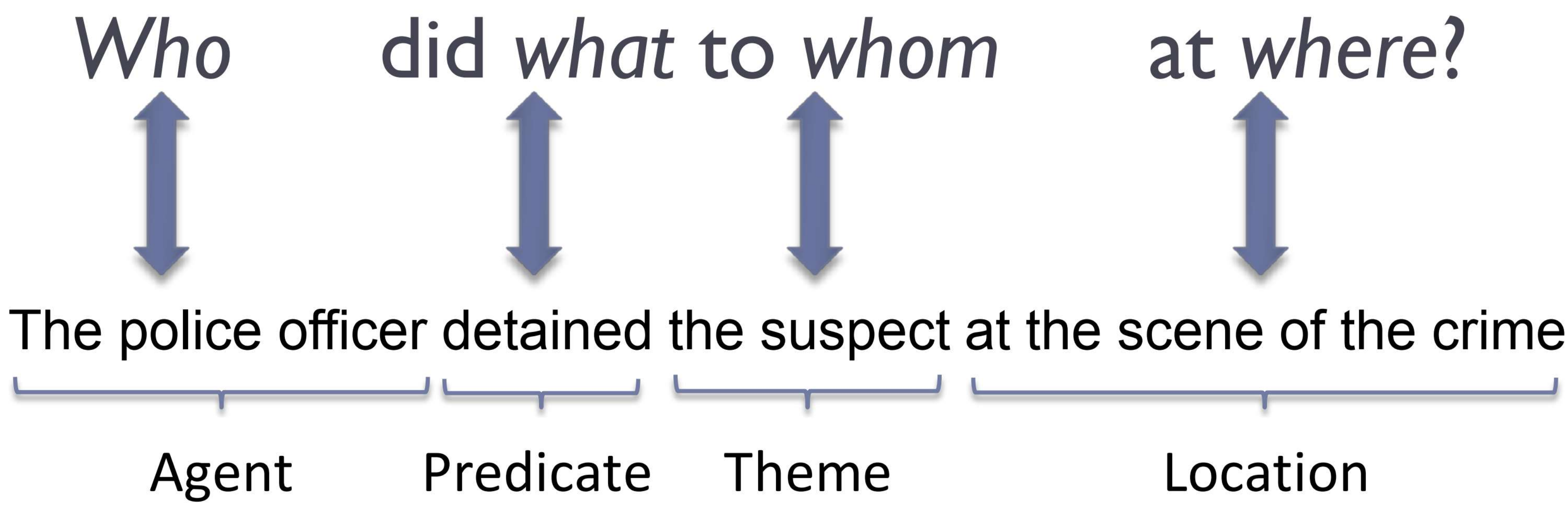
Semantic Role Labeling

Chong Xiang and Zhongqiao Gao

02/20/2020

Semantic Role Labeling



SRL introduction slides adapted from https://web.stanford.edu/~jurafsky/slp3/slides/22_SRL.pdf

B-arg0; O; B-arg1; I-arg0; B-arg2; I-arg2; TMP
B_arg0, B_v, B_arg2, B_arg2, B_arg1, B_arg1, B_argm_temp
B O B I O O O
y1 = B, y2 = O, y3 = B, y4 = I, y5 = B, y6 = I, y7 = O
B_ARG1 O B_ARG2 I_ARG2 O B_ARG3 O
Alice, gave, Bob's, mom, a, book, yesterday B_ARG0, B_V, B_ARG1, I_ARG1, B_ARG2, I_ARG2, B_ARG3
y1=B-ARG0, y2=B-v, y3=B-ARG1,y4=I-ARG1,y5=B-ARG2,y6=I-ARG2,y7=O
B O B I O B B
B, O, B, I, O, B, B
I'm not *quite* sure, but: y1=B-ARG0 y2=I-ARG0 y3=B-ARG1 y4=I-ARG1 y5=O y6=B-ARG2 y7=O
B-ARG1, V, O, I-ARG2, O, I-ARG3, O; I am very confused by the BIO sequence
B_{arg0} B_v B_{arg1} I_{arg1} B_{arg2} I_{arg2} O
Ir, O,Ir, Ir, O,O,O
y1 = B, y2 = O, y3 = B, y4 = I, y5 = B, y6 = I, y7 = O
B(arg0), B(v), B(arg1), I(arg1), B(arg2), I(arg2), O
B, O, B, I, B, I, B
y1=Alice, y2=Bob's mom, y3=a book, modifier=yesterday

The Proposition Bank (PropBank)

Give

Arg0: giver
Arg1: thing given
Arg2: entity given to

ArgM-TMP: when?
ArgM-LOC: where?
ArgM-DIR: where to/from?
ArgM-MNR: how?
ArgM-PRP/CAU: why?

Alice gave Bob's mom a book yesterday

The Proposition Bank (PropBank)

Give

Arg0: giver
Arg1: thing given
Arg2: entity given to

ArgM-TMP: when?
ArgM-LOC: where?
ArgM-DIR: where to/from?
ArgM-MNR: how?
ArgM-PRP/CAU: why?

[Arg0: Alice] gave [Arg2: Bob's mom] [Arg1: a book] [ArgM-TMP: yesterday]

BIO (Beginning-Inside-Outside) tagging

Alex is going to Los Angeles

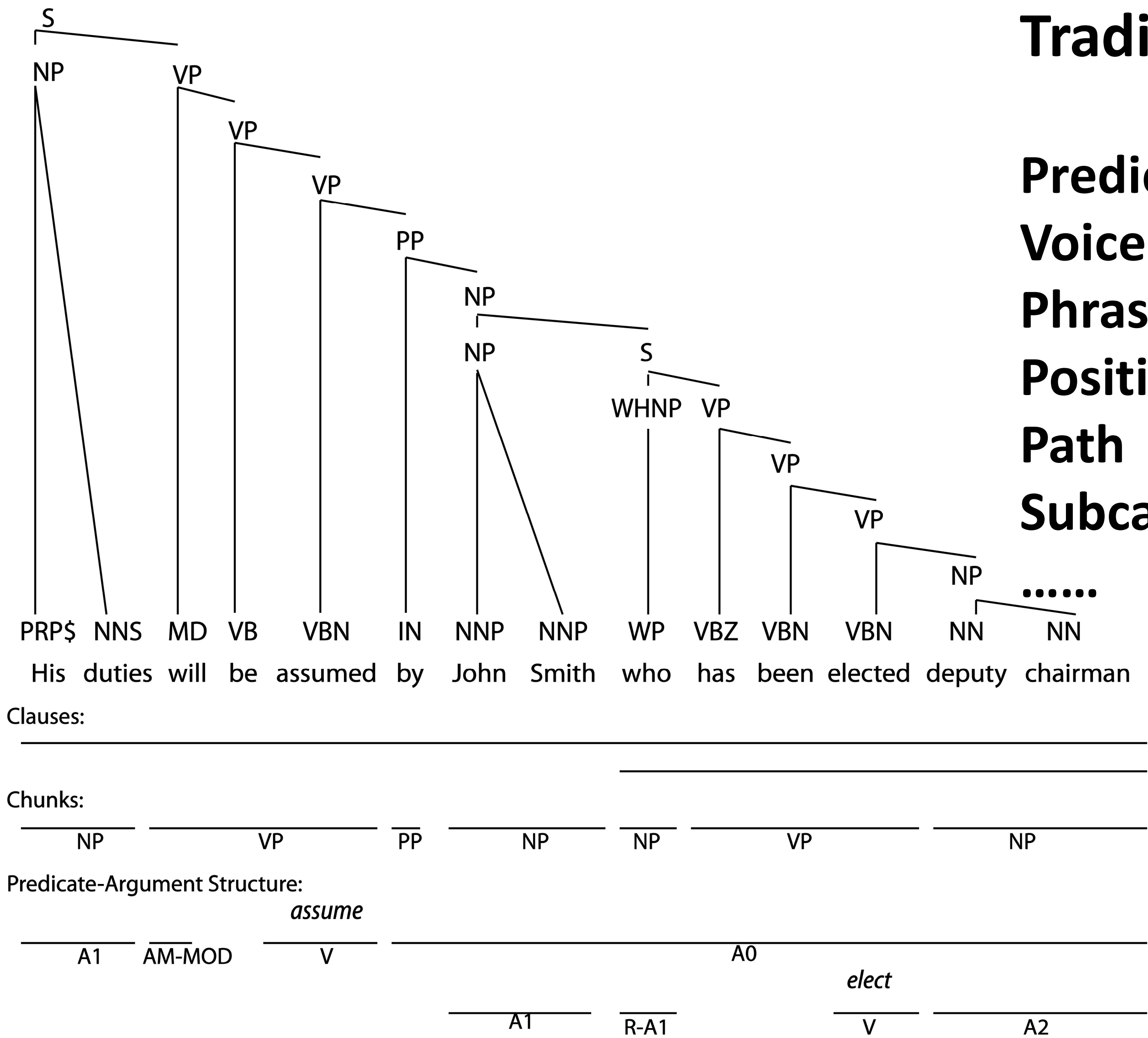
Alex: B-PER
is: O
going: O
to: O
Los: B-LOC
Angeles: I-LOC

[Arg0: Alice] gave [Arg2: Bob's mom] [Arg1: a book] [ArgM-TMP: yesterday]

Alice: B-Arg0
gave: B-v
Bob': B-Arg2
mom: I-Arg2
a: B-Arg1
book: I-Arg1
Yesterday: B-ArgM-TMP

CoNLL-2005

WORDS---->	NE--->	POS	PARTIAL_SYNT		FULL_SYNT------>	VS	TARGETS	PROPS------->	
The	*	DT	(NP*	(S*	(S(NP*	-	-	(A0*	(A0*
$	*	$	*	*	(ADJP(QP*	-	-	*	*
1.4	*	CD	*	*	*	-	-	*	*
billion	*	CD	*	*	*))	-	-	*	*
robot	*	NN	*	*	*	-	-	*	*
spacecraft	*	NN	*)	*	*)	-	-	*)	*)
faces	*	VBZ	(VP*)	*	(VP*	01	face	(V*)	*
a	*	DT	(NP*	*	(NP*	-	-	(A1*	*
six-year	*	JJ	*	*	*	-	-	*	*
journey	*	NN	*)	*	*	-	-	*	*
to	*	TO	(VP*	(S*	(S(VP*	-	-	*	*
explore	*	VB	*)	*	(VP*	01	explore	*	(V*)
Jupiter	(ORG*)	NNP	(NP*)	*	(NP(NP*)	-	-	*	(A1*
and	*	CC	*	*	*	-	-	*	*
its	*	PRP$	(NP*	*	(NP*	-	-	*	*
16	*	CD	*	*	*	-	-	*	*
known	*	JJ	*	*	*	-	-	*	*
moons	*	NNS	*)	*)	*)))))))	-	-	*)	*)
.	*	.	*	*)	*)	-	-	*	*



Traditional features:

Predicate and POS tag of predicate
Voice
Phrase type
Position
Path S↑NP↑PP↑VP↓VBN
Subcategorization
……

Figure copied from ***The Importance of Syntactic Parsing and Inference in Semantic Role Labeling***

- **Feature based Semantic Role Labeling**

  Syntax seems to be a prerequisite for SRL

- **Deep Semantic Role Labeling: What Works and What's Next ACL'2017**

  End-to-end model without syntactic input

- **Linguistically-Informed Self-Attention for Semantic Role Labeling ACL'2018**

  Explicitly model the syntactic information in neural network

- **Feature based Semantic Role Labeling**

  Syntax seems to be a prerequisite for SRL

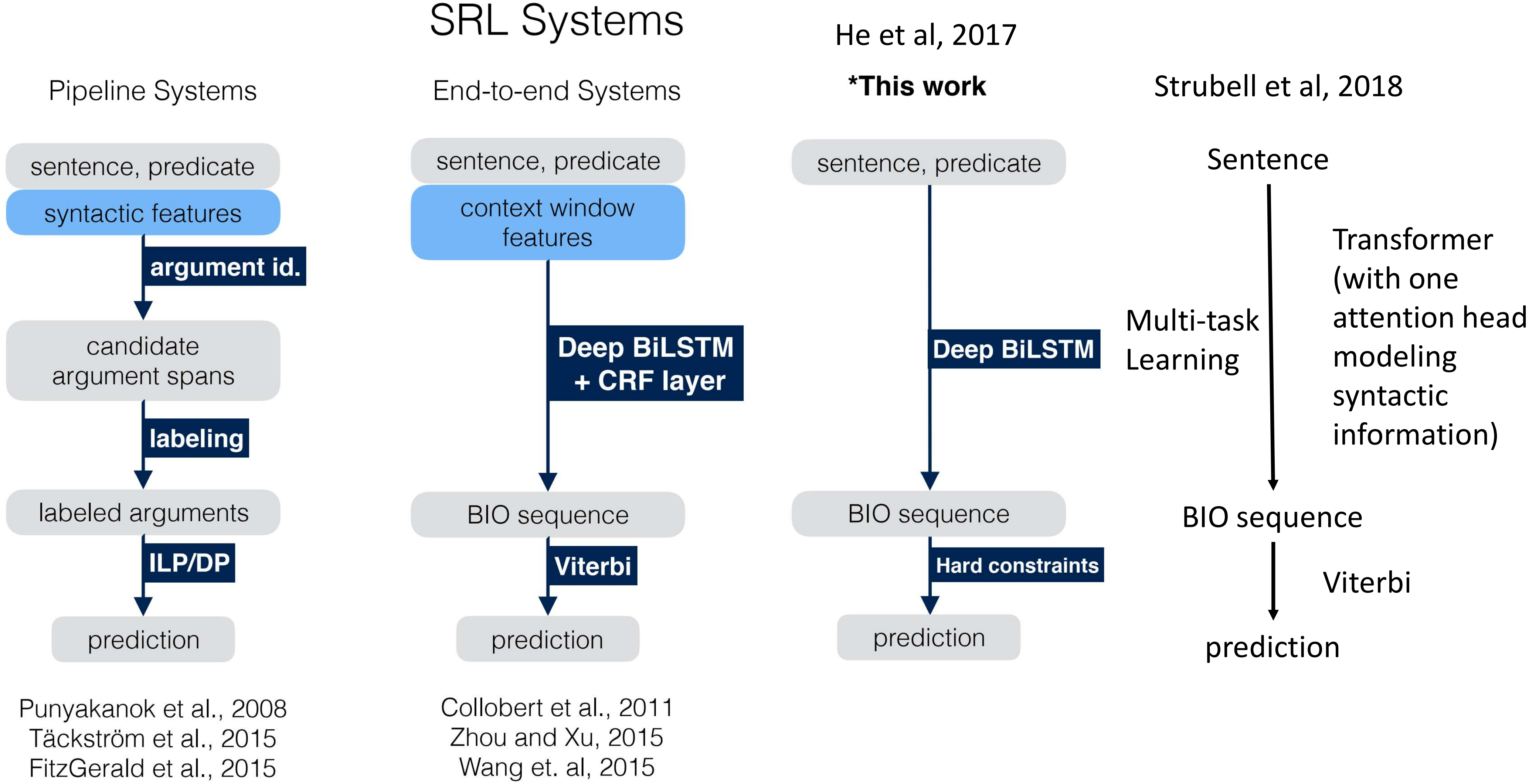
- **Deep Semantic Role Labeling: What Works and What's Next ACL'2017**

  End-to-end model without syntactic input

- **Linguistically-Informed Self-Attention for Semantic Role Labeling ACL'2018**

  Explicitly model the syntactic information in neural network

SRL Systems



He et al, 2017
Pipeline Systems
End-to-end Systems
*This work
Strubell et al, 2018
sentence, predicate
syntactic features
argument id.
candidate argument spans
labeling
labeled arguments
ILP/DP
prediction
sentence, predicate
context window features
Deep BiLSTM + CRF layer
BIO sequence
Viterbi
prediction
sentence, predicate
Deep BiLSTM
BIO sequence
Hard constraints
prediction
Sentence
Multi-task Learning
Transformer (with one attention head modeling syntactic information)
BIO sequence
Viterbi
prediction
Punyakanok et al., 2008
Täckström et al., 2015
FitzGerald et al., 2015
Collobert et al., 2011
Zhou and Xu, 2015
Wang et. al, 2015

- **Feature based Semantic Role Labeling**

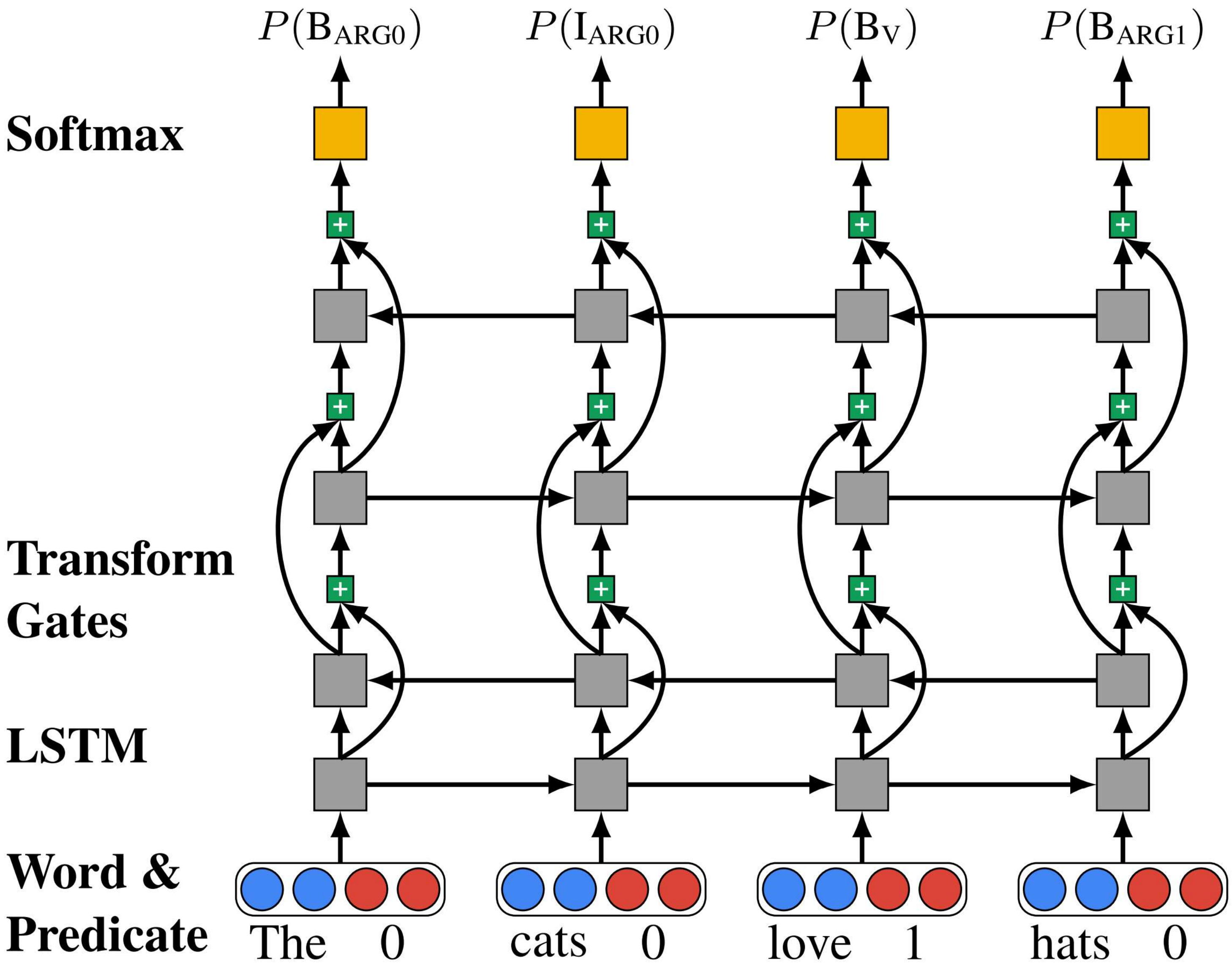
  Syntax seems to be a prerequisite for SRL

- **Deep Semantic Role Labeling: What Works and What's Next ACL'2017**

  End-to-end model without syntactic input

- **Linguistically-Informed Self-Attention for Semantic Role Labeling ACL'2018**

  Explicitly model the syntactic information in neural network



Basic idea: BiLSTM

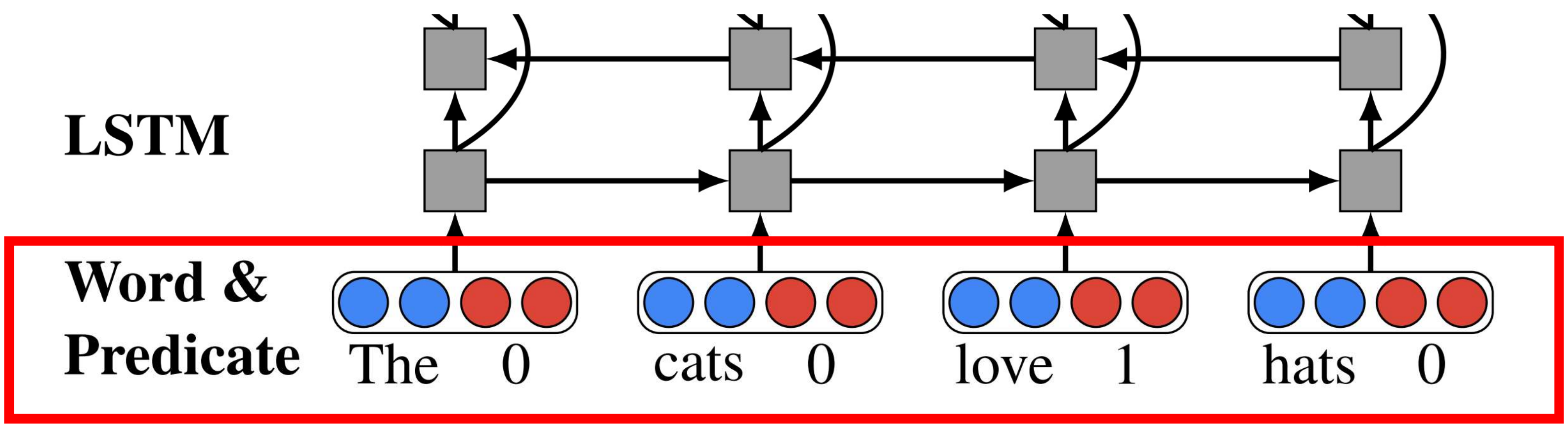
Design detail?

Training Technique?

Input: Glove word embedding and binary predicate mask

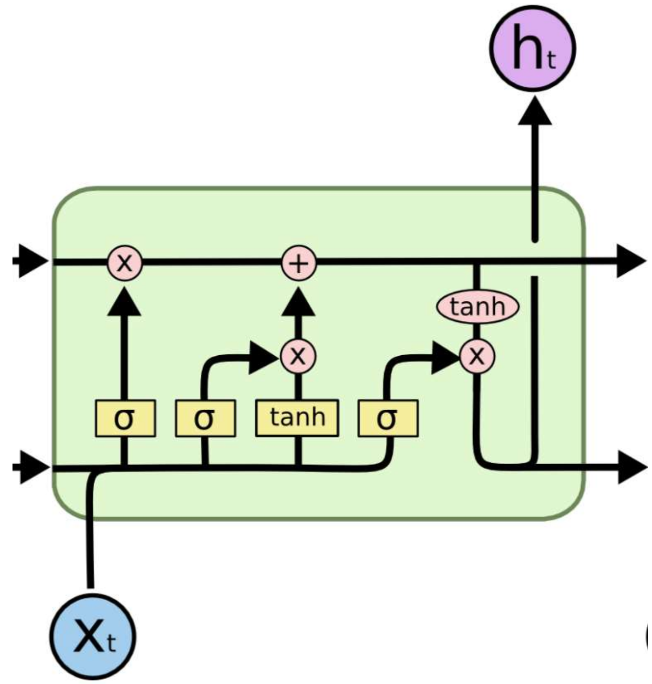
A simplification of ACL'2015 (word, predicate, predicate context, and region mark)

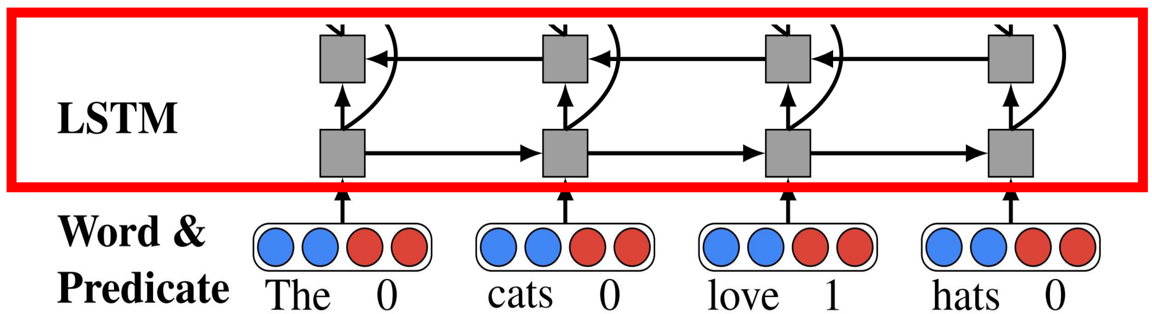
$$\boldsymbol{x}_{l,t} = \begin{cases} [\mathbf{W}_{\text{emb}}(w_t), \mathbf{W}_{\text{mask}}(t = v)] & l = 1 \\ \boldsymbol{h}_{l-1,t} & l > 1 \end{cases}$$



Basic structure: BiLSTM

$$\boldsymbol{i}_{l,t} = \sigma(\mathbf{W}_{\mathrm{i}}^{l}[\boldsymbol{h}_{l,t+\delta_l}, \boldsymbol{x}_{l,t}] + \boldsymbol{b}_{\mathrm{i}}^{l})$$
$$\boldsymbol{o}_{l,t} = \sigma(\mathbf{W}_{\mathrm{o}}^{l}[\boldsymbol{h}_{l,t+\delta_l}, \boldsymbol{x}_{l,t}] + \boldsymbol{b}_{\mathrm{o}}^{l})$$
$$\boldsymbol{f}_{l,t} = \sigma(\mathbf{W}_{\mathrm{f}}^{l}[\boldsymbol{h}_{l,t+\delta_l}, \boldsymbol{x}_{l,t}] + \boldsymbol{b}_{\mathrm{f}}^{l} + 1)$$
$$\tilde{\boldsymbol{c}}_{l,t} = \tanh(\mathbf{W}_{\mathrm{c}}^{l}[\boldsymbol{h}_{l,t+\delta_l}, \boldsymbol{x}_{l,t}] + \boldsymbol{b}_{\mathrm{c}}^{l})$$
$$\boldsymbol{c}_{l,t} = \boldsymbol{i}_{l,t} \circ \tilde{\boldsymbol{c}}_{l,t} + \boldsymbol{f}_{l,t} \circ \boldsymbol{c}_{t+\delta_l}$$
$$\boldsymbol{h}_{l,t} = \boldsymbol{o}_{l,t} \circ \tanh(\boldsymbol{c}_{l,t})$$
$$\delta_l = \begin{cases} 1 & \text{if } l \text{ is even} \\ -1 & \text{otherwise} \end{cases}$$





Network Training: Highway Connections and Recurrent Dropout

Highway Connection: somewhat like residual network

$$\boldsymbol{r}_{l,t} = \sigma(\mathbf{W}_{\mathrm{r}}^{l}[\boldsymbol{h}_{l,t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_{\mathrm{r}}^{l})$$ ← Transform gate

$$\boldsymbol{h}'_{l,t} = \boldsymbol{o}_{l,t} \circ \tanh(\boldsymbol{c}_{l,t})$$ ← LSTM output without gate

$$\boldsymbol{h}_{l,t} = \boldsymbol{r}_{l,t} \circ \boldsymbol{h}'_{l,t} + (1 - \boldsymbol{r}_{l,t}) \circ \mathbf{W}_{\mathrm{h}}^{l} \boldsymbol{x}_{l,t}$$ ← Gated output

Recurrent Dropout

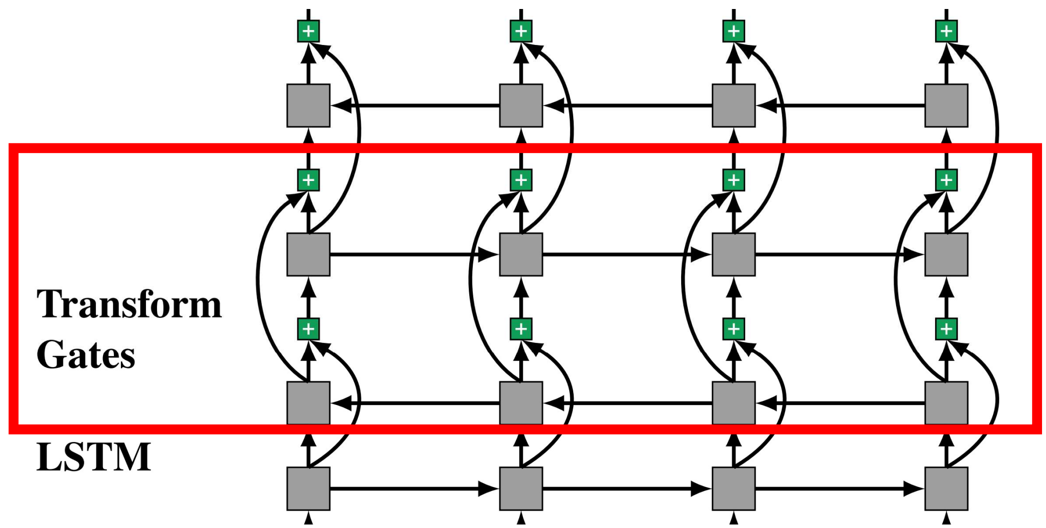
Gated output

$$\widetilde{\boldsymbol{h}}_{l,t} = \boldsymbol{r}_{l,t} \circ \boldsymbol{h}'_{l,t} + (1 - \boldsymbol{r}_{l,t}) \circ \mathbf{W}_{\mathrm{h}}^{l} \boldsymbol{x}_{l,t}$$

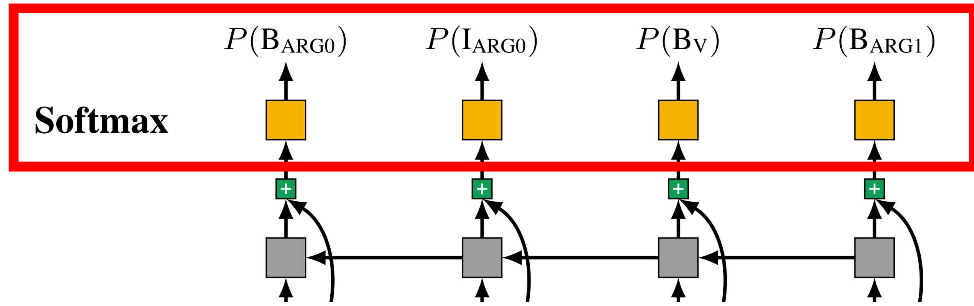
$$\boldsymbol{h}_{l,t} = \boldsymbol{z}_l \circ \widetilde{\boldsymbol{h}}_{l,t}$$

Masked/dropout output

Orthonormal initialization



Output: Constrained A* decoding



Another difference with ACL'2015 (CRF)

1. Model the dependencies between the output tags
2. Add constraints to reject invalid output
3. Use A* searching algorithm to find the "optimal" tag sequence

$$p(y_t \mid \boldsymbol{x}) \propto \exp(\mathbf{W}^{y}_{\text{tag}} \boldsymbol{h}_{L,t} + \boldsymbol{b}_{\text{tag}})$$ ⟵ Softmax output

$$f(\boldsymbol{w}, y_{1:t}) = \sum_{i=1}^{t} \log p(y_i \mid \boldsymbol{w}) - \sum_{c \in \mathcal{C}} c(\boldsymbol{w}, y_{1:i})$$ ⟵ Confidence value with penalization

$$g(\boldsymbol{w}, y_{1:t}) = \sum_{i=t+1}^{n} \max_{y_i \in \boldsymbol{T}} \log p(y_i \mid \boldsymbol{w})$$ ⟵ A* heuristic

Constraint example

BIO Constraints

Reject invalid BIO transitions, such as B_{ARG0} followed by I_{ARG1}

SRL Constraints

• **Unique core roles (U):** Each core role (ARG0-ARG5, ARGA) should appear at most once for each predicate.

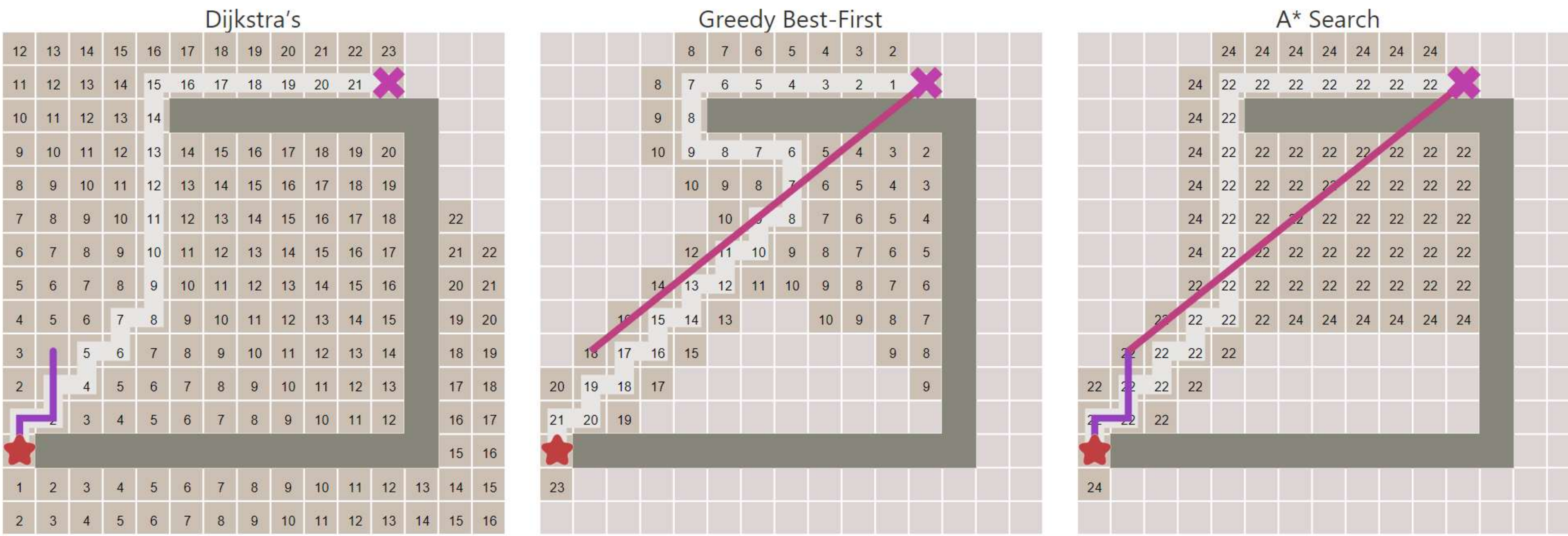
• **Continuation roles (C):** A continuation role C-X can exist only when its base role X is realized before it.

• **Reference roles (R):** A reference role R-X can exist only when its base role X is realized (not necessarily before R-X).

Syntactic Constraints

We can enforce consistency with a given parse tree by rejecting or penalizing arguments that are not constituents.

Intuition on A* searching



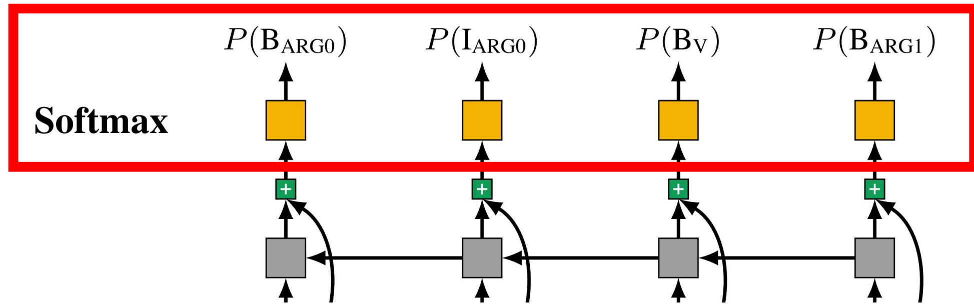
Record the distance from the starting point

Record the distance away from the destination

Consider both distances

Output: Constrained A* decoding

Another difference with ACL'2015 (CRF)

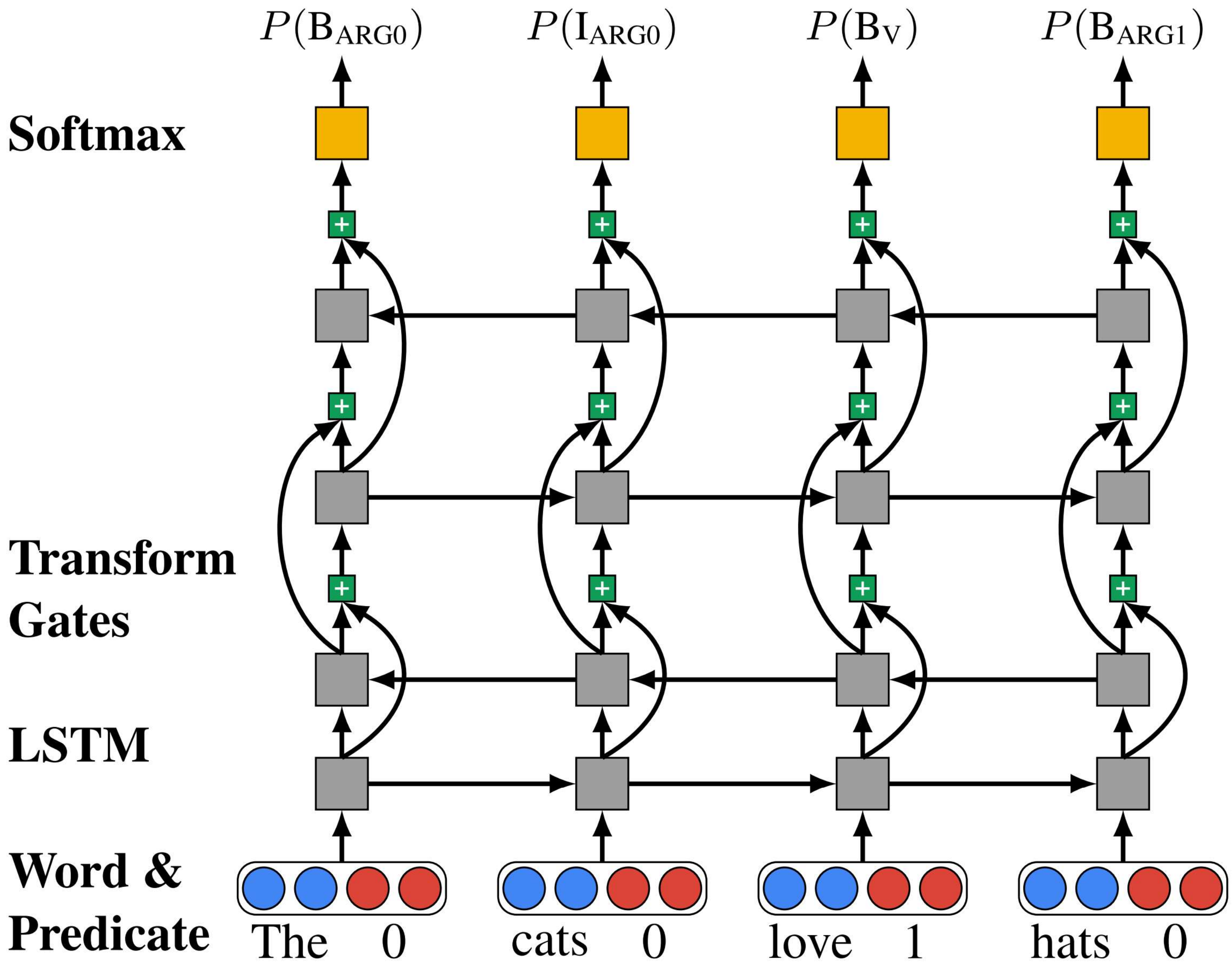


1. Not used during the training
2. A* searching algorithm reduced to Viterbi and able find the optimal solution?

$$p(y_t \mid \boldsymbol{x}) \propto \exp(\mathbf{W}_{\text{tag}}^{y} \boldsymbol{h}_{L,t} + \boldsymbol{b}_{\text{tag}})$$ ← Softmax output

$$f(\boldsymbol{w}, y_{1:t}) = \sum_{i=1}^{t} \log p(y_i \mid \boldsymbol{w}) - \sum_{c \in \mathcal{C}} c(\boldsymbol{w}, y_{1:i})$$ ← Confidence value with penalization

$$g(\boldsymbol{w}, y_{1:t}) = \sum_{i=t+1}^{n} \max_{y_i \in \boldsymbol{T}} \log p(y_i \mid \boldsymbol{w})$$ ← A* heuristic



$P(\mathrm{B}_{\mathrm{ARG0}})$
$P(\mathrm{I}_{\mathrm{ARG0}})$
$P(\mathrm{B}_{\mathrm{V}})$
$P(\mathrm{B}_{\mathrm{ARG1}})$
Softmax
Transform
Gates
LSTM
Word &
Predicate
The 0
cats 0
love 1
hats 0

	CoNLL-2005 (PropBank)	CoNLL-2012 (OntoNotes)
Size	40k sentences	140k sentences
Domains	• WSJ / newswire • Brown (test-only)	• telephone conversations • newswire • newsgroups • broadcast news • broadcast conversation • weblogs
Annotated predicates	Verbs	Added some nominal predicates "George III is the king of England"

CoNLL-2005

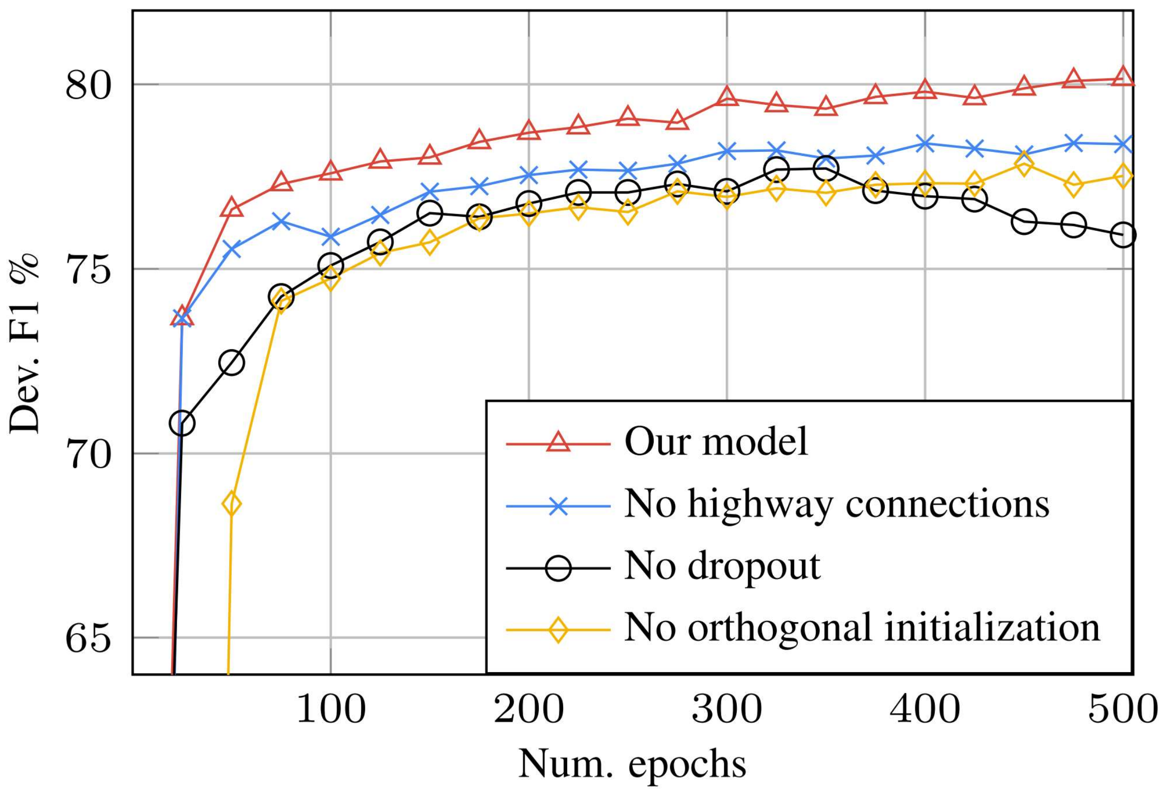
WORDS---->	NE--->	POS	PARTIAL_SYNT		FULL_SYNT------>	VS	TARGETS	PROPS------->	
The	*	DT	(NP*	(S*	(S(NP*	-	-	(A0*	(A0*
$	*	$	*	*	(ADJP(QP*	-	-	*	*
1.4	*	CD	*	*	*	-	-	*	*
billion	*	CD	*	*	*))	-	-	*	*
robot	*	NN	*	*	*	-	-	*	*
spacecraft	*	NN	*)	*	*)	-	-	*)	*)
faces	*	VBZ	(VP*)	*	(VP*	01	face	(V*)	*
a	*	DT	(NP*	*	(NP*	-	-	(A1*	*
six-year	*	JJ	*	*	*	-	-	*	*
journey	*	NN	*)	*	*	-	-	*	*
to	*	TO	(VP*	(S*	(S(VP*	-	-	*	*
explore	*	VB	*)	*	(VP*	01	explore	*	(V*)
Jupiter	(ORG*)	NNP	(NP*)	*	(NP(NP*)	-	-	*	(A1*
and	*	CC	*	*	*	-	-	*	*
its	*	PRP$	(NP*	*	(NP*	-	-	*	*
16	*	CD	*	*	*	-	-	*	*
known	*	JJ	*	*	*	-	-	*	*
moons	*	NNS	*)	*)	*)))))))	-	-	*)	*)
.	*	.	*	*)	*)	-	-	*	*

Model Performance

	Development				WSJ Test				Brown Test				Combined
Method	P	R	F1	Comp.	P	R	F1	Comp.	P	R	F1	Comp.	F1
Ours (PoE)	**83.1**	**82.4**	**82.7**	**64.1**	**85.0**	**84.3**	**84.6**	**66.5**	**74.9**	**72.4**	**73.6**	**46.5**	**83.2**
Ours	81.6	81.6	81.6	62.3	83.1	83.0	83.1	64.3	72.9	71.4	72.1	44.8	81.6
Zhou	79.7	79.4	79.6	-	82.9	82.8	82.8	-	70.7	68.2	69.4	-	81.1
FitzGerald (Struct.,PoE)	81.2	76.7	78.9	55.1	82.5	78.2	80.3	57.3	74.5	70.0	72.2	41.3	-
Täckström (Struct.)	81.2	76.2	78.6	54.4	82.3	77.6	79.9	56.0	74.3	68.6	71.3	39.8	-
Toutanova (Ensemble)	-	-	78.6	58.7	81.9	78.8	80.3	60.1	-	-	68.8	40.8	-
Punyakanok (Ensemble)	80.1	74.8	77.4	50.7	82.3	76.8	79.4	53.8	73.4	62.9	67.8	32.3	77.9

Table 1: Experimental results on CoNLL 2005, in terms of precision (P), recall (R), F1 and percentage of completely correct predicates (Comp.). We report results of our best single and ensemble (PoE) model. The comparison models are Zhou and Xu (2015), FitzGerald et al. (2015), Täckström et al. (2015), Toutanova et al. (2008) and Punyakanok et al. (2008).

Contributions of Three Training Techniques



1. Without any of the three, seems unable to beat state-of-the-art

2. Orthogonal initialization is very important at the early stage of training

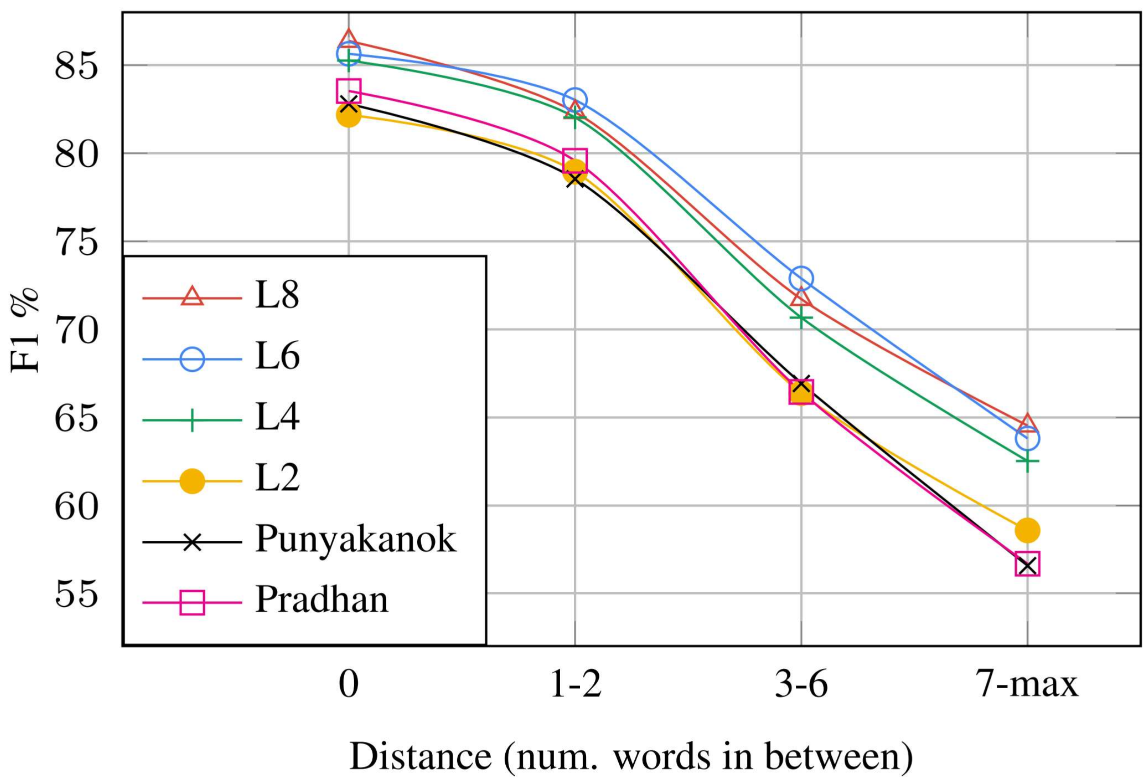
End-to-End SRL

Train a separate predicate detection model

	Predicate Detection			End-to-end SRL (Single)			End-to-end SRL (PoE)			
Dataset	P	R	F1	P	R	F1	P	R	F1	Δ F1
CoNLL 2005 Dev.	97.4	97.4	97.4	80.3	80.4	80.3	81.8	81.2	81.5	-1.2
WSJ Test	94.5	98.5	96.4	80.2	82.3	81.2	82.0	83.4	82.7	-1.9
Brown Test	89.3	95.7	92.4	67.6	69.6	68.5	69.7	70.5	70.1	-3.5
CoNLL 2012 Dev.	88.7	90.6	89.7	74.9	76.2	75.5	76.5	77.8	77.2	-6.2
CoNLL 2012 Test	93.7	87.9	90.7	78.6	75.1	76.8	80.2	76.6	78.4	-5.0

Table 3: Predicate detection performance and end-to-end SRL results using predicted predicates. Δ F1 shows the absolute performance drop compared to our best ensemble model with gold predicates.

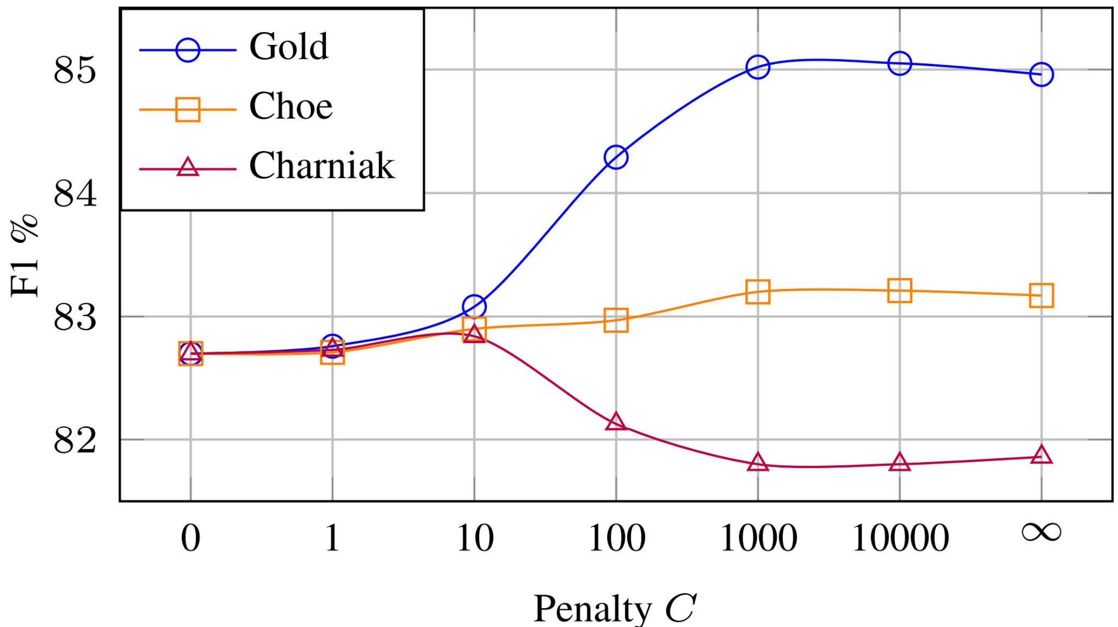
Long-range Dependency



Performance deteriorates as distance increases.

Adding Syntactic Constraints

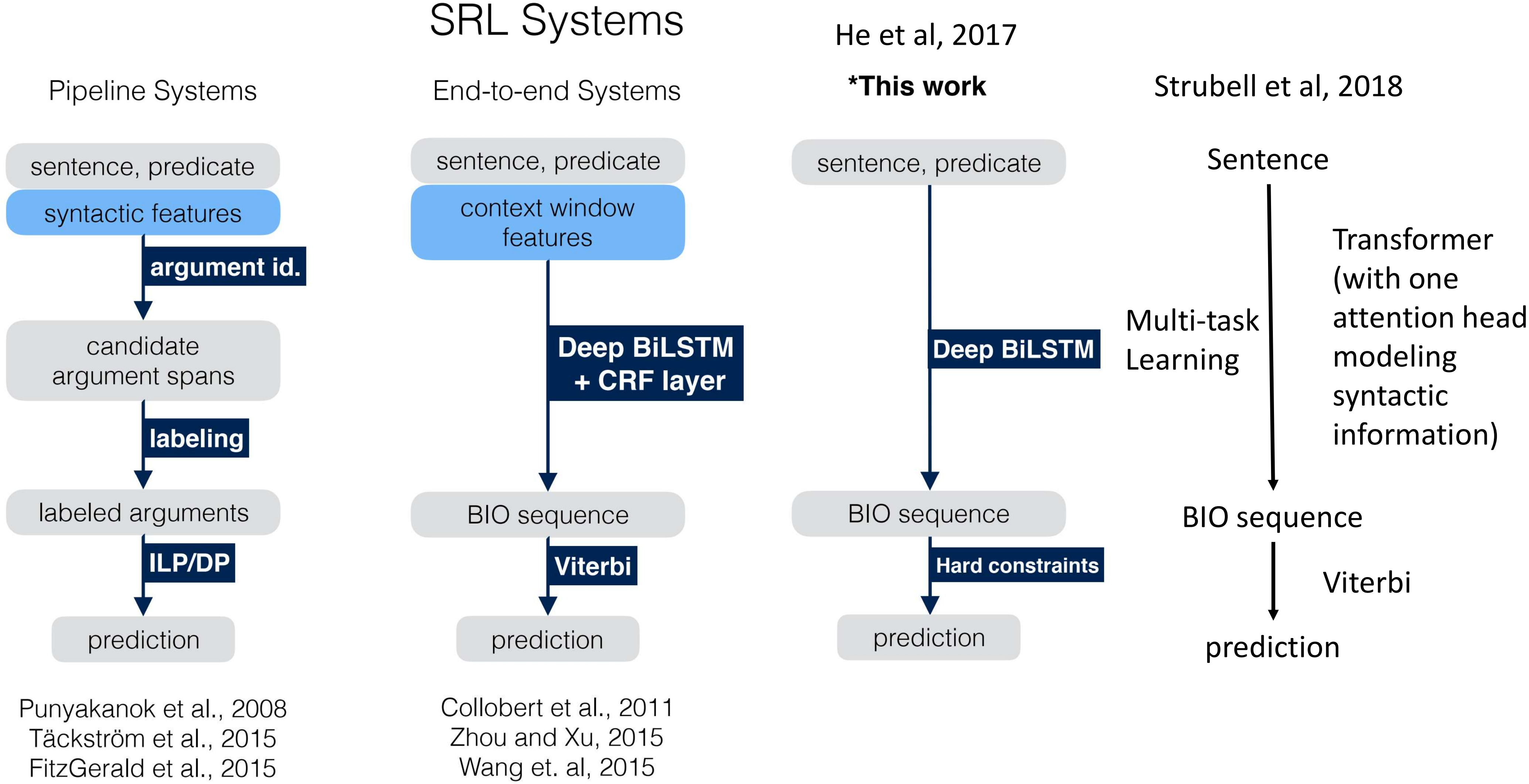
$$f(\boldsymbol{w}, y_{1:t}) = \sum_{i=1}^{t} \log p(y_i \mid \boldsymbol{w}) - \sum_{c \in \mathcal{C}} c(\boldsymbol{w}, y_{1:i})$$



1. **Syntax helps!**
2. **We need a better automatic parser (syntax modeling), though**

Gold: Penn Treebank constituents.
Choe: Parsing as language modeling, Choe and Charniak, 2016 (SOTA)
Charniak: A maximum-entropy-inspired parser, Charniak, 2000

SRL Systems



Pipeline Systems
sentence, predicate
syntactic features
argument id.
candidate argument spans
labeling
labeled arguments
ILP/DP
prediction
Punyakanok et al., 2008
Täckström et al., 2015
FitzGerald et al., 2015
End-to-end Systems
sentence, predicate
context window features
Deep BiLSTM + CRF layer
BIO sequence
Viterbi
prediction
Collobert et al., 2011
Zhou and Xu, 2015
Wang et. al, 2015
He et al, 2017
*This work
sentence, predicate
Deep BiLSTM
BIO sequence
Hard constraints
prediction
Strubell et al, 2018
Sentence
Multi-task Learning
Transformer (with one attention head modeling syntactic information)
BIO sequence
Viterbi
prediction

- **Feature based Semantic Role Labeling**

  Syntax plays an important role

- **Deep Semantic Role Labeling: What Works and What's Next ACL'2017**

  End-to-end model without syntactic input

- **Linguistically-Informed Self-Attention for Semantic Role Labeling ACL'2018**

  Explicitly model the syntactic information in neural network

LISA

Linguistically-Informed Self-Attention for Semantic Role Labeling





Emma Strubell[1]   Patrick Verga[1]   Daniel Andor[2]   David Weiss[2]   Andrew McCallum[1]

Linguistically-Informed Self-Attention

- **Multi-task learning**
  - Part-of-speech tagging
  - Labeled dependency parsing
  - Predicate detection
  - Semantic role spans & labeling
- **Syntactically-informed self-attention**
  - Multi-head self-attention supervised by syntax

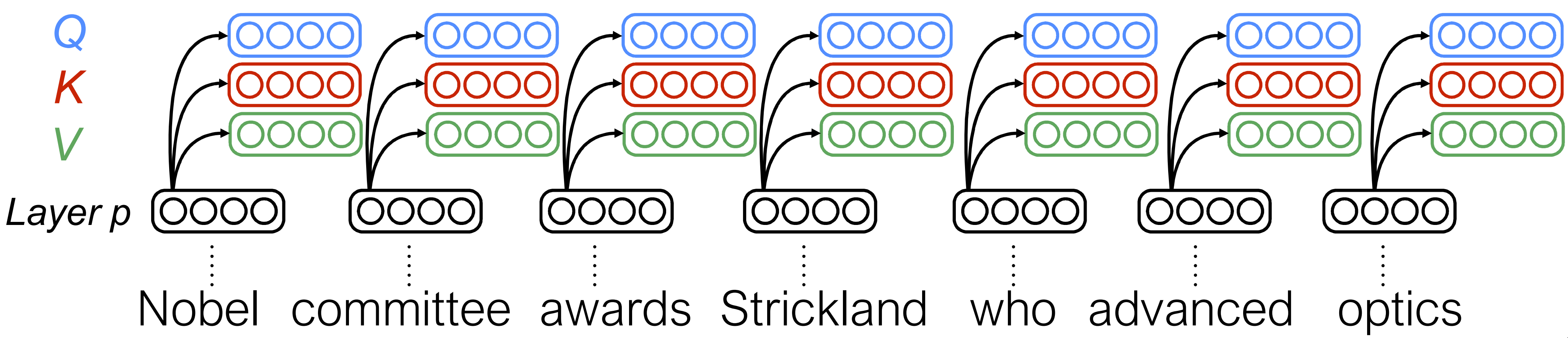
Outline

- LISA: Linguistically-informed self attention
    - Multi-head self-attention
    - Syntactically-informed self-attention

      [Vaswani et al. 2017]
    - Multi-task learning, single-pass inference
- Experimental results & error analysis

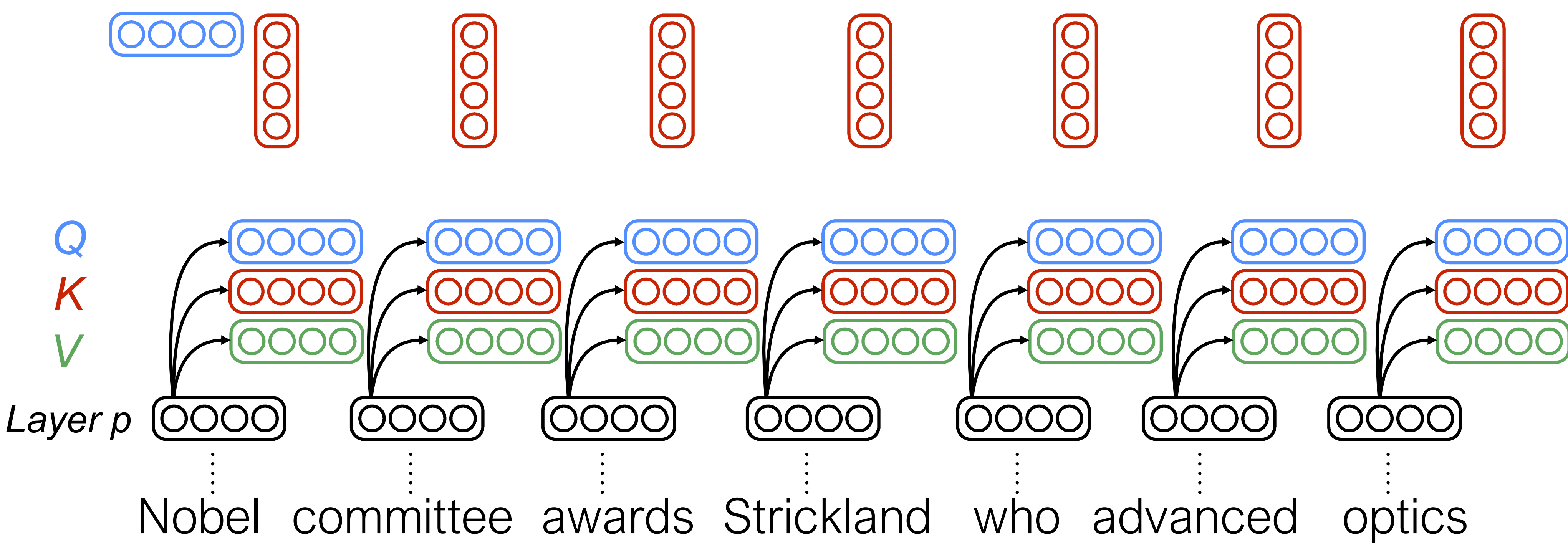
[Vaswani et al. 2017]

Self-attention



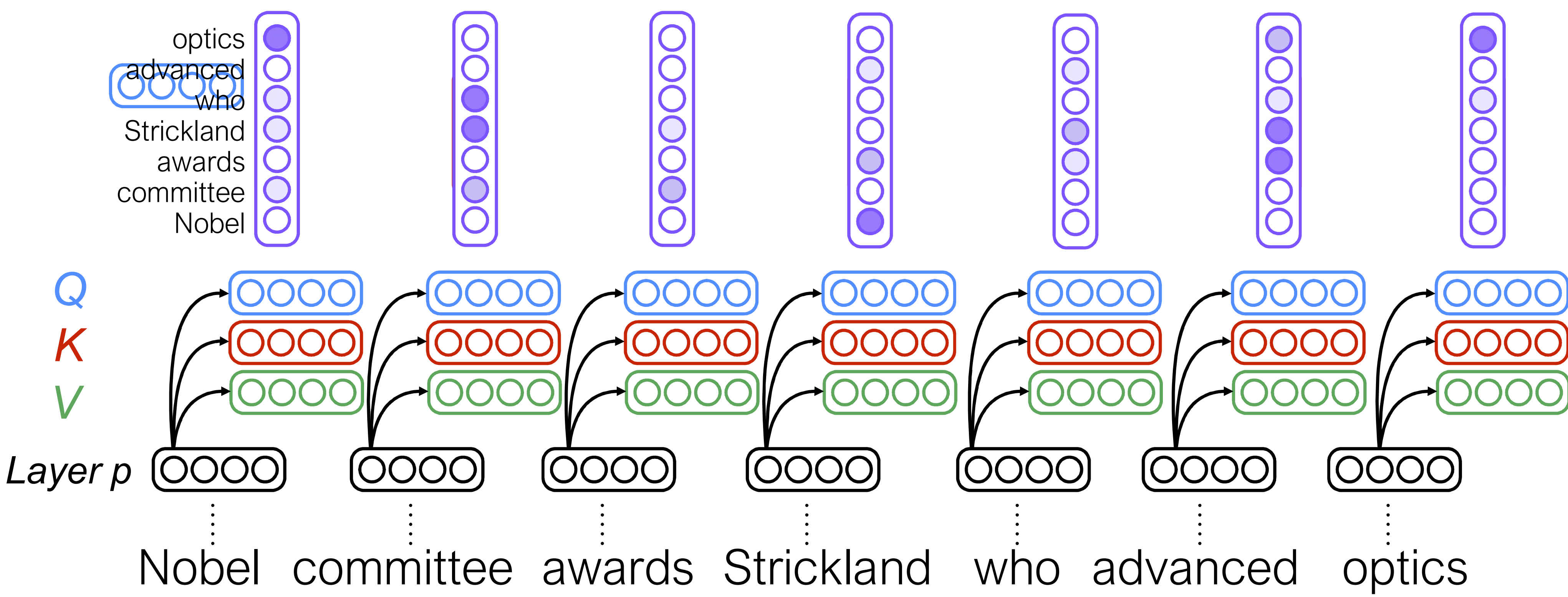
Self-attention

[Vaswani et al. 2017]



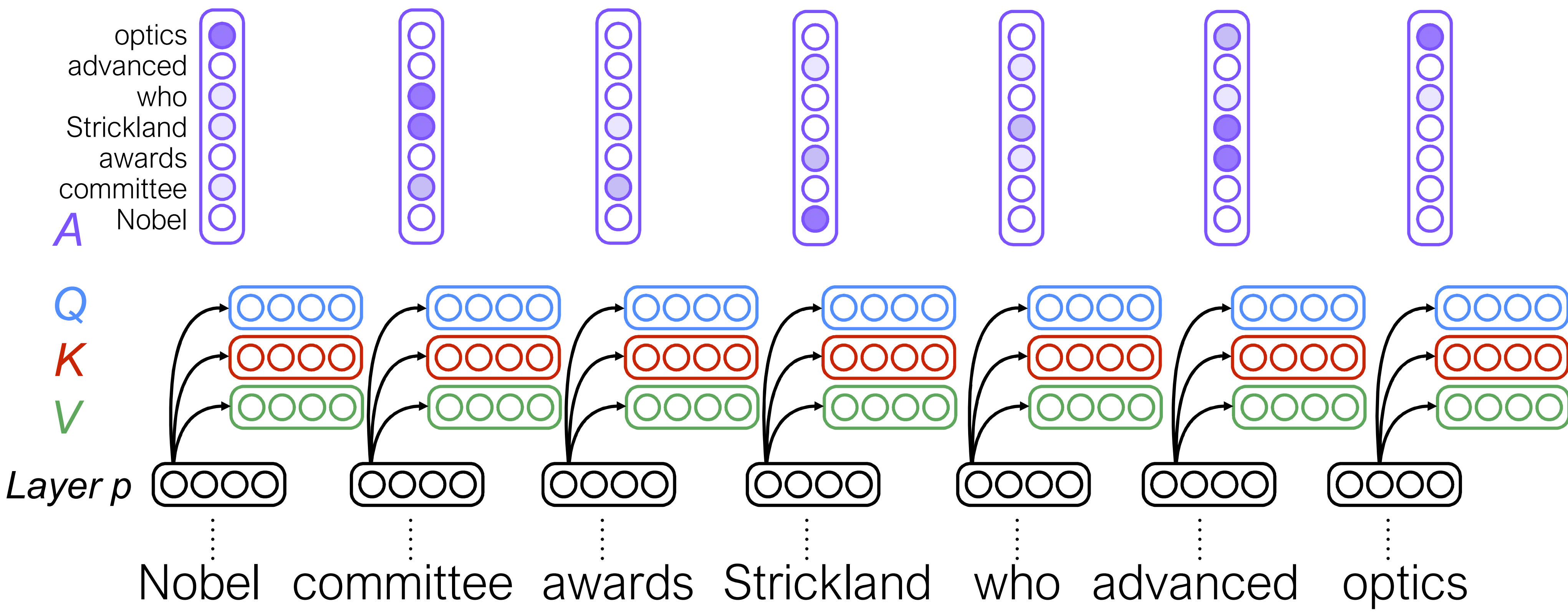
Self-attention

[Vaswani et al. 2017]



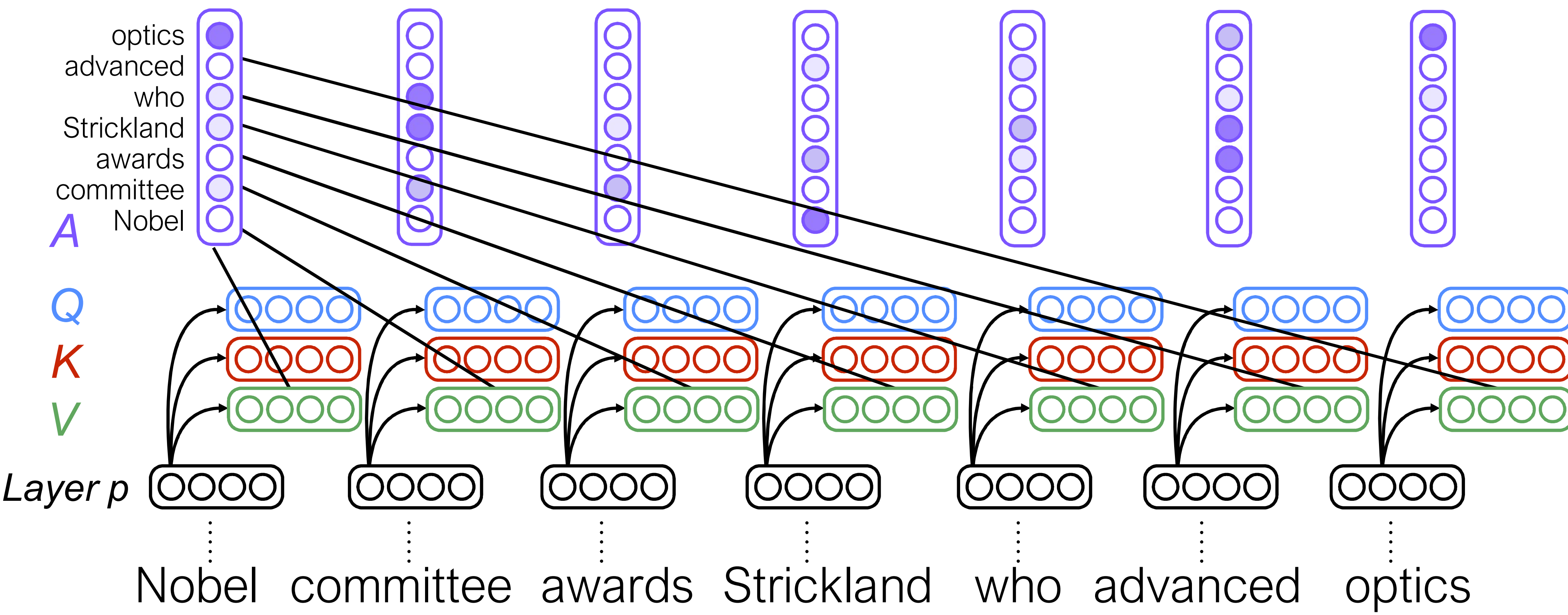
[Vaswani et al. 2017]

Self-attention

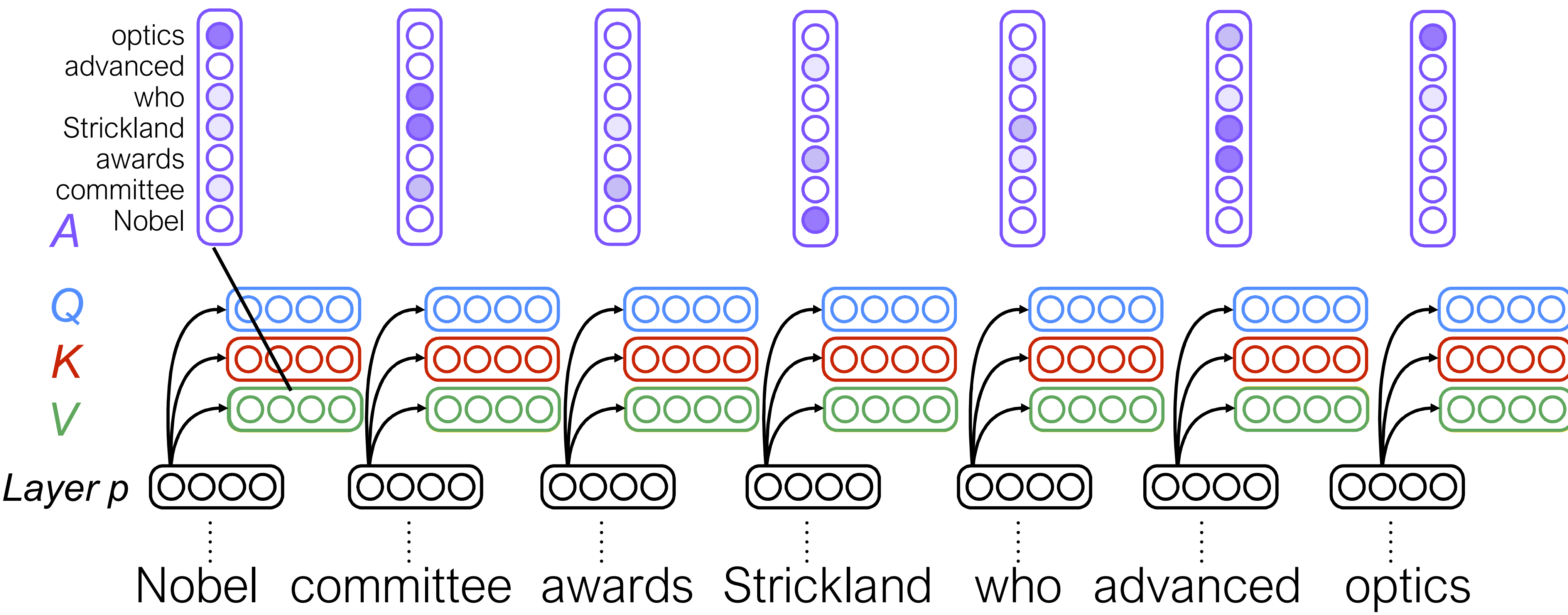


[Vaswani et al. 2017]

Self-attention

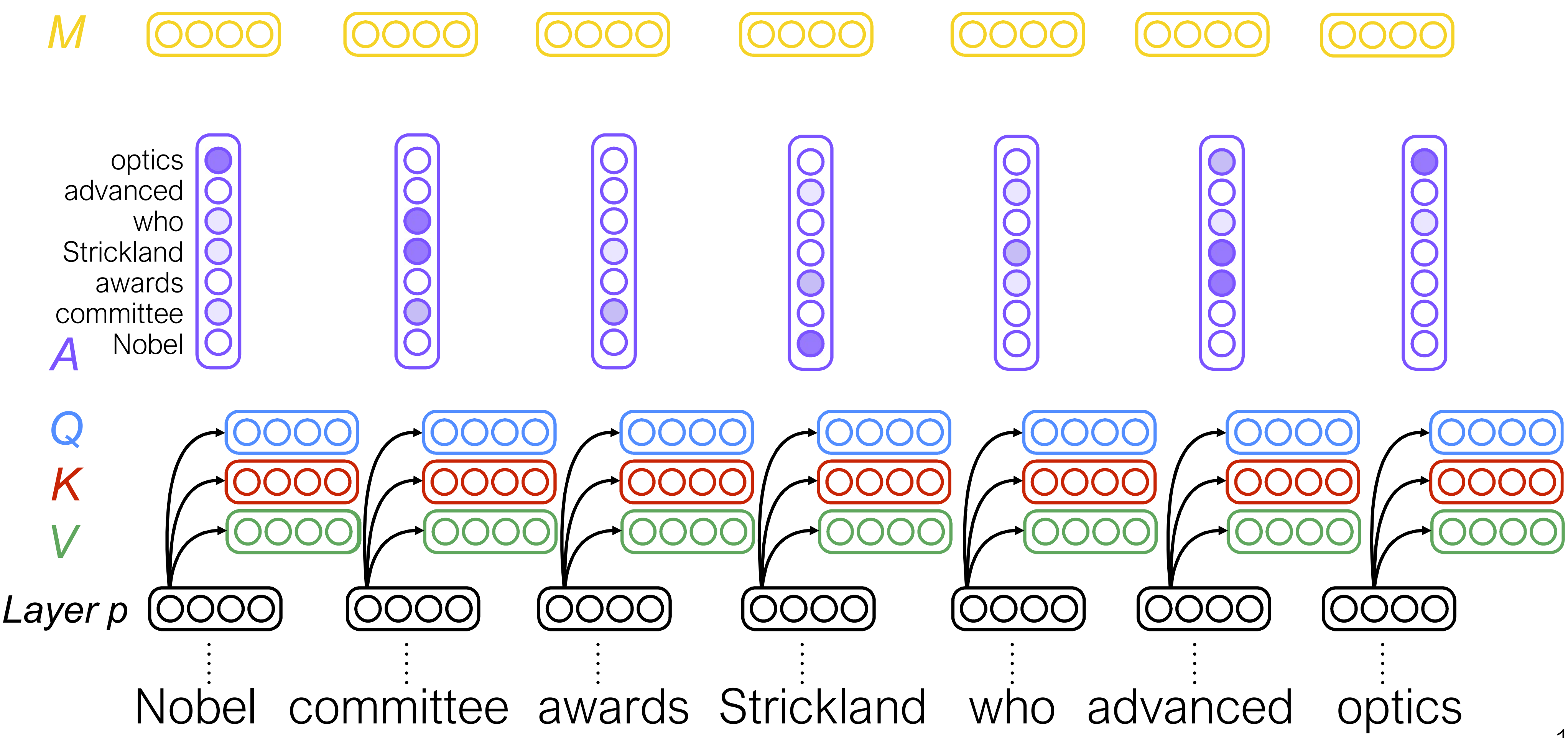


Self-attention



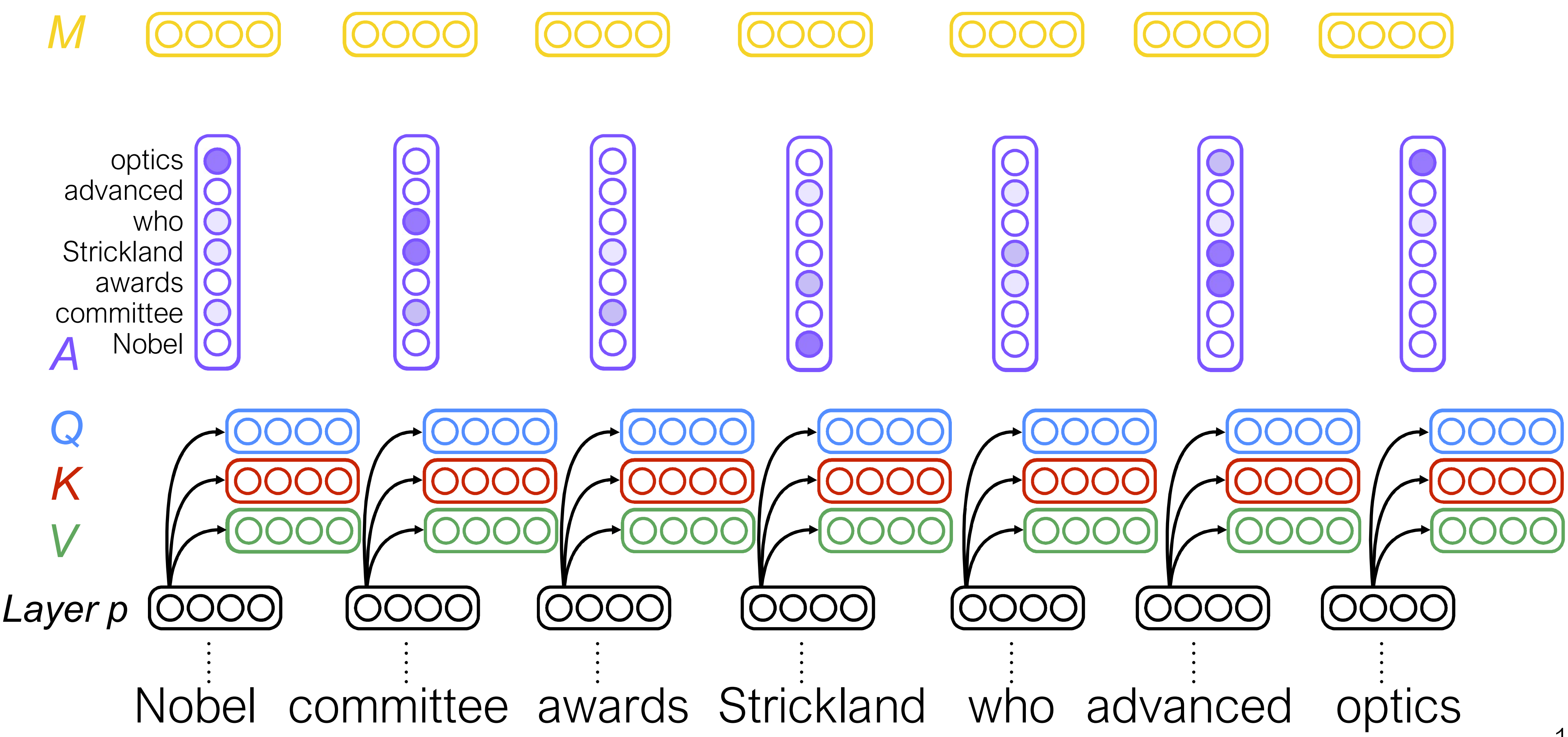
[Vaswani et al. 2017]

Self-attention

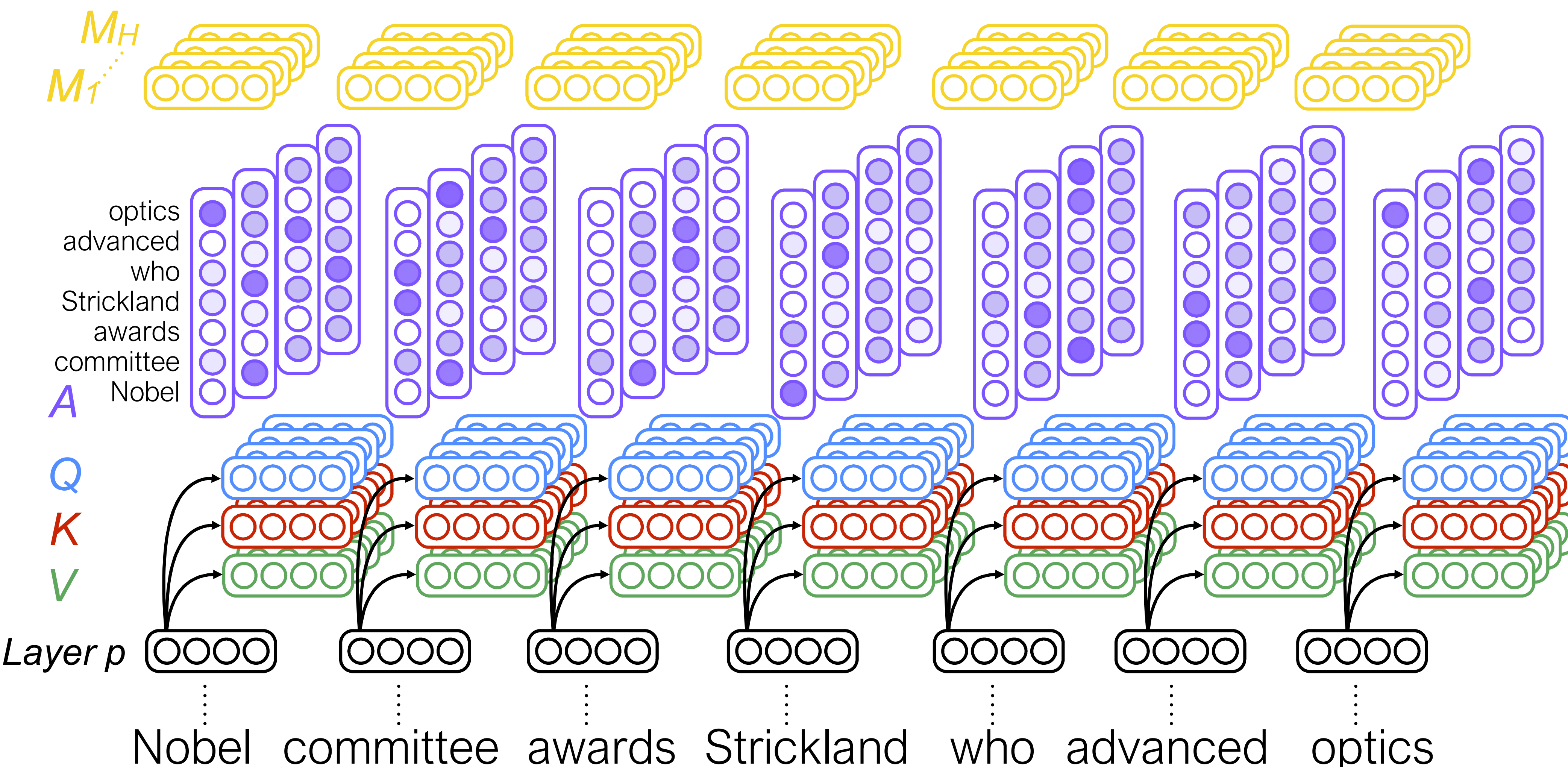


Self-attention

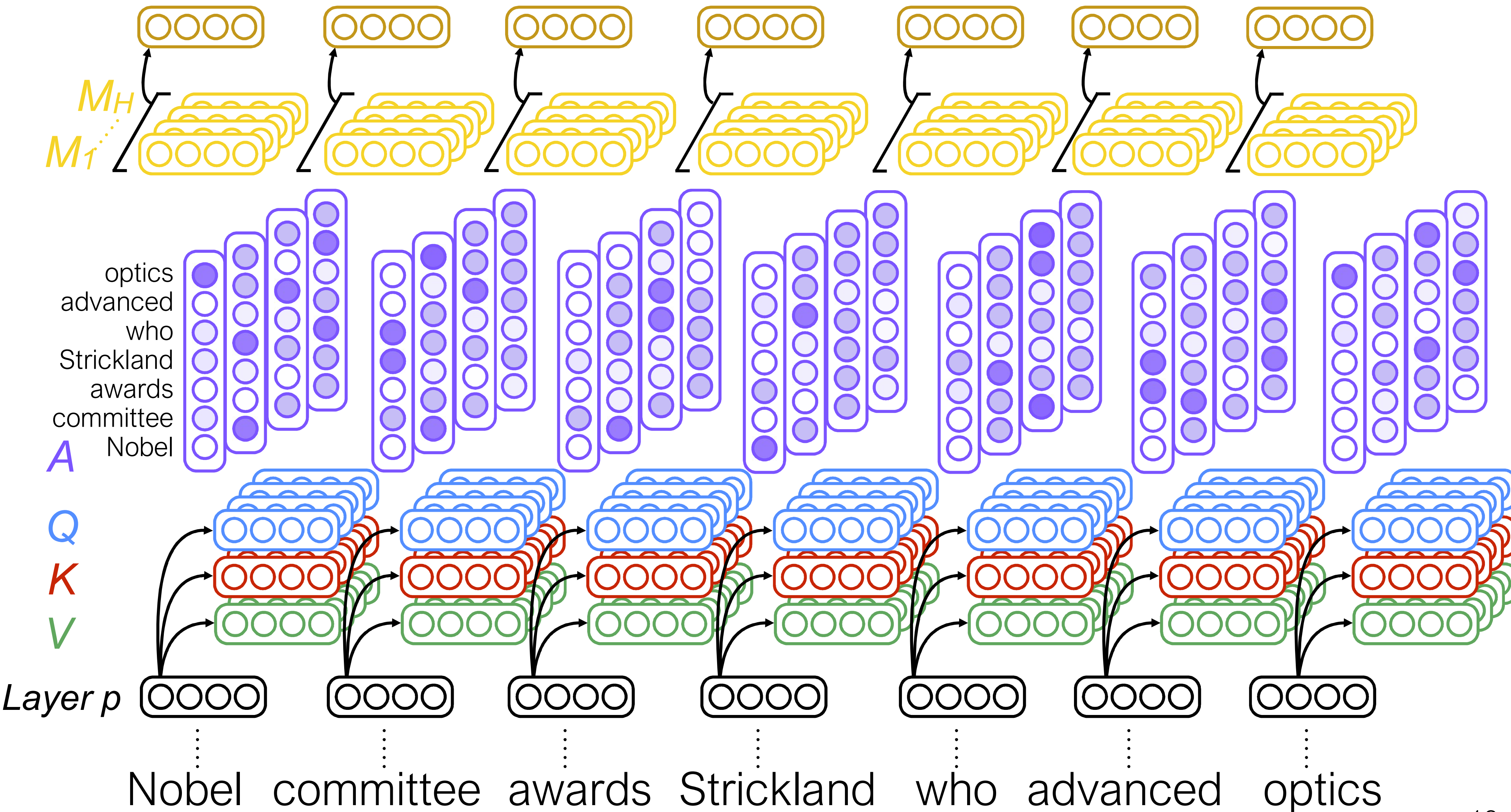
[Vaswani et al. 2017]



Multi-head self-attention

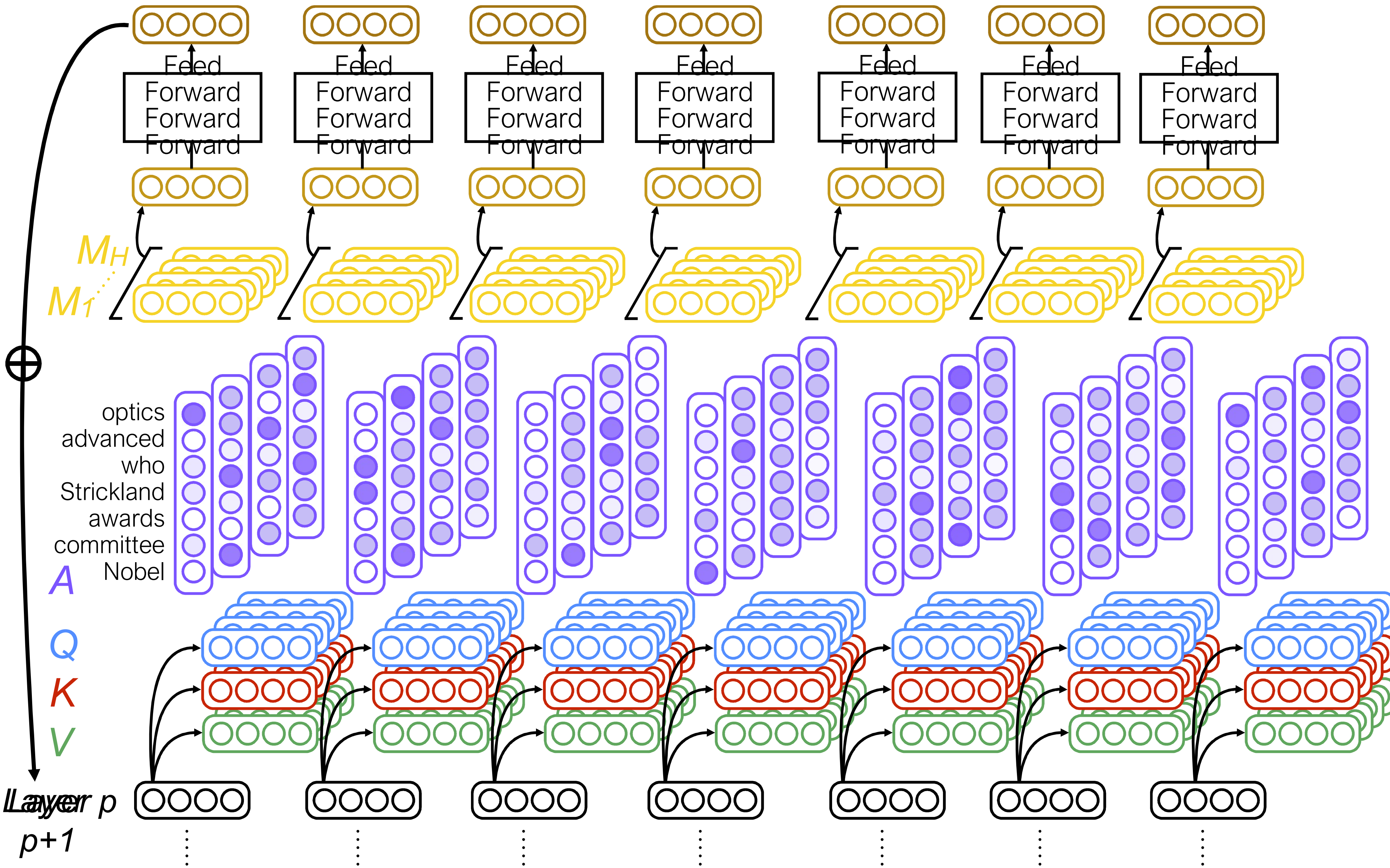


Multi-head self-attention



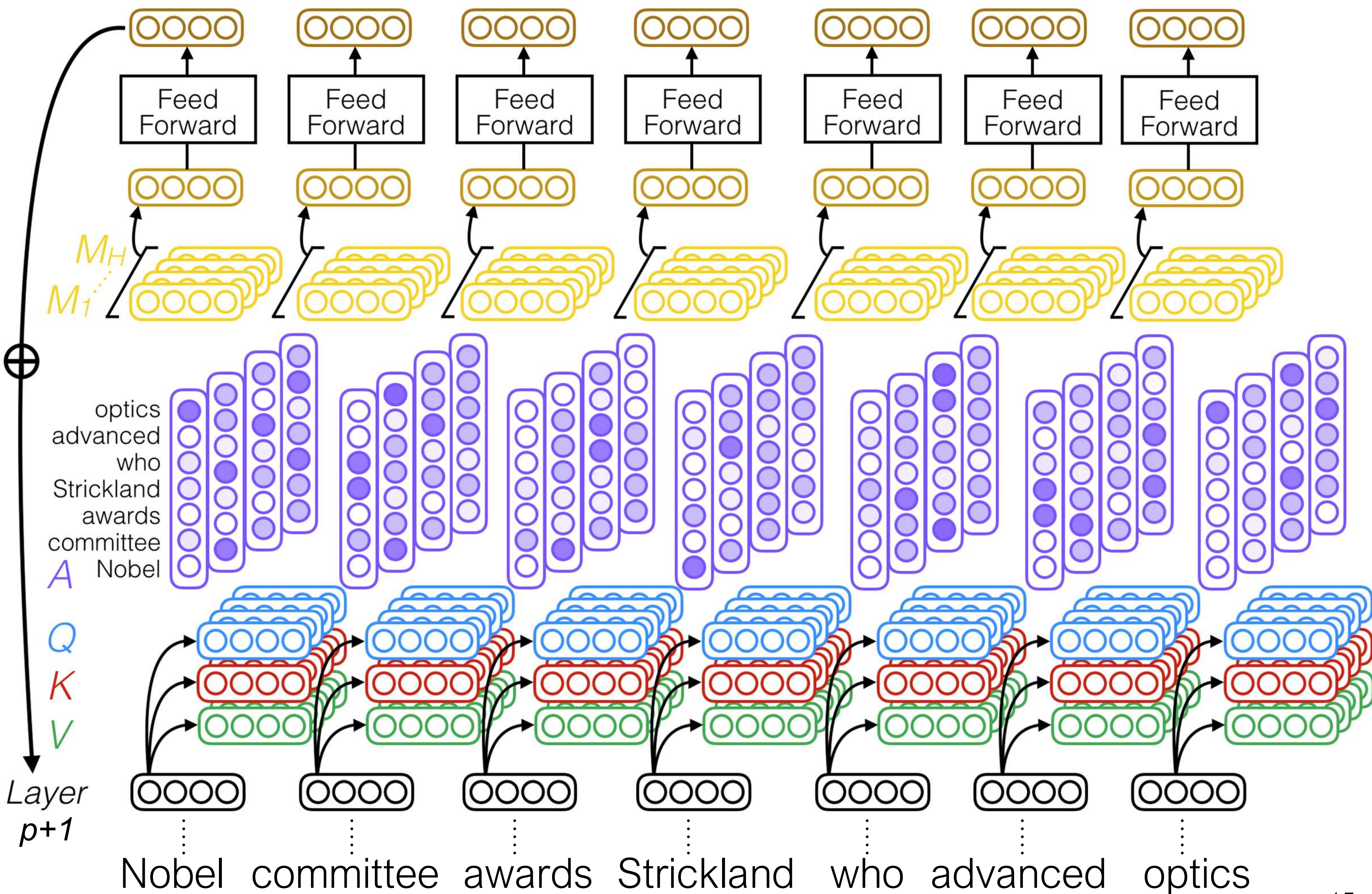
[Vaswani et al. 2017]

Multi-head self-attention



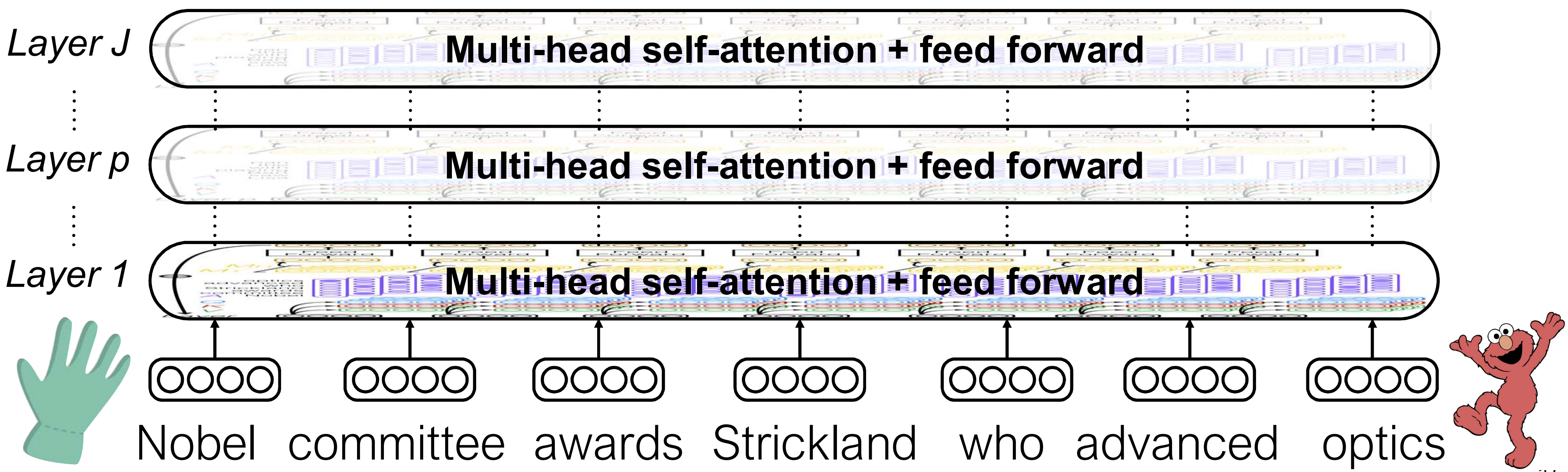
[Vaswani et al. 2017]

Multi-head self-attention



[Vaswani et al. 2017]

Multi-head self-attention

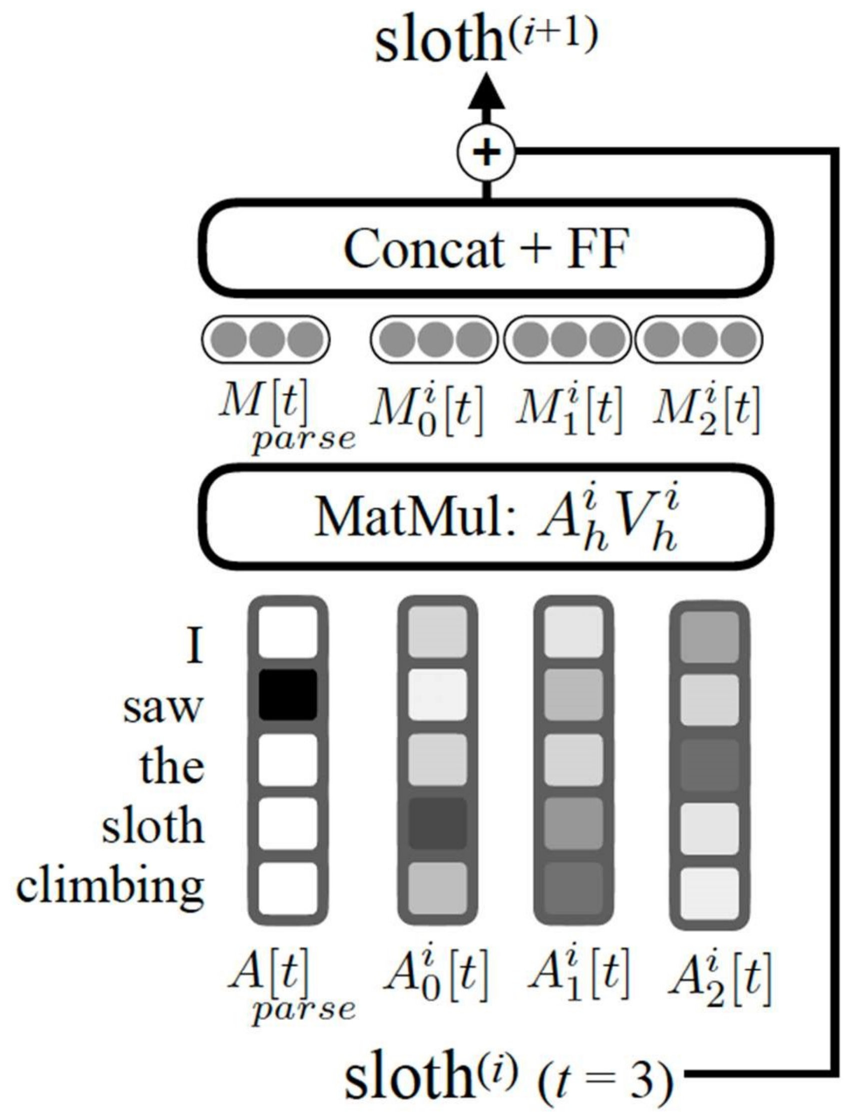


Self-attention

$$s_t^{(j)} = LN(s_t^{(j-1)} + T^{(j)}(s_t^{(j-1)}))$$

$$A_h^{(j)} = \mathrm{softmax}(d_k^{-0.5} Q_h^{(j)} {K_h^{(j)}}^T)$$

$$M_h^{(j)} = A_h^{(j)} V_h^{(j)}$$



Outline

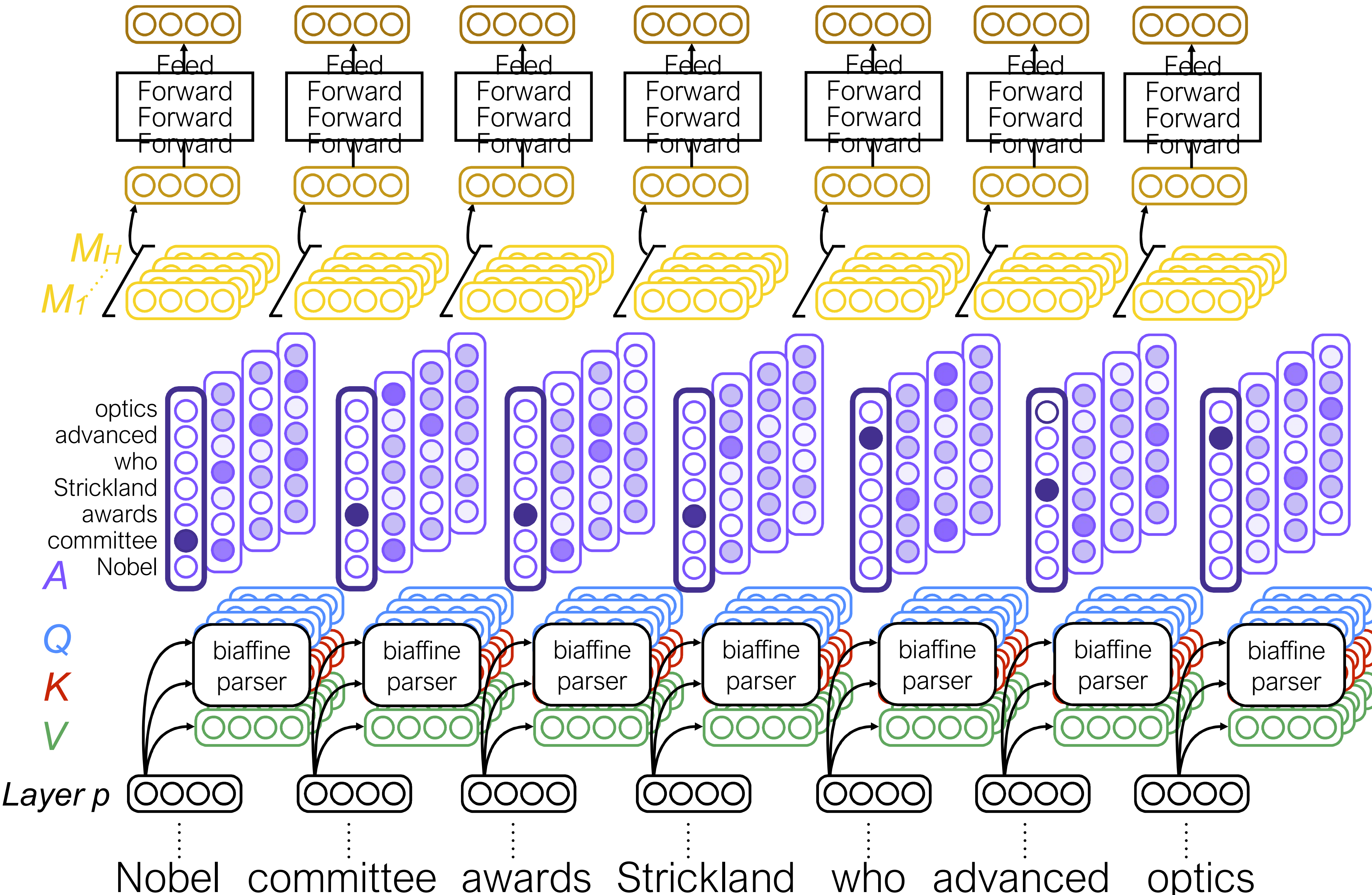
- LISA: Linguistically-informed self attention
  - Multi-head self-attention
  - Syntactically-informed self-attention

    [Vaswani et al. 2017]
  - Multi-task learning, single-pass inference
- Experimental results & error analysis

How to incorporate syntax?

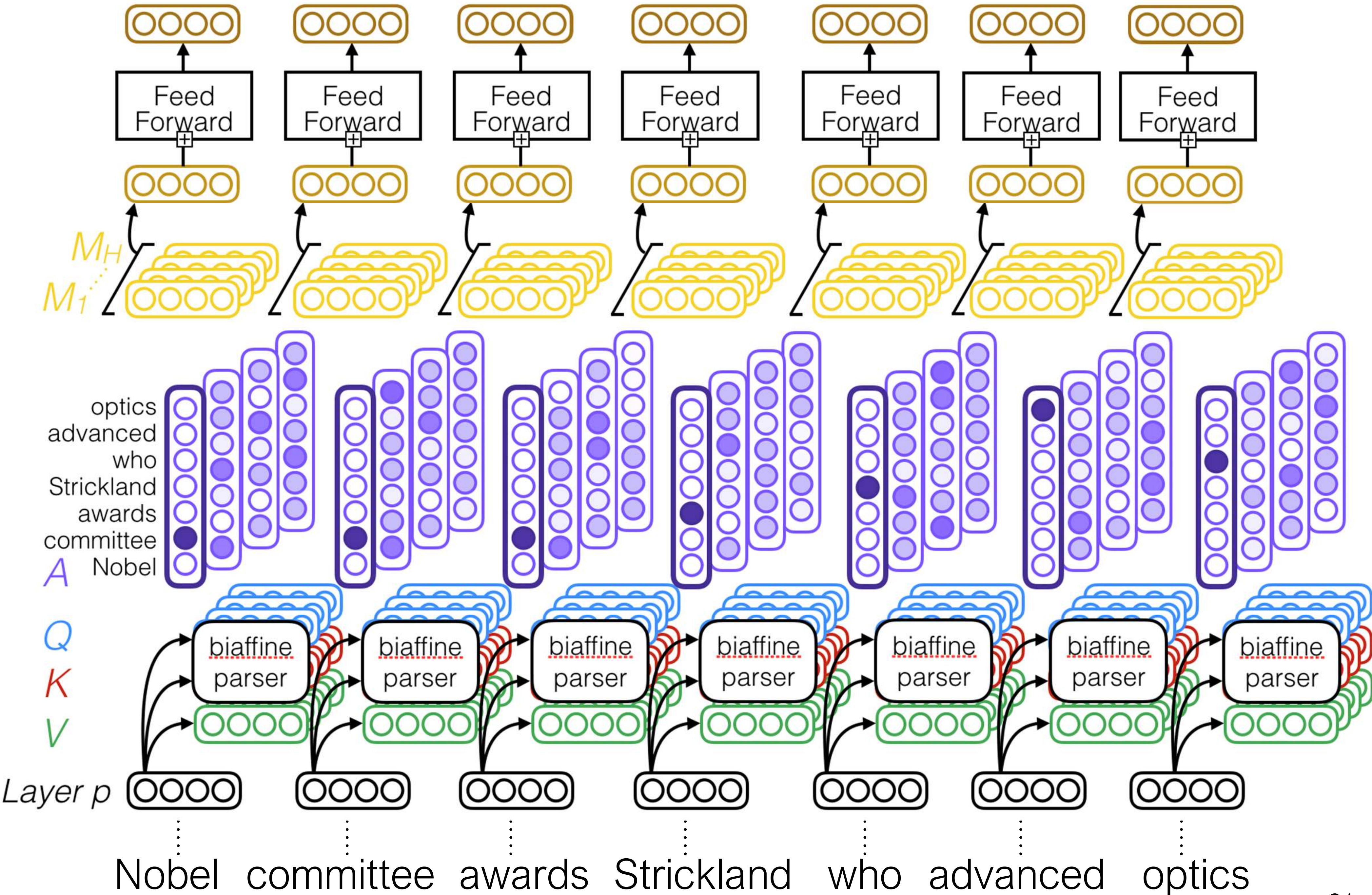
- Multi-task learning [Caruana 1993; Collobert et al. 2011]:
  - Overfits to training domain like single-task end-to-end NN.
  - Must re-train SRL model to leverage new (improved) syntax.
- Dependency path embeddings [Roth & Lapata 2016]; Graph CNN over parse [Marcheggiani & Titov 2017]
  - Restricted context: path to predicate or fixed-width window.
- Syntactically-informed self-attention
  - In one head, token attends to its likely syntactic parent(s).
  - Global context: In next layer, tokens observe all other parents.
  - At test time: can use own predicted parse, *OR*
    supply syntax to improve SRL model without re-training.

Syntactically-informed self-attention

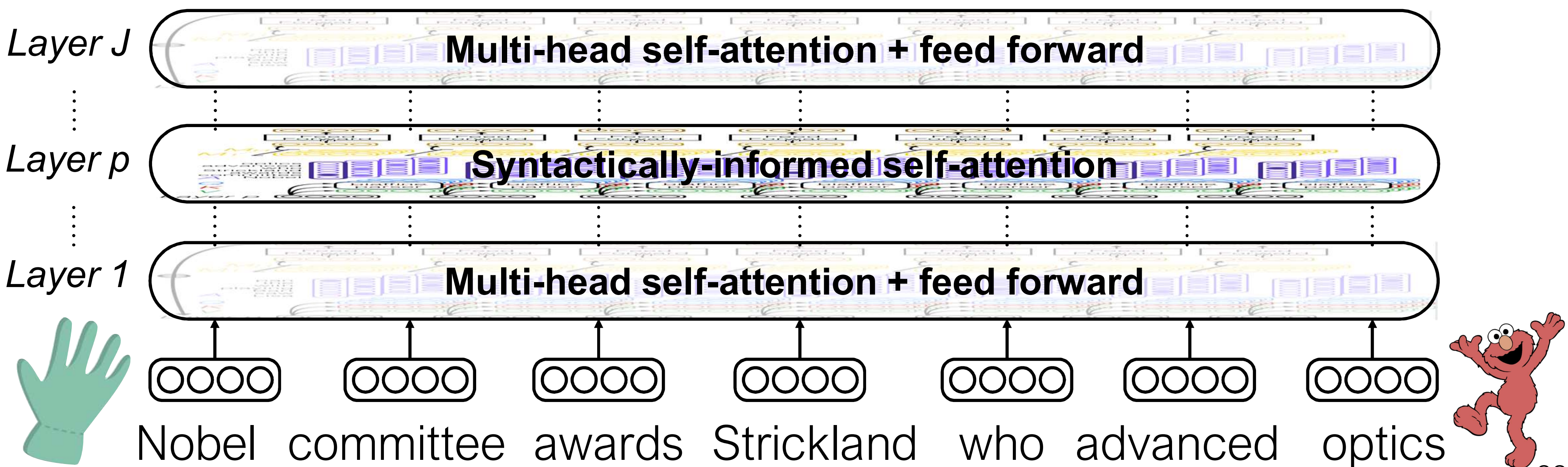


[Dozat and Manning 2017]

Syntactically-informed self-attention



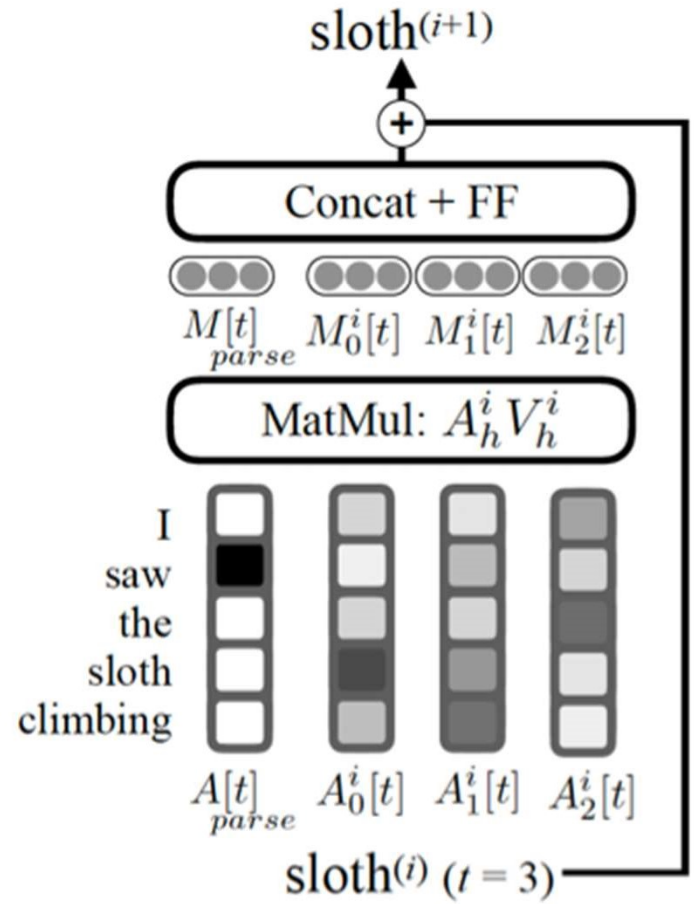
Syntactically-informed self-attention



Syntactically-informed self-attention

$$A_{parse} = \text{softmax}(Q_{parse} U_{heads} K_{parse}^T)$$

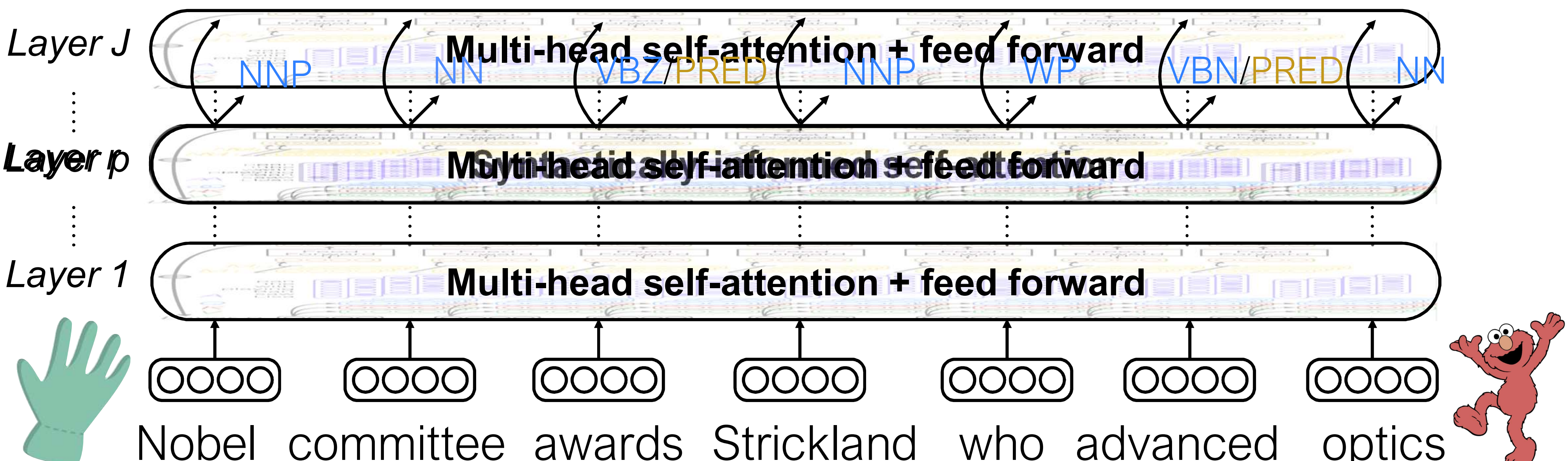
$$P(q = \text{head}(t) \mid \mathcal{X}) = A_{parse}[t, q]$$



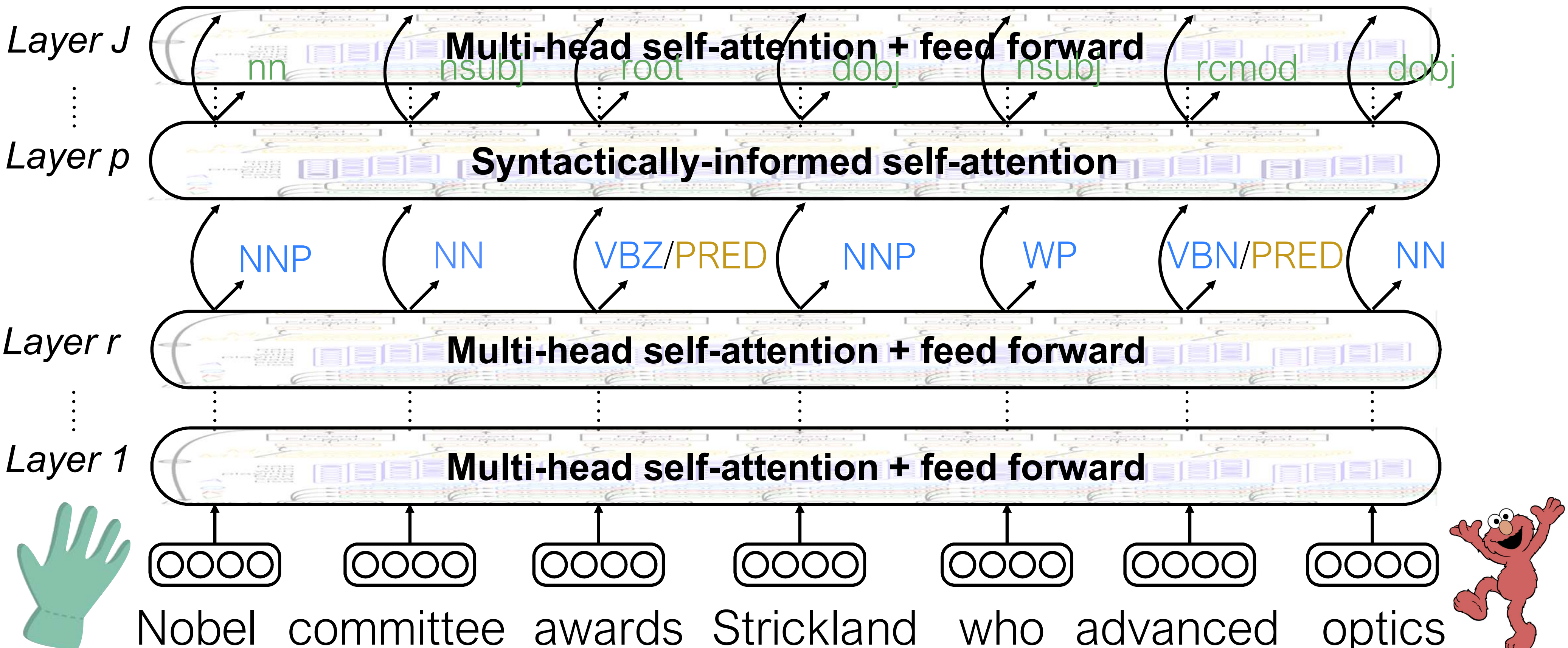
Outline

- LISA: Linguistically-informed self attention
  - Multi-head self-attention
  - Syntactically-informed self-attention
  - Multi-task learning, single-pass inference
- Experimental results & error analysis

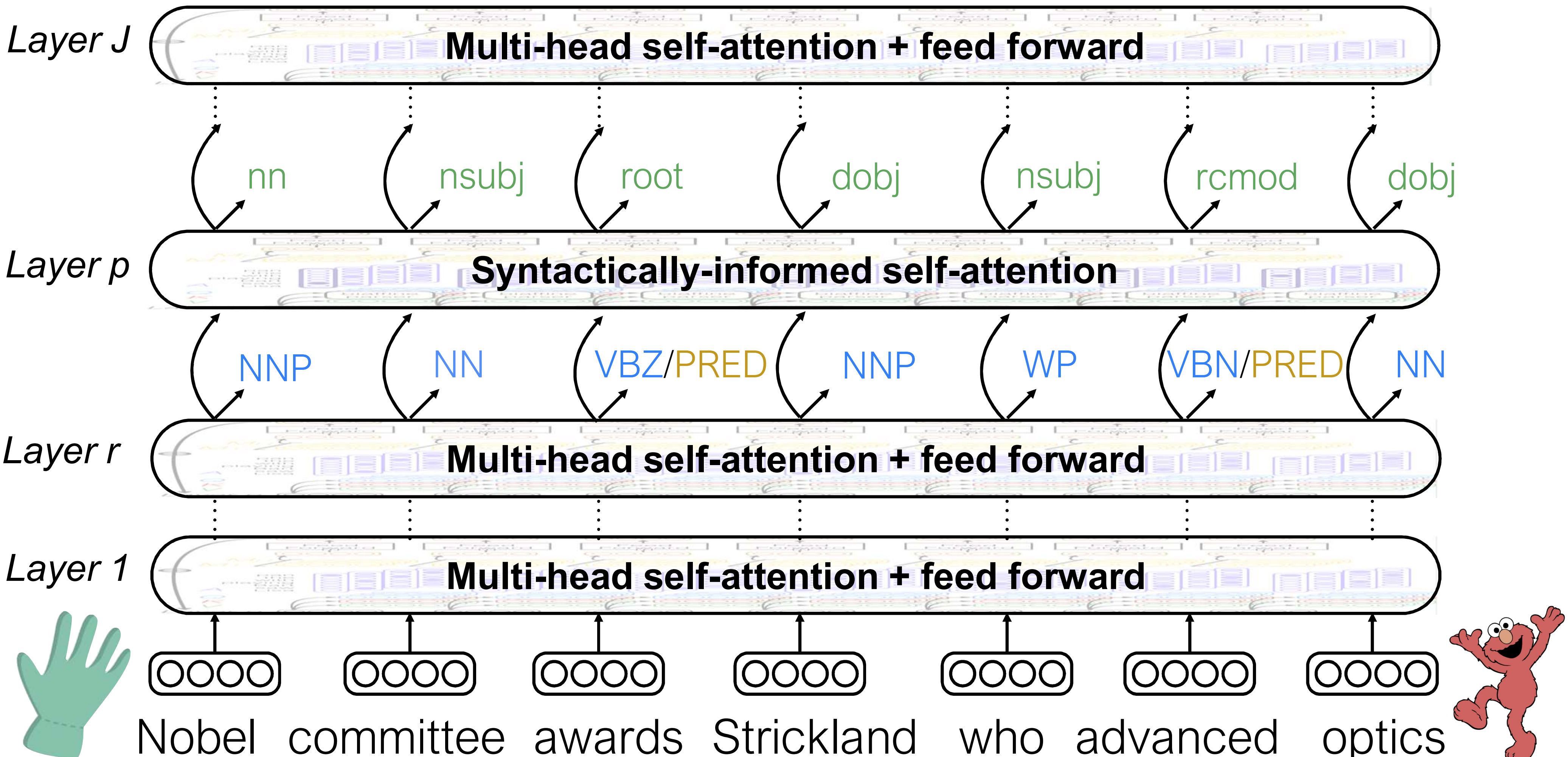
LISA: Linguistically-Informed Self-Attention



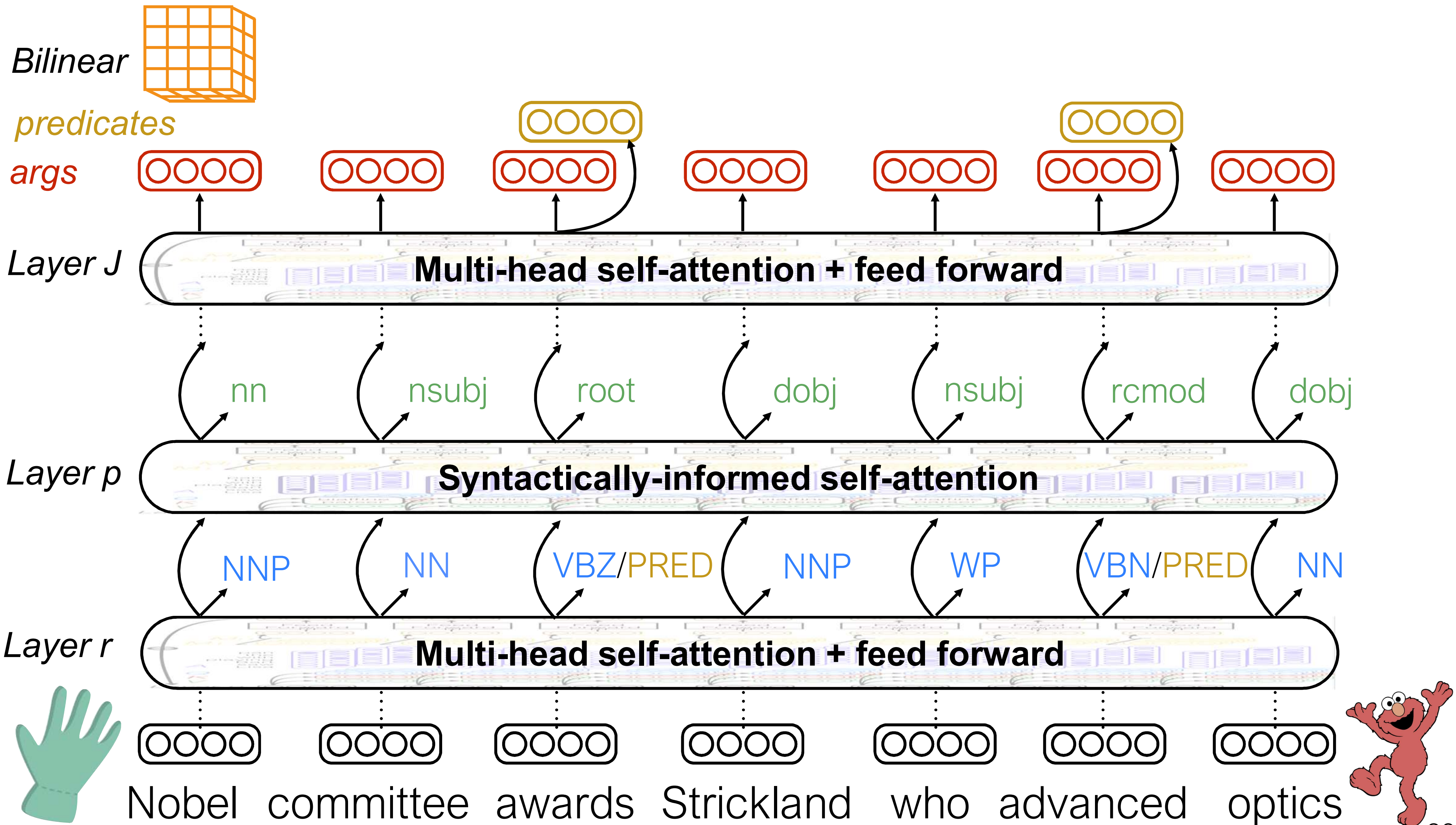
LISA: Linguistically-Informed Self-Attention



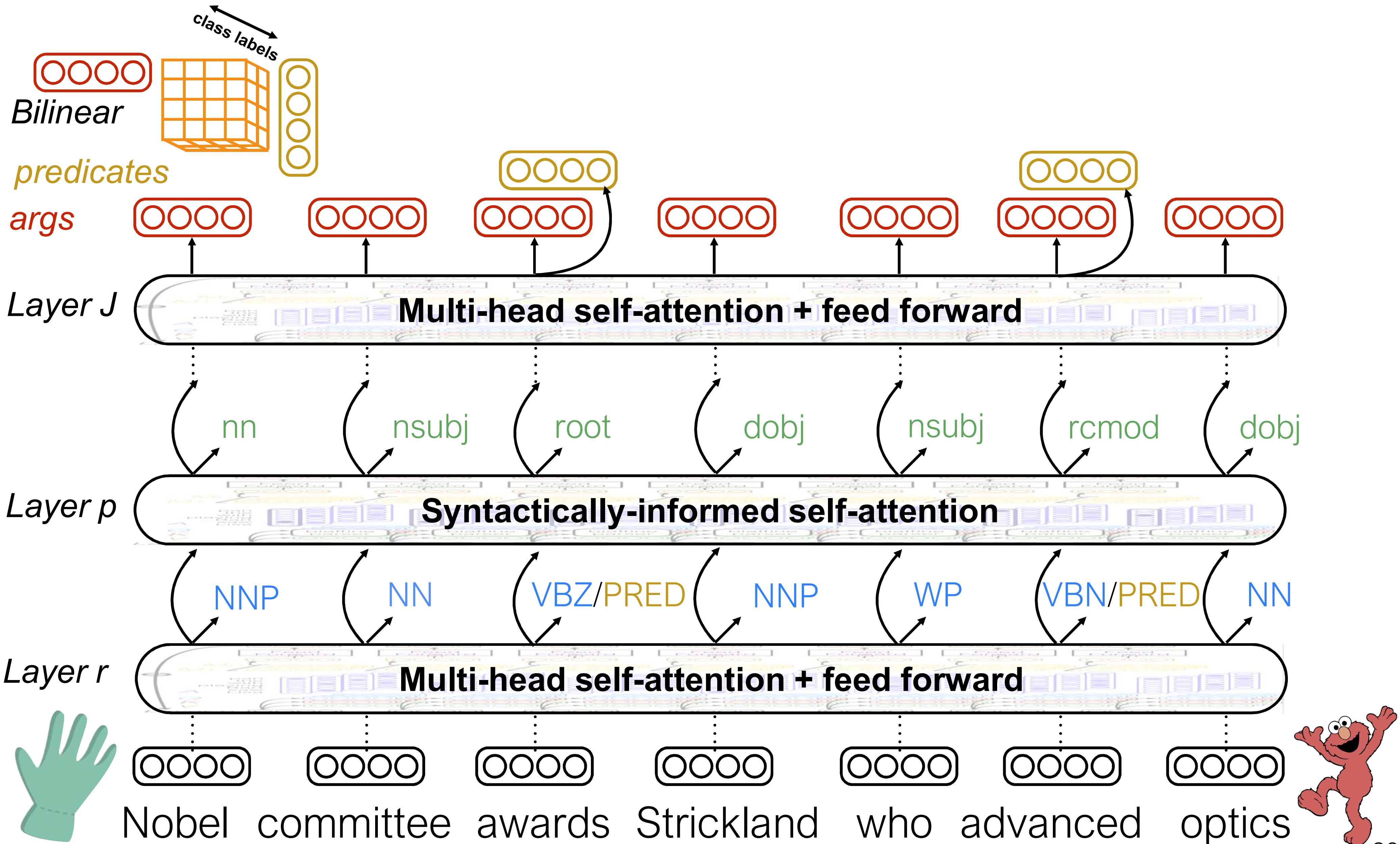
LISA: Linguistically-Informed Self-Attention



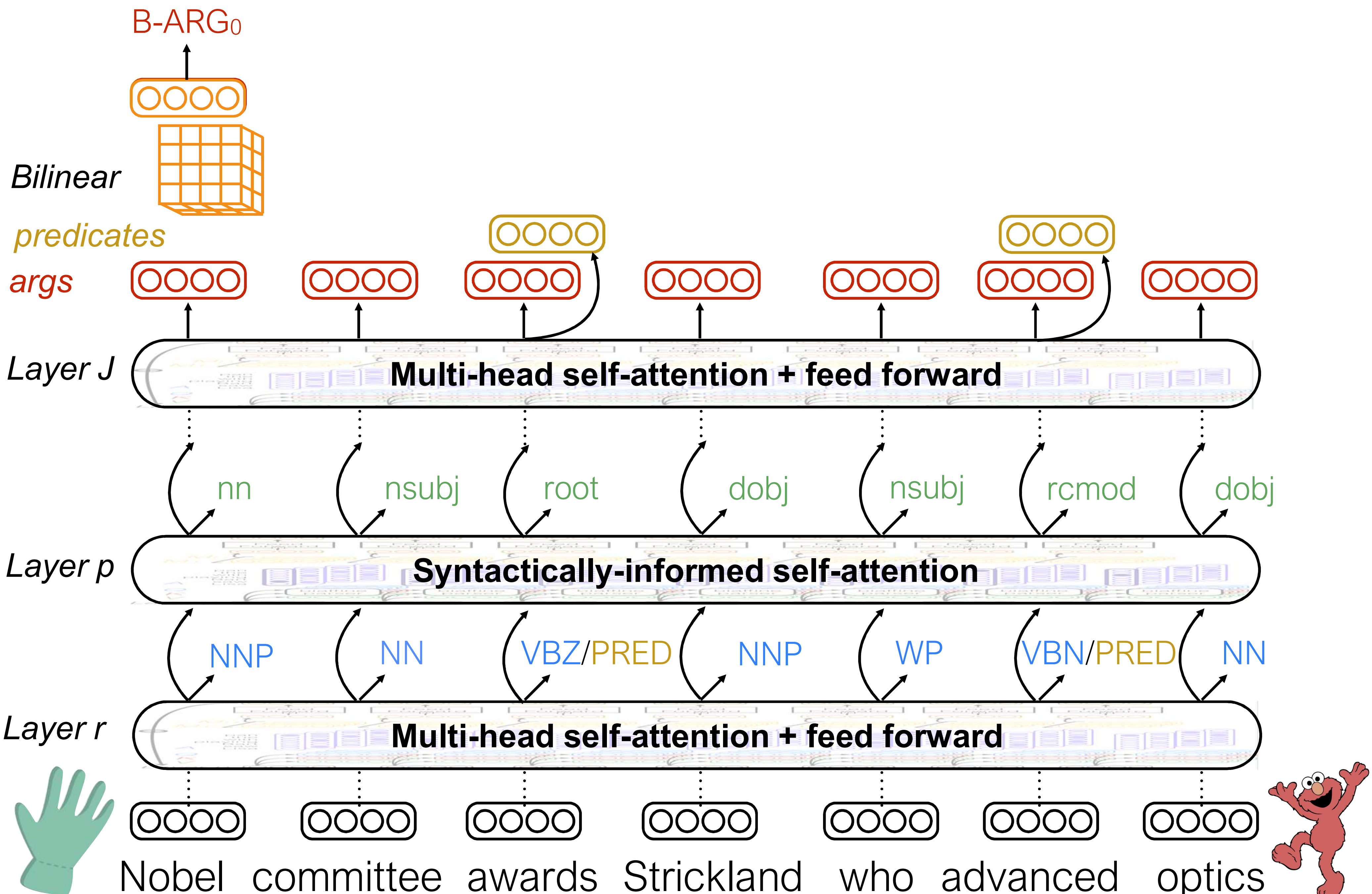
LISA: Linguistically-Informed Self-Attention



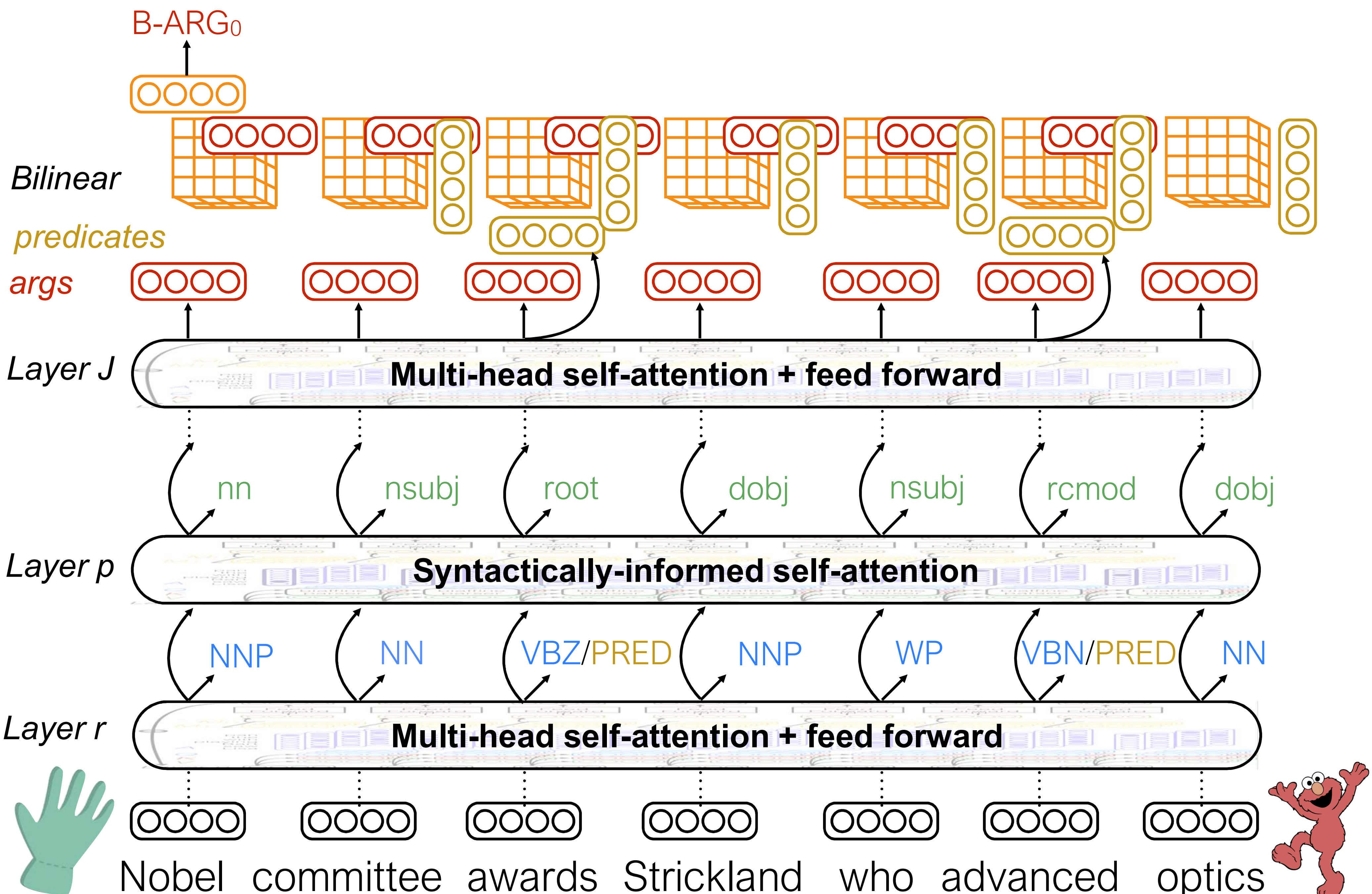
LISA: Linguistically-Informed Self-Attention



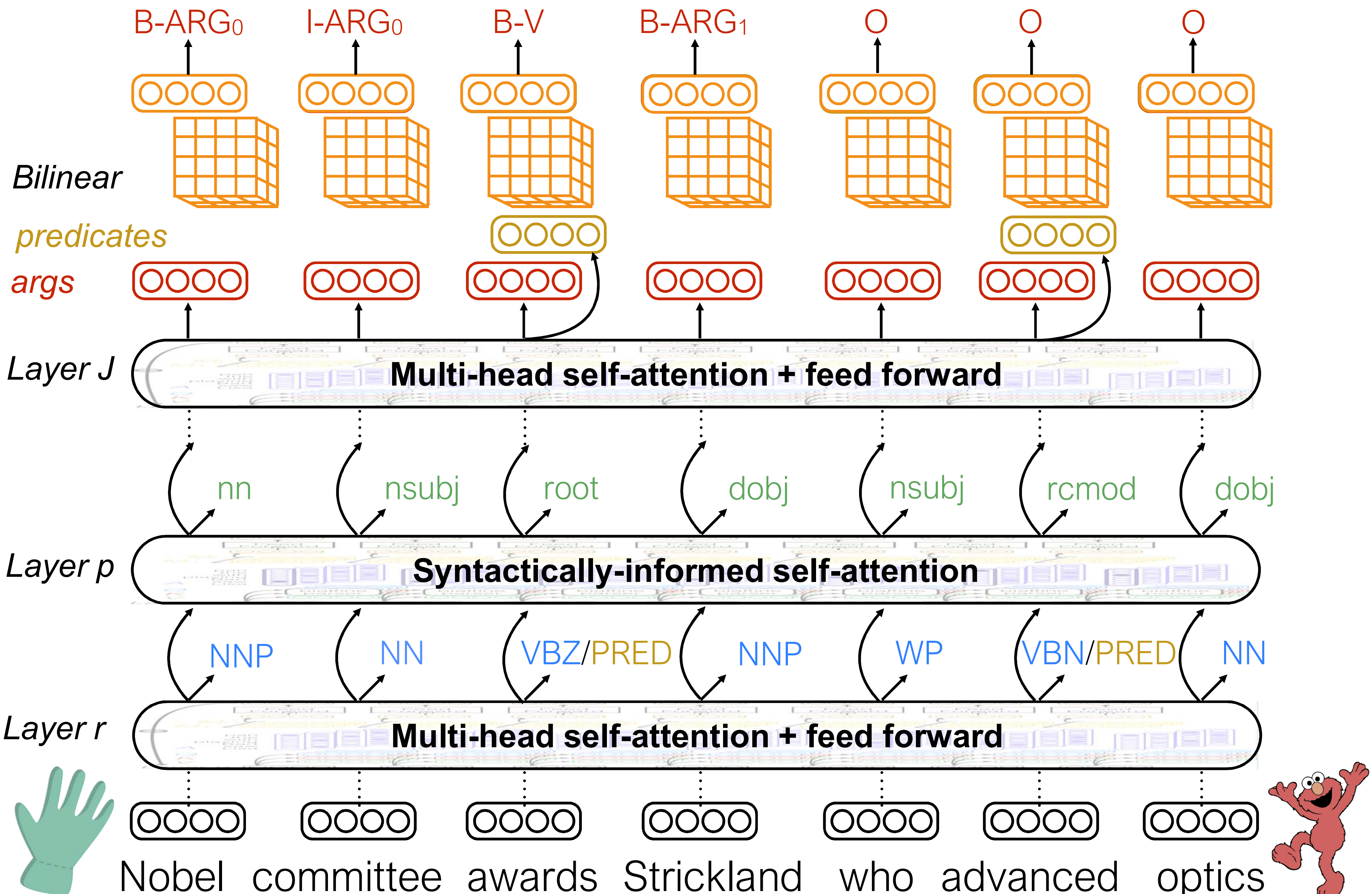
LISA: Linguistically-Informed Self-Attention



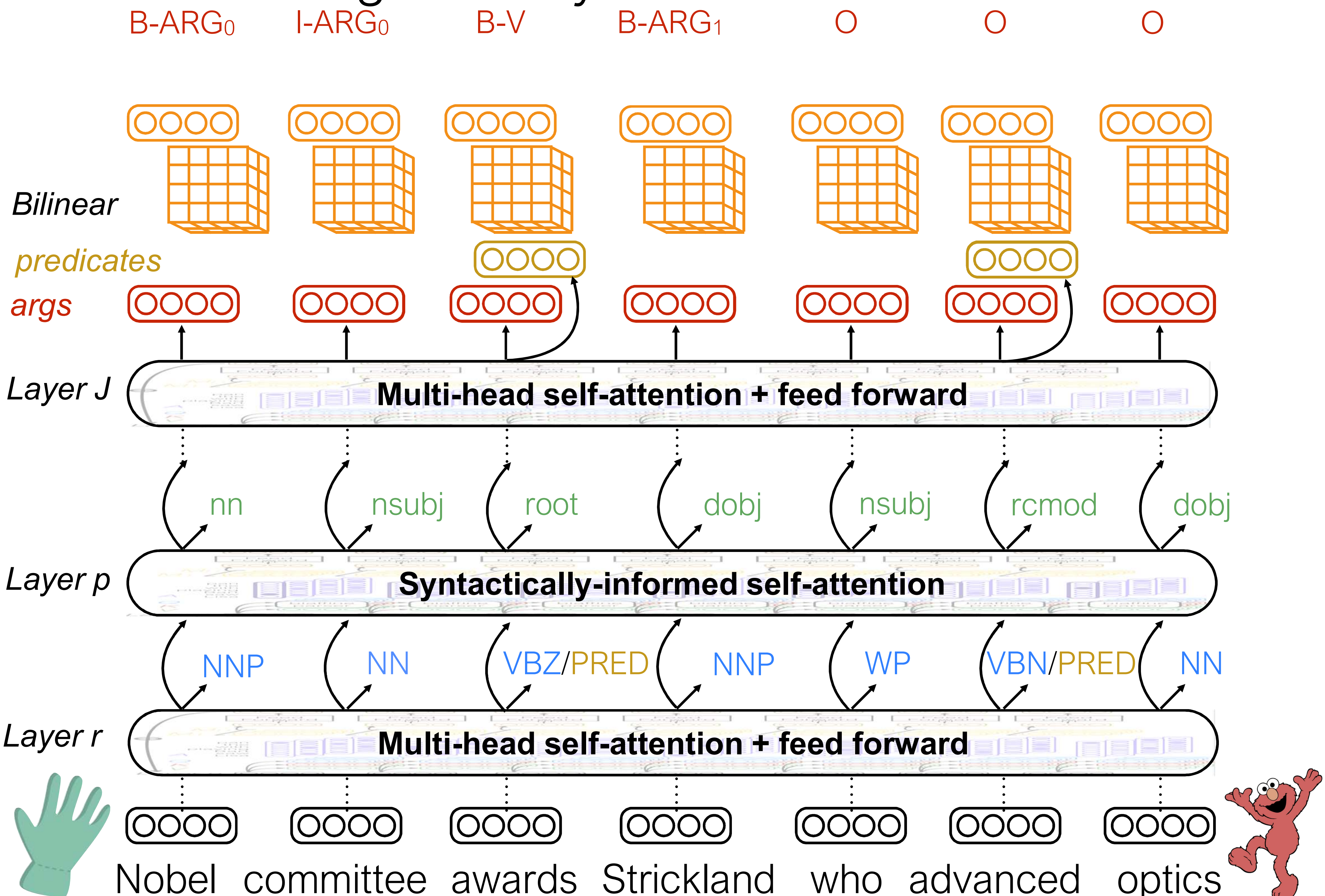
LISA: Linguistically-Informed Self-Attention



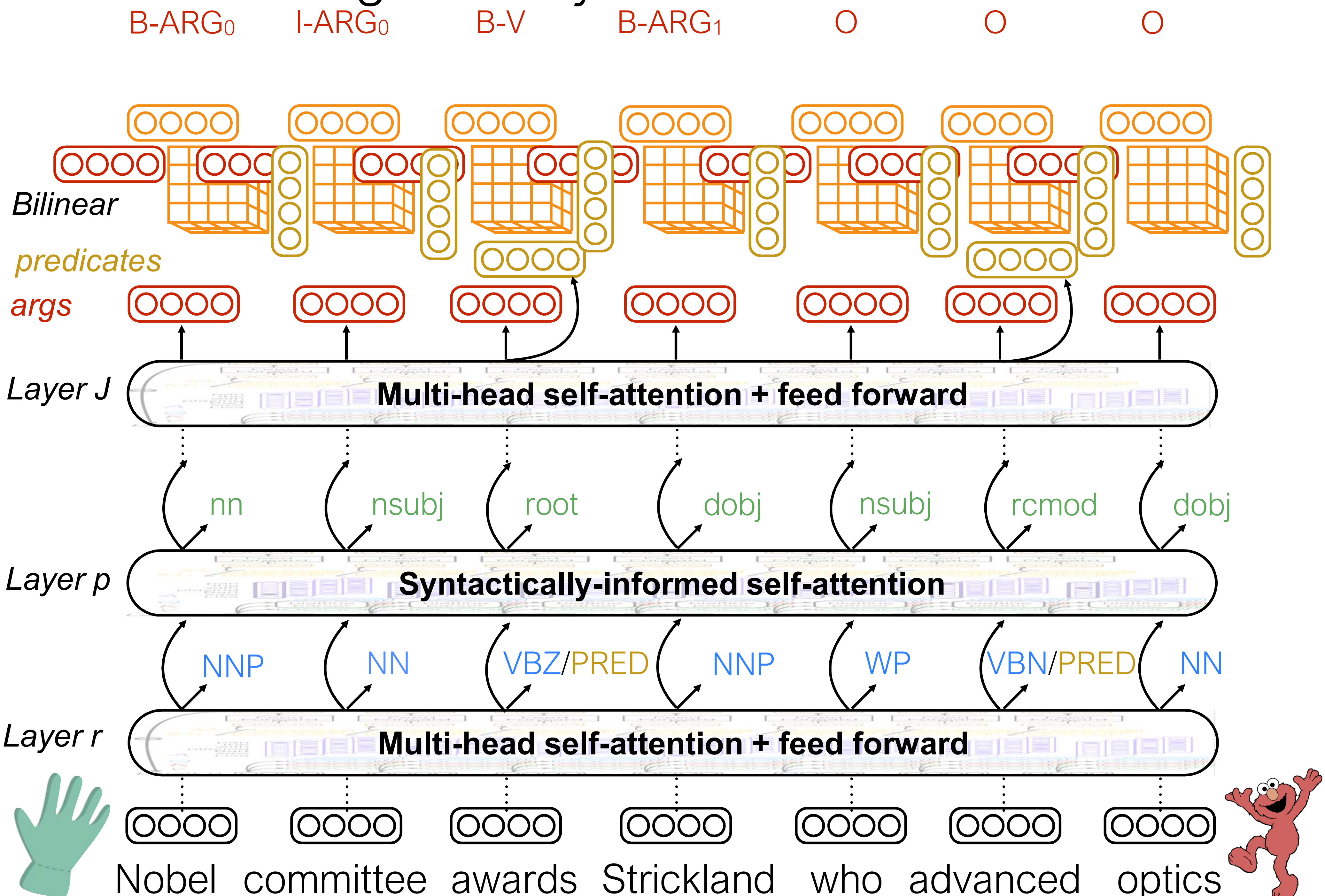
LISA: Linguistically-Informed Self-Attention



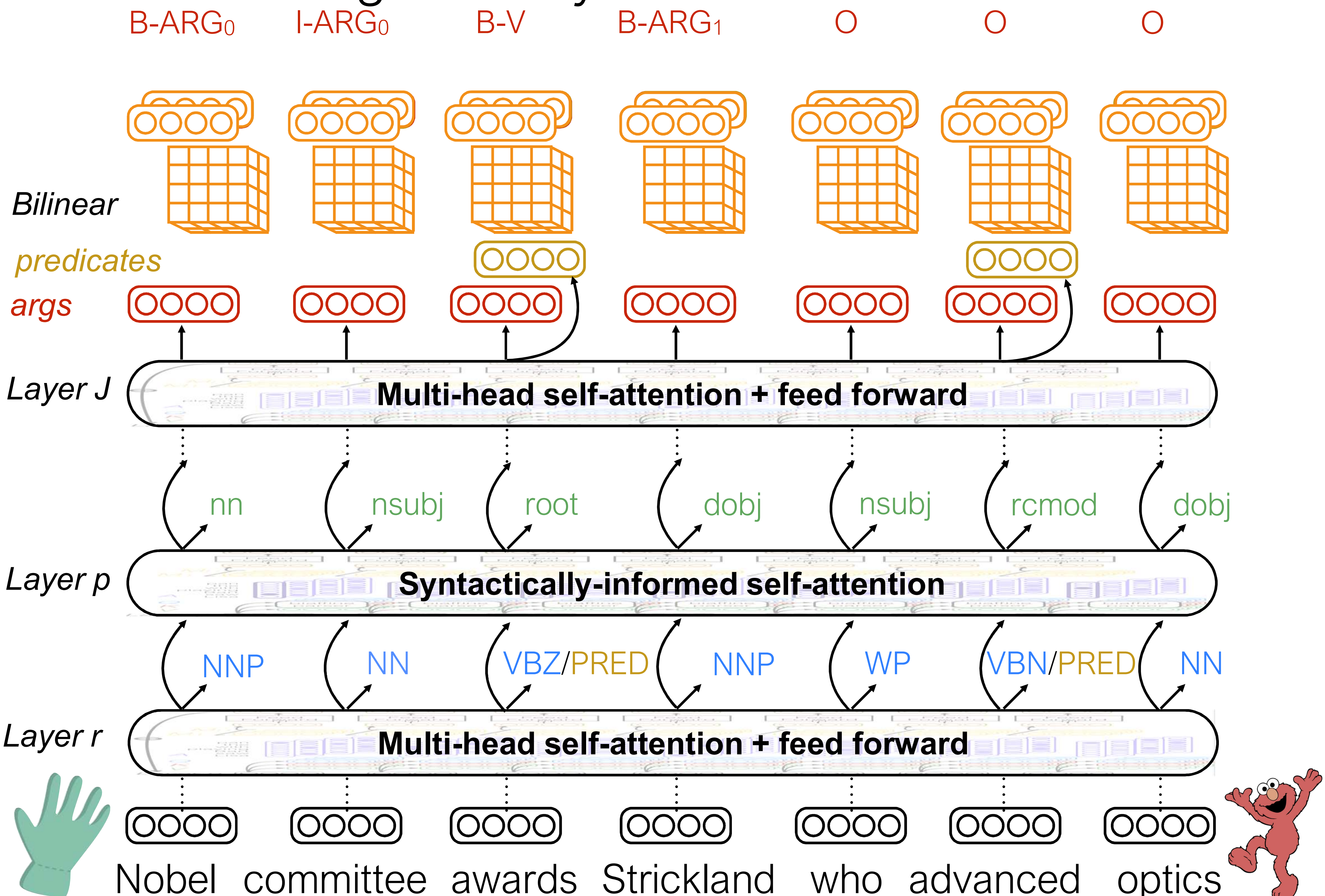
LISA: Linguistically-Informed Self-Attention



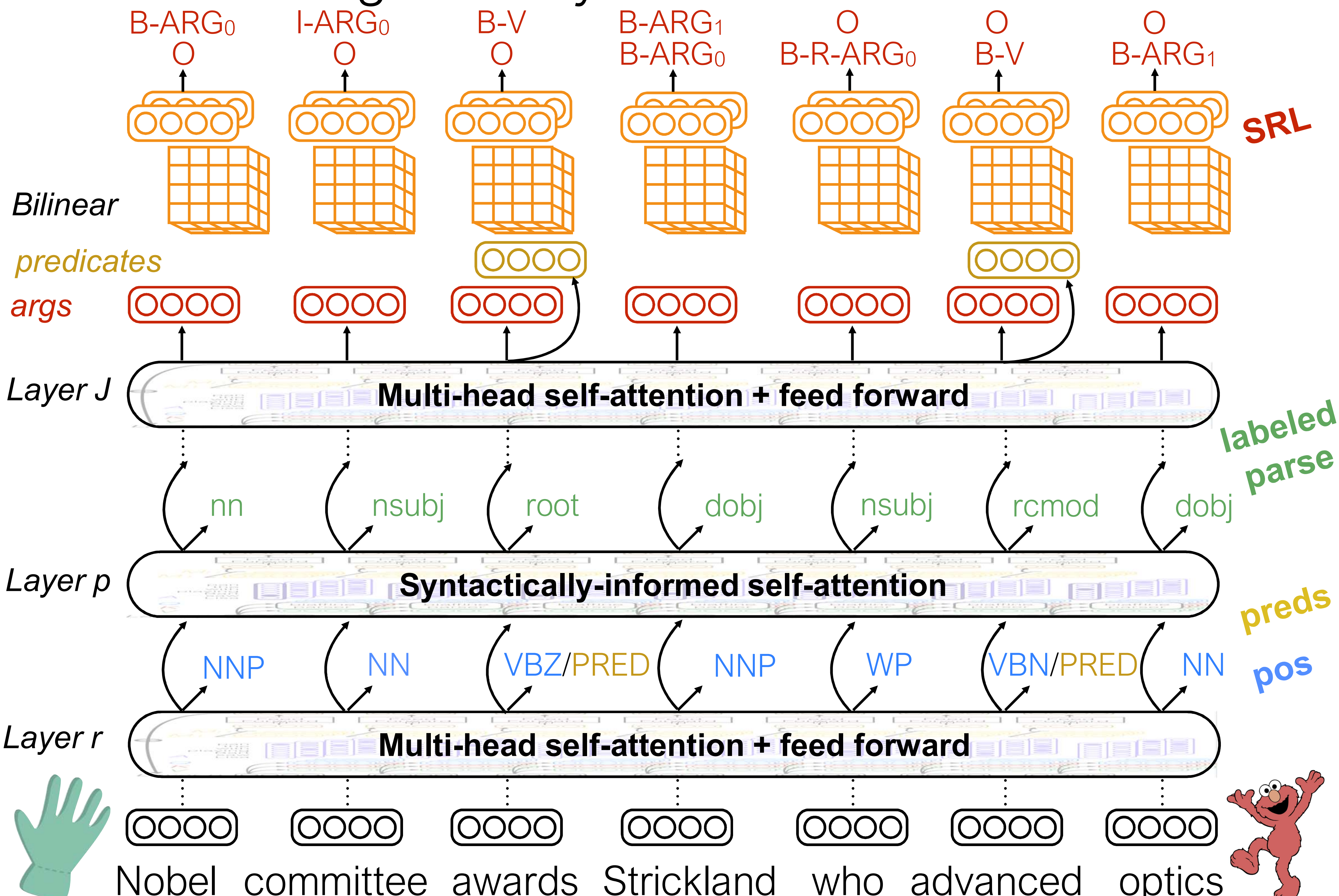
LISA: Linguistically-Informed Self-Attention



LISA: Linguistically-Informed Self-Attention



LISA: Linguistically-Informed Self-Attention



LISA: Linguistically-Informed Self-Attention

$$\frac{1}{T}\sum_{t=1}^{T}\Big[\sum_{f=1}^{F}\log P(y_{ft}^{role} \mid \mathcal{P}_G, \mathcal{V}_G, \mathcal{X})$$

$$+ \log P(y_t^{prp} \mid \mathcal{X})$$

$$+ \lambda_1 \log P(\mathrm{head}(t) \mid \mathcal{X})$$

$$+ \lambda_2 \log P(y_t^{dep} \mid \mathcal{P}_G, \mathcal{X})\Big]$$

Outline

- LISA: Linguistically-informed self attention
  - Multi-head self-attention
  - Syntactically-informed self-attention
  - Multi-task learning, single-pass inference
- Experimental results & error analysis

Experimental results

	CoNLL-2005	CoNLL-2012
domains	Train, dev: news Test: news, novels	Train, dev, test: 7 domains (news, telephone, bible, ...)
word embeddings	GloVe [Pennington et al. 2014] ELMo [Peters et al. 2018]	GloVe [Pennington et al. 2014] ELMo [Peters et al. 2018]
predicates	predicted; gold	predicted
baselines	He et al. 2017 He et al. 2018 Tan et al. 2018	He et al. 2018
our models	SA LISA LISA+D&M, +Gold Lisa_Gold	SA LISA LISA+D&M, +Gold Lisa_Gold

Experimental results

He et al. 2017	PoE
He et al. 2018	jointly predict all predicates and argument spans
SA	does not incorporate syntactic information
LISA	Predicted parser
+D&M	injecting state-of-the-art predicted parses at test time (+D&M)
+Gold	the gold syntactic parse at test time (+Gold)

Experimental results

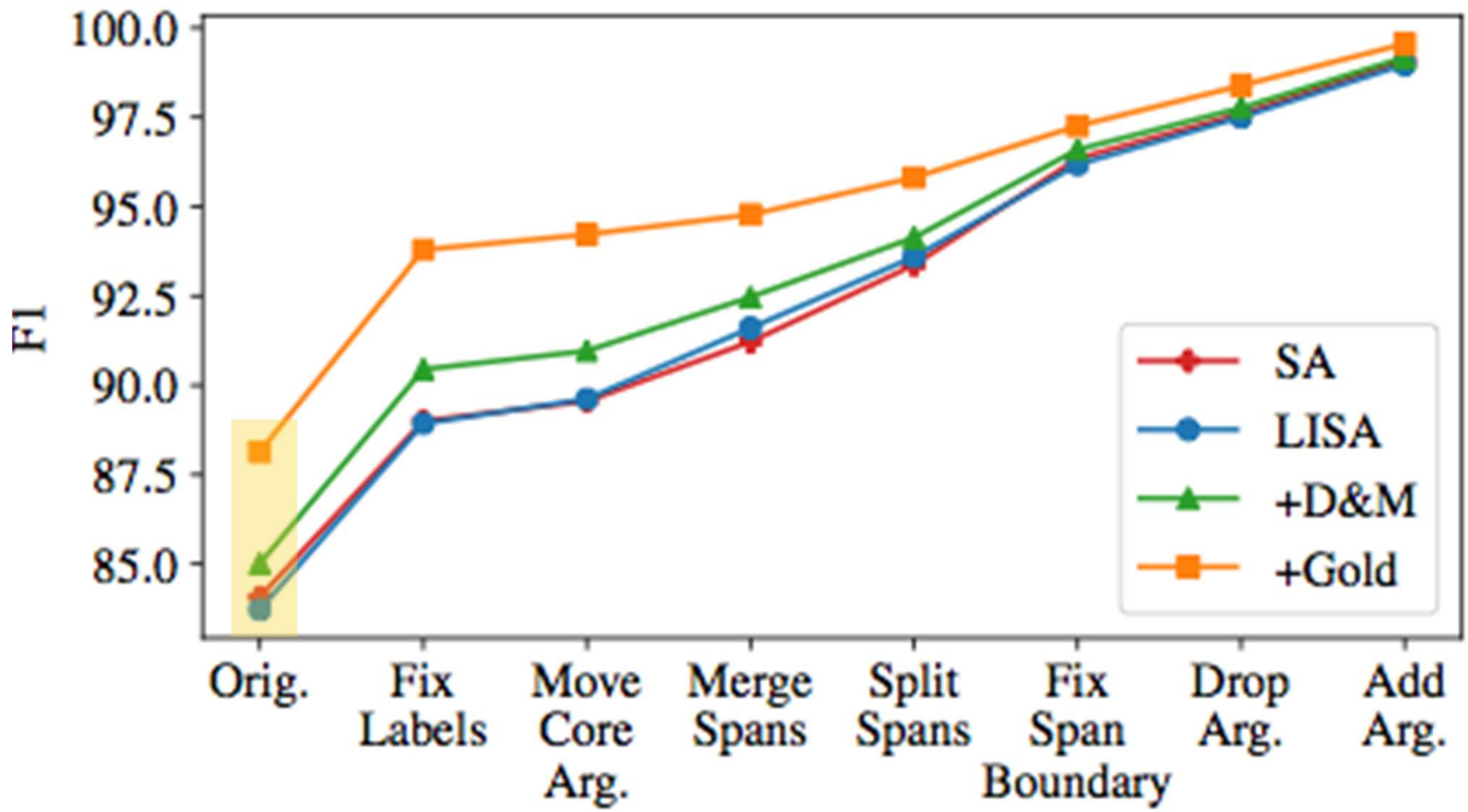
	 **GloVe**		 **ELMo**	
	in-domain	**out-of-domain**	**in-domain**	**out-of-domain**
He et al. 2017	82.7	70.1	---	---
He et al. 2018	82.5	70.8	86.0	76.1
SA	83.72	71.51	86.09	76.35
LISA	83.61	71.91	86.55	78.05
+D&M	84.99 94.9 UAS	74.66 90.3 UAS	86.90 96.3 UAS	78.25 93.4 UAS
	+2.49 F1	+3.86 F1	+0.9 F1 ?	+2.15 F1

Experimental results: CoNLL-2005

	GloVe	ELMo 
	in-domain (dev)	in-domain (dev)

96.5 UAS!

Experimental results: Analysis



Error Breakdown

Oracle Transformations

1) Fix Label:

[We] *fly* to NYC tomorrow.

~~ARG0~~
ARG1

2) Move core arg:

ARG0 ← ARG0

[They] wrote **[an email]** to *cancel* it.

3) Split/ Merge span:

ARG1
I *eat* **[pasta with delight]**.
ARG1 ARGM-MNR
[pasta] [with delight]

ARG1 ARGM-MNR
I *eat* **[pasta] [with broccoli]**.
ARG1
[pasta with broccoli]

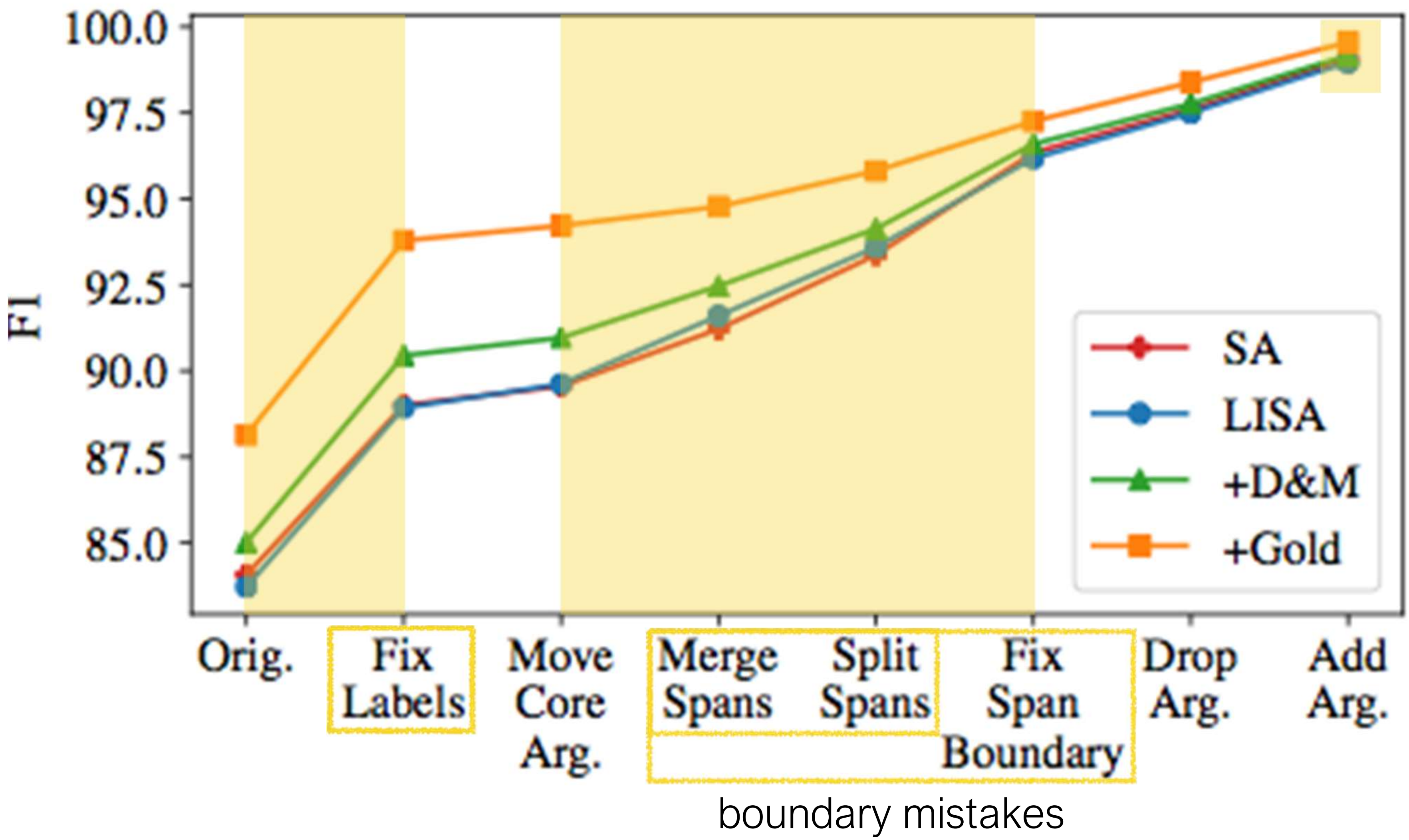
4) Fix span boundary:

ARG1
[“No broccoli”,] I said.
[“No broccoli”] ,

5) Drop/ add arg:

Hesitantly, they *declined* to elaborate **[on that matter]**.
~~ARG1~~
+ ARGM-MNR
[Hesitantly], they *declined* to elaborate on that matter.

Experimental results: Analysis



Summary

- **LISA**: Multi-task learning + multi-head self attention trained to attend to syntactic parents
  - Achieves state-of-the-art F1 on PropBank SRL
  - Linguistic structure improves generalization
  - Fast: encodes sequence ***only once*** to predict predicates, parts-of-speech, labeled dependency parse, SRL

Slide reference: Emma Strubell
on Linguistically-Informed Self-Attention for Semantic Role Labeling (best paper, EMNLP 2018)