TOWARDS SIMPLER TREE SUBSTITUTION GRAMMARS

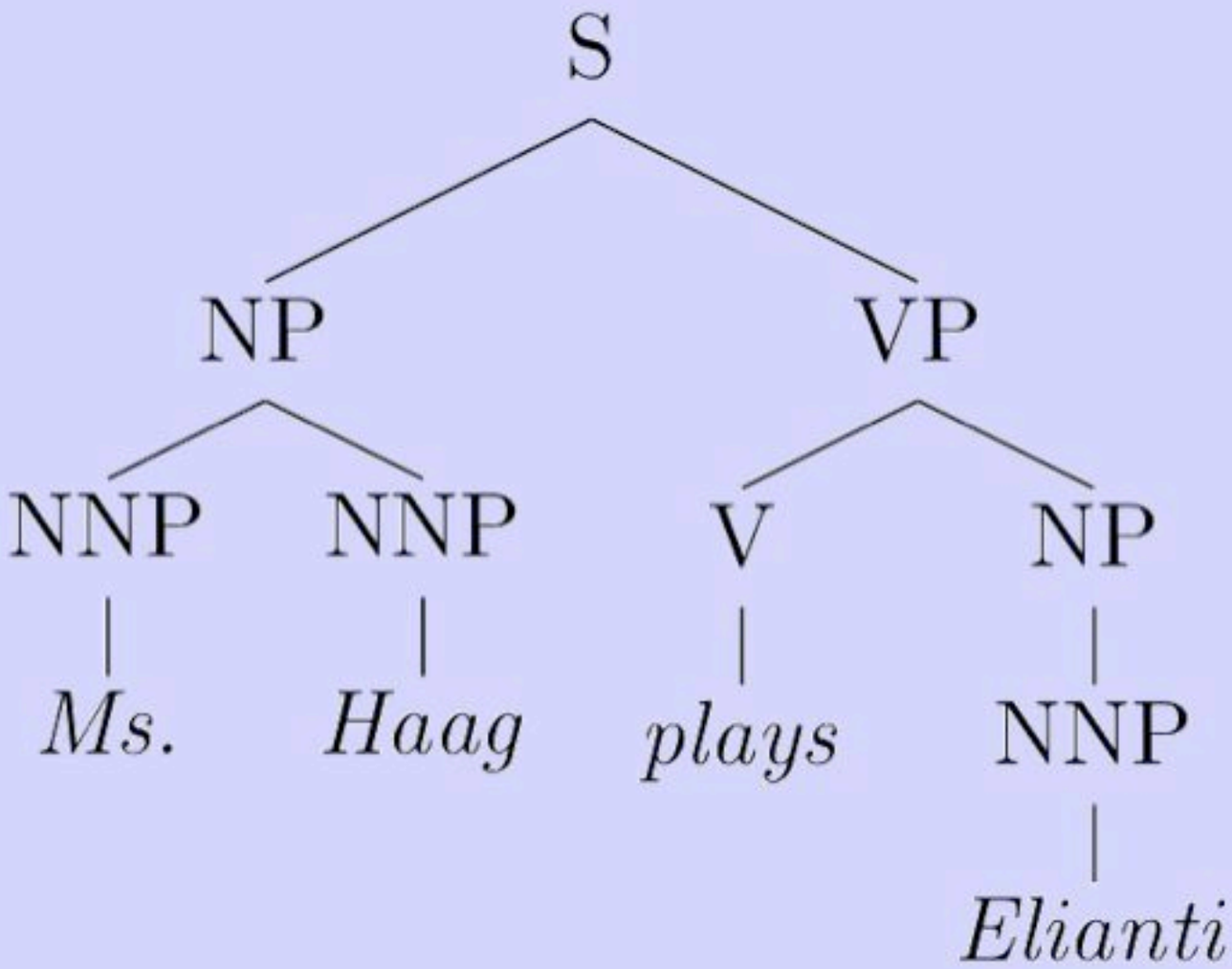
MSc Thesis Defense

Federico Sangati

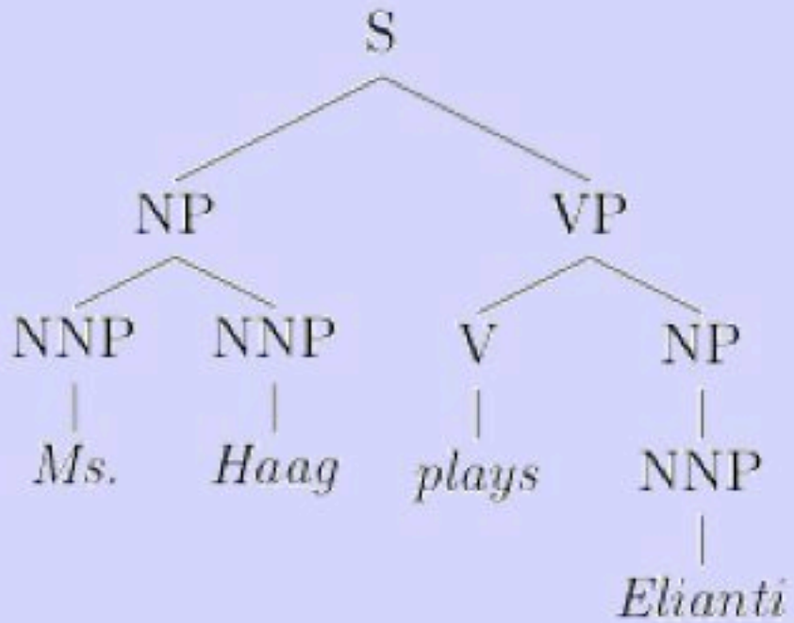
OUTLINE

- *Syntactic Structures*
- *STSGs intro*
- *Comparing different STSGs*
- *LTSGs*
- *Automatically inducing head dependency*
- *Results*
- *Conclusions*

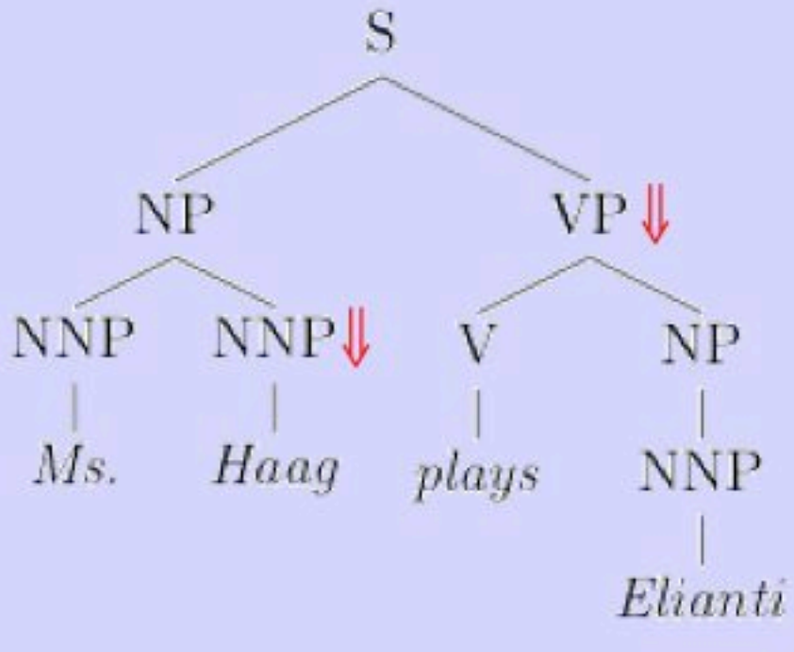
Syntactic Structures



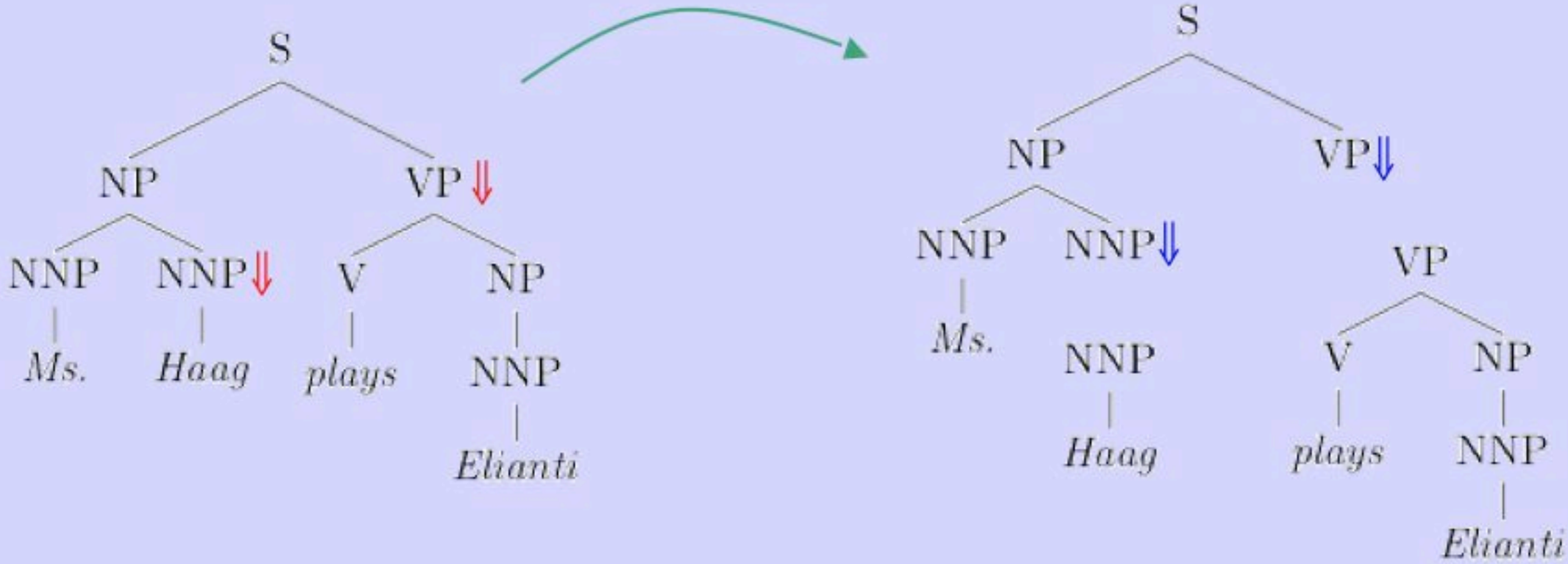
TSGs



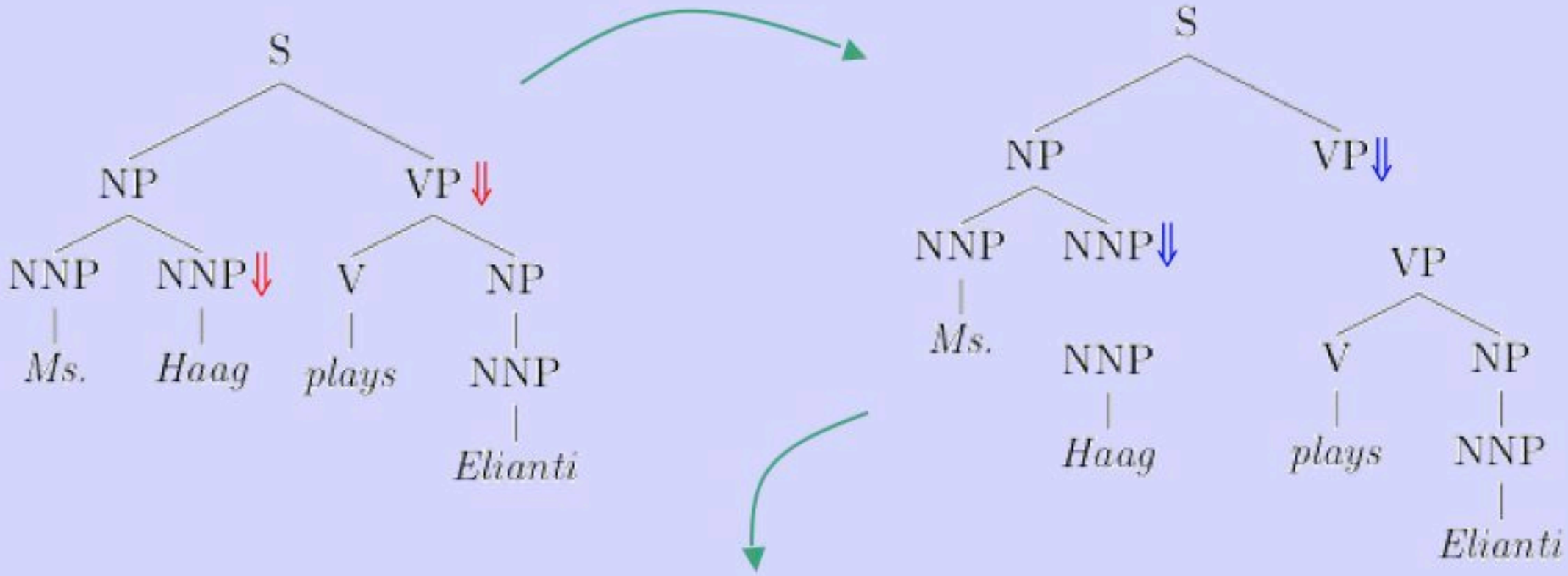
TSGs

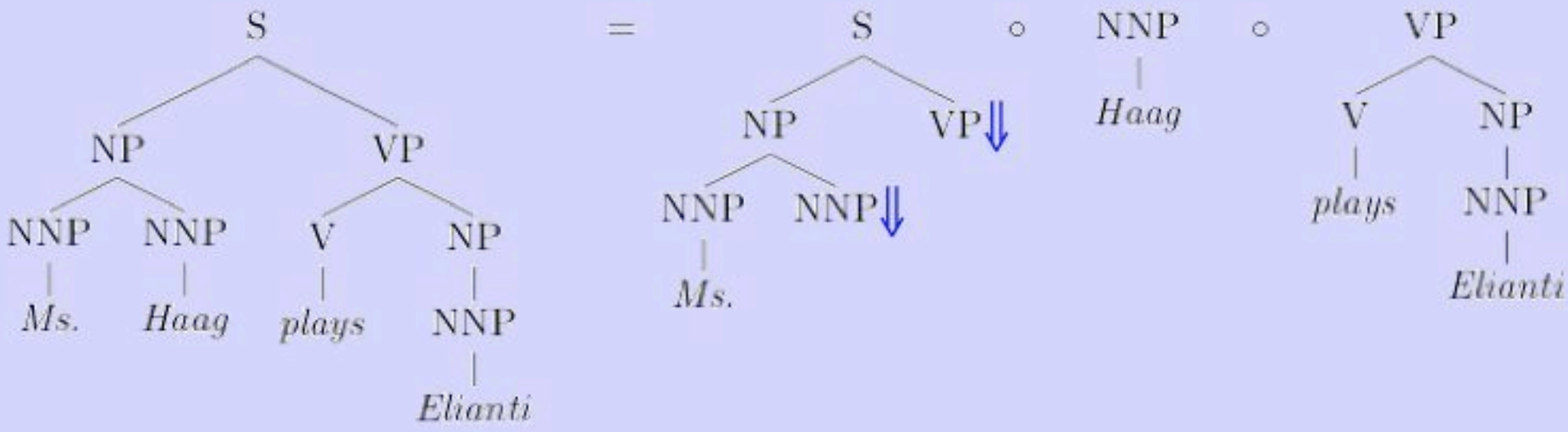


TSGs


S
NP
VP
NNP
NNP
V
NP
Ms.
Haag
plays
NNP
Elianti
S
NP
VP
NNP
NNP
Ms.
NNP
Haag
VP
V
NP
plays
NNP
Elianti

TSGs


S
NP
VP
NNP
NNP
V
NP
Ms.
Haag
plays
NNP
Elianti
S
NP
VP
NNP
NNP
Ms.
NNP
Haag
VP
V
NP
plays
NNP
Elianti


S
NP
VP
NNP
NNP
Ms.
Haag
plays
NNP
Elianti
=
S
NP
VP
NNP
NNP
Ms.
∘
NNP
Haag
∘
VP
V
NP
plays
NNP
Elianti

TSGs

Given an *input corpus* of parsed sentences, TSGs define:

- how to extract from it a bag T of *elementary trees*
- how to combine them to form (*new) parse trees*

Given a new sentence our TSG can produce possibly many parse trees yielding the sentence. We need to select only one (*disambiguation*).

Stochastic TSGs

STSGs define a probabilistic model over the possible parse trees which can be produced by the grammar.

For every parse tree t,

$d : \tau_1 \circ \tau_2 \circ \ldots \circ \tau_n = t$

$P(d) = F(\tau_1) \cdot F(\tau_2) \cdot \ldots \cdot F(\tau_n)$ where $F(\tau) = \frac{f(\tau)}{\sum_{\tau' \in r(\tau)} f(\tau')}$

$d_1, d_2, \ldots, d_m = t$

$P(t) = P(d_1) + P(d_2) + \ldots + P(d_m)$

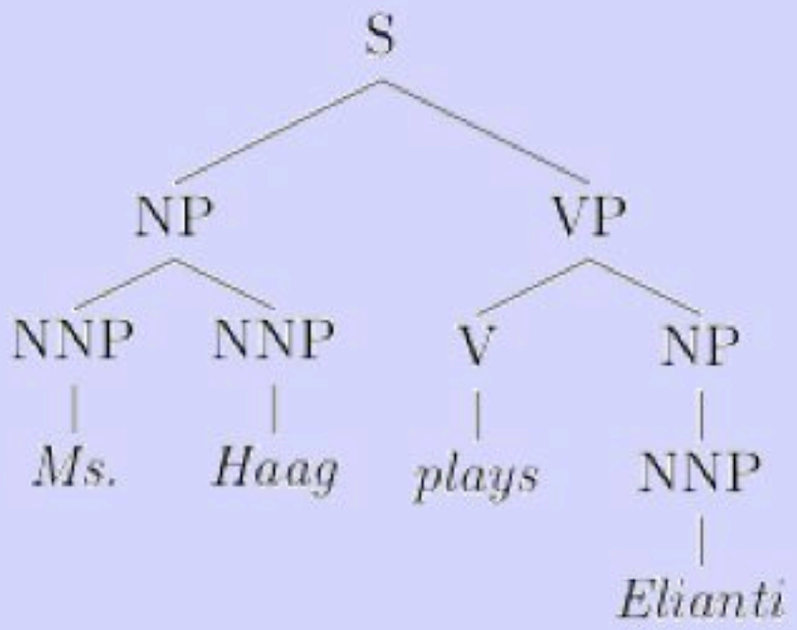
If we have a test sentence, we generate all possible parse trees yielding that sentence and select the one with max probability.

STSGs

STSGs

simple ⟷ complex

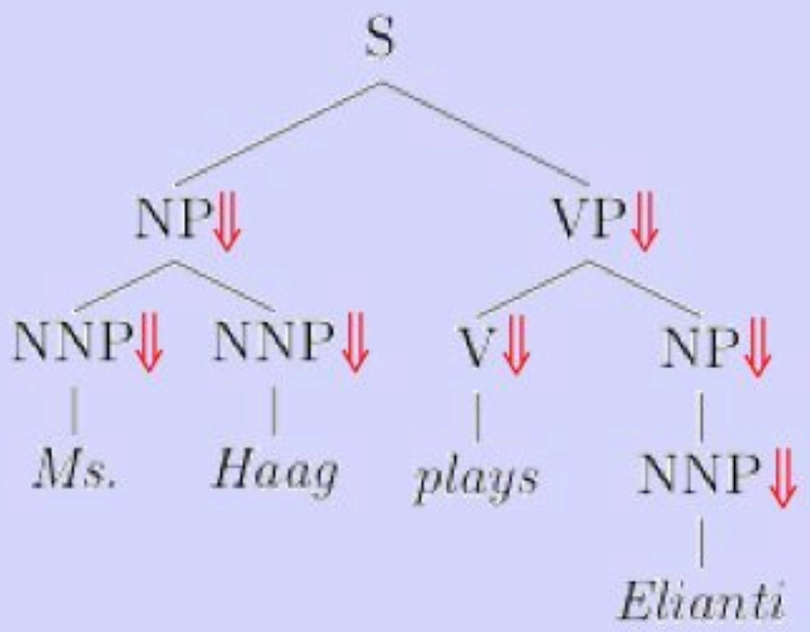
CFG



STSGs

simple ⟷ complex

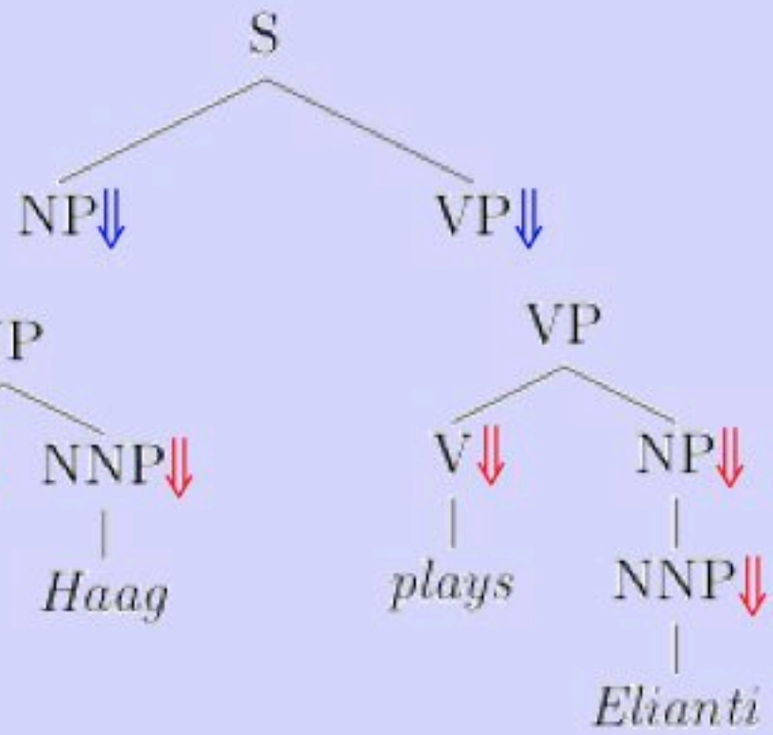
CFG



STSGs

simple ⟷ complex

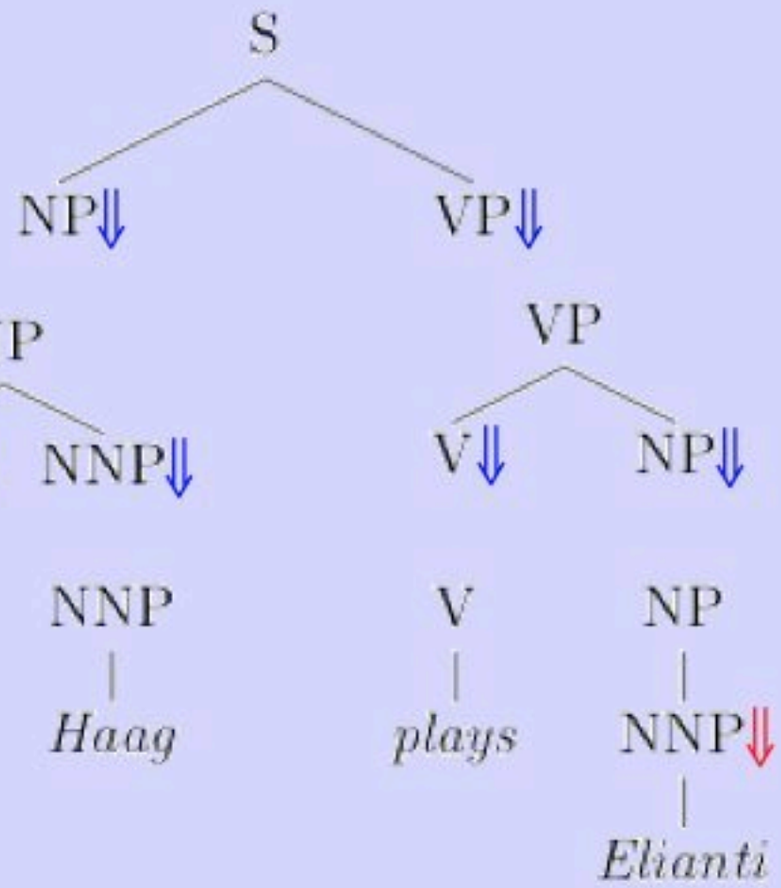
CFG



STSGs

simple ⟷ complex

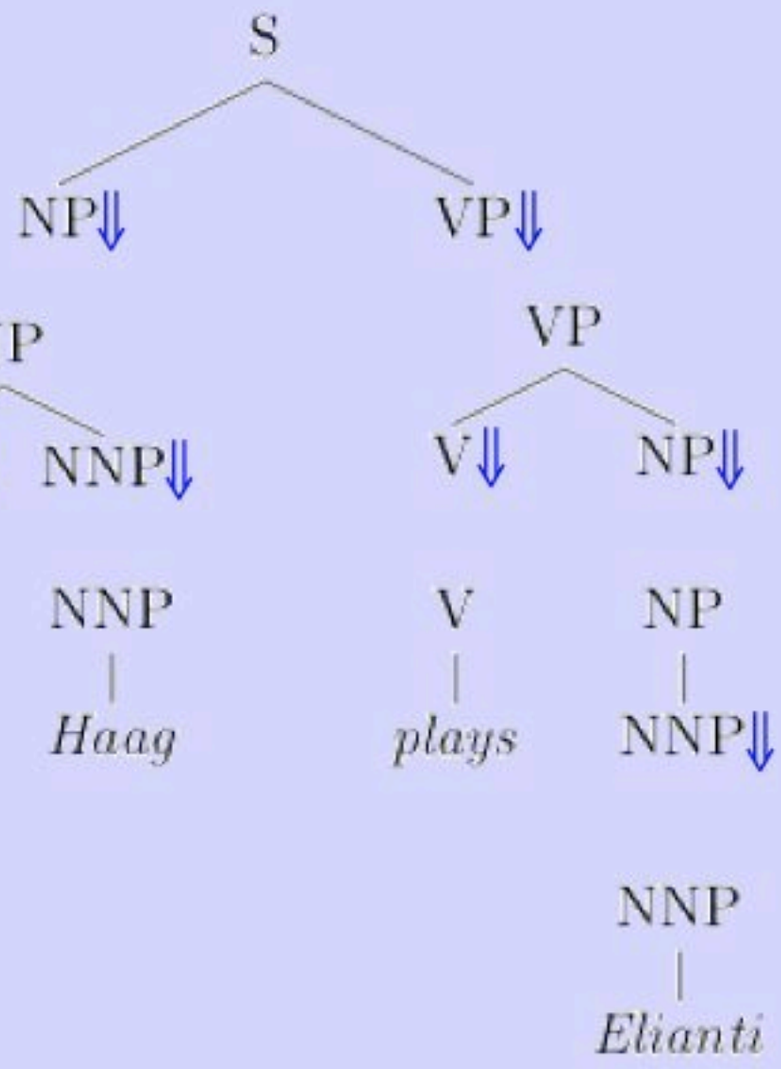
CFG



STSGs

simple
complex

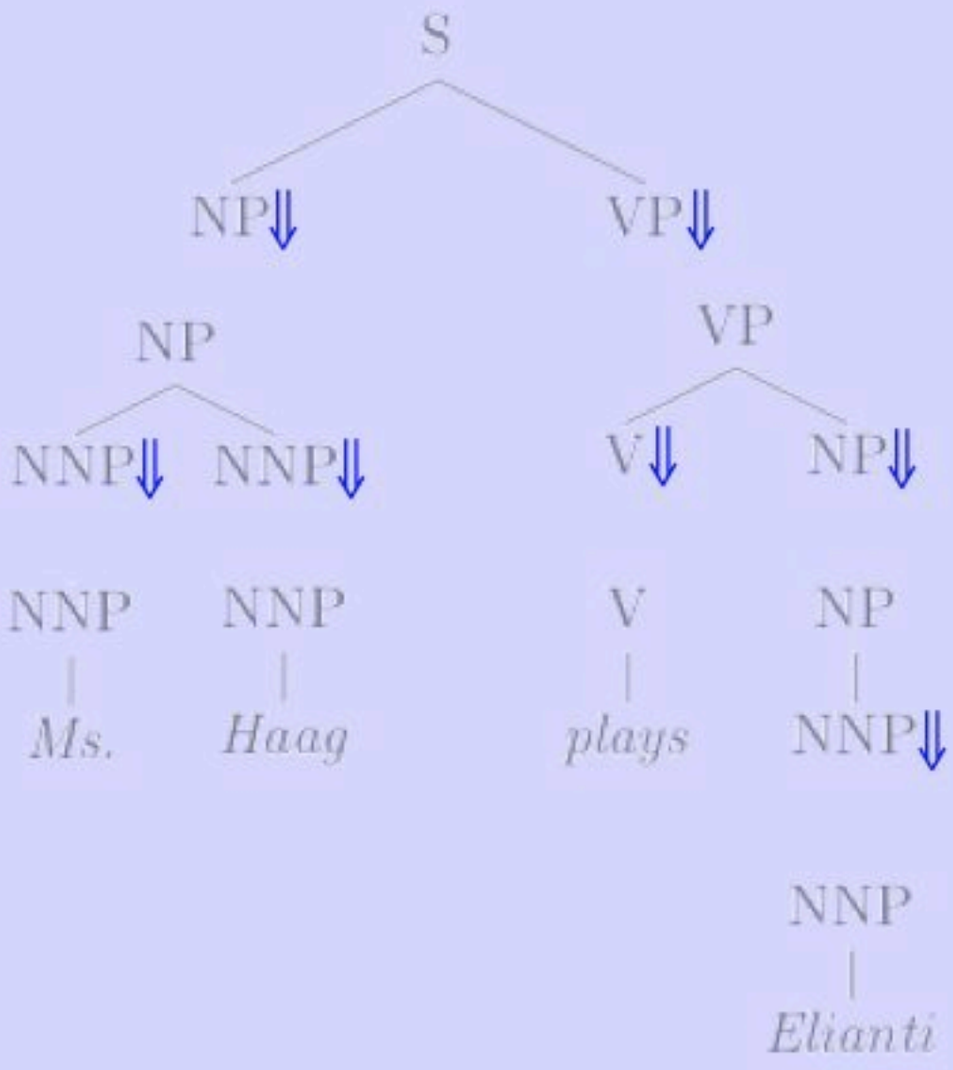
CFG


S
NP⇓
VP⇓
NP
NNP⇓
NNP⇓
VP
V⇓
NP⇓
NNP
NNP
V
NP
Ms.
Haag
plays
NNP⇓
NNP
Elianti

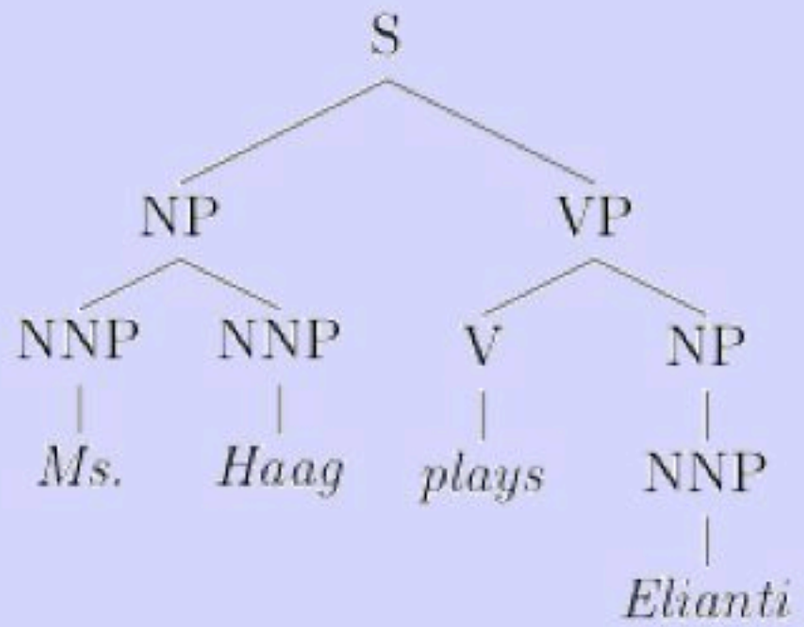
STSGs

simple ⟷ complex

CFG



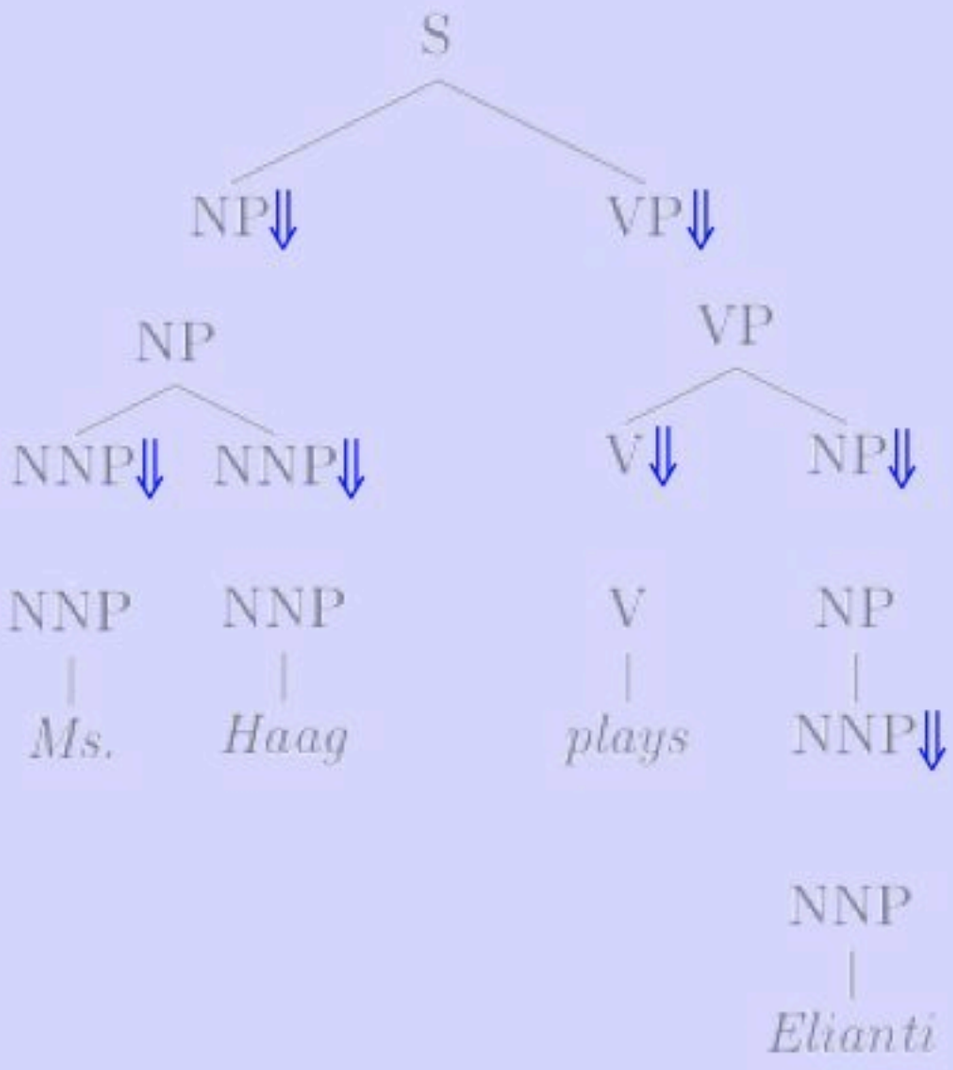
DOP



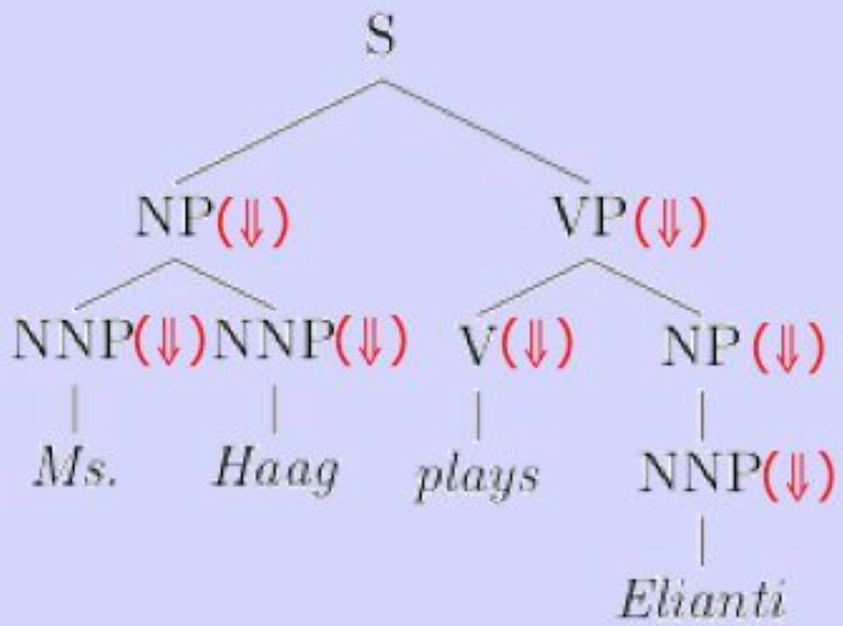
STSGs

simple ⟷ complex

CFG



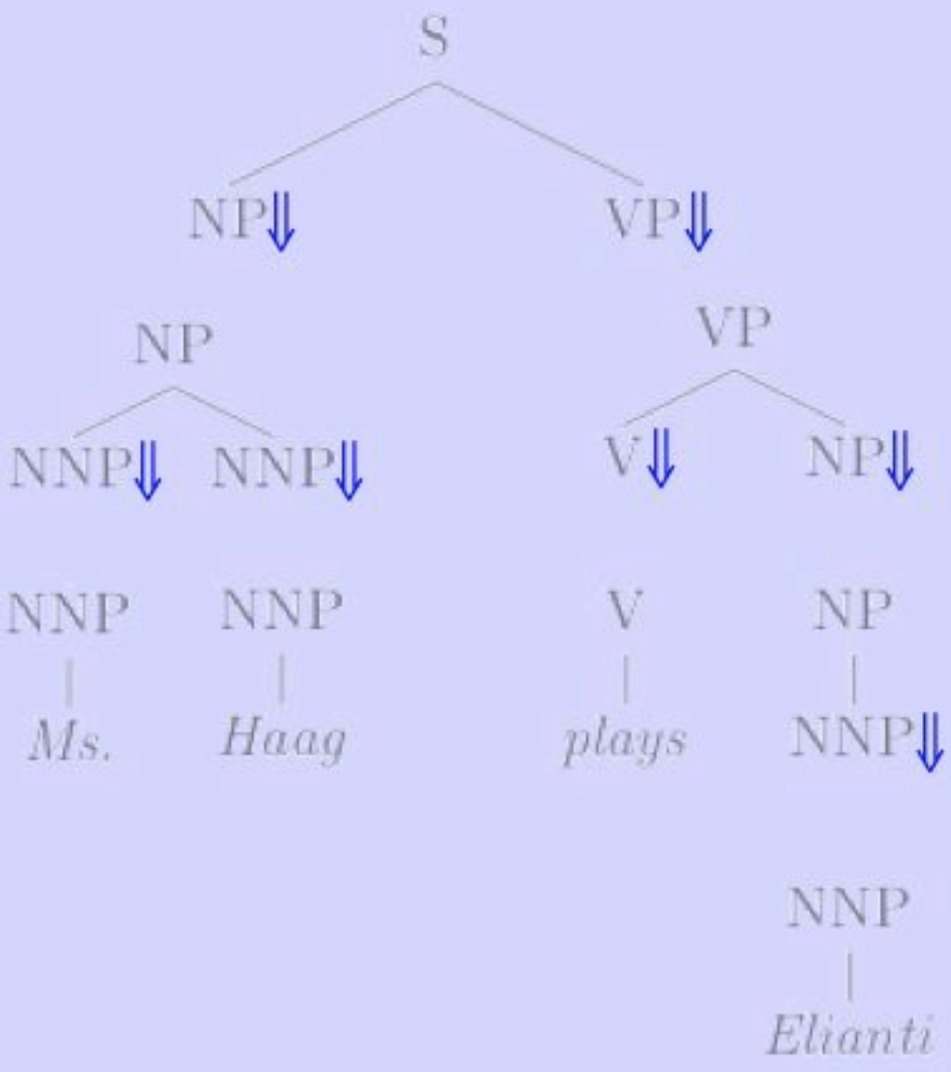
DOP



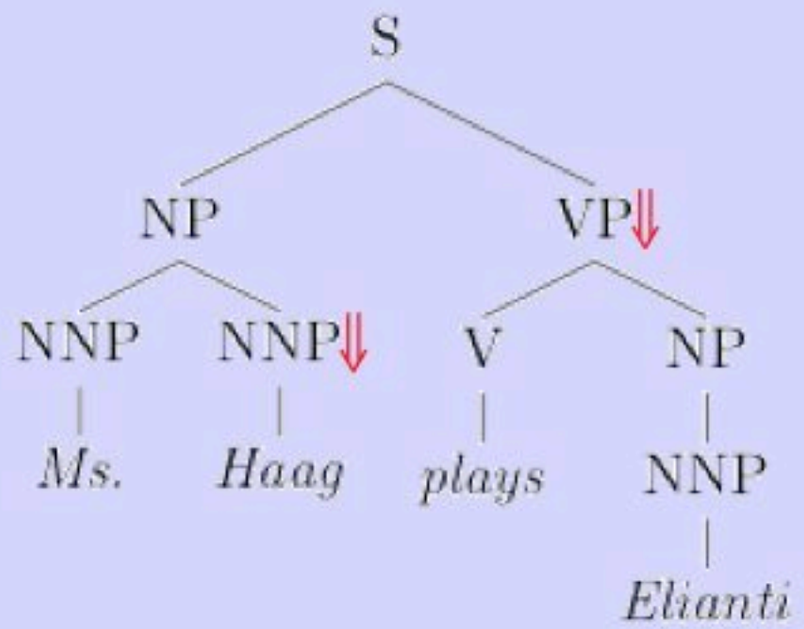
STSGs

simple ⟷ complex

CFG



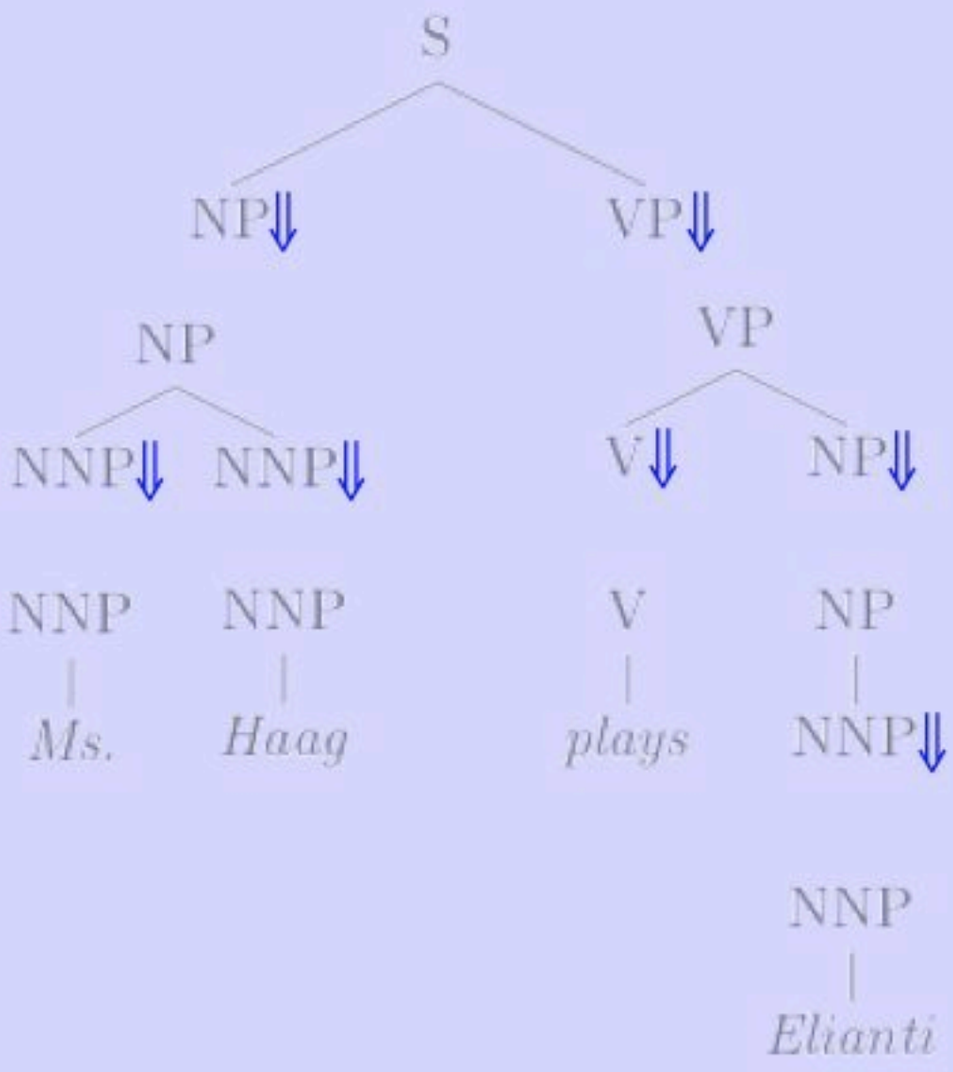
DOP



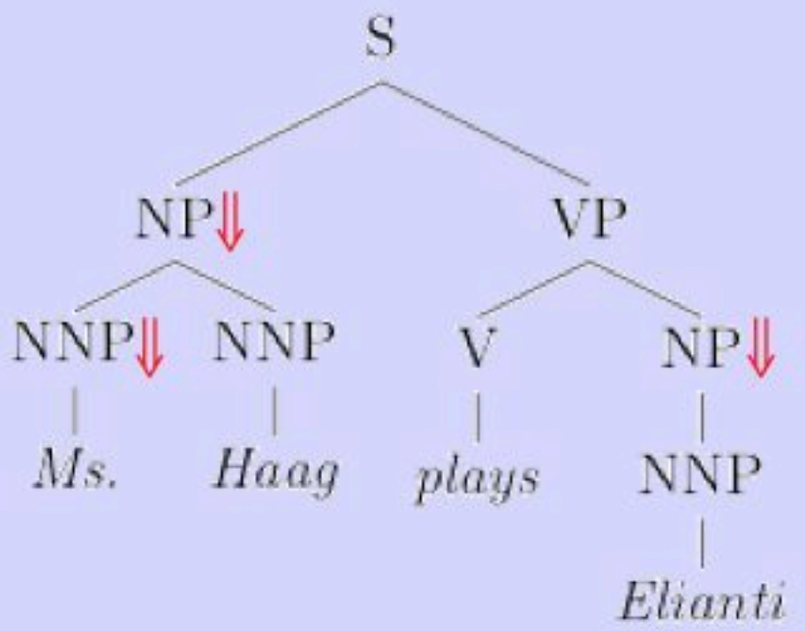
STSGs

simple ⟷ complex

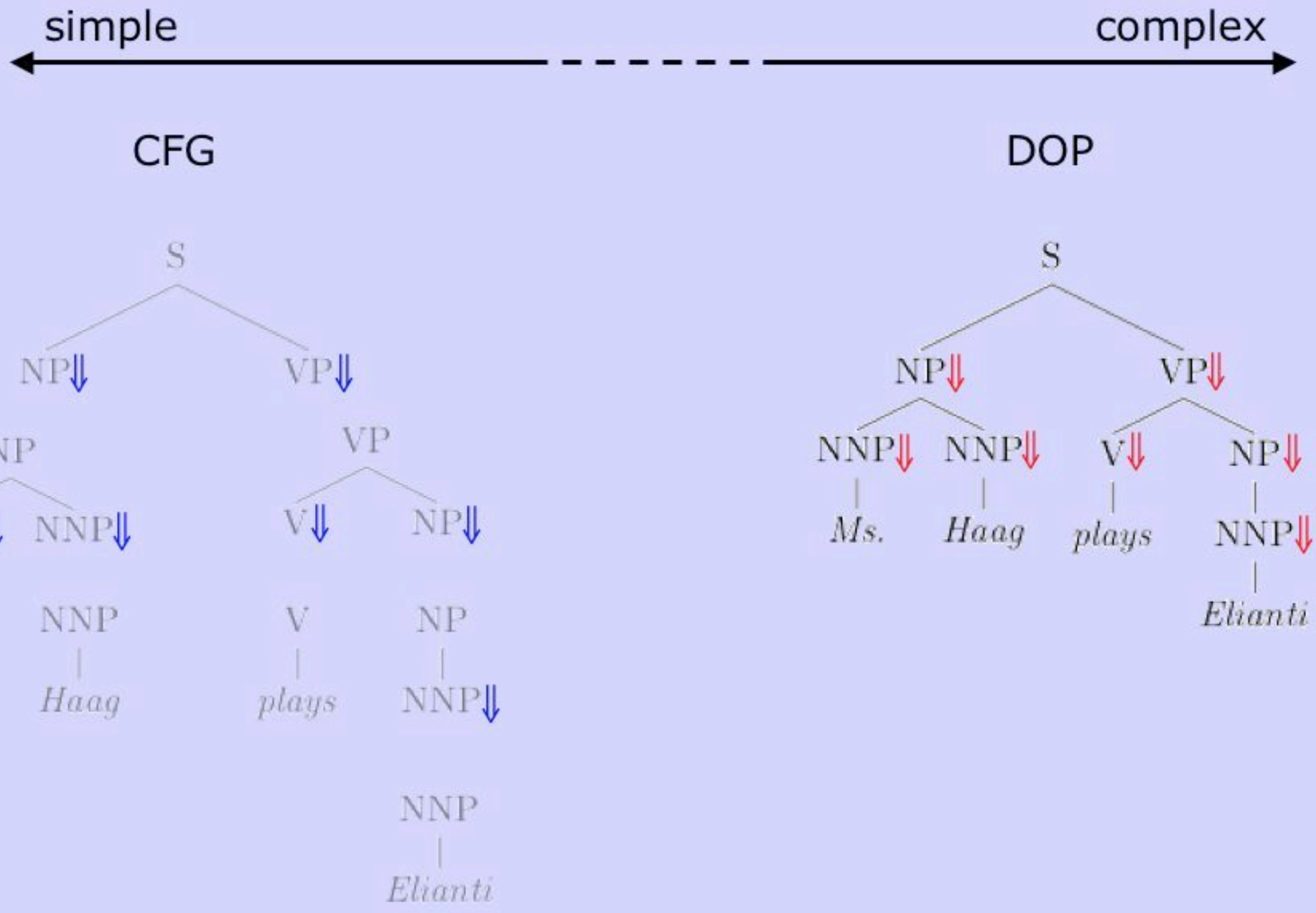
CFG



DOP



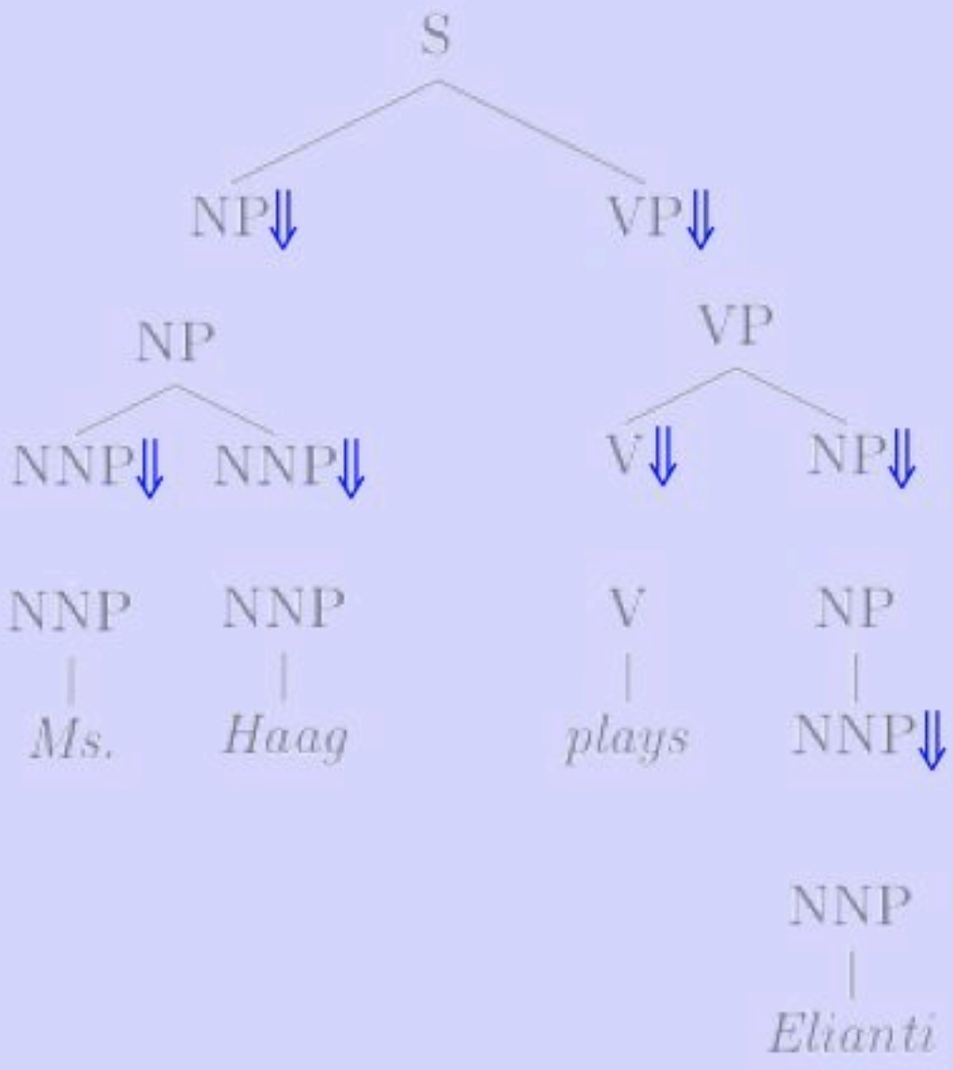
STSGs


simple
complex
CFG
DOP
S
NP
VP
NNP
NNP
V
NP
NNP
Ms.
Haag
plays
Elianti

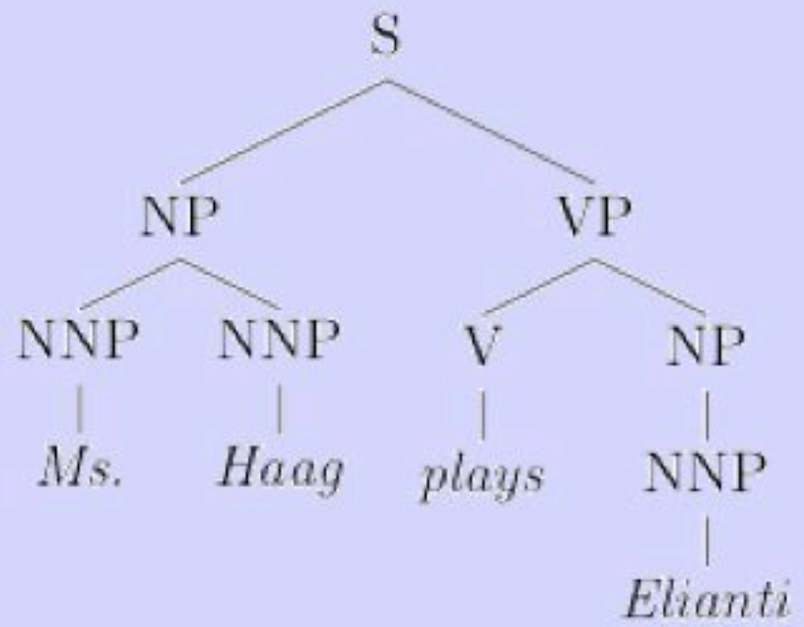
STSGs

simple ⟷ complex

CFG



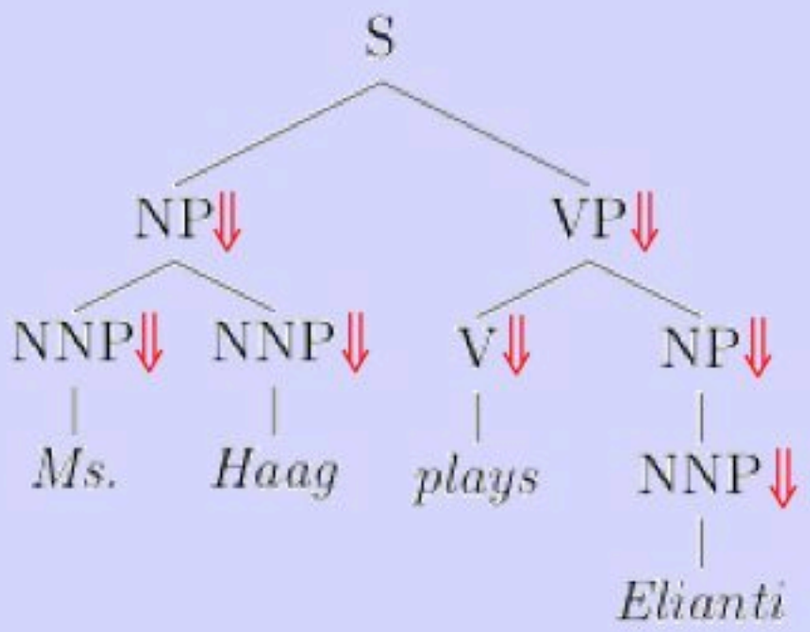
DOP



STSGs

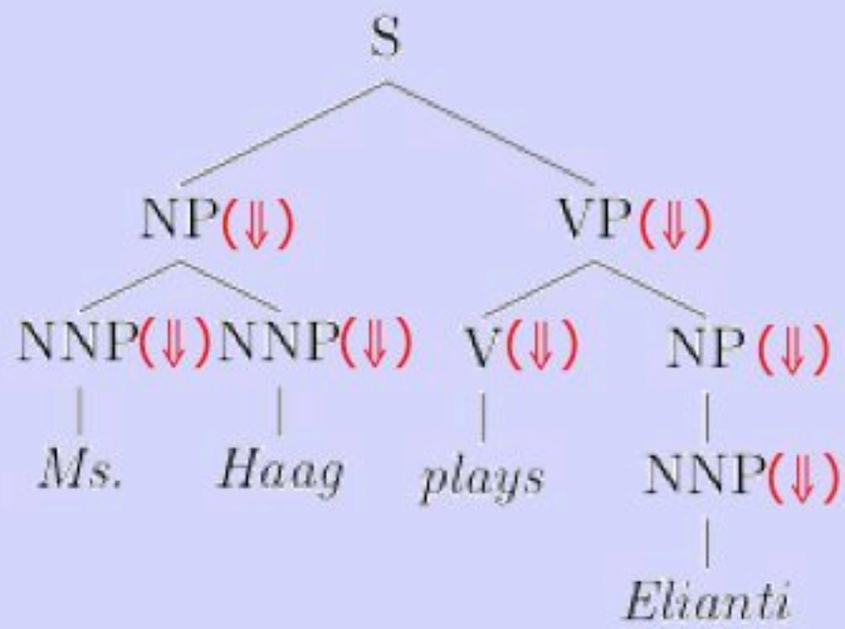
simple ⟷ complex

CFG



1 derivation

DOP

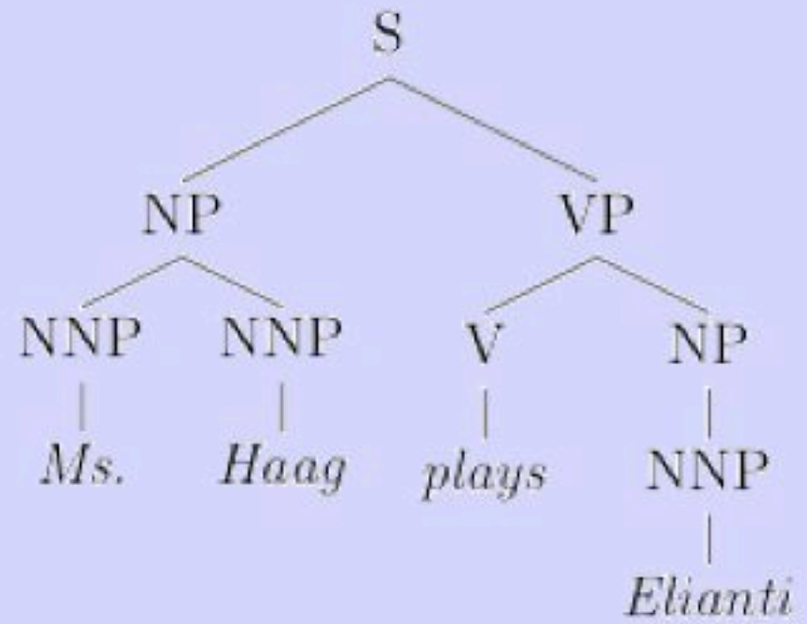


2^N derivations

STSGs

simple ⟷ complex

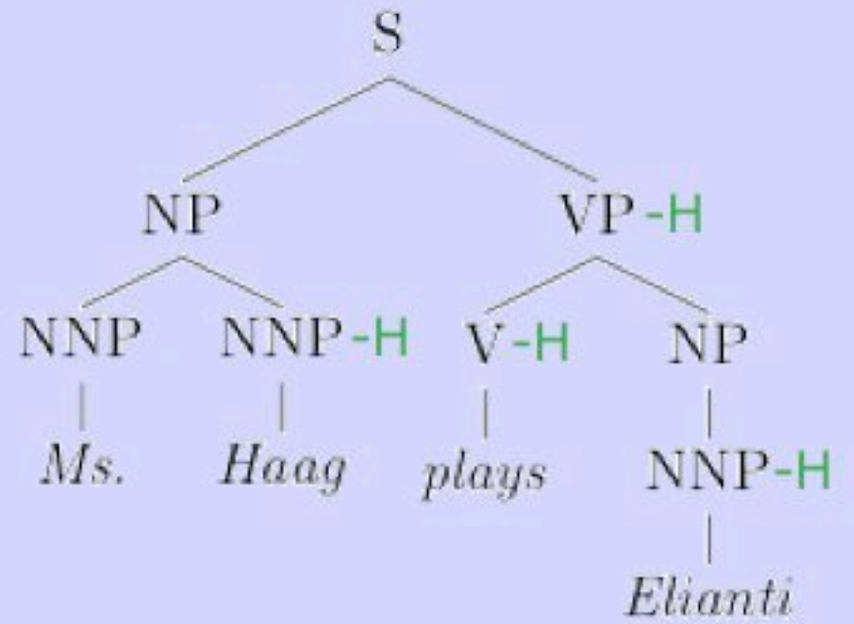
CFG — CM-LTSG — DOP



STSGs

simple ⟷ complex

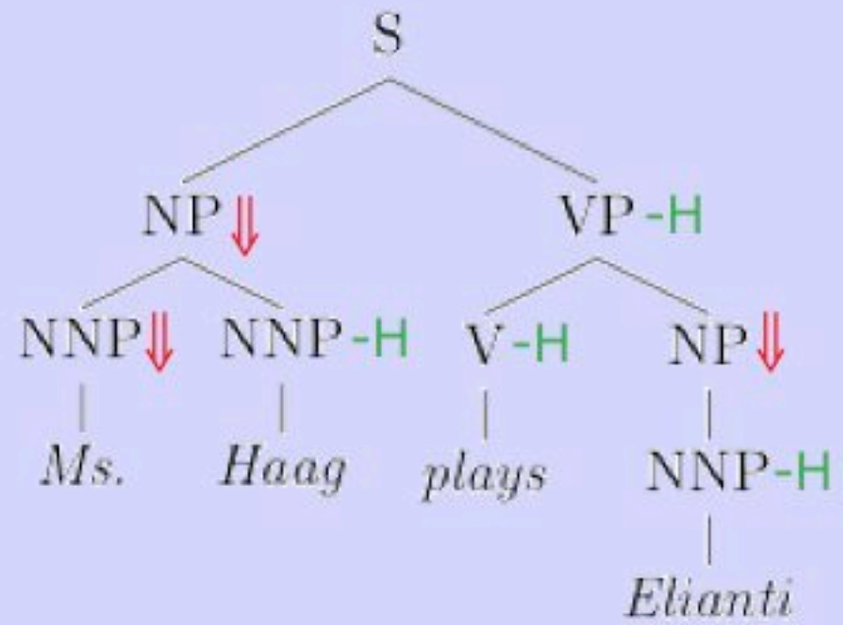
CFG — CM-LTSG — DOP



STSGs

simple ⟷ complex

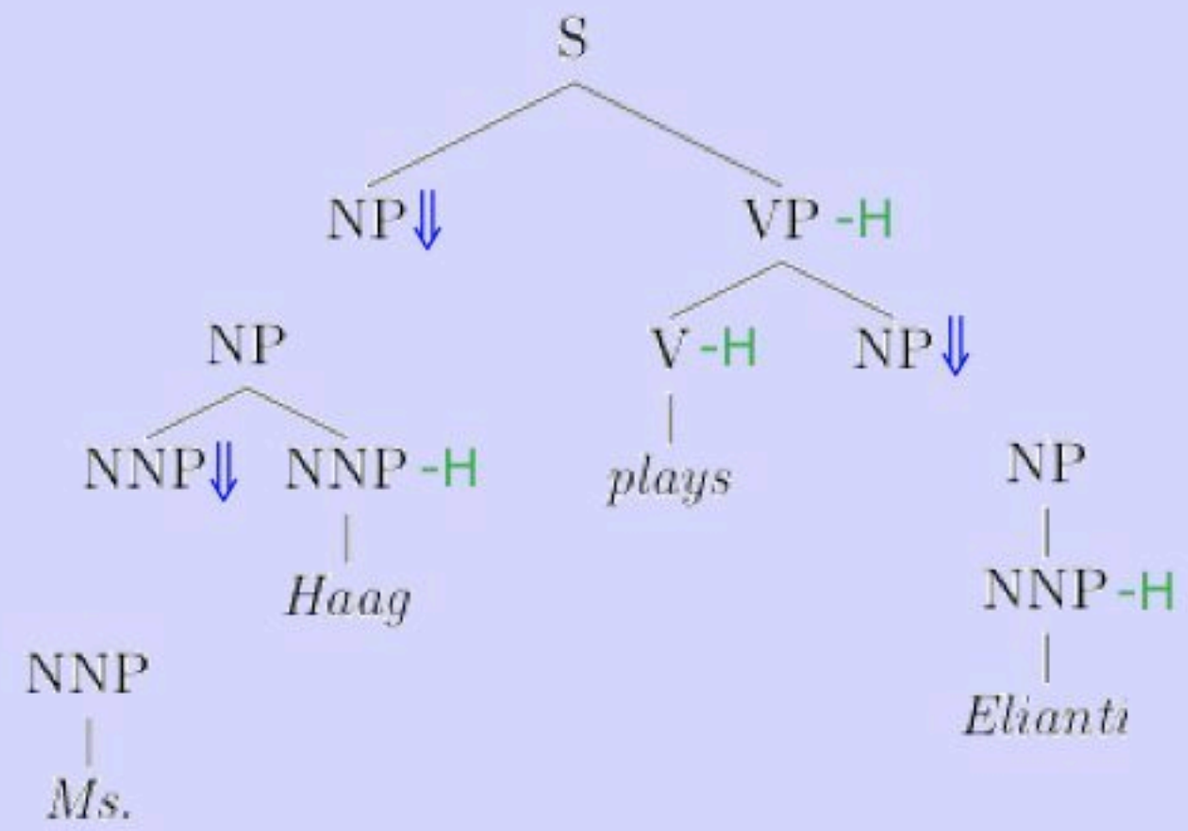
CFG | CM-LTSG | DOP



STSGs

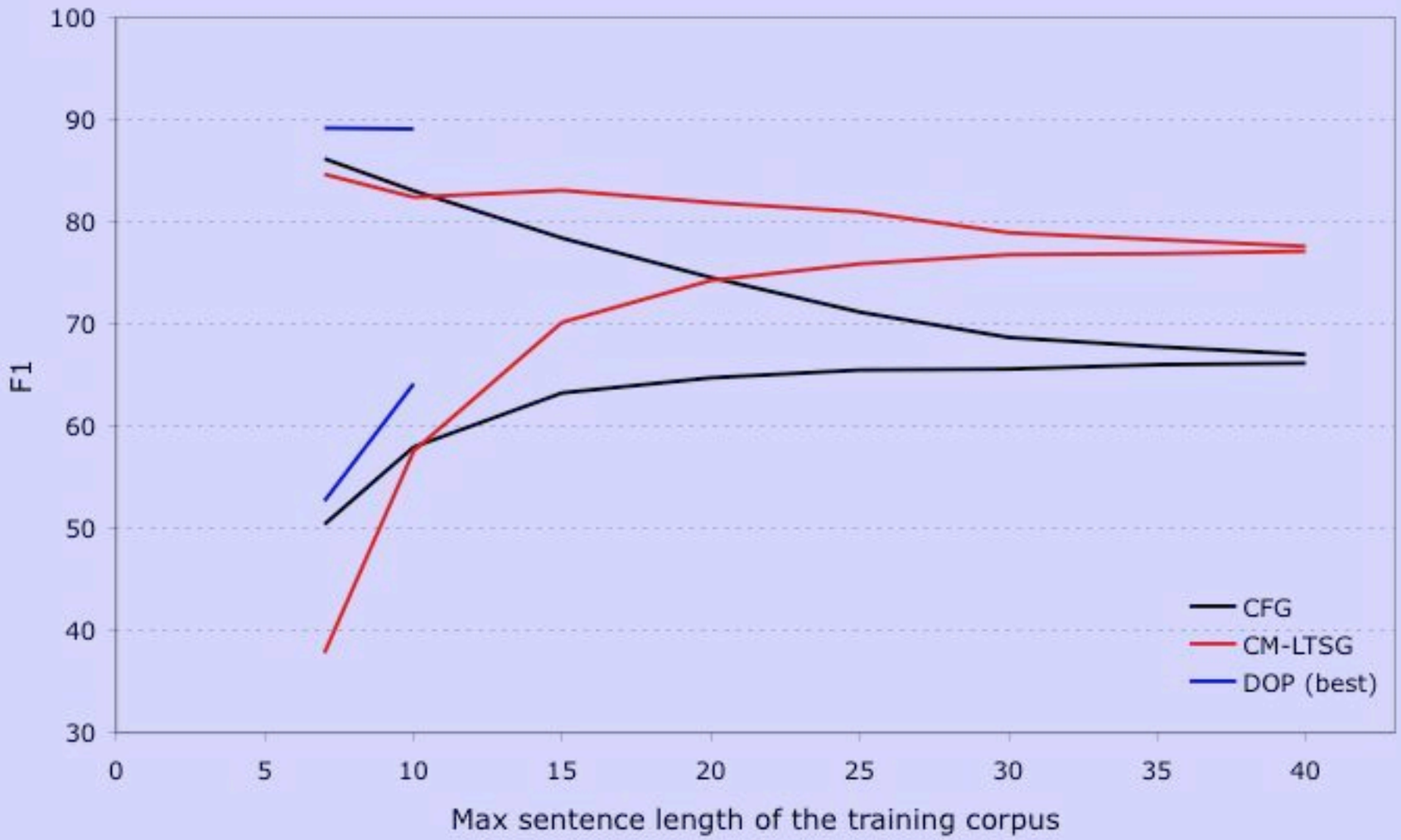
simple ⟷ complex

CFG | CM-LTSG | DOP

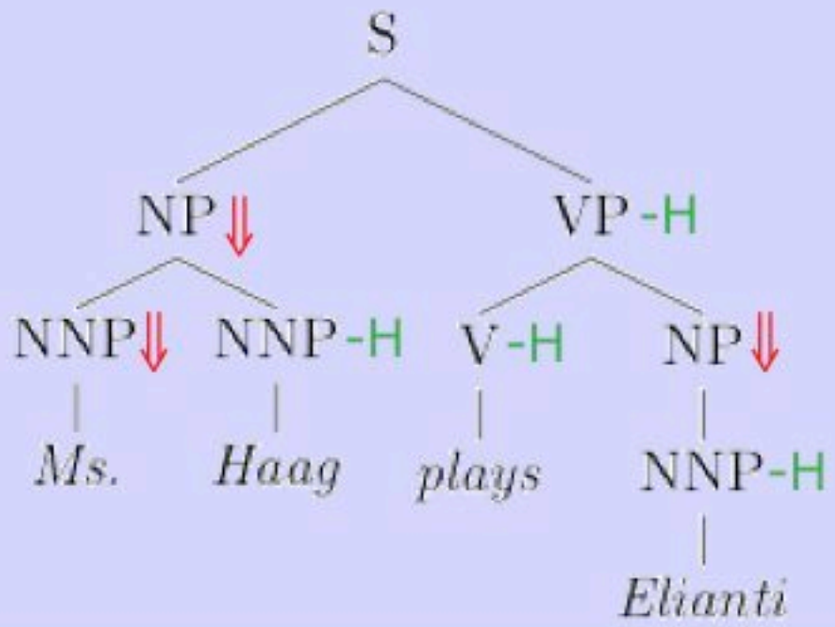


COMPARING DIFFERENT STSGs (Grammar Size)

COMPARING DIFFERENT STSGs (Performance)

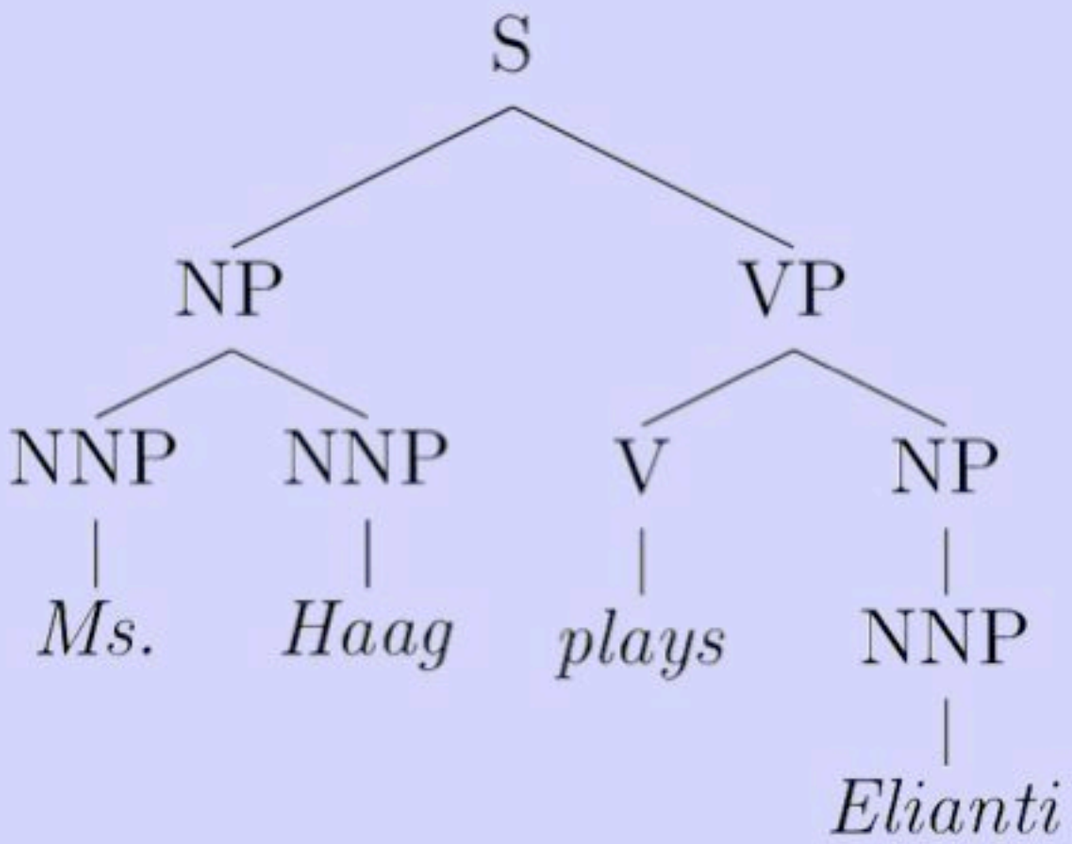


CM-LTSG

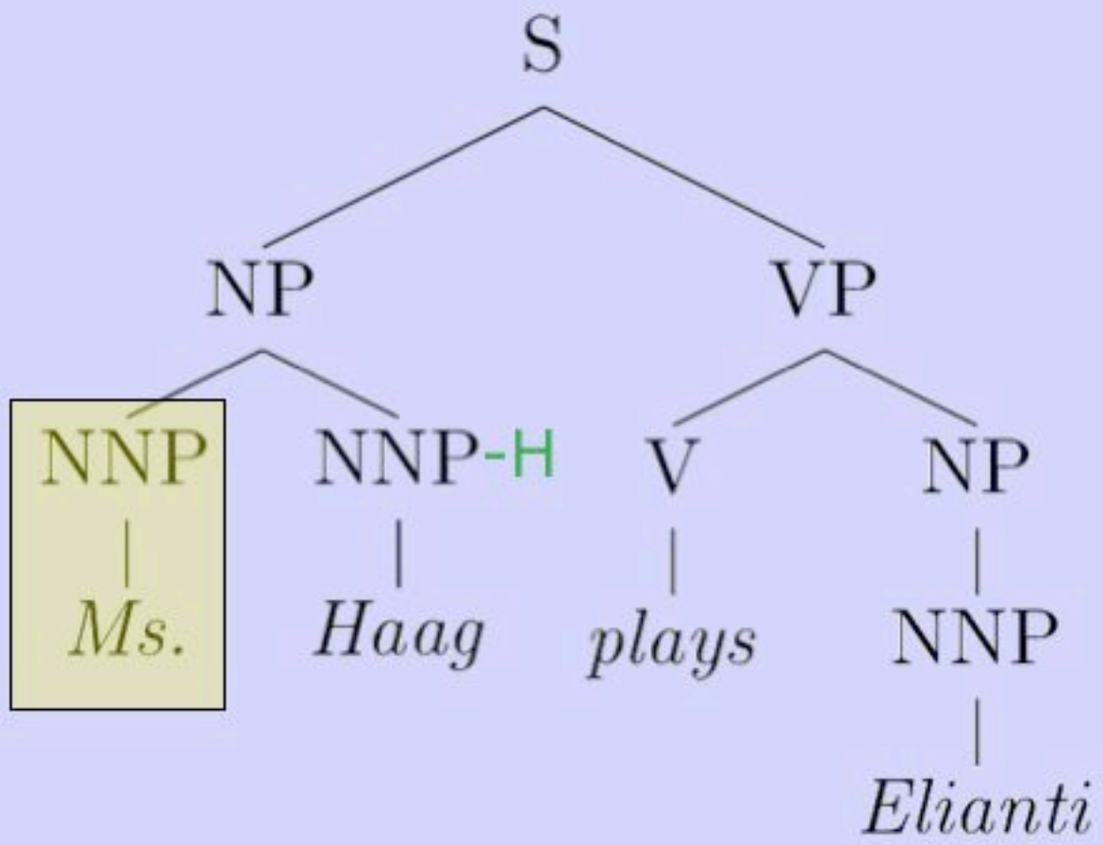


Parent Non-terminal	Direction	Priority List
ADJP	Left	NNS QP NN $ ADVP JJ VBN VBG ADJP JJR NP JJS DT FW RBR RBS SBAR RB
ADVP	Right	RB RBR RBS FW ADVP TO CD JJR JJ IN NP JJS NN
CONJP	Right	CC RB IN
FRAG	Right	
INTJ	Left	
LST	Right	LS :
NAC	Left	NN NNS NNP NNPS NP NAC EX $ CD QP PRP VBG JJ JJS JJR ADJP FW
PP	Right	IN TO VBG VBN RP FW
PRN	Left	
PRT	Right	RP
QP	Left	$ IN NNS NN JJ RB DT CD NCD QP JJR JJS
RRC	Right	VP NP ADVP ADJP PP
S	Left	TO IN VP S SBAR ADJP UCP NP

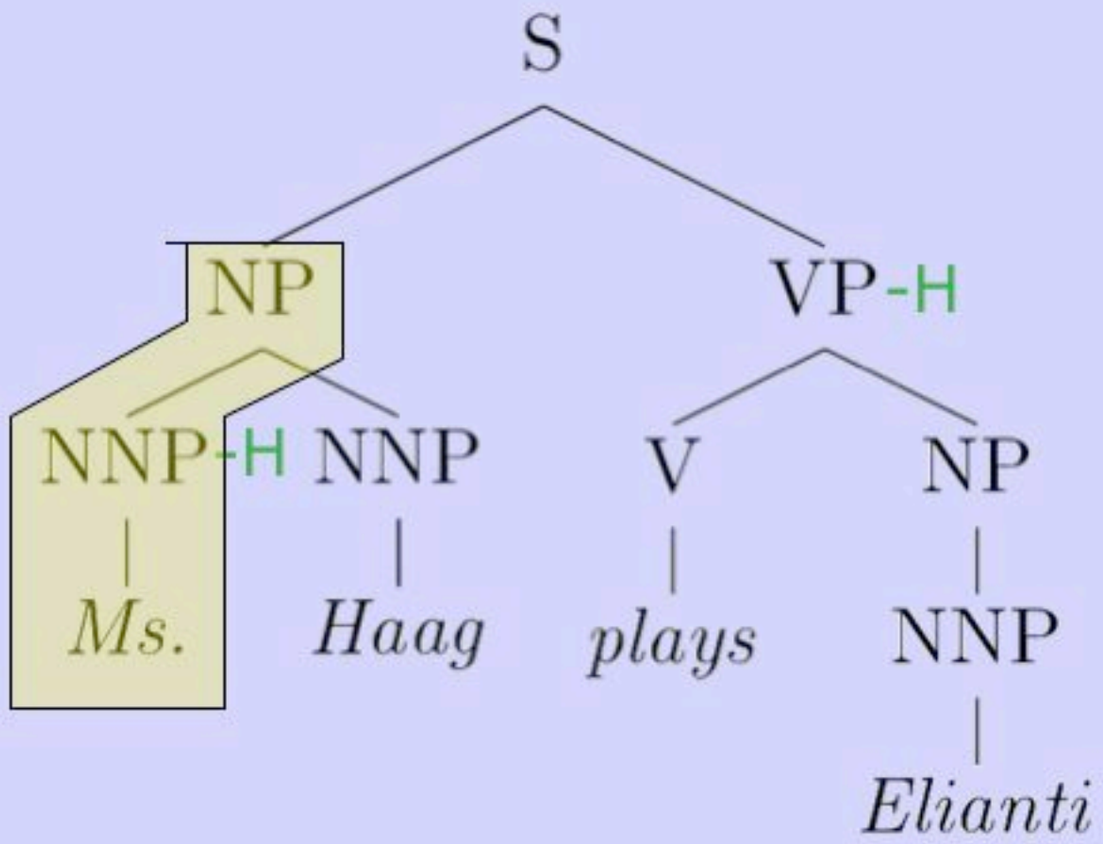
LTSG



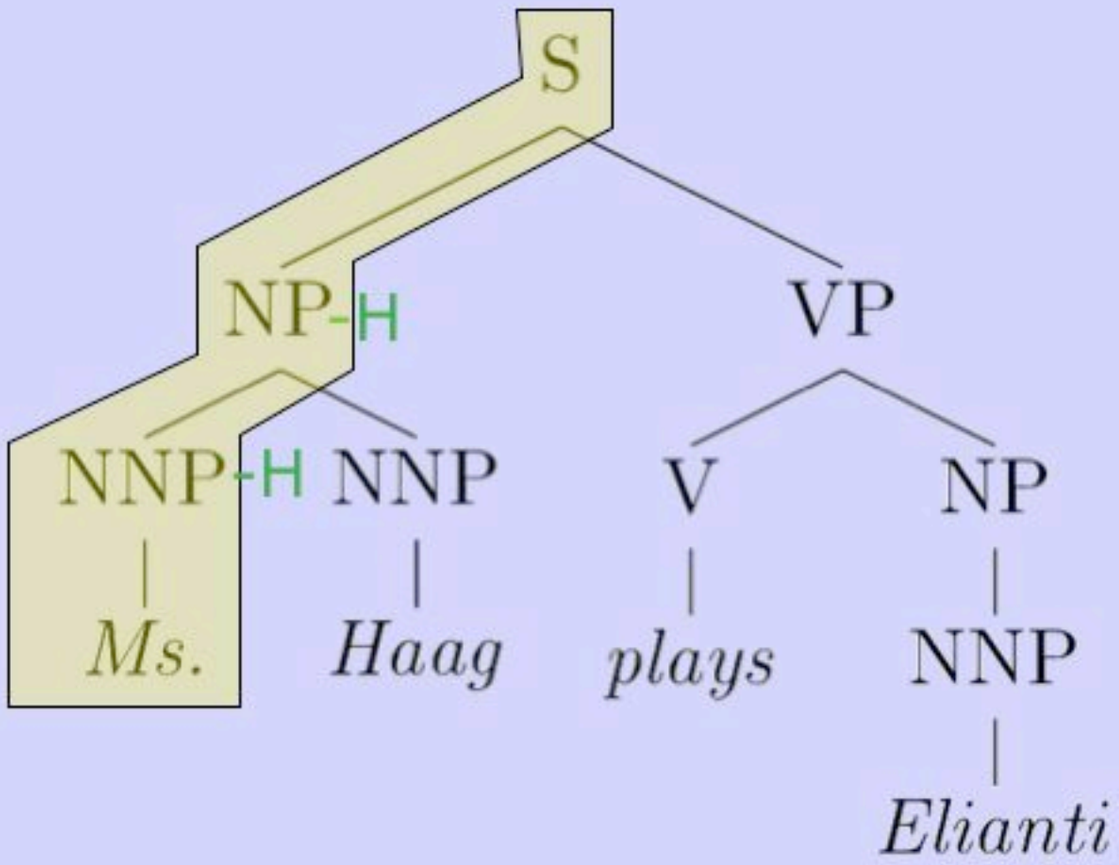
LTSG



LTSG



LTSG



AUTOMATICALLY INDUCING HEAD DEPENDENCY

CM-LTSG (76.63%)

Naive Strategies

- *First Left*
- *First Right*
- *Random*

Principle Based Strategies

- *MLE*
- *Greedy strategy*
- *Entropy based strategy*

AUTOMATICALLY INDUCING HEAD DEPENDENCY

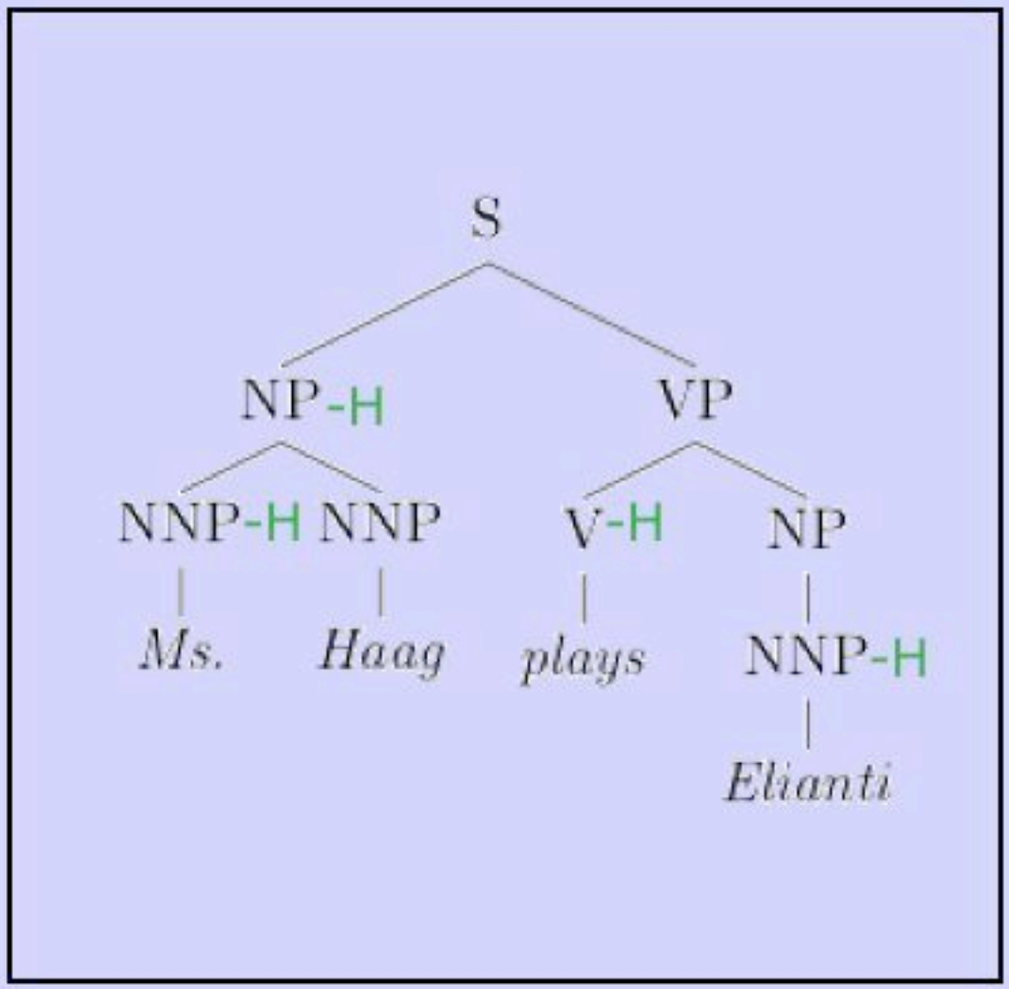
CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right*
- *Random*

Principle Based Strategies

- *MLE*
- *Greedy strategy*
- *Entropy based strategy*



AUTOMATICALLY INDUCING HEAD DEPENDENCY

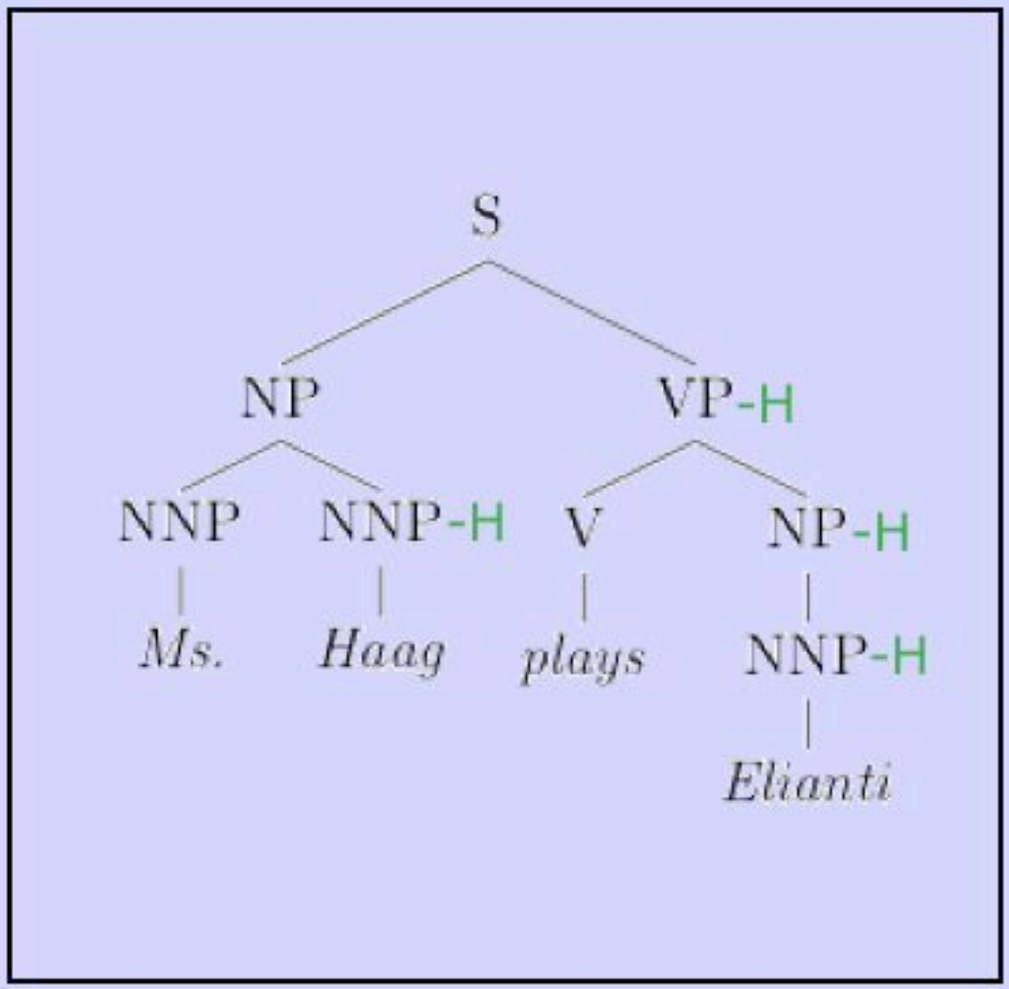
CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random*

Principle Based Strategies

- *MLE*
- *Greedy strategy*
- *Entropy based strategy*



AUTOMATICALLY INDUCING HEAD DEPENDENCY

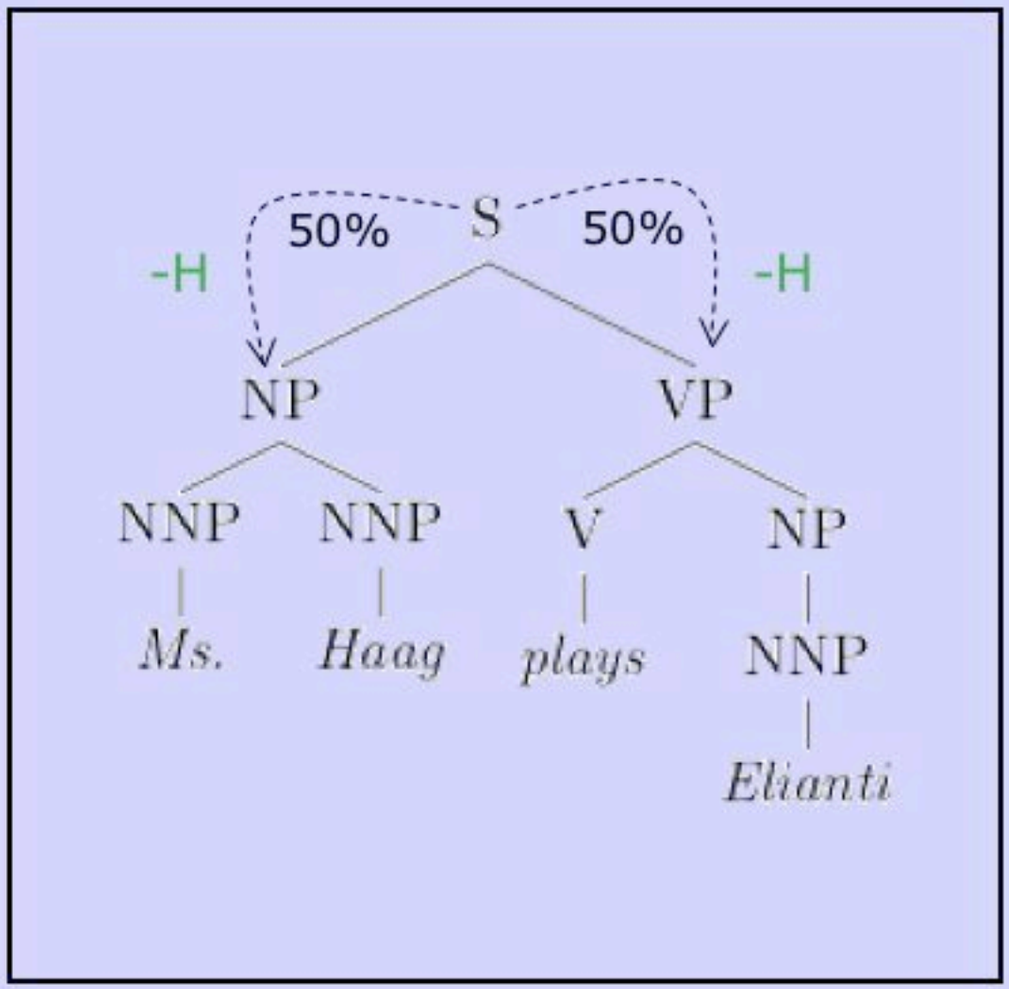
CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE*
- *Greedy strategy*
- *Entropy based strategy*



AUTOMATICALLY INDUCING HEAD DEPENDENCY

CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy*
- *Entropy based strategy*

- Extracts all possible e-trees with one lexical item
- Finds the probabilistic distributions over the fragments which maximizes the likelihood of the observed data
- Implemented using EM

AUTOMATICALLY INDUCING HEAD DEPENDENCY

CM-LTSG (76.63%)

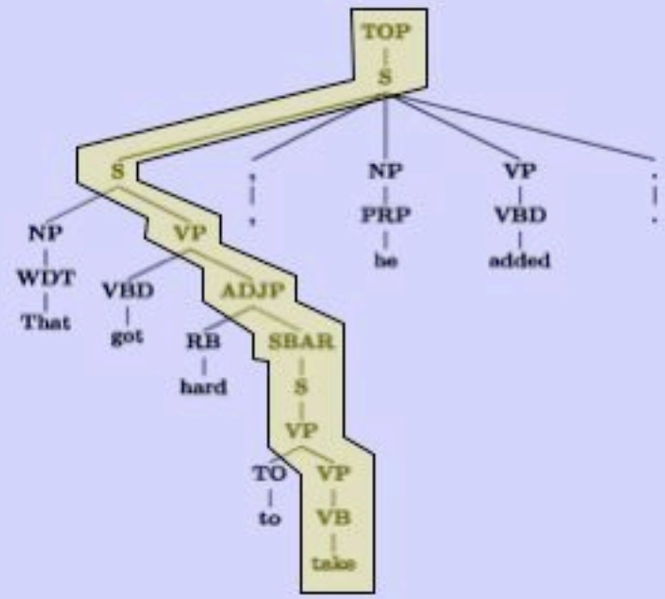
Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy*
- *Entropy based strategy*

Overfitting problem !



AUTOMATICALLY INDUCING HEAD DEPENDENCY

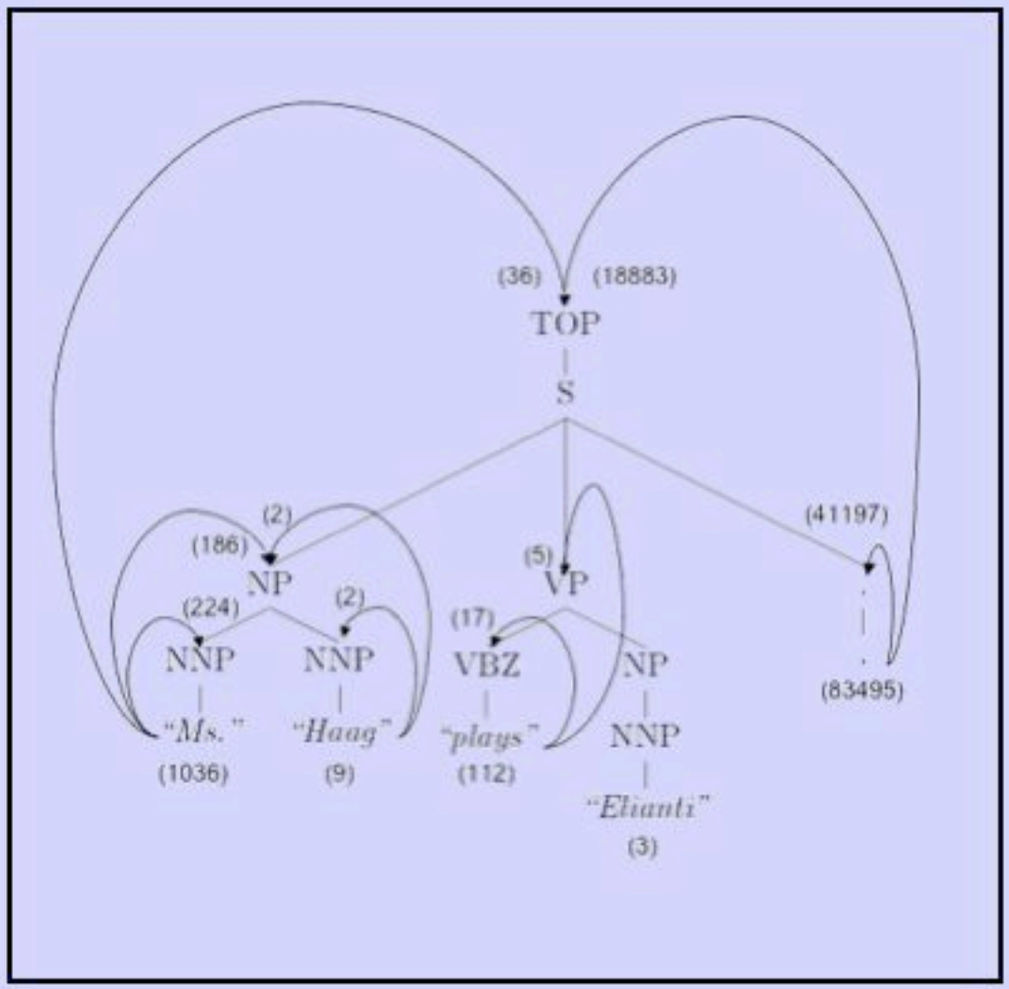
CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy* (80.23%)
- *Entropy based strategy*



AUTOMATICALLY INDUCING HEAD DEPENDENCY

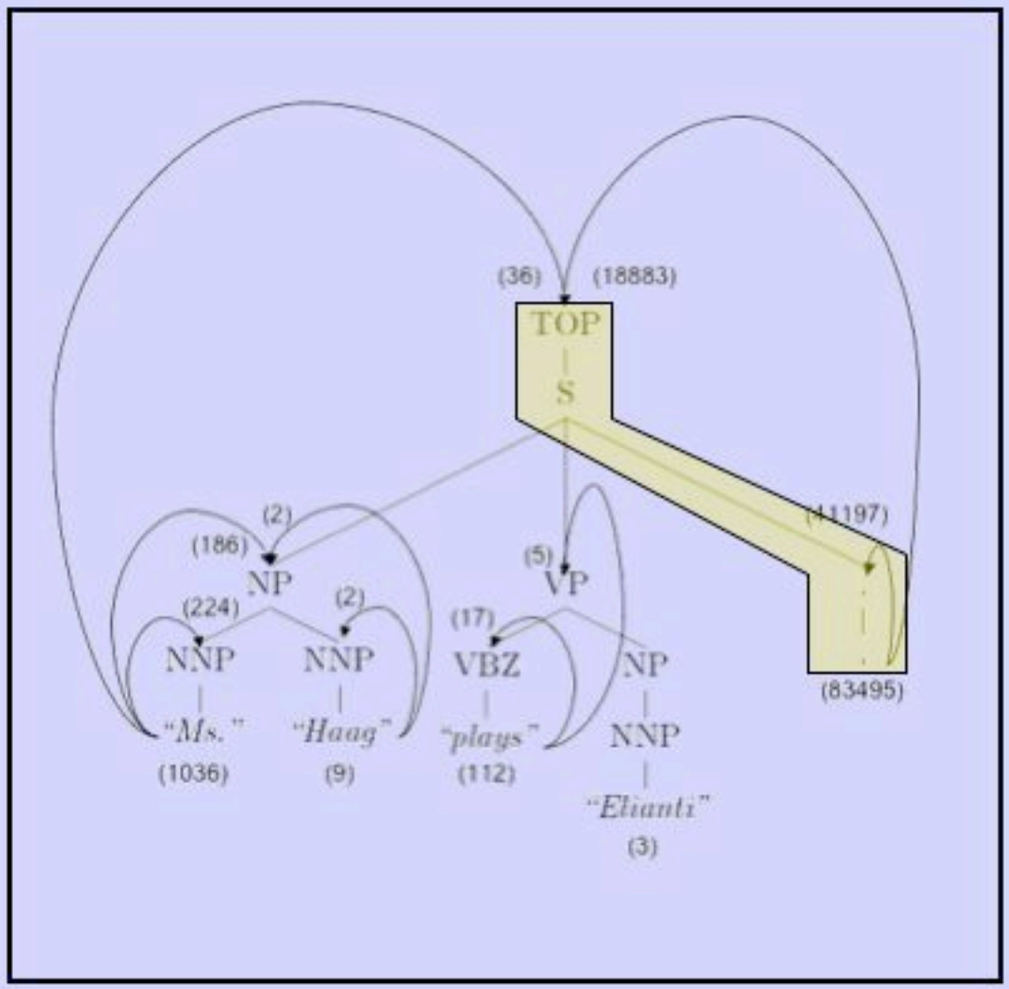
CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy* (80.23%)
- *Entropy based strategy*



AUTOMATICALLY INDUCING HEAD DEPENDENCY

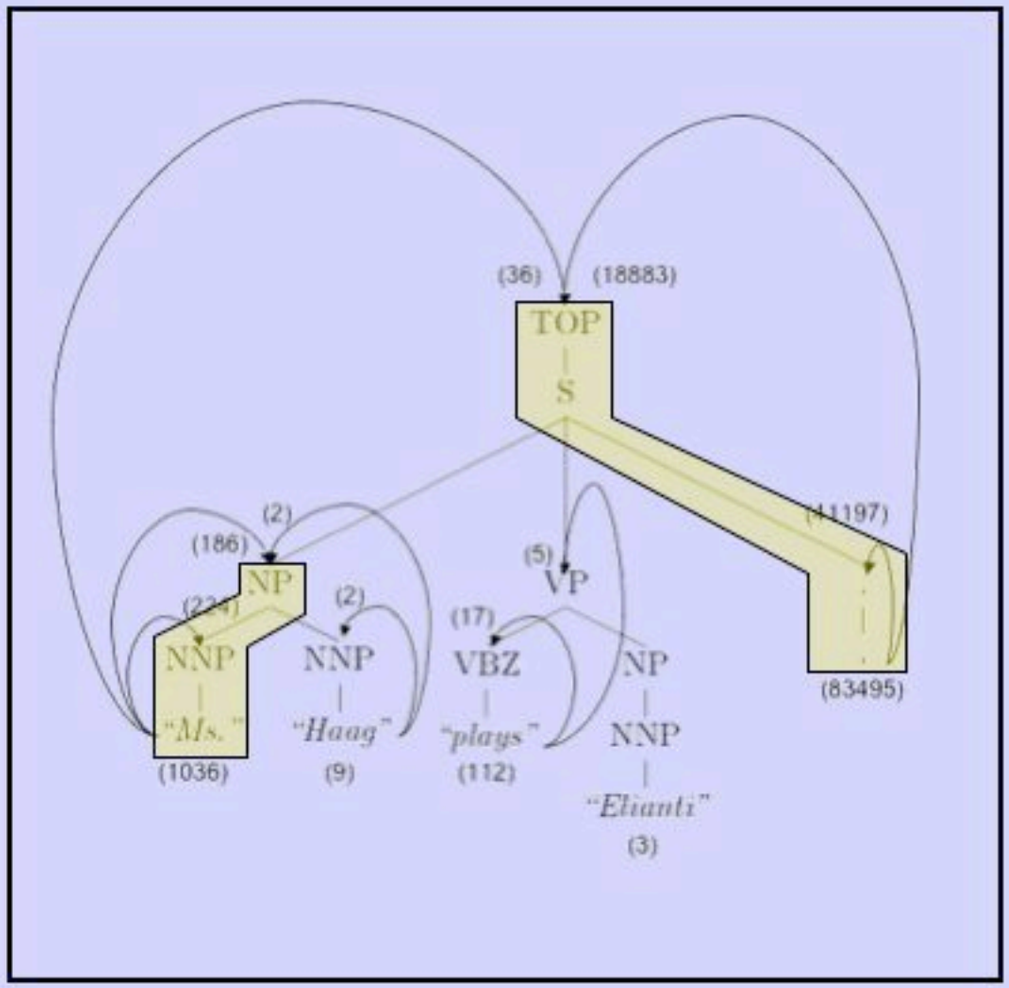
CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy* (80.23%)
- *Entropy based strategy*



AUTOMATICALLY INDUCING HEAD DEPENDENCY

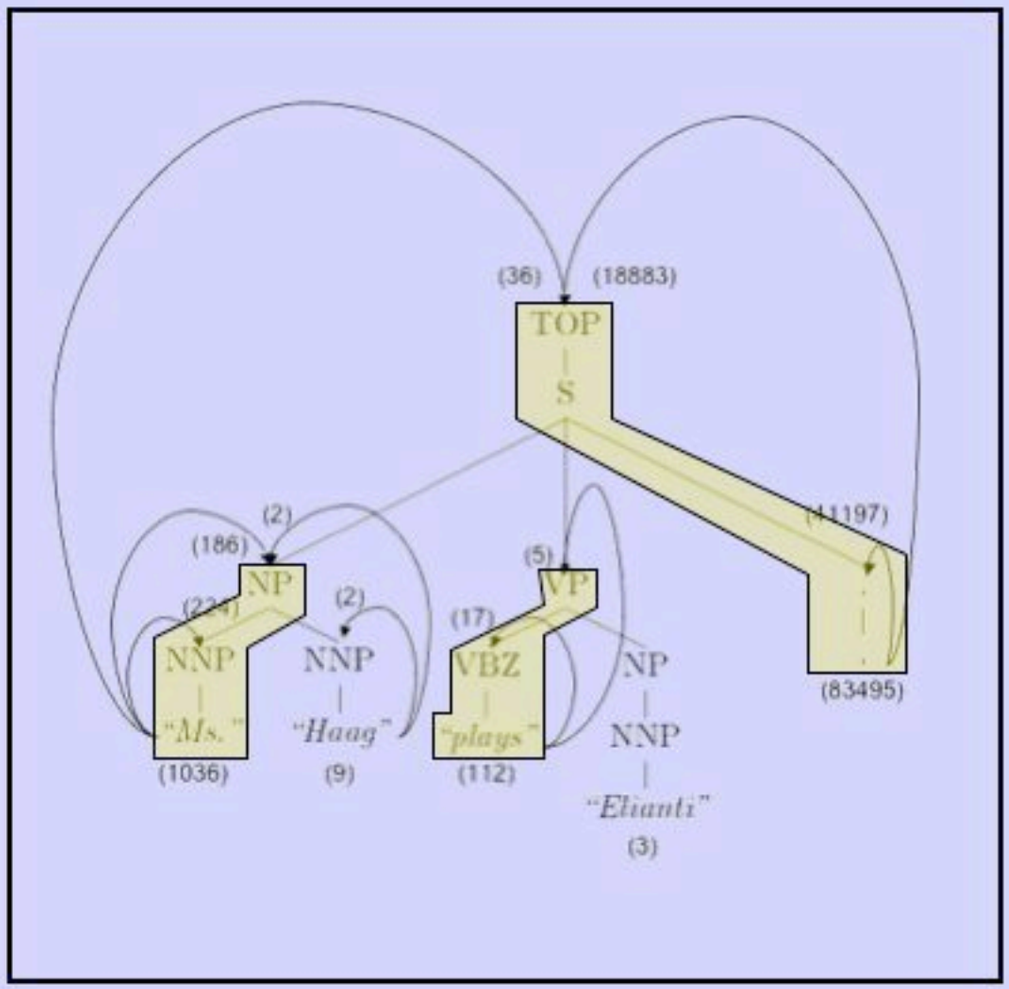
CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy* (80.23%)
- *Entropy based strategy*



AUTOMATICALLY INDUCING HEAD DEPENDENCY

CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy* (80.23%)
- *Entropy based strategy* (80.40%)

Hill climbing strategy

- starts from a given head annotation (i.e. random)
- aims at reducing the uncertainty of the structures which can be associated to each lexical item

AUTOMATICALLY INDUCING HEAD DEPENDENCY

CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy* (80.23%)
- *Entropy based strategy* (80.40%)

```
1    (S NP (VP (VBZ "plays") ADVP))
1    (S NP (VP (VBZ "plays") NP ADVP))
1    (VP (VBZ "plays") NP PP)
1    (S NP (VP (VBZ "plays") ADVP PP))
10   (VBZ "plays")
1    (S NP (VP (VBZ "plays")))
2    (VP (VBZ "plays") NP)
```

```
17   (VBZ "plays")
```

AUTOMATICALLY INDUCING HEAD DEPENDENCY

CM-LTSG (76.63%)

Naive Strategies

- *First Left* (76.57%)
- *First Right* (71.30%)
- *Random* (79.84%)

Principle Based Strategies

- *MLE* (69.11%)
- *Greedy strategy* (80.23%)
- *Entropy based strategy* (80.40%)

Entropy (Information Theory)

discrete stochastic variable τ_l

taking n possible values $\tau_{l_1}, \tau_{l_2}, \ldots, \tau_{l_n}$

with probabilities $p(\tau_{l_1}), p(\tau_{l_2}), \ldots, p(\tau_{l_n})$

$$H(\tau_l) = -\sum_{i=1}^{n} p(\tau_{l_i}) \log_2 p(\tau_{l_i})$$

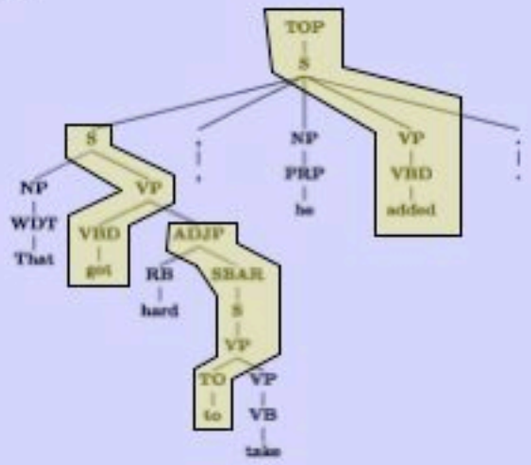
$$H(T) = \sum_{l=1}^{|\mathscr{L}|} H(\tau_l)$$

RESULTS

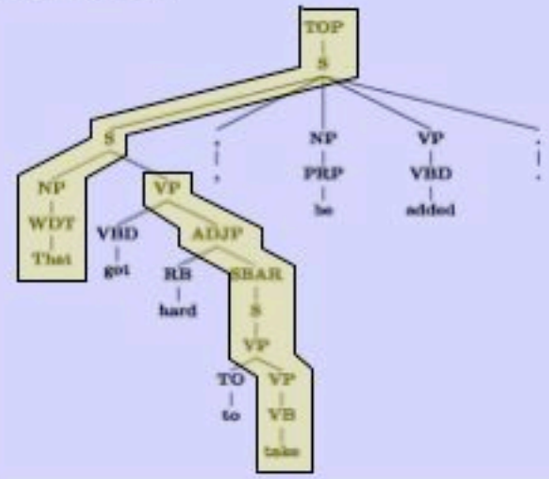
	LR	LP	F_1
Entropy (Rand.)	80.72	80.08	80.40
Greedy (Bot.)	80.42	80.05	80.23
Random	80.21	79.47	79.84
Entropy (CM)	79.98	79.46	79.72
Entropy (Left)	80.01	79.40	79.70
CM	76.46	76.80	76.63
Left	76.65	76.48	76.57
Right	69.58	73.10	71.30

CONCLUSIONS

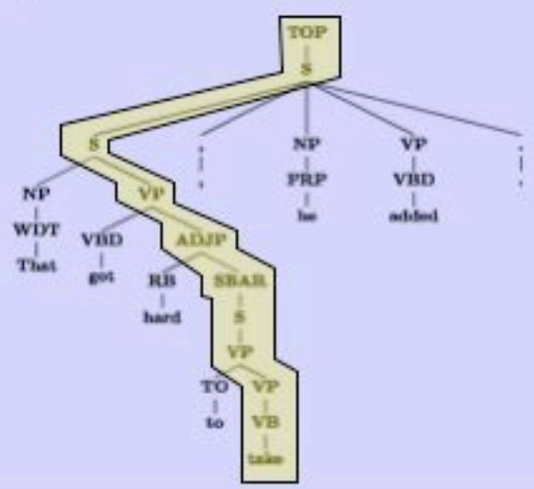
CM



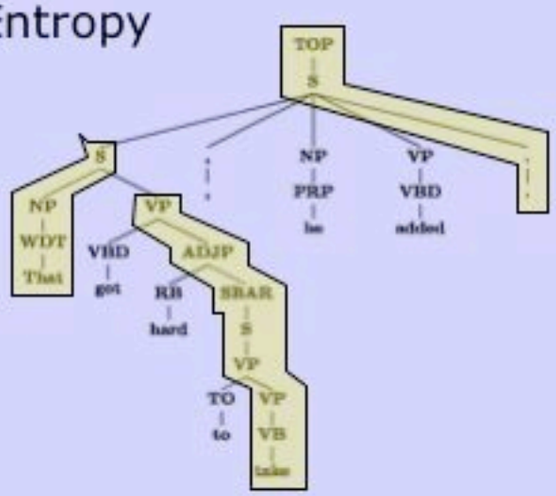
Random



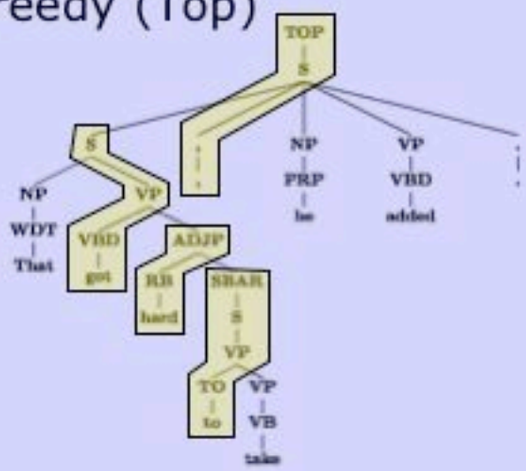
EM



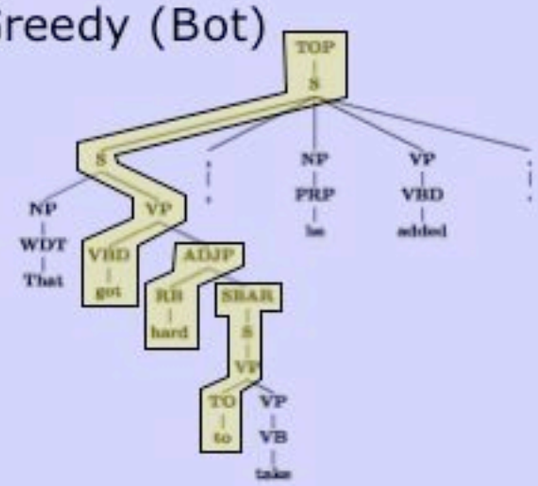
Entropy



Greedy (Top)



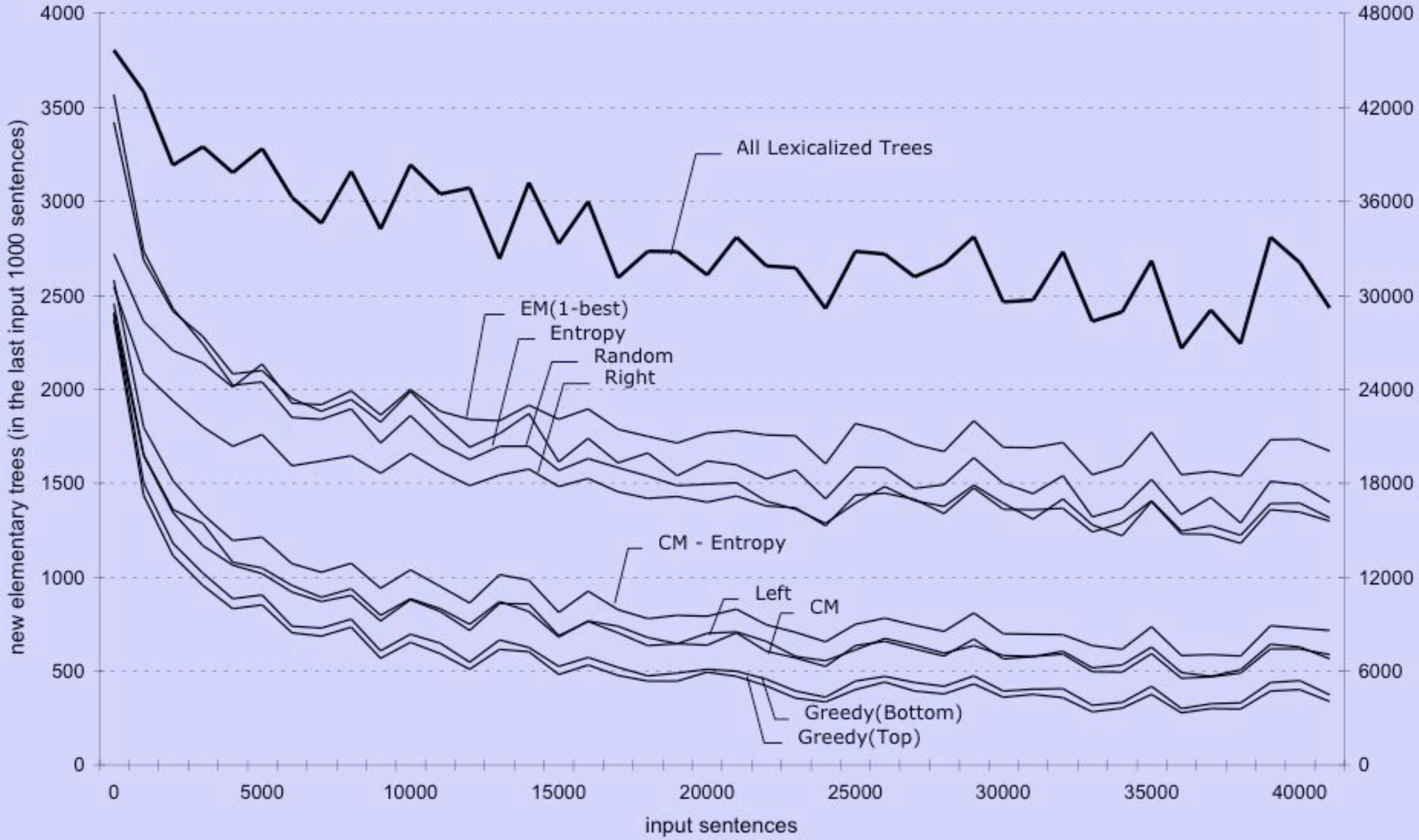
Greedy (Bot)



CONCLUSIONS

- *Results are promising but not state of the art*
- *Many possible LTSG - not all were evaluated*
- *Possible to come up with better head annotations*
- *Make the extracted fragment more general*

FURTHER ANALYSIS



FURTHER ANALYSIS

So what 's the best way to buy stocks ?

WHICH IS the best medicine for runaway health costs : competition or regulation ?

Who cares ?