3,5]	, 6], [8, 9],	$A_4 = A_3 \cup \{(8, \texttt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^s_{\mathtt{PMOD}} \implies$	([0,3,5]	[6, 9],	$A_5 = A_4 \cup \{(6, \texttt{PMOD}, 8)\}$	)

Arc-eager stack-based algorithm

#### **Illustration** Arc-standard transition sequence for English example sentence:

\_NMOD\_\_ NMOD\_\_ / [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9 β A $\sigma$ [0],1.  $[1, \ldots, 9],$ Ø 2. [0, 1], $[2, \ldots, 9],$ Shift  $\Longrightarrow$ Ø 3.  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathsf{NMOD}} \implies$ [0], $[2, \ldots, 9],$  $A_1 = \{(2, NMOD, 1)\}$ [0, 2],4. Shift  $\Longrightarrow$  $[3, \ldots, 9],$  $A_1$  $A_2 = A_1 \cup \{(3, SBJ, 2)\}$ [0], $[3, \ldots, 9],$ 5.  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathtt{SBJ}} \implies$ 6. [0, 3], $[4, \ldots, 9],$ Shift  $\Longrightarrow$  $A_2$ [0, 3, 4], $[5, \ldots, 9],$ 7. Shift  $\Longrightarrow$  $A_3 = A_2 \cup \{(5, NMOD, 4)\}$ 8. [0, 3], $[5, \ldots, 9],$  $Left-Arc_{NMOD} \implies$ 9. [0, 3, 5], $[6, \ldots, 9],$ Shift  $\Longrightarrow$  $A_3$ 10. Shift  $\Longrightarrow$ [0, 3, 5, 6],[7, 8, 9], $A_3$ 11. [0, 3, 5, 6, 7],[8, 9],Shift  $\Longrightarrow$  $A_3$ 12.  $Left-Arc_{NMOD} \implies$ [0, 3, 5, 6],[8, 9], $A_4 = A_3 \cup \{(8, NMOD, 7)\}$ 13.  $\mathsf{Right} ext{-}\mathsf{Arc}^s_{\mathtt{PMOD}} \implies$  $A_5 = A_4 \cup \{(6, PMOD, 8)\}$ [0, 3, 5],[6, 9], $A_6 = A_5 \cup \{(5, NMOD, 6)\}$ 14.  $Right-Arc_{nmod}^s \Longrightarrow$ [0, 3],[5, 9],

[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9 B A  $\sigma$ 

			U	ρ	A	
1.		(	[0],	$[1,\ldots,9],$	Ø )	
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$		
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\} $	
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$ )	
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\} $	
6.	$SHIFT \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_2$	
7.	$SHIFT \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_2$	
8.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_3 = A_2 \cup \{(5, \texttt{NMOD}, 4)\}  )$	
9.	$SHIFT \implies$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_3$	
10.	$SHIFT \implies$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_3$	
11.	$SHIFT \implies$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_3$	
12.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[8, 9],	$A_4 = A_3 \cup \{(8, \texttt{NMOD}, 7)\}  )$	
13.	$Right ext{-}Arc^s_{\mathtt{PMOD}} \implies$	(	[0, 3, 5],	[6, 9],	$A_5 = A_4 \cup \{(6, \texttt{PMOD}, 8)\}  )$	
14.	$Right ext{-}Arc^s_{\mathtt{NMOD}} \implies$	(	[0, 3],	[5, 9],	$A_6 = A_5 \cup \{(5, \texttt{NMOD}, 6)\}  )$	
15.		(	[0].	[3, 9],	$A_7 = A_6 \cup \{(3, OBJ, 5)\}$	

[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9

			$\sigma$	$\beta$	A	
1.		(	[0],	$[1,\ldots,9],$	Ø )	)
2.	$SHIFT \implies$	(	[0,1],	$[2,\ldots,9],$	Ø )	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, NMOD, 1)\}$	)
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\}$	)
6.	$SHIFT \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_2$	)
7.	$SHIFT \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_2$	)
8.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_3 = A_2 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$SHIFT \implies$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_3$	)
10.	$Shift \implies$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_3$	)
11.	$Shift \implies$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_3$	)
12.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3, 5, 6],	[8, 9],	$A_4 = A_3 \cup \{(8, \texttt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^s_{\mathtt{PMOD}} \implies$	(	[0, 3, 5],	[6, 9],	$A_5 = A_4 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	$Right ext{-}Arc^s_{\mathtt{NMOD}} \implies$	(	[0, 3],	[5, 9],	$A_6 = A_5 \cup \{(5, NMOD, 6)\}$	)
15.	$Right\text{-}Arc^s_{\mathtt{OBJ}} \implies$	(	[0],	[3, 9],	$A_7 = A_6 \cup \{(3, \text{OBJ}, 5)\}$	)
16.	$SHIFT \implies$	(	[0, 3],	[9],	$A_7$	)

```
OBJ_NMOD_\_\ \_NMOD-\_\ \
[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>
                                                                                β
                                                                                                     A
                                                       \sigma
                                                       [0],
   1.
                                                                                [1, \ldots, 9],
                                                                                                     Ø
   2.
                                                       [0, 1],
                                                                                 [2, \ldots, 9],
                          Shift \Longrightarrow
                                                                                                     Ø
   3.
             \mathsf{Left}	ext{-}\mathsf{Arc}_{\mathsf{NMOD}} \implies
                                                       [0],
                                                                                [2, \ldots, 9],
                                                                                                     A_1 = \{(2, NMOD, 1)\}
                                                       [0, 2],
   4.
                          Shift \Longrightarrow
                                                                                [3, \ldots, 9],
                                                                                                     A_1
                                                                                                     A_2 = A_1 \cup \{(3, SBJ, 2)\}
                                                       [0],
                                                                                [3, \ldots, 9],
   5.
               Left-Arc_{SRJ} \Longrightarrow
   6.
                                                       [0, 3].
                                                                                [4, \ldots, 9],
                          Shift \Longrightarrow
                                                                                                     A_2
                                                       [0, 3, 4],
                                                                                [5, \ldots, 9],
   7.
                          Shift \Longrightarrow
                                                                                                     A_3 = A_2 \cup \{(5, \texttt{NMOD}, 4)\}
   8.
                                                       [0, 3],
                                                                                [5, \ldots, 9],
             Left-Arc_{NMOD} \implies
   9.
                                                       [0, 3, 5],
                                                                                [6, \ldots, 9],
                          Shift \Longrightarrow
                                                                                                     A_3
  10.
                          Shift \Longrightarrow
                                                       [0, 3, 5, 6],
                                                                                [7, 8, 9],
                                                                                                     A_3
 11.
                                                       [0, 3, 5, 6, 7],
                                                                                [8, 9],
                          Shift \Longrightarrow
                                                                                                     A_3
 12.
             \mathsf{Left}	ext{-}\mathsf{Arc}_{\mathsf{NMOD}} \implies
                                                       [0, 3, 5, 6],
                                                                                [8, 9],
                                                                                                     A_4 = A_3 \cup \{(8, NMOD, 7)\}
           \mathsf{Right}	ext{-}\mathsf{Arc}^s_{\mathtt{PMOD}} \implies
                                                                                                     A_5 = A_4 \cup \{(6, PMOD, 8)\}
 13.
                                                       [0, 3, 5],
                                                                                [6, 9],
           Right-Arc_{nmod}^{s} \Longrightarrow
 14.
                                                       [0, 3],
                                                                                [5, 9],
                                                                                                     A_6 = A_5 \cup \{(5, NMOD, 6)\}
 15.
             Right-Arc^s_{OBJ} \Longrightarrow
                                                       [0],
                                                                                [3, 9],
                                                                                                     A_7 = A_6 \cup \{(3, OBJ, 5)\}
                                                       [0, 3],
                                                                                [9],
 16.
                          Shift \Longrightarrow
                                                                                                     A_8 = A_7 \cup \{(3, P, 9)\}
 17.
               \mathsf{Right}\text{-}\mathsf{Arc}^s_{\mathtt{p}} \implies
                                                       [0],
                                                                                [3],
```

 $A_6 = A_5 \cup \{(5, NMOD, 6)\}$ 

 $A_7 = A_6 \cup \{(3, OBJ, 5)\}$ 

 $A_9 = A_8 \cup \{(0, ROOT, 3)\}$ 

 $A_8 = A_7 \cup \{(3, P, 9)\}$ 

 $A_7$ 

Arc-standard stack-based algorithm

14.

15.

16.

17.

18.

 $Right-Arc_{nmod}^s \Longrightarrow$ 

 $Right-Arc^s_{OBJ} \Longrightarrow$ 

 $Right-Arc_{p}^{s} \Longrightarrow$ 

 $Right-Arc_{rot}^s \Longrightarrow$ 

Shift  $\Longrightarrow$ 

Arc-eager stack-based algorithm

#### **Illustration** Arc-standard transition sequence for English example sentence:

```
OBJ_NMOD_\\ \rightarrow NMOD_\\ \land{figure}
[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>
                                                                                   β
                                                                                                        A
                                                        \sigma
                                                         [0],
                                                                                   [1, \ldots, 9],
   1.
                                                                                                        Ø
   2.
                                                        [0, 1],
                                                                                   [2, \ldots, 9],
                           Shift \Longrightarrow
                                                                                                        Ø
    3.
              \mathsf{Left}	ext{-}\mathsf{Arc}_{\mathsf{NMOD}} \implies
                                                         [0],
                                                                                   [2, \ldots, 9],
                                                                                                        A_1 = \{(2, NMOD, 1)\}
                                                         [0, 2],
   4.
                           Shift \Longrightarrow
                                                                                   [3, \ldots, 9],
                                                                                                        A_1
                                                                                                        A_2 = A_1 \cup \{(3, SBJ, 2)\}
                                                         [0],
    5.
               Left-Arc_{SRJ} \Longrightarrow
                                                                                   [3, \ldots, 9],
   6.
                                                         [0, 3].
                                                                                   [4, \ldots, 9],
                           Shift \Longrightarrow
                                                                                                        A_2
                                                         [0, 3, 4],
                                                                                   [5, \ldots, 9],
   7.
                           Shift \Longrightarrow
                                                                                                        A_3 = A_2 \cup \{(5, \texttt{NMOD}, 4)\}
   8.
                                                         [0, 3],
                                                                                   [5, \ldots, 9],
              Left-Arc_{NMOD} \implies
   9.
                                                         [0, 3, 5],
                                                                                   [6, \ldots, 9],
                           Shift \Longrightarrow
                                                                                                        A_3
  10.
                                                         [0, 3, 5, 6],
                                                                                   [7, 8, 9],
                                                                                                        A_3
                           Shift \Longrightarrow
                                                         [0, 3, 5, 6, 7],
                                                                                   [8, 9],
  11.
                           Shift \Longrightarrow
                                                                                                        A_3
 12.
              \mathsf{Left}	ext{-}\mathsf{Arc}_{\mathsf{NMOD}} \implies
                                                         [0, 3, 5, 6],
                                                                                   [8, 9],
                                                                                                        A_4 = A_3 \cup \{(8, NMOD, 7)\}
            \mathsf{Right}	ext{-}\mathsf{Arc}^s_{\mathtt{PMOD}} \implies
                                                                                                        A_5 = A_4 \cup \{(6, PMOD, 8)\}
 13.
                                                        [0, 3, 5],
                                                                                   [6, 9],
```

[5, 9],

[3, 9],

[9],

[3],

[0],

[0, 3],

[0, 3],

[0],

[0],

Π,

7 / 16

```
[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>
```

				$\sigma$	$\beta$	A	
1.			(	[0],	$[1,\ldots,9],$	Ø	)
2.	Shift	$\Longrightarrow$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	Left-Arc <sub>nmod</sub>	$\Longrightarrow$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	Shift	$\Longrightarrow$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{SBJ}$	$\Longrightarrow$	(	[0],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	Shift	$\Longrightarrow$	(	[0, 3],	$[4,\ldots,9],$	$A_2$	)
7.	Shift	$\Longrightarrow$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_2$	)
8.	Left-Arc <sub>nmod</sub>	$\Longrightarrow$	(	[0, 3],	$[5,\ldots,9],$	$A_3 = A_2 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	Shift	$\Longrightarrow$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_3$	)
10.	Shift	$\Longrightarrow$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_3$	)
11.	Shift	$\Longrightarrow$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_3$	)
12.	Left-Arc <sub>nmod</sub>	$\Longrightarrow$	(	[0, 3, 5, 6],	[8, 9],	$A_4 = A_3 \cup \{(8, \texttt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^s_{\mathtt{PMOD}}$	$\Longrightarrow$	(	[0, 3, 5],	[6, 9],	$A_5 = A_4 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	$Right ext{-}Arc^s_{\mathtt{NMOD}}$	$\Longrightarrow$	(	[0, 3],	[5, 9],	$A_6 = A_5 \cup \{(5, \texttt{NMOD}, 6)\}$	)
15.	$Right ext{-}Arc^s_{OBJ}$	$\Longrightarrow$	(	[0],	[3, 9],	$A_7 = A_6 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
16.	Shift	$\Longrightarrow$	(	[0, 3],	[9],	$A_7$	)
17.	$Right ext{-}Arc^s_{\mathtt{P}}$	$\Longrightarrow$	(	[0],	[3],	$A_8 = A_7 \cup \{(3, P, 9)\}$	)
18.	$Right ext{-}Arc^s_{\mathtt{ROOT}}$	$\Longrightarrow$	(	[],	[0],	$A_9 = A_8 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
19.	Shift	$\Longrightarrow$	(	[0],	[],	$A_9$	)

				$\sigma$	$\beta$	A	
1.			(	[0],	$[1,\ldots,9],$	Ø	)
2.	Shift	$\Longrightarrow$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	Left-Arc <sub>nmod</sub>	$\Longrightarrow$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	Shift	$\Longrightarrow$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{SBJ}$	$\Longrightarrow$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3,\mathtt{SBJ},2)\}$	)
6.	Shift	$\Longrightarrow$	(	[0, 3],	$[4,\ldots,9],$	$A_2$	)
7.	Shift	$\Longrightarrow$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_2$	)
8.	Left-Arc <sub>nmod</sub>	$\Longrightarrow$	(	[0, 3],	$[5,\ldots,9],$	$A_3 = A_2 \cup \{(5, \texttt{NMOD}, 4)\}$	)
9.	Shift	$\Longrightarrow$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_3$	)
10.	Shift	$\Longrightarrow$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_3$	)
11.	Shift	$\Longrightarrow$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_3$	)
12.	Left-Arc <sub>nmod</sub>	$\Longrightarrow$	(	[0, 3, 5, 6],	[8, 9],	$A_4 = A_3 \cup \{(8, \texttt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^s_{\mathtt{PMOD}}$	$\Longrightarrow$	(	[0, 3, 5],	[6, 9],	$A_5 = A_4 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	$Right ext{-}Arc^s_{\mathtt{NMOD}}$	$\Longrightarrow$	(	[0, 3],	[5, 9],	$A_6 = A_5 \cup \{(5, \texttt{NMOD}, 6)\}$	)
15.	Right-Arc $_{\mathtt{OBJ}}^{s}$	$\Longrightarrow$	(	[0],	[3, 9],	$A_7 = A_6 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
16.	Shift	$\Longrightarrow$	(	[0, 3],	[9],	$A_7$	)
17.	Right-Arc $_{\mathtt{P}}^{s}$	$\Longrightarrow$	(	[0],	[3],	$A_8 = A_7 \cup \{(3, P, 9)\}$	)
18.	$Right ext{-}Arc^s_{\mathtt{ROOT}}$	$\Longrightarrow$	(	[],	[0],	$A_9 = A_8 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
19.	Shift	$\Longrightarrow$	(	[0],	[],	$A_9$	)

## Arc-eager stack-based algorithm

#### **Transitions** LEFT-ARC1 $(\sigma|i,j|\beta,A) \Rightarrow (\sigma,j|\beta,A\cup\{(j,l,i)\})$ RIGHT-ARC1 $(\sigma|i,j|\beta,A) \Rightarrow (\sigma|i|j,\beta,A \cup \{(i,l,j)\})$ REDUCE $(\sigma|i,\beta,A) \Rightarrow (\sigma,\beta,A)$ $(\sigma, i | \beta, A) \Rightarrow (\sigma | i, \beta, A)$ SHIFT **Preconditions** LEFT-ARC1 $\neg[i=0]$ $\neg \exists k \exists l' [(k, l', i) \in A]$ RIGHT-ARC<sup>e</sup> $\neg \exists k \exists l'[(k,l',j) \in A]$ REDUCE $\exists k \exists l [(k,l,i) \in A]$

Figure: Transitions for the arc-eager stack-based parsing algorithm.

66

The arc-eager parser differs from the arc-standard one by attaching right dependents (using RIGHT-ARC<sup>e</sup> transitions) as soon as possible, that is, before the right dependent has found all its right dependents.

As a consequence, the Right-Arc $_{I}^{e}$  transitions cannot replace the head-dependent structure with the head, as in the arc-standard system, but must store both the head and the dependent on the stack for further processing. The dependent can be popped from the stack at a later time through the REDUCE transition, which completes the reduction of this structure.

Arc-eager stack-based algorithm

# **Illustration** Arc-eager transition sequence for English example sentence:

[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>

	$\sigma$	$\beta$ A	
1.	( [0],	$[1,\ldots,9],\varnothing$	)

# [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9

			σ	β	A	
1. 2.	Shift $\Longrightarrow$	(	[0], [0, 1],	$[1, \ldots, 9],$ $[2, \ldots, 9],$	Ø Ø	

Arc-eager stack-based algorithm

# **Illustration** Arc-eager transition sequence for English example sentence:

[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2, \ldots, 9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)

Arc-eager stack-based algorithm

# **Illustration** Arc-eager transition sequence for English example sentence:

[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)

Arc-eager stack-based algorithm

### **Illustration** Arc-eager transition sequence for English example sentence:

[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9

			$\sigma$	eta	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)

Arc-eager stack-based algorithm

			$\sigma$	$\beta$	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)

Arc-eager stack-based algorithm

# **Illustration** Arc-eager transition sequence for English example sentence:

 $[\mathsf{ROOT}]_0 \ \mathsf{Economic}_1 \ \mathsf{news}_2 \ \mathsf{had}_3 \ \mathsf{little}_4 \ \mathsf{effect}_5 \ \mathsf{on}_6 \ \mathsf{financial}_7 \ \mathsf{markets}_8 \ .9$ 

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	Shift $\Longrightarrow$	(	[0, 3, 4],	$[5, \ldots, 9],$	$A_3$	)

Arc-eager stack-based algorithm

# **Illustration** Arc-eager transition sequence for English example sentence:

 $[\mathsf{ROOT}]_0 \ \mathsf{Economic}_1 \ \mathsf{news}_2 \ \mathsf{had}_3 \ \mathsf{little}_4 \ \mathsf{effect}_5 \ \mathsf{on}_6 \ \mathsf{financial}_7 \ \mathsf{markets}_8 \ ._9$ 

			$\sigma$	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	$SHIFT \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0, 3],	$[5, \ldots, 9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)

Arc-eager stack-based algorithm

# **Illustration** Arc-eager transition sequence for English example sentence:

NMOD— SBJ— OBJ—NMOD— [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies$	(	[0, 1],	$[2, \ldots, 9],$	Ø	)
3.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$SHIFT \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left ext{-}Arc_{\mathtt{NMOD}} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	Rіght-Arc $_{\mathtt{OBJ}}^{e} \implies$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)

		(	σ	β	A	
1.		( [	[0],	$[1,\ldots,9],$	Ø	)
2.	$Shift \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$SHIFT \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	( [	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	( [	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	$SHIFT \implies$	( [	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left ext{-}Arc_{NMOD} \implies$	( [	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	Rіснт-Arc $_{\mathtt{OBJ}}^{e} \implies$	( [	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^e_{\mathtt{NMOD}} \implies$	( [	[0, 3, 5, 6],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \mathtt{NMOD}, 6)\}$	)

Arc-eager stack-based algorithm

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$Shift \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	$Shift \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	Rіght-Arc $_{\mathtt{OBJ}}^{e} \Longrightarrow$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^e_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \mathtt{NMOD}, 6)\}$	)
11.	$Shift \implies$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_6$	)

8.

 $Left-Arc_{NMOD} \implies$ 

Arc-eager stack-based algorithm

#### **Illustration** Arc-eager transition sequence for English example sentence:

NMOD—\\_\_NMOD— [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9 A $\sigma$ 1. [0], $[1, \ldots, 9],$ Ø 2.  $[2, \ldots, 9],$ Shift  $\Longrightarrow$ [0, 1], $A_1 = \{(2, NMOD, 1)\}$ 3.  $Left-Arc_{NMOD} \implies$ [0],4. [0, 2], $A_1$ Shift  $\Longrightarrow$  $A_2 = A_1 \cup \{(3, SBJ, 2)\}$ 5.  $\mathsf{Left} ext{-}\mathsf{Arc}_{\mathsf{SBJ}} \implies$ [0], $\mathsf{Right}\text{-}\mathsf{Arc}^e_{\mathtt{ROOT}} \implies$  $A_3 = A_2 \cup \{(0, ROOT, 3)\}$ 6. [0, 3],7. [0, 3, 4], $[5, \ldots, 9],$ Shift  $\Longrightarrow$ 

 $[6,\ldots,9],$  $A_5 = A_4 \cup \{(3, OBJ, 5)\}$ Right-Arc $_{\mathtt{OB,I}}^{e} \Longrightarrow$ 9. [0, 3, 5], $\mathsf{Right} ext{-}\mathsf{Arc}^e_{\mathtt{NMOD}} \implies$ [0, 3, 5, 6],[7, 8, 9], $A_6 = A_5 \cup \{(5, NMOD, 6)\}$ 10. 11.  $SHIFT \Longrightarrow$ [0, 3, 5, 6, 7],[8, 9], $A_6$ 

[0, 3],

 $[5,\ldots,9],$ 

 $A_4 = A_3 \cup \{(5, NMOD, 4)\}$ 

12.  $A_7 = A_6 \cup \{(8, NMOD, 7)\}$  $Left-Arc_{NMOD} \implies$ [0, 3, 5, 6],[8, 9],

			$\sigma$	β	$\overline{A}$	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	Shift $\Longrightarrow$	(	[0,1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	$Shift \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	Right-Arc $_{\mathtt{OBJ}}^{e} \Longrightarrow$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^e_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \mathtt{NMOD}, 6)\}$	)
11.	$Shift \implies$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_6$	)
12.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3, 5, 6],	[8, 9],	$A_7 = A_6 \cup \{(8, \texttt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^e_{\mathtt{PMOD}} \implies$	(	[0, 3, 5, 6, 8],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)

		$\sigma$	$\beta$	A	
1.	(	[0],	$[1,\ldots,9],$	Ø	)
2.	$SHIFT \implies ($	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies ($	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$SHIFT \implies ($	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies ($	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies ($	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	$SHIFT \implies ($	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc_{\mathtt{NMOD}} \implies ($	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$Right ext{-}Arc^e_{\mathtt{OBJ}} \implies ($	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^e_{\mathtt{NMOD}} \implies ($	[0, 3, 5, 6],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \mathtt{NMOD}, 6)\}$	)
11.	$SHIFT \implies ($	[0, 3, 5, 6, 7],	[8, 9],	$A_6$	)
12.	$Left\text{-}Arc_{\mathtt{NMOD}} \implies ($	[0, 3, 5, 6],	[8, 9],	$A_7 = A_6 \cup \{(8, \mathtt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^e_{\mathtt{PMOD}} \implies ($	[0, 3, 5, 6, 8],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	$Reduce \implies ($	[0, 3, 5, 6],	[9],	$A_8$	)

Arc-eager stack-based algorithm

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$Shift \implies$	(	[0,1],	$[2, \ldots, 9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$SHIFT \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	Right-Arc $_{\mathtt{OBJ}}^{e} \Longrightarrow$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^e_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \mathtt{NMOD}, 6)\}$	)
11.	Shift $\Longrightarrow$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_6$	)
12.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3, 5, 6],	[8, 9],	$A_7 = A_6 \cup \{(8, \mathtt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^e_{\mathtt{PMOD}} \implies$	(	[0, 3, 5, 6, 8],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	Reduce ⇒	(	[0, 3, 5, 6],	[9],	$A_8$	)
15.	Reduce $\Longrightarrow$	(	[0, 3, 5],	[9],	$A_8$	)

2 Stack-based Algorithms

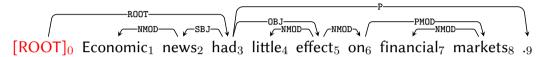
Arc-eager stack-based algorithm

### **Illustration** Arc-eager transition sequence for English example sentence:

			$\sigma$	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	Shift $\Longrightarrow$	(	[0,1],	$[2, \ldots, 9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	Ì	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift \implies$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	$Shift \implies$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$Right ext{-}Arc^e_{\mathtt{OBJ}} \implies$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^e_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \mathtt{NMOD}, 6)\}$	)
11.	$Shift \implies$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_6$	)
12.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3, 5, 6],	[8, 9],	$A_7 = A_6 \cup \{(8, \mathtt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^e_{\mathtt{PMOD}} \implies$	(	[0, 3, 5, 6, 8],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	Reduce $\Longrightarrow$	(	[0, 3, 5, 6],	[9],	$A_8$	)
15.	Reduce $\Longrightarrow$	(	[0, 3, 5],	[9],	$A_8$	)
16.	Reduce $\Longrightarrow$	(	[0, 3],	[9],	$A_8$	)

```
 \begin{array}{c|c} -ROOT \\ \hline & \\ \nearrow -NMOD \\ \hline & \\ \nearrow -NMOD \\ \hline & \\ \hline \end{array} \begin{array}{c} -PMOD \\ \hline & \\ \nearrow -NMOD \\ \hline \end{array} 
[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .9
```

			σ	β	A	
1.		(	[0],	$[1,\ldots,9],$	Ø	)
2.	$Shift \implies$	(	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left ext{-}Arc_{NMOD} \implies$	(	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	Shift $\Longrightarrow$	(	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left ext{-}Arc_{\mathtt{SBJ}} \implies$	(	[0],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies$	(	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	Shift $\Longrightarrow$	(	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	Rіght-Arc $_{ t OBJ}^e \Longrightarrow$	(	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^e_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[7, 8, 9],	$A_6=A_5\cup\{(5,\mathtt{NMOD},6)\}$	)
11.	Shift $\Longrightarrow$	(	[0, 3, 5, 6, 7],	[8, 9],	$A_6$	)
12.	$Left ext{-}Arc_{NMOD} \implies$	(	[0, 3, 5, 6],	[8, 9],	$A_7 = A_6 \cup \{(8, \mathtt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^e_{\mathtt{PMOD}} \implies$	(	[0, 3, 5, 6, 8],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	Reduce $\Longrightarrow$	(	[0, 3, 5, 6],	[9],	$A_8$	)
15.	Reduce $\Longrightarrow$	(	[0, 3, 5],	[9],	$A_8$	)
16.	Reduce $\Longrightarrow$	(	[0, 3],	[9],	$A_8$	)
17.	Rіght-Arc $_{\mathtt{P}}^{e} \Longrightarrow$	(	[0, 3, 9],	[],	$A_9 = A_8 \cup \{(3, P, 9)\}$	)



		$\sigma$	β	A	
1.	(	[0],	$[1,\ldots,9],$	Ø	)
2.	Shift $\Longrightarrow$ (	[0, 1],	$[2,\ldots,9],$	Ø	)
3.	$Left\text{-}Arc_{NMOD} \implies ($	[0],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	Shift $\Longrightarrow$ (	[0, 2],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc_{\mathtt{SBJ}} \implies ($	[0],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^e_{\mathtt{ROOT}} \implies ($	[0, 3],	$[4,\ldots,9],$	$A_3=A_2\cup\{(0,\mathtt{ROOT},3)\}$	)
7.	Shift $\Longrightarrow$ (	[0, 3, 4],	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc_{NMOD} \implies ($	[0, 3],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$Right ext{-}Arc^e_{\mathtt{OBJ}} \implies ($	[0, 3, 5],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^e_{\mathtt{NMOD}} \implies ($	[0, 3, 5, 6],	[7, 8, 9],	$A_6=A_5\cup\{(5,\mathtt{NMOD},6)\}$	)
11.	Shift $\Longrightarrow$ (	[0, 3, 5, 6, 7],	[8, 9],	$A_6$	)
12.	$Left\text{-}Arc_{NMOD} \implies ($	[0, 3, 5, 6],	[8, 9],	$A_7 = A_6 \cup \{(8, \mathtt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^e_{\mathtt{PMOD}} \implies ($	[0, 3, 5, 6, 8],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	Reduce $\Longrightarrow$ (	[0, 3, 5, 6],	[9],	$A_8$	)
15.	$Reduce \implies ($	[0, 3, 5],	[9],	$A_8$	)
16.	Reduce $\Longrightarrow$ (	[0, 3],	[9],	$A_8$	)
17.	$Right ext{-}Arc^e_{\mathtt{P}} \implies ($	[0, 3, 9],	[],	$A_9 = A_8 \cup \{(3, P, 9)\}$	)

# **List-based Algorithms**

#### Definition

A list-based configuration for a sentence  $x = (w_0, w_1, \dots, w_n)$  is a quadruple  $c = (\lambda_1, \lambda_2, \beta, A)$ , where

- 1.  $\lambda_1$  is a list of tokens  $i_1 < k_1$  (for some  $k_1 < n$ )
  - will represent as a list with head to right (nodes in decreasing order)
- 2.  $\lambda_2$  is a list of tokens  $i_2 \leq k_2$  (for some  $k_2$ , such that  $k_1 < k_2 \leq n$ )
  - will represent as a list with head to left (nodes in increasing order)
- 3.  $\beta$  is a buffer of tokens  $i > k_2$ ,
  - will represent as a list with head to left
- 4. A is a set of dependency arcs such that  $G = (\{0, 1, \dots, n\}, A)$  is a dependency graph for x.

Write  $\lambda_1.\lambda_2$  for the concatenation of lists  $\lambda_1$  and  $\lambda_2$ . Ex., [0,1].[2,3,4] = [0,1,2,3,4].

#### Definition

A list-based transition system is a quadruple  $S = (C, T, c_{\text{start}}, C_{\text{terminal}})$ , where

- 1. C is the set of all list-based configurations,
- 2.  $c_{\text{start}}(x = (w_0, w_1, \dots, w_n)) = ([0], [], [1, \dots, n], \emptyset),$
- 3. T is a set of transitions, each of which is a function  $t: C \to C$ ,
- 4.  $C_{\text{terminal}} = \{c \in C \mid c = (\lambda_1, \lambda_2, [], A)\}.$

(Note, only difference from stack-based system is: two lists instead of a single stack)

#### **Transitions** LEFT-ARC $_{1}^{n}$ $(\lambda_1|i,\lambda_2,j|\beta,A) \Rightarrow (\lambda_1,i|\lambda_2,j|\beta,A\cup\{(j,l,i)\})$ $(\lambda_1|i,\lambda_2,i|\beta,A) \Rightarrow (\lambda_1,i|\lambda_2,i|\beta,A\cup\{(i,l,i)\})$ RIGHT-ARC $_{1}^{n}$ $No-Arc^n$ $(\lambda_1|i,\lambda_2,\beta,A) \Rightarrow (\lambda_1,i|\lambda_2,\beta,A)$ $SHIFT^{\lambda}$ $(\lambda_1, \lambda_2, i | \beta, A) \Rightarrow (\lambda_1, \lambda_2 | i, [], \beta, A)$ **Preconditions** LEFT-ARC $_{1}^{n}$ $\neg[i = 0]$ $\neg \exists k \exists l' [(k, l', i) \in A]$ $\neg [i \rightarrow^* j]_A$ RIGHT-ARC $_{1}^{n}$ $\neg \exists k \exists l' [(k, l', i) \in A]$ $\neg [i \rightarrow^* i]_A$

Figure: Transitions for the arc-eager stack-based parsing algorithm.

The fact that both the head and the dependent are kept in either  $\lambda_2$  or  $\beta$  makes it possible to construct non-projective dependency graphs, because the No-Arc<sup>n</sup> transition allows a node to be passed from  $\lambda_1$  to  $\lambda_2$  even if it does not (yet) have a head.

Projective list-based algorithm

# **Illustration** Transition sequence for non-projective Czech example sentence:

 $[\mathsf{ROOT}]_0 \ \mathsf{Z}_1 \ \mathsf{nich}_2 \ \mathsf{je}_3 \ \mathsf{jen}_4 \ \mathsf{jedna}_5 \ \mathsf{na}_6 \ \mathsf{kvalitu}_7 \ ._8$ 

		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )

Projective list-based algorithm

# **Illustration** Transition sequence for non-projective Czech example sentence:

 $[\mathsf{ROOT}]_0$   $\mathsf{Z}_1$  nich $_2$  je $_3$  jen $_4$  jedna $_5$  na $_6$  kvalitu $_7$  . $_8$ 

		$\lambda_1$	$\lambda_2$	$oldsymbol{eta}$	A	
1.	(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies ($	[0, 1],	[],	$[2,\ldots,8],$		)

Projective list-based algorithm

# **Illustration** Transition sequence for non-projective Czech example sentence:

 $[\mathsf{ROOT}]_0 \ \mathsf{Z}_1 \ \mathsf{nich}_2 \ \mathsf{je}_3 \ \mathsf{jen}_4 \ \mathsf{jedna}_5 \ \mathsf{na}_6 \ \mathsf{kvalitu}_7 \ ._8$ 

			$\lambda_1$	$\lambda_2$	eta	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \implies$	(	[0],	[1],	$[2, \ldots, 8],$	$A_1=(1,\mathtt{Atr},2)$	)

Projective list-based algorithm

# **Illustration** Transition sequence for non-projective Czech example sentence:

$$[ROOT]_0$$
  $Z_1$   $nich_2$   $je_3$   $jen_4$   $jedna_5$   $na_6$   $kvalitu_7$  .8

			$\lambda_1$	$\lambda_2$	eta	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \implies$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \Longrightarrow$	(	[0, 1, 2],	[]	$[3,\ldots,8],$	$A_1$	)

Projective list-based algorithm

# **Illustration** Transition sequence for non-projective Czech example sentence:

 $[\mathsf{ROOT}]_0 \ \mathsf{Z}_1 \ \mathsf{nich}_2 \ \mathsf{je}_3 \ \mathsf{jen}_4 \ \mathsf{jedna}_5 \ \mathsf{na}_6 \ \mathsf{kvalitu}_7 \ ._8$ 

			$\lambda_1$	$\lambda_2$	eta	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \implies$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)

Projective list-based algorithm

 $[\mathsf{ROOT}]_0 \ \mathsf{Z}_1 \ \mathsf{nich}_2 \ \mathsf{je}_3 \ \mathsf{jen}_4 \ \mathsf{jedna}_5 \ \mathsf{na}_6 \ \mathsf{kvalitu}_7 \ ._8$ 

			$\lambda_1$	$\lambda_2$	$oldsymbol{eta}$	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[]	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \implies$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \implies$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No ext{-}Arc^n \implies$	(	[0],	[1, 2],	$[3, \ldots, 8],$	$A_1$	)

			$\lambda_1$	$\lambda_2$	$\beta$	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \Longrightarrow$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \Longrightarrow$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No ext{-}Arc^n \implies$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$	)
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \mathtt{Pred}, 3)$	)

Projective list-based algorithm



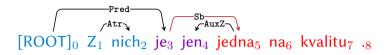
			$\lambda_1$	$\lambda_2$	$\beta$	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \implies$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No\text{-}Arc^n \Longrightarrow$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$	)
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \mathtt{Pred}, 3)$	)
8.	$Shift^\lambda \implies$	(	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$	)

Projective list-based algorithm

			$\lambda_1$	$\lambda_2$	$\beta$	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \implies$		[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No ext{-}Arc^n \implies$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$	)
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)$	)
8.	$Shift^\lambda \implies$	(	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$	)
9.	$Shift^\lambda \implies$	(	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$	)

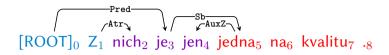
Projective list-based algorithm

			$\lambda_1$	$\lambda_2$	$\beta$	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \implies$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No ext{-}Arc^n \implies$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$	)
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)$	)
8.	$Shift^\lambda \implies$	(	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$	)
9.	$Shift^\lambda \implies$	(	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$	)
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies$	(	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)$	)



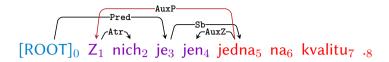
			$\lambda_1$	$\lambda_2$	$oldsymbol{eta}$	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \Longrightarrow$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \Longrightarrow$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No\text{-}Arc^n \Longrightarrow$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$	)
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)$	)
8.	$Shift^\lambda \Longrightarrow$	(	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$	)
9.	$Shift^\lambda \Longrightarrow$	(	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$	)
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies$	(	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)$	)
11.	Rіght-Arc $_{\mathtt{Sb}}^{n} \Longrightarrow$	(	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4=A_3\cup(3,\operatorname{Sb},5)$	)

Projective list-based algorithm

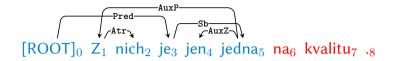


			$\lambda_1$	$\lambda_2$	$oldsymbol{eta}$	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \implies$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No ext{-}Arc^n \implies$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$	)
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \mathtt{Pred}, 3)$	)
8.	$Shift^\lambda \implies$	(	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$	)
9.	$Shift^\lambda \Longrightarrow$	(	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$	)
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies$	(	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)$	)
11.	$Right\text{-}Arc^n_{Sb} \Longrightarrow$	(	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5)$	)
12.	$\operatorname{No-Arc}^n \Longrightarrow$	(	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$	)

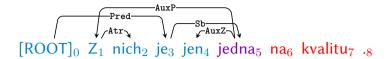
Non-projective list-based algorithm



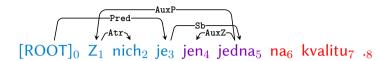
			$\lambda_1$	$\lambda_2$	eta	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \Longrightarrow$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \implies$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \Longrightarrow$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No ext{-}Arc^n \implies$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$	)
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \mathtt{Pred}, 3)$	)
8.	$Shift^\lambda \implies$	(	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$	)
9.	$Shift^\lambda \Longrightarrow$	(	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$	)
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies$	(	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)$	)
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies$	(	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5)$	)
12.	$No\text{-}Arc^n \Longrightarrow$	(	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$	)
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies$	(	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)$	)



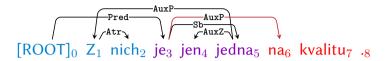
			$\lambda_1$	$\lambda_2$	eta	A	
1.		(	[0],	[],	$[1,\ldots,8],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø	)
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1=(1,\mathtt{Atr},2)$	)
4.	$Shift^\lambda \implies$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$	)
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$	)
6.	$No ext{-}Arc^n \implies$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$	)
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \mathtt{Pred}, 3)$	)
8.	$Shift^\lambda \Longrightarrow$	(	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$	)
9.	$Shift^\lambda \implies$	(	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$	)
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies$	(	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)$	)
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies$	(	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5)$	)
12.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$	)
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies$	(	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)$	)
14.	$Shift^\lambda \implies$	(	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$	)



		$\lambda_1$	$\lambda_2$	eta	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \implies ($	[0,1],	[],	$[2, \ldots, 8],$	Ø )
3.	$Right\text{-}Arc^n_{\mathtt{Atr}} \Longrightarrow ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2)$
4.	$Shift^\lambda \implies \ ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$
5.	$No ext{-}Arc^n \implies ($	[0,1],	[2],	$[3,\ldots,8],$	$A_1$
6.	$No ext{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$ )
7.	$Right ext{-}Arc^n_{Pred} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \mathtt{Pred}, 3)  )$
8.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \Longrightarrow \ ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)$ )
11.	$Right\text{-}Arc^n_{\mathtt{Sb}} \Longrightarrow ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No ext{-}Arc^n \implies ($	[0,1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies ($	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$

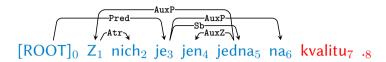


		$\lambda_1$	$\lambda_2$	eta	A	
1.		([0],	[],	$[1,\ldots,8],$	Ø )	
2.	$Shift^\lambda \implies$	([0,1],	[],	$[2,\ldots,8],$	Ø )	
3.	Rіgнт-Arc $_{\mathtt{Atr}}^{n} \Longrightarrow$	([0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $	
4.	$Shift^\lambda \implies$	([0,1,1]	[2], [],	$[3,\ldots,8],$	$A_1$	
5.	$No ext{-}Arc^n \implies$	([0,1],	[2],	$[3,\ldots,8],$	$A_1$	
6.	$No ext{-}Arc^n \implies$	([0],	[1, 2],	$[3,\ldots,8],$	$A_1$	
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	( [],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$	
8.	$Shift^\lambda \Longrightarrow$	$([0,\ldots$	[], [],	$[4,\ldots,8],$	$A_2$	
9.	$Shift^\lambda \implies$	$([0,\ldots$		$[5,\ldots,8],$	$A_2$	
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies$	$([0,\ldots$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$	
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies$	([0,1,1]	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \mathtt{Sb}, 5) \qquad )$	
12.	$No ext{-}Arc^n \implies$	([0,1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$	
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies$	([0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$	
14.	$Shift^\lambda \implies$	$([0,\ldots$	[], [],	[6, 7, 8],	$A_5$	
15.	$No ext{-}Arc^n \implies$	$([0,\ldots$	[5],	[6, 7, 8],	$A_5$	
16.	$No ext{-}Arc^n \implies$	$([0,\ldots$	[4, 5], [4, 5],	[6, 7, 8],	$A_5$	



			$\lambda_1$	$\lambda_2$	β	A
1.		(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,8],$	Ø )
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	(	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$Shift^\lambda \implies$	(	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$ )
5.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$ )
6.	$No ext{-}Arc^n \implies$	(	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$ )
7.	$Right ext{-}Arc^n_{Pred} \implies$	(	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$Shift^\lambda \implies$	(	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \implies$	(	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies$	(	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies$	(	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No ext{-}Arc^n \implies$	(	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies$	(	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)$ )
14.	$Shift^\lambda \implies$	(	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No ext{-}Arc^n \implies$	(	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No ext{-}Arc^n \implies$	(	$[0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$
17.	$Right ext{-}Arc^n_{\mathtt{AuxP}} \implies$	(	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$

Projective list-based algorithm

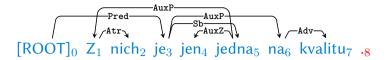


		$\lambda_1$	$\lambda_2$	$\beta$	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \implies \ ($	[0, 1],	[],	$[2,\ldots,8],$	Ø )
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$Shift^\lambda \implies \ ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$ )
5.	$No ext{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$ )
6.	$No ext{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$ )
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right\text{-}Arc^n_{\mathtt{Sb}} \implies ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No ext{-}Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)$ )
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No ext{-}Arc^n \implies ($	$[0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$
17.	$Right\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6=A_5\cup(3,\mathtt{AuxP},6)$ )
18.	$Shift^\lambda \implies \ ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$

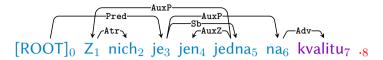


		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \Longrightarrow ($	[0, 1],	[],	$[2,\ldots,8],$	$\varnothing$
3.	$Right ext{-}Arc^n_\mathtt{Atr} \implies ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$SHIFT^\lambda \Longrightarrow ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$
5.	$No ext{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$
6.	$No ext{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$Shift^\lambda \implies ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$SHIFT^\lambda \Longrightarrow ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right ext{-}Arc^n_\mathtt{Sb} \implies ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No ext{-}Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies ($	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)$ )
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No ext{-}Arc^n \implies ($	$[0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$
17.	$Right ext{-}Arc^n_{\mathtt{Aux}  ext{P}} \implies ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$
18.	$SHIFT^\lambda \implies \ ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right ext{-}Arc^n_{\mathtt{Adv}} \Longrightarrow$ (	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad)$

Projective list-based algorithm



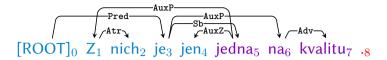
		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$SHIFT^\lambda \Longrightarrow ($	[0, 1],	[],	$[2,\ldots,8],$	Ø )
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \implies ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$SHIFT^\lambda \Longrightarrow ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$
5.	$No-Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$
6.	$No-Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$SHIFT^\lambda \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$SHIFT^\lambda \Longrightarrow ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No\text{-}Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No-Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No\text{-}Arc^n \implies ($		[4, 5],	[6, 7, 8],	$A_5$
17.	$Right\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$
18.	$SHIFT^\lambda \Longrightarrow ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right-Arc^n_{Adv} \Longrightarrow ($	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad)$
20.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,7],$	[],	[8],	$A_7$



		$\lambda_1$	-	$\lambda_2$	β	A
1.		([0]	],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \implies$		, 1],	[],	$[2, \ldots, 8],$	Ø )
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$	( [0]	],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$Shift^\lambda \implies$	(0)	, 1, 2],	[],	$[3,\ldots,8],$	$A_1$ )
5.	$No ext{-}Arc^n \implies$	(0)	, 1],	[2],	$[3,\ldots,8],$	$A_1$ )
6.	$No ext{-}Arc^n \implies$	(0]	],	[1, 2],	$[3,\ldots,8],$	$A_1$ )
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	( [],		[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$Shift^\lambda \implies$	(0)	$,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \implies$	(0)	$,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies$	(0)	$,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right ext{-}Arc^n_\mathtt{Sb} \implies$	(0)	, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No ext{-}Arc^n \implies$	(0)	, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies$	(0]	],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$Shift^\lambda \implies$	(0)	$,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No ext{-}Arc^n \implies$	(0)	$,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No ext{-}Arc^n \implies$	(0)	$,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$
17.	Rіght-Arc $_{\mathtt{Aux}\mathtt{P}}^{n}\Longrightarrow$	(0)	, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)$ )
18.	$Shift^\lambda \implies$	(0)	$,\ldots,6],$	[],	[7, 8],	$A_6$
19.	Rіснт-Arc $_{\mathtt{Ady}}^n \Longrightarrow$	(0)	$,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad )$
20.	$Shift^\lambda \implies$	(0)	$,\ldots,7],$	[],	[8],	$A_7$
21.	$No ext{-}Arc^n \implies$	(0)	$,\ldots,6],$	[7],	[8],	$A_7$

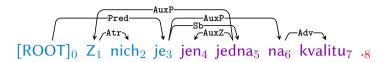


				0	
		$\lambda_1$	$\lambda_2$	β	A
1.	(	([0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \Longrightarrow$ (	([0,1],	[],	$[2,\ldots,8],$	Ø )
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \implies 0$	([0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$Shift^\lambda \Longrightarrow$ (	([0,1,2],	[],	$[3,\ldots,8],$	$A_1$
5.	$No\text{-}Arc^n \Longrightarrow$	([0,1],	[2],	$[3,\ldots,8],$	$A_1$ )
6.	$No-Arc^n \implies 0$	([0],	[1, 2],	$[3,\ldots,8],$	$A_1$
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \Longrightarrow$	( [],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	Shift $^{\lambda} \Longrightarrow$ (	$([0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \Longrightarrow$ (	$([0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies 0$	$([0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies 0$	([0,1,2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No-Arc^n \implies 0$	([0,1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies$	( [0],	$[1,\ldots,4],$		$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)$ )
14.	Shift $^{\lambda} \Longrightarrow$ (	$([0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No\text{-}Arc^n \Longrightarrow$	$([0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No\text{-}Arc^n \Longrightarrow$	$([0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$
17.	$Right\text{-}Arc^n_{\mathtt{AuxP}} \implies$	([0,1,2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$
18.	$Shift^\lambda \Longrightarrow$ (	$([0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right ext{-}Arc^n_{\mathtt{Adv}} \Longrightarrow$	$([0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad)$
20.	$Shift^\lambda \Longrightarrow 0$	$([0,\ldots,7],$	[],	[8],	$A_7$
21.	$No\text{-}Arc^n \Longrightarrow$	$([0,\ldots,6],$	[7],	[8],	$A_7$
22.	$No\text{-}Arc^n \implies 0$	$([0,\ldots,5],$	[6, 7],	[8],	$A_7$



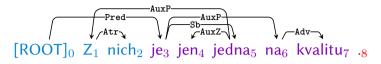
		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \Longrightarrow ($	[0, 1],	[],	$[2,\ldots,8],$	Ø )
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \implies ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$Shift^\lambda \implies ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$ )
5.	$No ext{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$ )
6.	$No ext{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$ )
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \implies ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right ext{-}Arc^n_{Sb} \implies ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No ext{-}Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)$ )
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No\text{-}Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No\text{-}Arc^n \implies ($	$[0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$
17.	$Right-Arc^n_{AuxP} \Longrightarrow ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)$ )
18.	$Shift^\lambda \implies ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right ext{-}Arc^n_{\mathtt{Adv}} \Longrightarrow$ (	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad)$
20.	$Shift^\lambda \implies ($	$[0,\ldots,7],$	[],	[8],	$A_7$
21.	$No-Arc^n \implies ($	$[0,\ldots,6],$	[7],	[8],	$A_7$
22.	$No ext{-}Arc^n \implies ($	$[0,\ldots,5],$	[6, 7],	[8],	$A_7$
23.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5, 6, 7],	[8],	$A_7$

Projective list-based algorithm

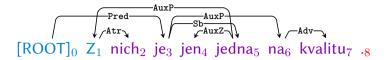


		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \Longrightarrow ($	[0, 1],	[],	$[2,\ldots,8],$	$\varnothing$
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \implies ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$Shift^\lambda \implies ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$
5.	$No ext{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$
6.	$No ext{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	
7.	$Right\text{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$Shift^{\lambda} \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \implies ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	nuxa	$[0,\ldots,3],$	[4],		$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No\text{-}Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	
13.	$Left\text{-}Arc^n_{\mathtt{Aux}} \implies ($		$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$SHIFT^\lambda \implies ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$ )
15.	$No ext{-}Arc^n \implies ($		[5],	[6, 7, 8],	$A_5$
16.	$No-Arc^n \implies ($		[4, 5],	[6, 7, 8],	$A_5$
17.	$Right-Arc^n_{Aux^p_i} \Longrightarrow ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$
18.	$SHIFT^\lambda \implies ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right ext{-}Arc^n_{\mathtt{Adv}} \implies ($	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, Adv, 7) \qquad)$
20.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,7],$	[],	[8],	$A_7$
21.	$No ext{-}Arc^n \implies ($	$[0,\ldots,6],$	[7],	[8],	$A_7$
22.	$No-Arc^n \Longrightarrow ($	$[0,\ldots,5],$	[6, 7],	[8],	$A_7$
23.	$No-Arc^n \Longrightarrow ($	$[0,\ldots,4],$	[5, 6, 7],	[8],	$A_7$
24.	$No-Arc^n \implies ($	$[0,\ldots,3],$	$[4,\ldots,7],$	[8],	$A_7$

Projective list-based algorithm

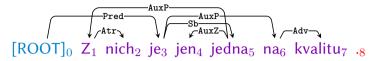


		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$SHIFT^\lambda \Longrightarrow ($	[0, 1],	<u> </u>	$[2,\ldots,8],$	Ø )
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$ (	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$SHIFT^\lambda \Longrightarrow ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$ )
5.	$No ext{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$ )
6.	$No\text{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$SHIFT^{\lambda} \Longrightarrow ($	$[0,\ldots,3],$	[]	$[4,\ldots,8],$	$A_2$
9.	$SHIFT^\lambda \implies ($	$[0,\ldots,4],$	[]	$[5,\ldots,8],$	,
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	
11.	$Right-Arc^n_{Sb} \Longrightarrow ($	[0, 1, 2],	[3, 4],		$A_4 = A_3 \cup (3, \mathtt{Sb}, 5) $
12.	$No-Arc^n \implies ($	[0, 1],	[2, 3, 4],		
13.	$Left\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($		$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$ )
15.	$No\text{-}Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$ )
16.	$No-Arc^n \implies ($		[4, 5],	[6, 7, 8],	$A_5$
17.	$Right-Arc^n_{Aux^p} \implies ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$
18.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right ext{-}Arc^n_\mathtt{Adv} \implies ($	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad)$
20.	$SHIFT^\lambda \implies ($	$[0,\ldots,7],$	[],	[8],	$A_7$
21.	$No-Arc^n \implies ($	$[0,\ldots,6],$	[7],	[8],	$A_7$
22.	$No-Arc^n \Longrightarrow ($		[6, 7],	[8],	$A_7$
23.	$No-Arc^n \Longrightarrow ($	$[0,\ldots,4],$	[5, 6, 7],	[8],	$A_7$
24.	$No-Arc^n \Longrightarrow ($	$[0,\ldots,3],$	$[4,\ldots,7],$	[8],	$A_7$
25.	$No\text{-}Arc^n \implies ($	[0, 1, 2],	$[3,\ldots,7],$	[8],	$A_7$

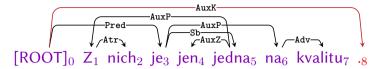


		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \implies ($	[0, 1],	[],	$[2,\ldots,8],$	Ø )
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow$ (	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$Shift^\lambda \Longrightarrow ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$
5.	$No ext{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$
6.	$No ext{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	
7.	$Right\text{-}Arc^n_{\mathtt{Pred}} \Longrightarrow ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$Shift^{\lambda} \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \implies ($	$[0,\ldots,4],$	[]	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	
11.	$Right\text{-}Arc^n_{\mathtt{Sb}} \implies ($	[0, 1, 2],	[3, 4],		$A_4 = A_3 \cup (3, \mathtt{Sb}, 5) $
12.	$No-Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	*
13.	$Left\text{-}Arc^n_{\mathtt{Aux}} \implies ($		$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No-Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No-Arc^n \implies ($		[4, 5],		$A_5$
17.	$Right-Arc^n_{AuxP} \Longrightarrow ($		[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$
18.	$SHIFT^{\lambda} \Longrightarrow ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right-Arc^n_{Adv} \Longrightarrow ($	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, Adv, 7) \qquad )$
20.	$SHIFT^{\lambda} \Longrightarrow ($	$[0,\ldots,7],$	[],	[8],	$A_7$
21.	$No-Arc^n \Longrightarrow ($	$[0,\ldots,6],$	[7],	[8],	$A_7$
22.	$No-Arc^n \implies ($	$[0,\ldots,5],$	[6,7],	[8],	$A_7$
23.	$No-Arc^n \implies ($		[5,6,7],	[8],	$A_7$
24.	$No-Arc^n \implies ($	$[0,\ldots,3],$	$[4, \ldots, 7],$	[8],	$A_7$
25.	$No-Arc^n \Longrightarrow ($	[0, 1, 2],	$[3,\ldots,7],$	[8],	$A_7$
26.	$No-Arc^n \implies ($	[0, 1],	$[2,\ldots,7],$	[8],	$A_7$

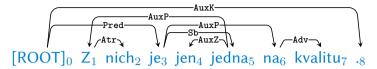
Projective list-based algorithm



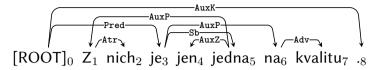
		$\lambda_1$	$\lambda_2$	β	$\overline{A}$
				,	
1.	, (	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$SHIFT^\lambda \Longrightarrow ($	[0, 1],	[],	$[2,\ldots,8],$	Ø )
3.	$Right-Arc^n_{Atr} \Longrightarrow ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$SHIFT^\lambda \Longrightarrow ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$ )
5.	$No\text{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$ )
6.	$No-Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$SHIFT^{\lambda} \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$SHIFT^\lambda \implies ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	,
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No\text{-}Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	
13.	$Left\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($		$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No-Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No\text{-}Arc^n \implies ($	$[0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$
17.	$Right\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$
18.	$Shift^\lambda \implies ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right ext{-}Arc^n_{\mathtt{Adv}} \implies ($	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad)$
20.	$Shift^\lambda \implies ($	$[0,\ldots,7],$	[],	[8],	$A_7$
21.	$No-Arc^n \implies ($	$[0,\ldots,6],$	[7],	[8],	$A_7$
22.	$No ext{-}Arc^n \implies ($	$[0,\ldots,5],$	[6, 7],	[8],	$A_7$
23.	$No ext{-}Arc^n \implies ($		[5, 6, 7],	[8],	$A_7$
24.	$No-Arc^n \implies ($	$[0,\ldots,3],$	$[4,\ldots,7],$	[8],	$A_7$
25.	$No\text{-}Arc^n \Longrightarrow ($	[0, 1, 2],	$[3,\ldots,7],$	[8],	$A_7$
26.	$No ext{-}Arc^n \implies ($	[0, 1],	$[2,\ldots,7],$	[8],	$A_7$
27.	$No\text{-}Arc^n \implies ($	[0],	$[1,\ldots,7],$	[8],	$A_7$



		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$Shift^\lambda \Longrightarrow ($	[0, 1],	[],	$[2, \ldots, 8],$	Ø )
3.	$Right ext{-}Arc^n_\mathtt{Atr} \implies ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$SHIFT^\lambda \Longrightarrow ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$
5.	$No-Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$ )
6.	$No-Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$ )
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \mathtt{Pred}, 3)  )$
8.	$SHIFT^\lambda \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$
11.	$Right-Arc^n_{Sb} \Longrightarrow ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No-Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$
13.	$Left\text{-}Arc^n_{\mathtt{AuxP}} \implies ($	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No\text{-}Arc^n \implies ($	$[0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$ )
17.	$Right ext{-}Arc^n_{\mathtt{AuxP}} \implies ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)$ )
18.	$SHIFT^\lambda \Longrightarrow ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right ext{-}Arc^n_{\mathtt{Ady}} \Longrightarrow$ (	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad)$
20.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,7],$	[],	[8],	$A_7$
21.	$No\text{-}Arc^n \implies ($	$[0,\ldots,6],$	[7],	[8],	$A_7$
22.	$No ext{-}Arc^n \implies ($	$[0,\ldots,5],$	[6, 7],	[8],	$A_7$
23.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5, 6, 7],	[8],	$A_7$
24.	$No ext{-}Arc^n \implies ($	$[0,\ldots,3],$	$[4,\ldots,7],$	[8],	$A_7$
25.	$No-Arc^n \implies ($	[0, 1, 2],	$[3,\ldots,7],$	[8],	$A_7$
26.	$No ext{-}Arc^n \implies ($	[0, 1],	$[2,\ldots,7],$	[8],	$A_7$
27.	$No ext{-}Arc^n \implies ($	[0],	$[1,\ldots,7],$	[8],	$A_7$
28.	$Right ext{-}Arc^n_{\mathtt{AuxK}} \implies ($	[],	$[0,\ldots,7],$	[8],	$A_8 = A_7 \cup (0, \mathtt{AuxK}, 8)  )$



		$\lambda_1$	$\lambda_2$	β	A	
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )	
2.	$Shift^\lambda \Longrightarrow ($	[0, 1],	[],	$[2,\ldots,8],$	Ø )	
3.	$Right ext{-}Arc^n_{\mathtt{Atr}} \Longrightarrow ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $	
4.	$Shift^\lambda \Longrightarrow \ ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$ )	
5.	$No ext{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$ )	
6.	$No ext{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$ )	
7.	$Right ext{-}Arc^n_{Pred} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$	
8.	Shift $^{\lambda} \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$ )	
9.	$Shift^\lambda \implies \ ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$ )	
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	$A_3 = A_2 \cup (5, \mathtt{AuxZ}, 4)  )$	
11.	$Right ext{-}Arc^n_{\mathtt{Sb}} \implies ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $	
12.	$No ext{-}Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	$A_4$	
13.	$Left\text{-}Arc^n_{\mathtt{Aux}\mathtt{P}} \implies ($	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)$ )	
14.	$Shift^\lambda \implies \ ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$ )	
15.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$ )	
16.	$No ext{-}Arc^n \implies ($	$[0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$ )	
17.	$Right ext{-}Arc^n_{\mathtt{Aux}  extstyle{P}} \implies ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)$ )	
18.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$	
19.	$Right ext{-}Arc^n_{\mathtt{Ady}} \Longrightarrow ($	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad )$	
20.	$Shift^\lambda \Longrightarrow \ ($	$[0,\ldots,7],$	[],	[8],	$A_7$	
21.	$No ext{-}Arc^n \implies ($	$[0,\ldots,6],$	[7],	[8],	$A_7$	
22.	$No ext{-}Arc^n \implies ($	$[0,\ldots,5],$	[6, 7],	[8],	$A_7$	
23.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5, 6, 7],	[8],	$A_7$	,
24.	$No ext{-}Arc^n \implies ($	$[0,\ldots,3],$	$[4,\ldots,7],$	[8],	$A_7$	,
25.	$No\text{-}Arc^n \Longrightarrow ($	[0, 1, 2],	$[3,\ldots,7],$	[8],	$A_7$	,
26.	$No-Arc^n \Longrightarrow ($	[0, 1],	$[2,\ldots,7],$	[8],	$A_7$	,
27.	$No-Arc^n \implies ($	[0],	$[1,\ldots,7],$	[8],	$A_7$	,
28.	$Right-Arc^n_{Aux_i^K} \Longrightarrow ($	[],	$[0,\ldots,7],$	[8],	$A_8 = A_7 \cup (0, \mathtt{AuxK}, 8)  )$	
29.	$Shift^\lambda \implies \ ($	$[0,\ldots,8],$	[],	[],	$A_8$	



		$\lambda_1$	$\lambda_2$	β	A
1.	(	[0],	[],	$[1,\ldots,8],$	Ø )
2.	$SHIFT^\lambda \Longrightarrow ($	[0, 1],	[],	$[2,\ldots,8],$	Ø )
3.	$Right ext{-}Arc^n_\mathtt{Atr} \implies ($	[0],	[1],	$[2,\ldots,8],$	$A_1 = (1, \texttt{Atr}, 2) $
4.	$SHIFT^\lambda \Longrightarrow ($	[0, 1, 2],	[],	$[3,\ldots,8],$	$A_1$ )
5.	$No ext{-}Arc^n \implies ($	[0, 1],	[2],	$[3,\ldots,8],$	$A_1$
6.	$No ext{-}Arc^n \implies ($	[0],	[1, 2],	$[3,\ldots,8],$	$A_1$
7.	$Right ext{-}Arc^n_{\mathtt{Pred}} \implies ($	[],	[0, 1, 2],	$[3,\ldots,8],$	$A_2 = A_1 \cup (0, \texttt{Pred}, 3)  )$
8.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,3],$	[],	$[4,\ldots,8],$	$A_2$
9.	$Shift^\lambda \implies \ ($	$[0,\ldots,4],$	[],	$[5,\ldots,8],$	$A_2$
10.	$Left\text{-}Arc^n_{\mathtt{AuxZ}} \implies ($	$[0,\ldots,3],$	[4],	$[5,\ldots,8],$	- , , , ,
11.	$Right ext{-}Arc^n_\mathtt{Sb} \implies ($	[0, 1, 2],	[3, 4],	$[5,\ldots,8],$	$A_4 = A_3 \cup (3, \operatorname{Sb}, 5) $
12.	$No ext{-}Arc^n \implies ($	[0, 1],	[2, 3, 4],	$[5,\ldots,8],$	•
13.	$Left\text{-}Arc^n_{\mathtt{Aux}^{\mathbf{p}}} \implies ($	[0],	$[1,\ldots,4],$	$[5,\ldots,8],$	$A_5 = A_4 \cup (5, \mathtt{AuxP}, 1)  )$
14.	$Shift^\lambda \implies \ ($	$[0,\ldots,5],$	[],	[6, 7, 8],	$A_5$
15.	$No ext{-}Arc^n \implies ($	$[0,\ldots,4],$	[5],	[6, 7, 8],	$A_5$
16.	$No ext{-}Arc^n \implies ($	$[0,\ldots,3],$	[4, 5],	[6, 7, 8],	$A_5$
17.	$Right\text{-}Arc^n_{\mathtt{Aux}^\mathtt{P}} \implies ($	[0, 1, 2],	[3, 4, 5],	[6, 7, 8],	$A_6 = A_5 \cup (3, \mathtt{AuxP}, 6)  )$
18.	$Shift^\lambda \Longrightarrow ($	$[0,\ldots,6],$	[],	[7, 8],	$A_6$
19.	$Right ext{-}Arc^n_{\mathtt{Adv}} \implies ($	$[0,\ldots,5],$	[6],	[7, 8],	$A_7 = A_6 \cup (6, \mathtt{Adv}, 7) \qquad )$
20.	$Shift^\lambda \implies \ ($	$[0,\ldots,7],$	[],	[8],	$A_7$
21.	$No ext{-}Arc^n \implies ($	$[0,\ldots,6],$	[7],	[8],	$A_7$
22.	$No-Arc^n \implies ($	$[0,\ldots,5],$	[6, 7],	[8],	$A_7$
23.	$No-Arc^n \implies ($	$[0,\ldots,4],$	[5, 6, 7],	[8],	$A_7$
24.	$No-Arc^n \implies ($	$[0,\ldots,3],$	$[4,\ldots,7],$	[8],	$A_7$
25.	$No-Arc^n \implies ($	[0, 1, 2],	$[3,\ldots,7],$	[8],	$A_7$
26.	$No-Arc^n \implies ($	[0, 1],	$[2,\ldots,7],$	[8],	$A_7$
27.	$\operatorname{No-Arc}^n \Longrightarrow ($	[0],	$[1, \ldots, 7],$	[8],	$A_7$
28.	RIGHT-ARC $_{\text{Aux}_{\lambda}}^{n} \Longrightarrow ($	[],	$[0,\ldots,7],$	[8],	$A_8 = A_7 \cup (0, \mathtt{AuxK}, 8)  )$
29.	$SHIFT^\lambda \implies \ ($	$[0,\ldots,8],$	[],	[],	$A_8$

# **Transitions** LEFT-ARC, $(\lambda_1|i,\lambda_2,i|\beta,A) \Rightarrow (\lambda_1,[],i|\beta,A\cup\{(i,l,i)\})$ RIGHT-ARC<sup>p</sup> $(\lambda_1|i,\lambda_2,j|\beta,A) \Rightarrow (\lambda_1|i|j,[],\beta,A\cup\{(i,l,j)\})$ NO-ARC<sup>p</sup> $(\lambda_1|i,\lambda_2,\beta,A) \Rightarrow (\lambda_1,i|\lambda_2,\beta,A)$ $SHIFT^{\lambda}$ $(\lambda_1, \lambda_2, i | \beta, A) \Rightarrow (\lambda_1, \lambda_2 | i, [], \beta, A)$ **Preconditions** LEFT-ARC $_{1}^{p}$ $\neg [i = 0]$ $\neg \exists k \exists l' [(k, l', i) \in A]$ RIGHT-ARC<sub>1</sub><sup>p</sup> $\neg \exists k \exists l'[(k, l', j) \in A]$ No-Arc<sup>p</sup> $\exists k \exists l[(k,l,i) \in A]$

Figure: Transitions for the arc-eager stack-based parsing algorithm.

The projective, list-based parser uses the same basic strategy as its non-projective counterpart, but skips any pair (i, j) that could give rise to a non-projective dependency arc.

Skipping many node pairs makes it more efficient in practice, although the worst-case time complexity remains the same.

Illustration Transition sequence for projective English example sentence (nearly identical to the arc-eager stack-based sequence):

# [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>

	$\lambda_1$	$\lambda_2$	β	A	
1.	( [0],	[],	$[1,\ldots,9],$	Ø	)

Illustration Transition sequence for projective English example sentence (nearly identical to the arc-eager stack-based sequence):

# [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>

	$\lambda_1$	$\lambda_2$	β	A	
1.	( [0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies ( [0,1],$	[],	$[2,\ldots,9],$	Ø	)

Illustration Transition sequence for projective English example sentence (nearly identical to the arc-eager stack-based sequence):

-NMOD-[ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>

		$\lambda_1$	$\lambda_2$	β	A	
1.		( [0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies$	([0,1],	[],	$[2,\ldots,9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	([0],	[],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)

—NMOD— [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>

			$\lambda_1$	$\lambda_2$	β	A	
1.		(	[0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \Longrightarrow$	(	[0, 1],	[],	$[2, \ldots, 9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \Longrightarrow$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$	)

NMOD—SBJ [ROOT]<sub>0</sub> Economic<sub>1</sub> news<sub>2</sub> had<sub>3</sub> little<sub>4</sub> effect<sub>5</sub> on<sub>6</sub> financial<sub>7</sub> markets<sub>8</sub> .<sub>9</sub>

			$\lambda_1$	7	$\lambda_2$	β	A	
1.		(	[0],	[	],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[	],	$[2,\ldots,9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[	],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],	[	],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[	],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)

			$\lambda_1$	$\lambda_2$	β	A	
1.		(	[0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \Longrightarrow$	(	[0, 1],	[],	$[2, \ldots, 9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[]	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{SBJ} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right\text{-}Arc^p_{\mathtt{ROOT}} \Longrightarrow$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)

			$\lambda_1$	,	$\lambda_2$	β	A	
1.		(	[0],		],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],		],	$[2,\ldots,9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],		],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],		],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{SBJ} \implies$	(	[0],		],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right\text{-}Arc^p_{\mathtt{ROOT}} \Longrightarrow$	(	[0, 3],		],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$Shift^\lambda \Longrightarrow$				],	$[5,\ldots,9],$	$A_3$	)

			$\lambda_1$	$\lambda_2$	eta	A	
1.		(	[0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],		$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^p_{\mathtt{ROOT}} \implies$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3],	[],	$[5,\ldots,9],$	$A_4=A_3\cup\{(5,\mathtt{NMOD},4)\}$	)

			$\lambda_1$	$\lambda_2$	$\beta$	A	
1.		(	[0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2, \ldots, 9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3,\mathtt{SBJ},2)\}$	)
6.	$Right\text{-}Arc^p_{\mathtt{ROOT}} \Longrightarrow$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3],	[],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.				[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)

			$\lambda_1$	$\lambda_2$	β	A	
1.		(	[0],	Π.	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0,1],	[],	$[2,\ldots,9],$		)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[]	$[2,\ldots,9],$	$A_1=\{(2,\mathtt{NMOD},1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^p_{\mathtt{ROOT}} \implies$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[]	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3],	[],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$Right ext{-}Arc^p_{\mathtt{OBJ}} \Longrightarrow$	(	[0, 3, 5],	[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies$			[],	[7, 8, 9],	$A_6=A_5\cup\{(5,\mathtt{NMOD},6)\}$	)

Projective list-based algorithm Non-projective list-based algorithm

			$\lambda_1$	$\lambda_2$	eta	A	
1.		(	[0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2,\ldots,9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[],	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],	[]	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^p_{\mathtt{ROOT}} \Longrightarrow$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3],	[],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	Rіснт-Arc $_{\mathtt{OBJ}}^{p} \Longrightarrow$	(	[0, 3, 5],	[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[7, 8, 9],	$A_6=A_5\cup\{(5,\mathtt{NMOD},6)\}$	)
11.	$Shift^\lambda \implies$	(	[0, 3, 5, 6, 7],	[],	[8, 9],	$A_6$	)

Projective list-based algorithm

	10			1	0 0		
•			$\lambda_1$	$\lambda_2$	β	A	_
1.		(	[0],	[],	$[1,\ldots,9],$	Ø )	_
2.	$Shift^\lambda \Longrightarrow$	(	[0,1],	[],	$[2, \ldots, 9],$	Ø )	
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[],	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\}$	
4.	$Shift^\lambda \implies$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$	
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\} $	
6.	$Right ext{-}Arc^p_{\mathtt{ROOT}} \Longrightarrow$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, ROOT, 3)\}$	
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$	
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0,3],	[],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \texttt{NMOD}, 4)\} $	
9.	$Right\text{-}Arc^p_{\mathtt{OBJ}} \Longrightarrow$	(	[0, 3, 5],	[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, OBJ, 5)\} $	
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \texttt{NMOD}, 6)\}  )$	
11.	$Shift^\lambda \implies$	(	[0, 3, 5, 6, 7],	[],	[8, 9],	$A_6$	
12.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[8, 9],	$A_7 = A_6 \cup \{(8, NMOD, 7)\}$	

Non-projective list-based algorithm

Projective list-based algorithm

	30	_		-			
			$\lambda_1$	$\lambda_2$	$\beta$	A	
1.		(	[0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \implies$	(	[0,1],	[],	$[2, \ldots, 9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	$\bar{[]}$ ,	$[2,\ldots,9],$	$A_1 = \{(2, \mathtt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \Longrightarrow$	(	[0,2],	[],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, \mathtt{SBJ}, 2)\}$	)
6.	$Right-Arc_{ROOT}^p \Longrightarrow$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3],	[],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$Right ext{-}Arc^p_{\mathtt{OBJ}} \implies$	(	[0, 3, 5],	[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \mathtt{NMOD}, 6)\}$	)
11.	$Shift^\lambda \implies$	(	[0, 3, 5, 6, 7],	[],	[8, 9],	$A_6$	)
12.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[8, 9],	$A_7 = A_6 \cup \{(8, \mathtt{NMOD}, 7)\}$	)
13.	$Right-Arc_{PMOD}^{p} \Longrightarrow$	(	[0, 3, 5, 6, 8],	[],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)

Projective list-based algorithm

r.	tee: jo zeenemei	newsz naag .	111104	chiecti oni ii	inameta, inameta, ig	
		$\lambda_1$	$\lambda_2$	β	A	_
1.	(	[0],	[],	$[1,\ldots,9],$	Ø )	_
2.	$Shift^\lambda \Longrightarrow \ ($	[0,1],	[],	$[2, \ldots, 9],$	Ø )	
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies ($	[0],	[]	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\} $	
4.	$Shift^\lambda \Longrightarrow \ ($	[0,2],	[],	$[3,\ldots,9],$	$A_1$	
5.	$Left\text{-}Arc^p_{SBJ} \implies ($	[0],	[]	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\}$	
6.	$\operatorname{Right-Arc}_{\operatorname{ROOT}}^{p^{\operatorname{SB3}}} \Longrightarrow ($	[0, 3],	[]	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	
7.	$Shift^\lambda \Longrightarrow \ ($	[0,3,4],	[],	$[5,\ldots,9],$	$A_3$	
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies ($	[0,3],	[]	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, NMOD, 4)\}$	
9.	$Right\text{-}Arc^p_{OBJ} \Longrightarrow ($	[0,3,5],	[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \text{OBJ}, 5)\}$	
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies ($	[0, 3, 5, 6],	[]	[7, 8, 9],	$A_6 = A_5 \cup \{(5, NMOD, 6)\}$	
11.	$Shift^\lambda \Longrightarrow ($	[0,3,5,6,7],	[],	[8, 9],	$A_6$	
12.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies ($	[0, 3, 5, 6],	[]	[8, 9],	$A_7 = A_6 \cup \{(8, NMOD, 7)\}$	
13.	$Right\text{-}Arc^p_{\mathtt{PMOD}} \implies ($	[0, 3, 5, 6, 8],	[]	[9],	$A_8 = A_7 \cup \{(6, PMOD, 8)\}$	
14.	$No\text{-}Arc^p \implies ($	[0, 3, 5, 6],	[8],	[9],	$A_8$	

	too1ju Leonomei	• • •	ews <sub>2</sub> maa <sub>3</sub> r	rttic4 circ	5Ct <sub>5</sub> OH <sub>6</sub> H	Harretary markets8 .g
			$\lambda_1$	$\lambda_2$	β	A
1.		(	[0],	[],	$[1,\ldots,9],$	Ø )
2.	$Shift^\lambda \Longrightarrow$	(	[0,1],	[],	$[2, \ldots, 9],$	Ø )
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[]	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\} $
4.	$Shift^\lambda \Longrightarrow$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$ )
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\} $
6.	$Right ext{-}Arc^p_{\mathtt{ROOT}} \implies$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$ )
7.	Shift $^{\lambda} \Longrightarrow$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3],	[],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \texttt{NMOD}, 4)\}  )$
9.	Right-Arc $_{\mathtt{OBJ}}^{p} \Longrightarrow$	(	[0, 3, 5],	[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, OBJ, 5)\}$
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \texttt{NMOD}, 6)\}  )$
11.	Shift $^{\lambda} \Longrightarrow$	(	[0, 3, 5, 6, 7],	[],	[8, 9],	$A_6$
12.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[8, 9],	$A_7 = A_6 \cup \{(8, \texttt{NMOD}, 7)\}  )$
13.	$Right\text{-}Arc^p_{\mathtt{PMOD}} \Longrightarrow$	(	[0, 3, 5, 6, 8],	[],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$ )
14.	$No ext{-}Arc^p \implies$	(	[0, 3, 5, 6],	[8],	[9],	$A_8$
15.	$No ext{-}arc^p \implies$	(	[0, 3, 5],	[6, 8],	[9],	$A_8$

Ľ	COOTIN Leonomici	110	wsz nada i	rttic <sub>4</sub> circ	on <sub>0</sub> 11	marietar, markets8 .9	
			$\lambda_1$	$\lambda_2$	β	A	
1.		(	[0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \Longrightarrow$	(	[0,1],	[],	$[2, \ldots, 9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[],	$[2,\ldots,9],$	$A_1=\{(2,\mathtt{NMOD},1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^p_{\mathtt{ROOT}} \implies$	(	[0,3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0,3],	$\ddot{[}$	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$Right ext{-}Arc^p_{\mathtt{OBJ}} \Longrightarrow$	(	[0, 3, 5],	$\ddot{[}$	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \mathtt{NMOD}, 6)\}$	)
11.	$Shift^\lambda \implies$	(	[0, 3, 5, 6, 7],	[],	[8, 9],	$A_6$	)
12.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[8, 9],	$A_7 = A_6 \cup \{(8, \mathtt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^p_{\mathtt{PMOD}} \implies$	(	[0, 3, 5, 6, 8],	[],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	$No ext{-}Arc^p \implies$	(	[0, 3, 5, 6],	[8],	[9],	$A_8$	)
15.	$No ext{-}arc^p \implies$	(	[0, 3, 5],	[6, 8],	[9],	$A_8$	)
16.	$No ext{-}arc^p \implies$	(	[0, 3],	[5, 6, 8],	[9],	$A_8$	)

			$\lambda_1$	$\lambda_2$	eta	A
1.		(	[0],	[],	$[1,\ldots,9],$	Ø )
2.	$Shift^\lambda \implies$	(	[0, 1],	[],	$[2, \ldots, 9],$	Ø )
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[],	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\} $
4.	$Shift^\lambda \implies$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$ )
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2 = A_1 \cup \{(3, SBJ, 2)\}$
6.	$Right ext{-}Arc^p_{\mathtt{ROOT}} \implies$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3],	[],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \texttt{NMOD}, 4)\}  )$
9.	$Right ext{-}Arc^p_{\mathtt{OBJ}} \implies$	(	[0, 3, 5],	[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \text{OBJ}, 5)\}$
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[7, 8, 9],	$A_6 = A_5 \cup \{(5, \texttt{NMOD}, 6)\}  )$
11.	$Shift^\lambda \implies$	(	[0, 3, 5, 6, 7],	[],	[8, 9],	$A_6$
12.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[8, 9],	$A_7 = A_6 \cup \{(8, \text{NMOD}, 7)\}$
13.	$Right ext{-}Arc^p_{\mathtt{PMOD}} \implies$	(	[0, 3, 5, 6, 8],	[],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}  )$
14.	$No ext{-}Arc^p \implies$	(	[0, 3, 5, 6],	[8],	[9],	$A_8$
15.	$No ext{-}arc^p \implies$	(	[0, 3, 5],	[6, 8],	[9],	$A_8$
16.	$No ext{-}arc^p \implies$	(	[0, 3],	[5, 6, 8],	[9],	$A_8$
17.	$Right ext{-}Arc^p_\mathtt{P} \Longrightarrow$	(	[0, 3, 9],	[],	[],	$A_9 = A_8 \cup \{(3, P, 9)\}$

			$\lambda_1$	$\lambda_2$	eta	A	
1.		(	[0],	[],	$[1,\ldots,9],$	Ø	)
2.	$Shift^\lambda \Longrightarrow$	(	[0, 1],	[]	$[2, \ldots, 9],$	Ø	)
3.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0],	[],	$[2,\ldots,9],$	$A_1 = \{(2, \texttt{NMOD}, 1)\}$	)
4.	$Shift^\lambda \implies$	(	[0, 2],	[],	$[3,\ldots,9],$	$A_1$	)
5.	$Left\text{-}Arc^p_{\mathtt{SBJ}} \implies$	(	[0],	[],	$[3,\ldots,9],$	$A_2=A_1\cup\{(3,\mathtt{SBJ},2)\}$	)
6.	$Right ext{-}Arc^p_{\mathtt{ROOT}} \implies$	(	[0, 3],	[],	$[4,\ldots,9],$	$A_3 = A_2 \cup \{(0, \mathtt{ROOT}, 3)\}$	)
7.	$Shift^\lambda \implies$	(	[0, 3, 4],	[],	$[5,\ldots,9],$	$A_3$	)
8.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3],	[],	$[5,\ldots,9],$	$A_4 = A_3 \cup \{(5, \mathtt{NMOD}, 4)\}$	)
9.	$Right ext{-}Arc^p_{\mathtt{OBJ}} \Longrightarrow$	(	[0, 3, 5],	[],	$[6,\ldots,9],$	$A_5 = A_4 \cup \{(3, \mathtt{OBJ}, 5)\}$	)
10.	$Right ext{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[7, 8, 9],	$A_6=A_5\cup\{(5,\mathtt{NMOD},6)\}$	)
11.	$Shift^\lambda \implies$			[],	[8, 9],	$A_6$	)
12.	$Left\text{-}Arc^p_{\mathtt{NMOD}} \implies$	(	[0, 3, 5, 6],	[],	[8, 9],	$A_7 = A_6 \cup \{(8, \mathtt{NMOD}, 7)\}$	)
13.	$Right ext{-}Arc^p_{\mathtt{PMOD}} \implies$		[0, 3, 5, 6, 8],	[],	[9],	$A_8 = A_7 \cup \{(6, \texttt{PMOD}, 8)\}$	)
14.	$No ext{-}Arc^p \implies$	(	[0, 3, 5, 6],	[8],	[9],	$A_8$	)
15.	$No ext{-}arc^p \implies$	(	[0, 3, 5],	[6, 8],	[9],	$A_8$	)
16.	$No ext{-}arc^p \Longrightarrow$	(	[0, 3],	[5, 6, 8],	[9],	$A_8$	)
17.	$Right ext{-}Arc^p_{\mathtt{P}} \implies$	(	[0, 3, 9],	[],	[],	$A_9 = A_8 \cup \{(3, P, 9)\}$	)

**Evaluation of the four algorithms** in deterministic data-driven parsing: Use an oracle approximated by a classifier trained on treebank data to analyze of the accuracy and efficiency of these systems.

■ Data: CoNLL-X shared task multilingual dependency parsing

**Evaluation of the four algorithms** in deterministic data-driven parsing: Use an oracle approximated by a classifier trained on treebank data to analyze of the accuracy and efficiency of these systems.

■ Data: CoNLL-X shared task multilingual dependency parsing

Data sets. Tok = number of tokens ( $\times$ 1000); Sen = number of sentences ( $\times$ 1000); T/S = tokens per sentence (mean); Lem = lemmatization present; CPoS = number of coarse-grained part-of-speech tags; PoS = number of (fine-grained) part-of-speech tags; MSF = number of morphosyntactic features (split into atoms); Dep = number of dependency types; NPT = proportion of non-projective dependencies/tokens (%); NPS = proportion of non-projective dependency graphs/sentences (%).

Language	Tok	Sen	T/S	Lem	CPoS	PoS	MSF	Dep	NPT	NPS
Arabic	54	1.5	37.2	yes	14	19	19	27	0.4	11.2
Bulgarian	190	14.4	14.8	no	11	53	50	18	0.4	5.4
Chinese	337	57.0	5.9	no	22	303	0	82	0.0	0.0
Czech	1,249	72.7	17.2	yes	12	63	61	78	1.9	23.2
Danish	94	5.2	18.2	no	10	24	47	52	1.0	15.6
Dutch	195	13.3	14.6	yes	13	302	81	26	5.4	36.4
German	700	39.2	17.8	no	52	52	0	46	2.3	27.8
Japanese	151	17.0	8.9	no	20	77	0	7	1.1	5.3
Portuguese	207	9.1	22.8	yes	15	21	146	55	1.3	18.9
Slovene	29	1.5	18.7	yes	11	28	51	25	1.9	22.2
Spanish	89	3.3	27.0	yes	15	38	33	21	0.1	1.7
Swedish	191	11.0	17.3	no	37	37	0	56	1.0	9.8
Turkish	58	5.0	11.5	yes	14	30	82	25	1.5	11.6

Figure: Data sets

1,688

Portuguese

Slovene

Spanish

Swedish

Turkish

Average

1,240

1,361

Learning and parsing time for seven parsers on six languages, measured in seconds. NP-L = non-projective list-based; P-L = projective list-based; PP-L = pseudo-projective list-based; P-E = projective arc-eager stack-based; PP-E = pseudo-projective arc-eager stack-based; P-S = projective arc-standard stack-based; PP-S = pseudo-projective arc-standard stack-based.

			Learning	Time			
Language	NP-L	P-L	PP-L	Р-Е	PP-E	P-S	PP-S
Arabic	1,814	614	603	650	647	1,639	1,636
Bulgarian	6,796	2,918	2,926	2,919	2,939	3,321	3,391
Chinese	17,034	13,019	13,019	13,029	13,029	13,705	13,705
Czech	546,880	250,560	248,511	279,586	280,069	407,673	406,857
Danish	2,964	1,248	1,260	1,246	1,262	643	647
Dutch	7,701	3,039	2,966	3,055	2,965	7,000	6,812
German	48,699	16,874	17,600	16,899	17,601	24,402	24,705
Japanese	211	191	188	203	208	199	199
Portuguese	25,621	8,433	8,336	8,436	8,335	7,724	7,731
Slovene	167	78	90	93	99	86	90
Spanish	1,999	562	566	565	565	960	959
Swedish	2,410	942	1,020	945	1,022	1,350	1,402
Turkish	720	498	519	504	516	515	527
Average	105,713	46,849	46,616	51,695	51,876	74,798	74,691
			Parsing	Time			
Language	NP-L	P-L	PP-L	Р-Е	PP-E	P-S	PP-S
Arabic	213	103	131	108	135	196	243
Bulgarian	139	93	102	93	103	135	147
Chinese	1,008	855	855	855	855	803	803
Czech	5,244	3,043	5,889	3,460	6,701	3,874	7,437
Danish	109	66	83	66	83	82	106
Dutch	349	209	362	211	363	253	405
German	781	456	947	455	945	494	1,004
Japanese	10	8	8	9	10	7	7
<b>D</b> .	<b></b>	• • • •	40.4	• • • •	400	40=	

Figure: Parsing efficiency

1,496

Nivre, Joakim (Dec. 2008). "Algorithms for Deterministic Incremental Dependency Parsing". In: Computational Linguistics 34.4, pp. 513–553.