A Statistical Parsing Framework for Sentiment Classification

ProbModels@ILCC

Li Dong

Sentence-Level Sentiment Classification

- Input: sentence
- Output: polarity label (e.g., positive / negative)
- One of the most challenging problem is:
  - Sentiment composition

(1) The movie is not *good*. [negation]

(2) The movie is very *good*. [intensification]

(3) The movie is not *funny* at all. [negation + intensification]

(4) The movie is *just so so*, but i still *like* it. [contrast]

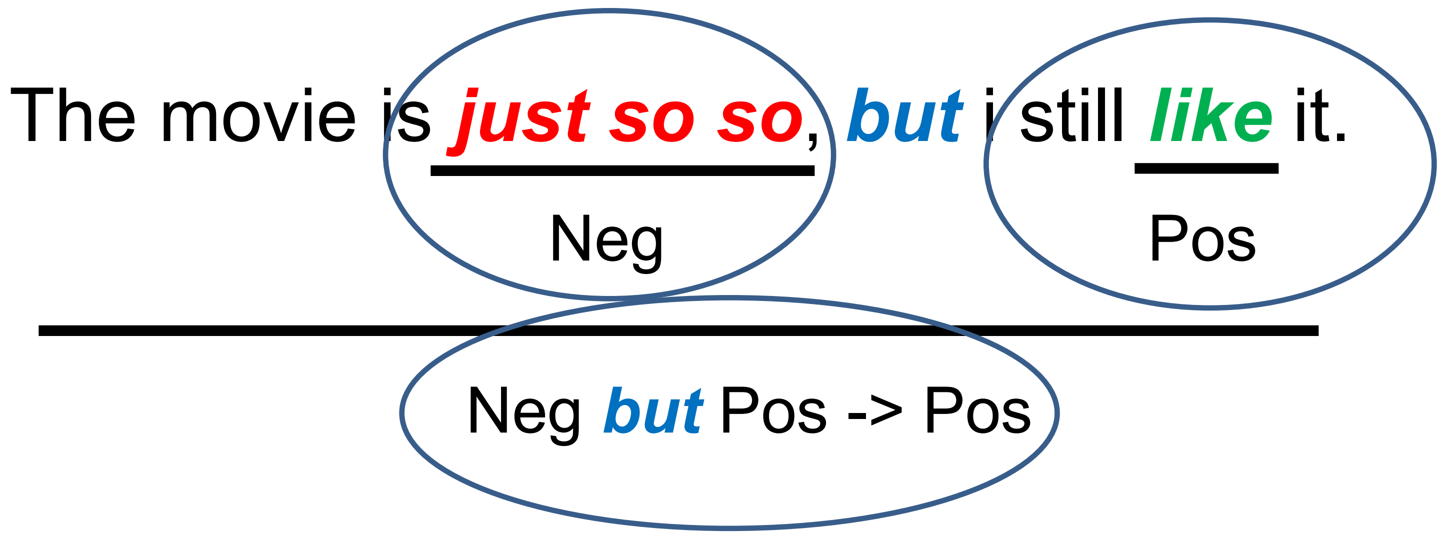
(5) The movie is not very *good*, but i still *like* it. [negation + intensification + contrast]

Two Mainstream Methods

- Lexicon-based
  - Lexicons (funny, dislike) + Rules (not *, * but *)
  - Pros: simple, interpretable
  - Cons: scalability
- Classifier-based
  - Classifier (SVM, MaxEnt) + Features (n-gram, POS)
  - Pros: data-driven, coverage
  - Cons: tricks to handle sentiment compositions

Revisit Sentiment Composition

- Two key components
  - Lexicon
  - Rule
- *Q1: Can we learn them from data?*



Revisit Sentiment Composition

- Sentiment composition is not only about polarity
  - P(Pos|Very good) > P(Pos|Good)
- *Q2: Can we model polarity strength?*

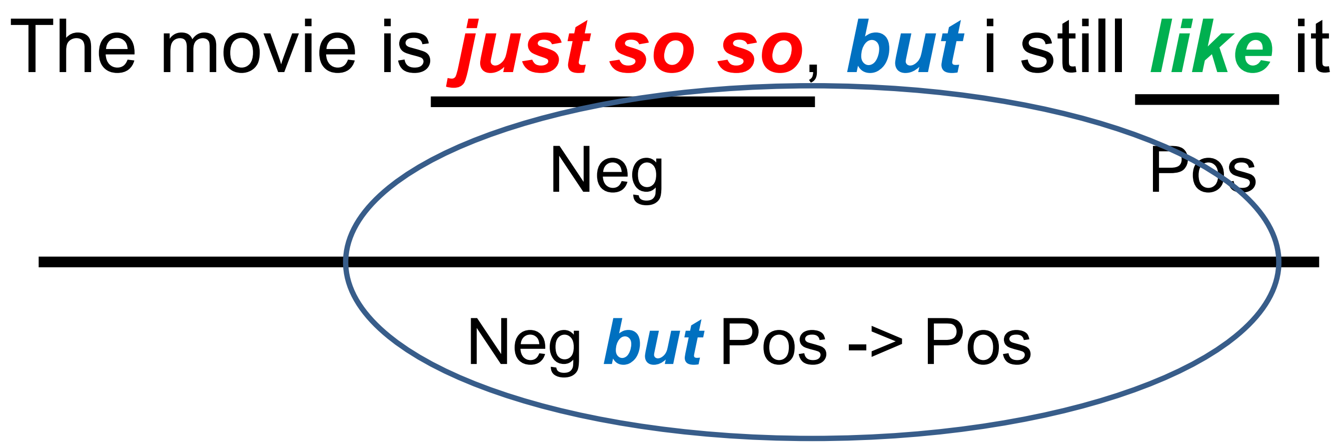
The movie is ***just so so***, ***but*** i still ***like*** it.

Neg Pos

Neg ***but*** Pos -> Pos

Revisit Sentiment Composition

- Latent sentiment structure
- *Q3: How do we define the sentiment structure?*



Revisit Sentiment Composition

- Latent sentiment structure
- *Q4: Can we only use (sentence, polarity) pairs to learn latent sentiment structure?*

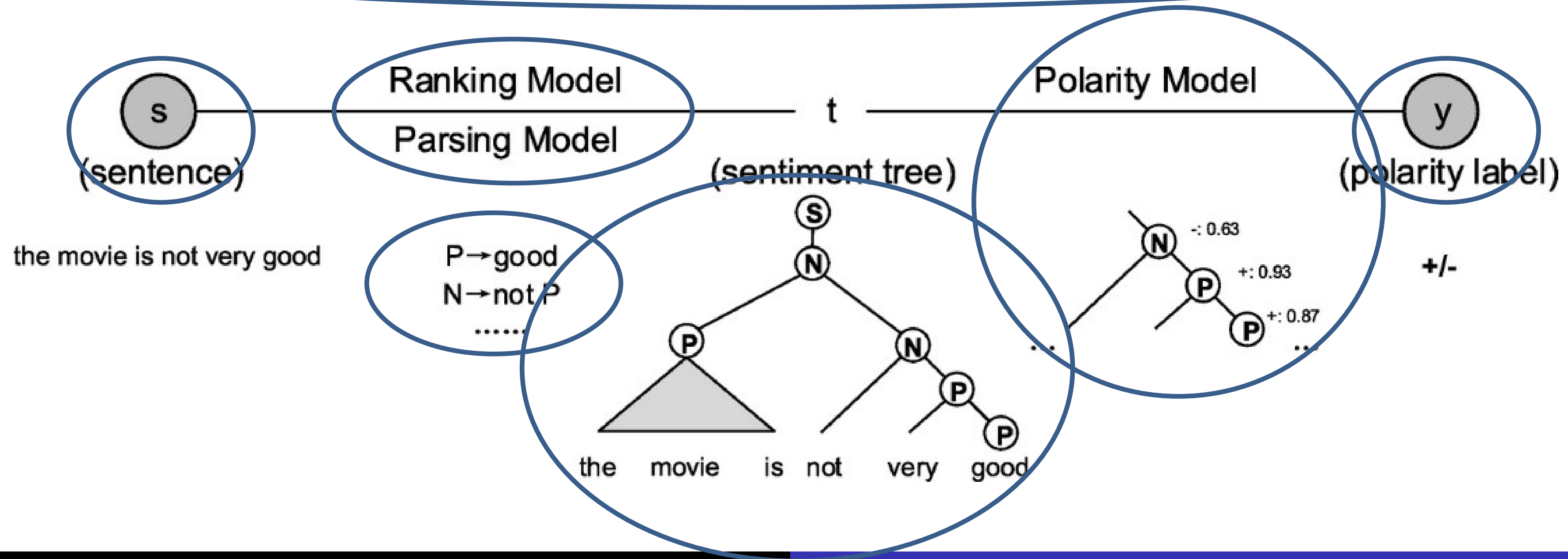
The movie is ***just so so***, ***but*** i still ***like*** it.

Neg Pos

Neg ***but*** Pos -> Pos

Overview

- Q1: learn lexicons and rules?
- Q2: calculate polarity strength?
- Q3: define sentiment structure?
- Q4: learn structure from (sentiment, polarity)?



Comparison with Semantic Parsing

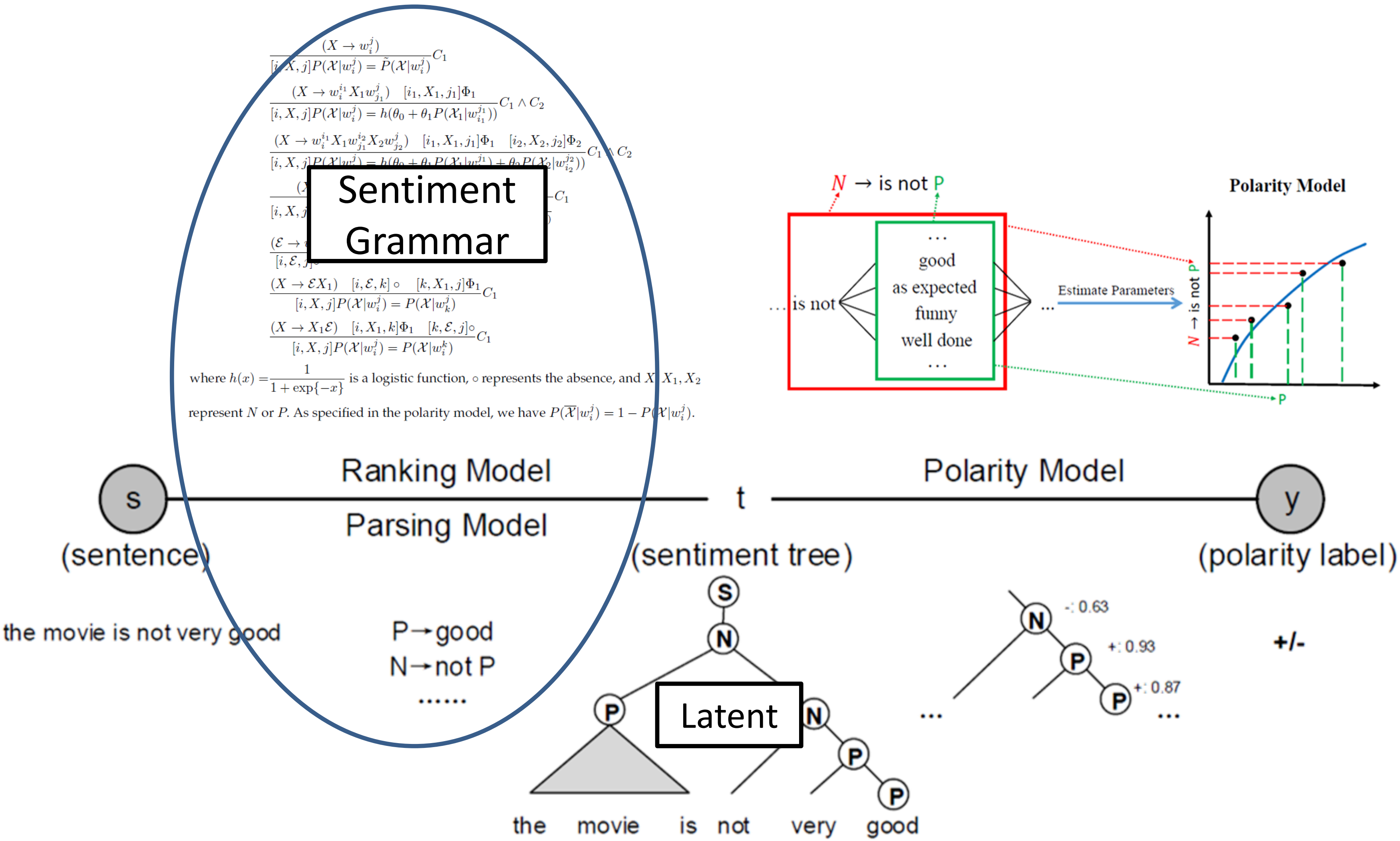
Sentiment parsing

$$p(label|sentence, model) = p(label|tree, polarity\ model) \quad \text{deterministic}$$
$$\mathrm{p}(tree|sentence, parsing\ model) \quad \text{probabilistic}$$

Semantic parsing

$$p(answer|question, model) = p(answer|representation, database)$$
$$p(representation|question, parsing\ model)$$

Sentiment parsing	Semantic parsing
sentiment lexicons	lexical triggers
(latent) sentiment tree	(latent) meaning representation
(sentence, polarity) pairs	(question, answer) pairs
calculate polarity strength	execute query



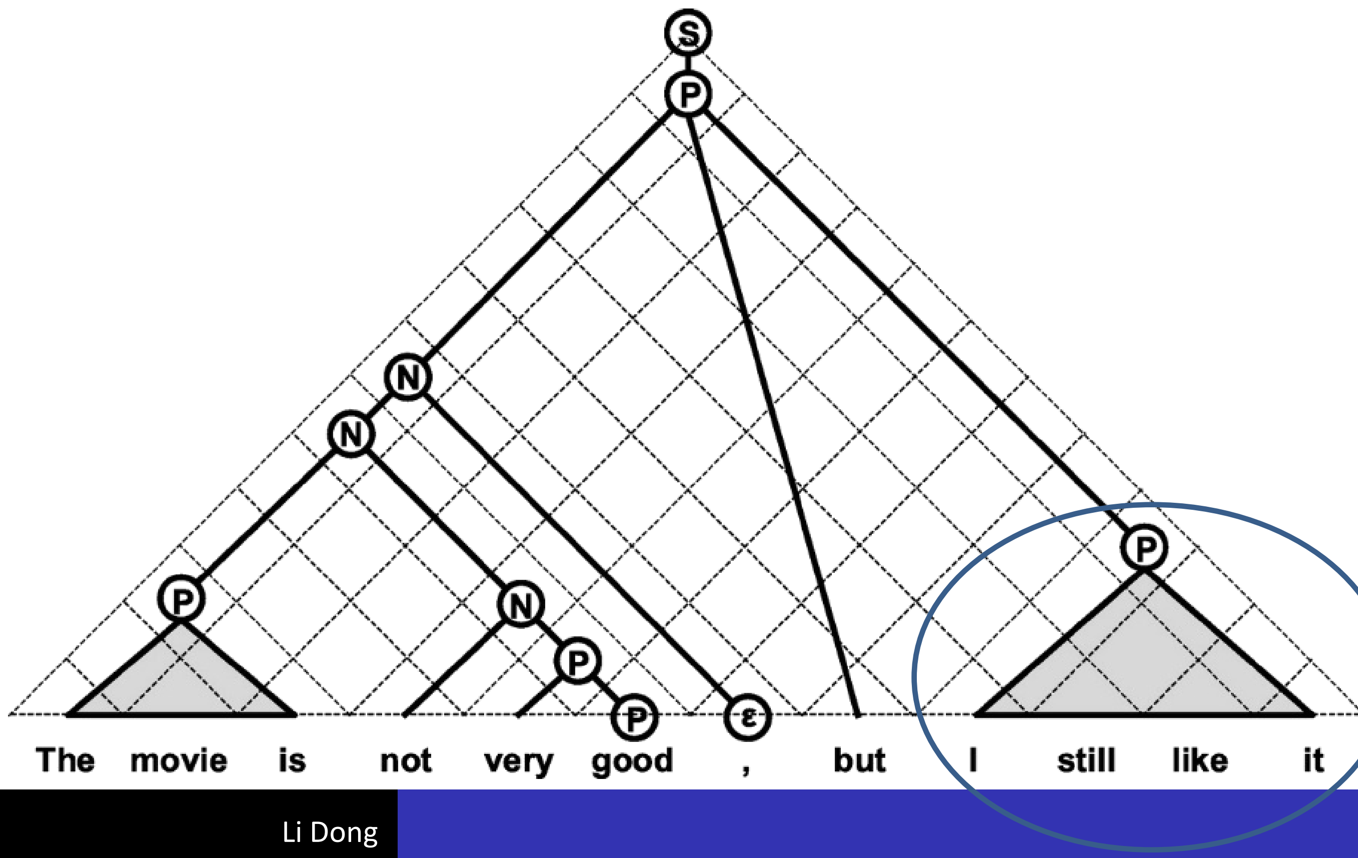
Sentiment Grammar
where $h(x) = \frac{1}{1+\exp\{-x\}}$ is a logistic function, ∘ represents the absence, and X, X_1, X_2 represent N or P. As specified in the polarity model, we have $P(\overline{\mathcal{X}}|w_i^j) = 1 - P(\mathcal{X}|w_i^j)$.
N → is not P
… is not
…
good
as expected
funny
well done
…
Estimate Parameters
Polarity Model
N → is not P
P
Ranking Model
Parsing Model
s
(sentence)
t
(sentiment tree)
Polarity Model
y
(polarity label)
the movie is not very good
P→good
N→not P
……
S
N
P
N
P
P
Latent
the movie is not very good
N -: 0.63
P +: 0.93
P +: 0.87
…
+/-

Sentiment Grammar

- Built upon Context-Free Grammar
- $G_s = < V_s, \Sigma_s, S, R_s >$
  - V_s={N,P,S,E}: non-terminal set
  - Σ_s : terminal set
  - S: start symbol
  - R_s : rewrite rule set

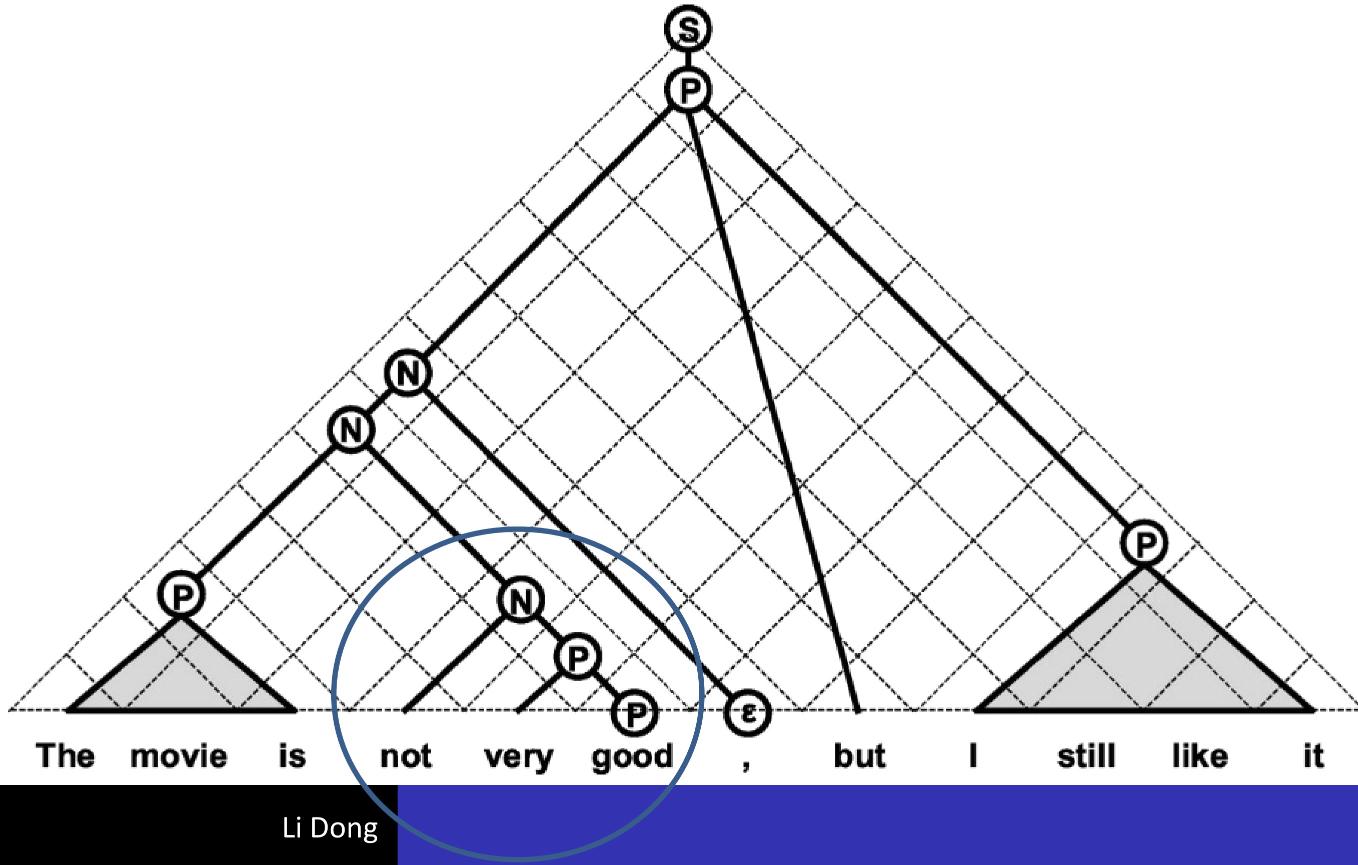
Sentiment Grammar

- Dictionary rules $X \rightarrow w_0^k$, where $X \in \{N, P\}, w_0^k = w_0 \dots w_{k-1},$
  - P -> I still like it
  - P -> good



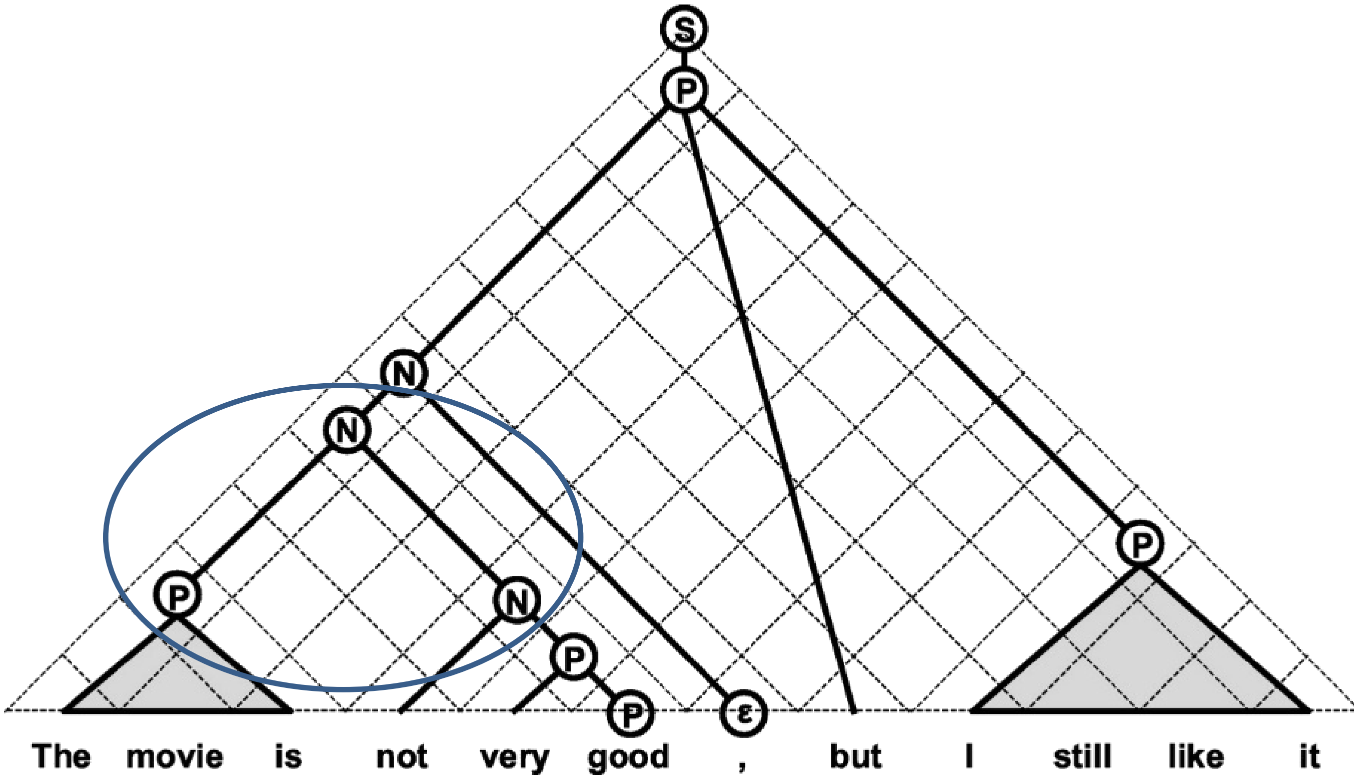
Sentiment Grammar

- Combination rules $X \rightarrow c, \text{where } c \in (V_s \cup \Sigma_s)^+,$
  - P->N but P
  - P->not P
  - P-> very P



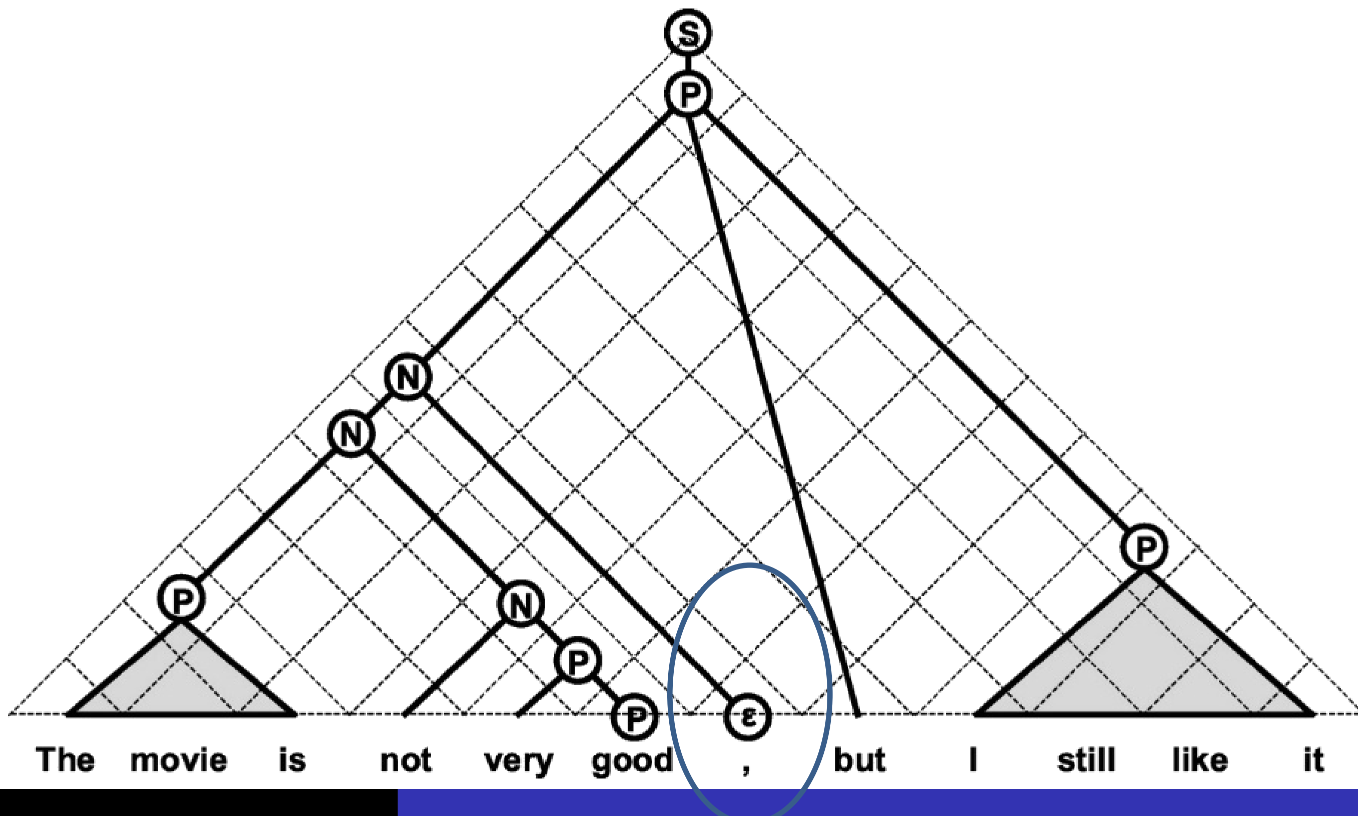
Sentiment Grammar

- Glue rules $X \rightarrow X_1 X_2$, where $X, X_1, X_2 \in \{N, P\}$
  - P->NP
  - N->NN



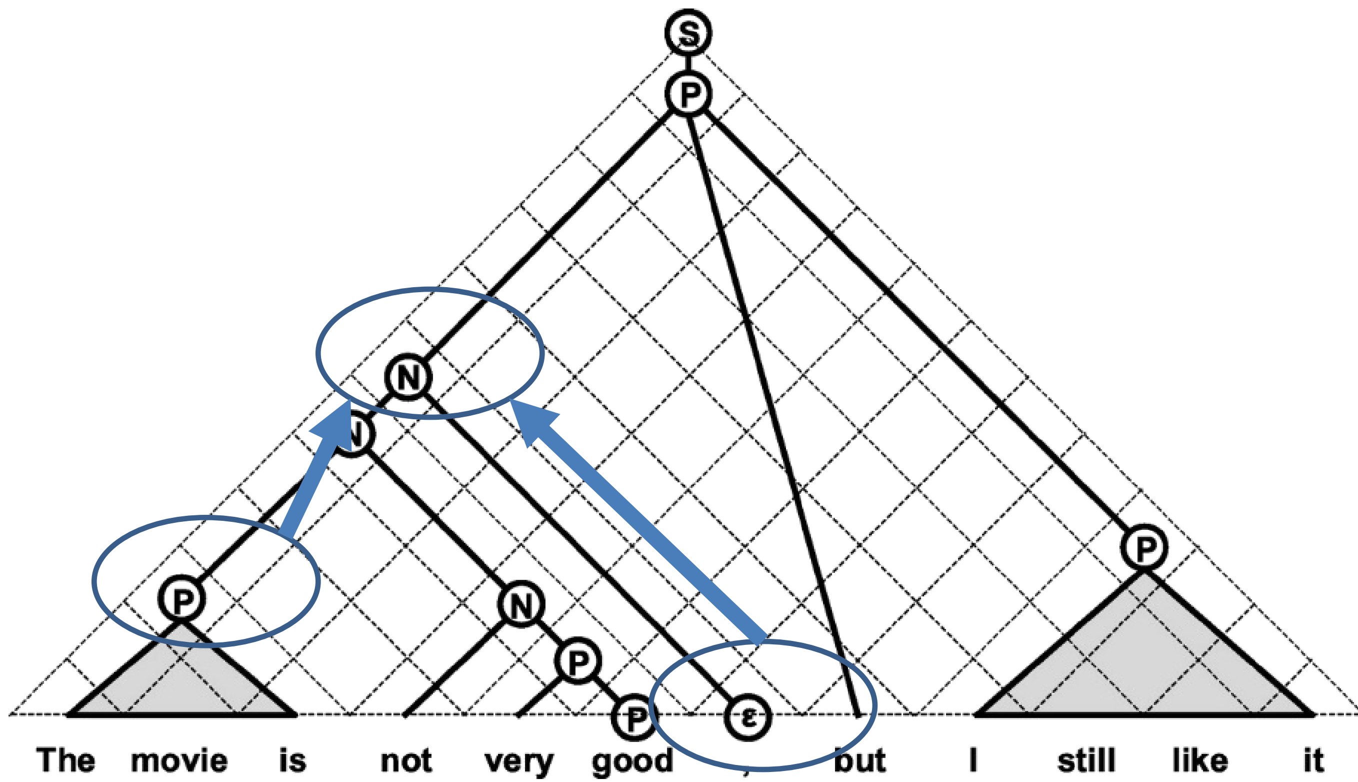
Sentiment Grammar

- OOV rules $\mathcal{E} \rightarrow w_0^k, \text{ where } w_0^k \in \Sigma^+$
  - Out-Of-Vocabulary text spans



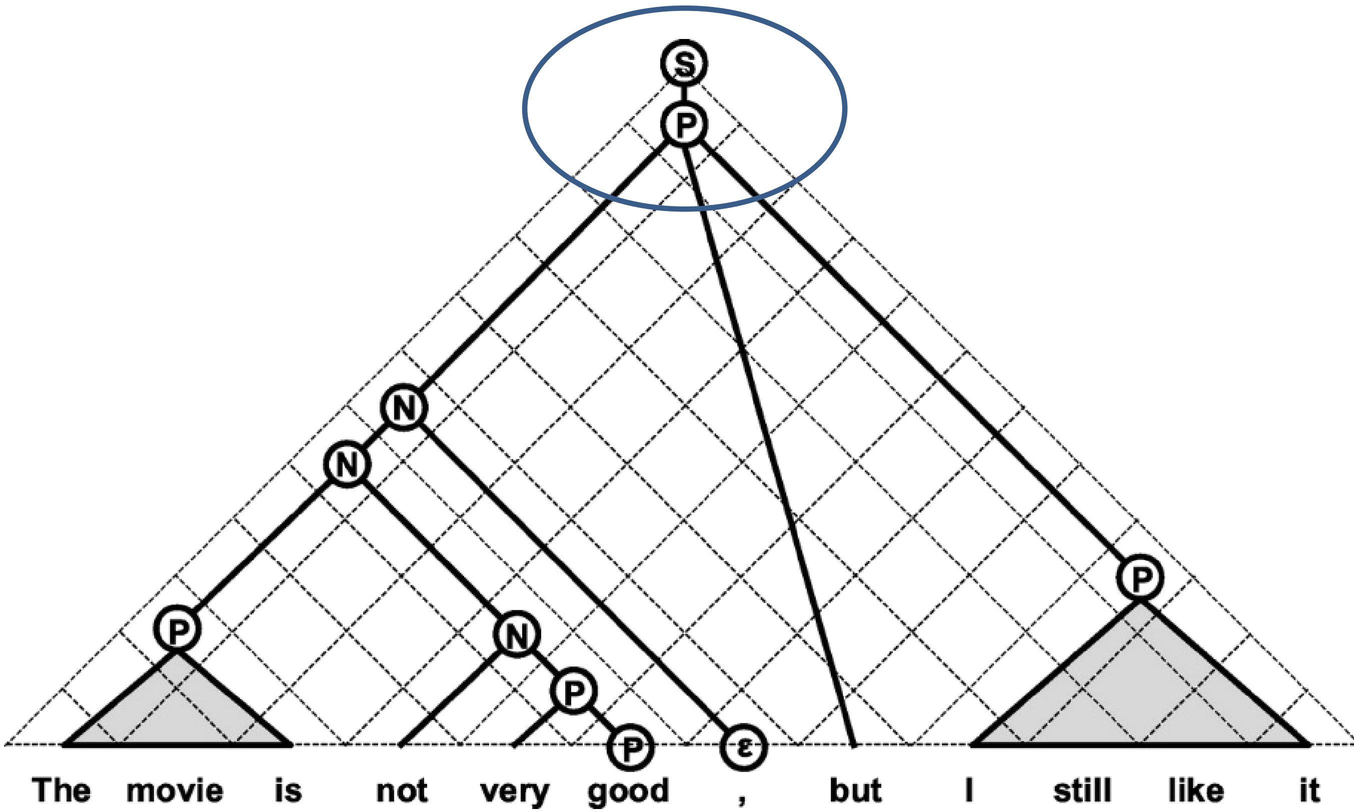
Sentiment Grammar

- Auxiliary rules $X \rightarrow \mathcal{E}X_1, X \rightarrow X_1\mathcal{E}$, where $X, X_1 \in \{N, P\}$
  - Combine out-of-vocabulary span and non-terminal



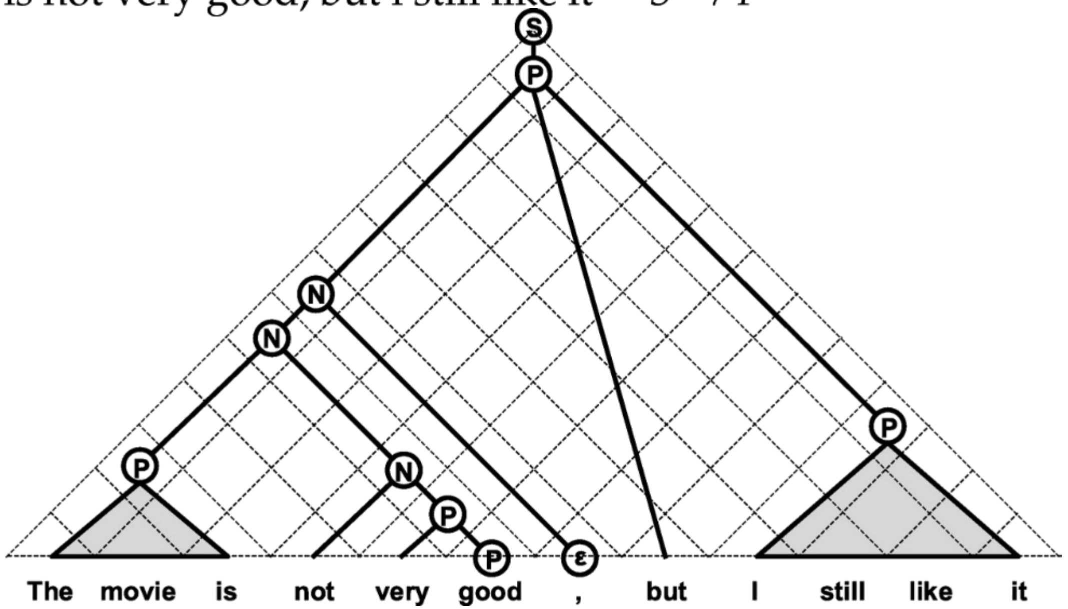
Sentiment Grammar

- Start rules $S \to Y$, where $Y \in \{N, P, \mathcal{E}\}$



Sentiment Grammar

Span	Rule	Strength	Polarity
$[0, P, 3]$: the movie is	$P \rightarrow$ the movie is	0.52	$\mathcal{P}$
$[5, P, 6]$: good	$P \rightarrow$ good	0.87	$\mathcal{P}$
$[6, \mathcal{E}, 7]$: ,	$\mathcal{E} \rightarrow$,	-	-
$[8, P, 11]$: i still like it	$P \rightarrow$ i still like it	0.85	$\mathcal{P}$
$[4, P, 6]$: very good	$P \rightarrow$ very P	0.93	$\mathcal{P}$
$[3, N, 6]$: not very good	$N \rightarrow$ not P	0.63	$\mathcal{N}$
$[0, N, 6]$: the movie is not very good	$N \rightarrow PN$	0.60	$\mathcal{N}$
$[0, N, 7]$: the movie is not very good,	$N \rightarrow N\mathcal{E}$	0.60	$\mathcal{N}$
$[0, P, 11]$: the movie is not very good, but i still like it	$P \rightarrow N$ but P	0.76	$\mathcal{P}$
$[0, S, 11]$: the movie is not very good, but i still like it	$S \rightarrow P$	0.76	$\mathcal{P}$



Parsing Model

- Present inference rules using deductive proof systems

Rule

$$\frac{(A \rightarrow BC) \quad [i, B, k] \quad [k, C, j]}{[i, A, j]}$$

Item

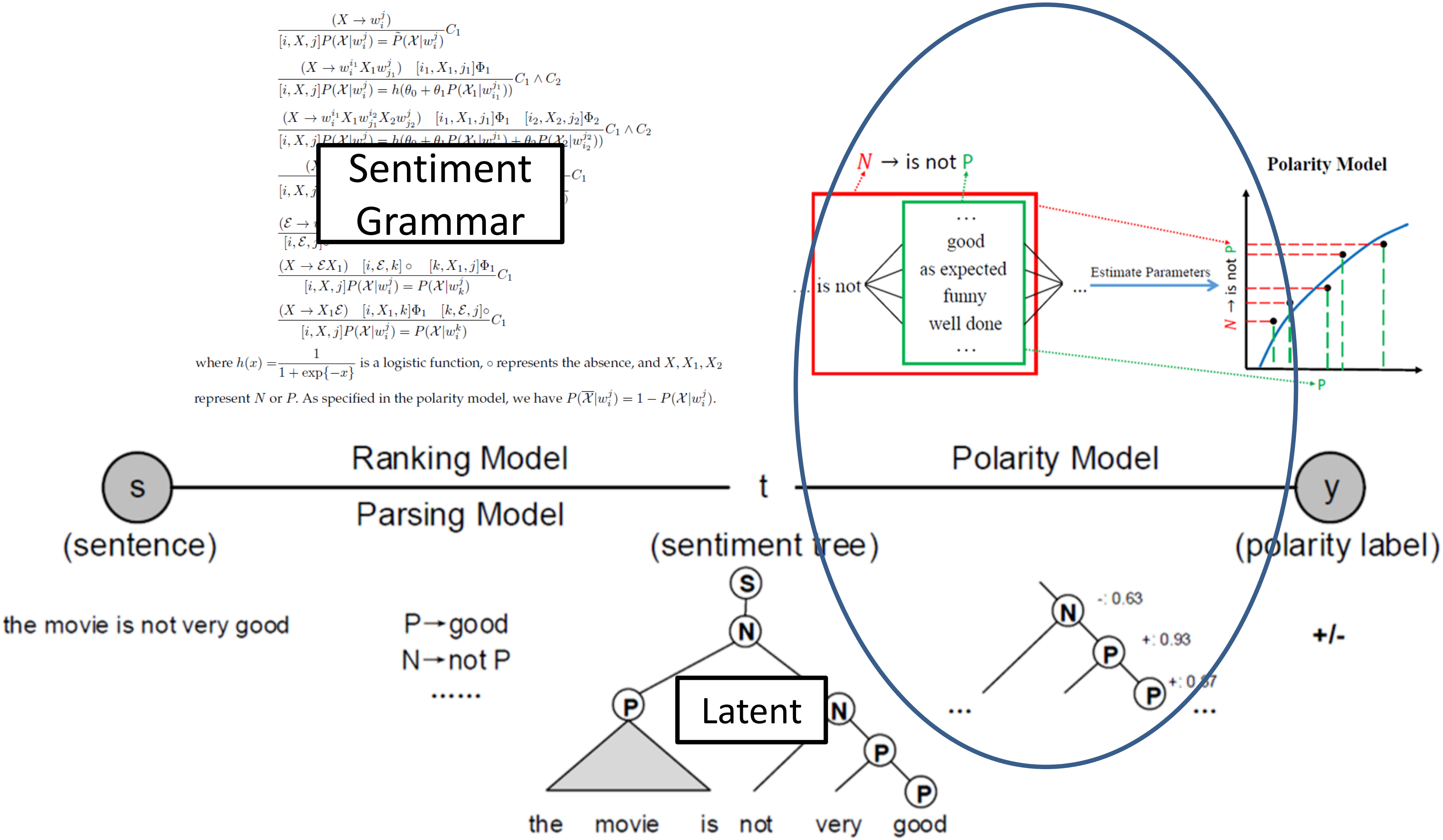
- If $A \rightarrow BC$ and $B \overset{*}{\Rightarrow} w_i^k$ and $C \overset{*}{\Rightarrow} w_k^j$
- Then $A \overset{*}{\Rightarrow} w_i^j$

(Shieber, Schabes, and Pereira 1995; Goodman 1999)

Inference Rules

P->good	$\frac{(X \rightarrow w_i^j)}{[i, X, j]}$
N->not P	$\frac{(X \rightarrow w_i^{i_1} X_1 w_{j_1}^j) \quad [i_1, X_1, j_1]}{[i, X, j]}$
P->N but P	$\frac{(X \rightarrow w_i^{i_1} X_1 w_{j_1}^{i_2} X_2 w_{j_2}^j) \quad [i_1, X_1, j_1] \quad [i_2, X_2, j_2]}{[i, X, j]}$
P->NP	$\frac{(X \rightarrow X_1 X_2) \quad [i, X_1, k] \quad [k, X_2, j]}{[i, X, j]}$
E->,	$\frac{(\mathcal{E} \rightarrow w_i^j)}{[i, \mathcal{E}, j]}$
P->EP	$\frac{(X \rightarrow \mathcal{E} X_1) \quad [i, \mathcal{E}, k] \quad [k, X_1, j]}{[i, X, j]}$
P->PE	$\frac{(X \rightarrow X_1 \mathcal{E}) \quad [i, X_1, k] \quad [k, \mathcal{E}, j]}{[i, X, j]}$

where X, X_1, X_2 represent N or P.


Sentiment Grammar
where $h(x) = \frac{1}{1+\exp\{-x\}}$ is a logistic function, $\circ$ represents the absence, and X, X_1, X_2 represent N or P. As specified in the polarity model, we have $P(\overline{\mathcal{X}}|w_i^j) = 1 - P(\mathcal{X}|w_i^j)$.
N → is not P
... is not
...
good
as expected
funny
well done
...
Estimate Parameters
Polarity Model
N → is not P
P
Ranking Model
Parsing Model
Polarity Model
s
(sentence)
t
(sentiment tree)
y
(polarity label)
the movie is not very good
P→good
N→not P
......
S
N
P
N
P
P
Latent
the movie is not very good
-: 0.63
+: 0.93
+/-

Polarity Model

- Calculate polarity strength from subspans

$$\frac{(r) \quad H_1 \quad \ldots \quad H_K}{[i, X, j]}$$

↓

$$\frac{(r) \quad H_1\Phi_1 \quad \ldots \quad H_K\Phi_K}{[i, X, j]\Phi}$$

$$\text{polarity strength } \Phi_k : \begin{cases} P(\mathcal{N}|w_{i_k}^{j_k}) \\ P(\mathcal{P}|w_{i_k}^{j_k}) \end{cases} \quad \text{for text span } w_{i_k}^{j_k}$$

- Polarity model: $\Phi(r, \Phi_1, \ldots, \Phi_K)$

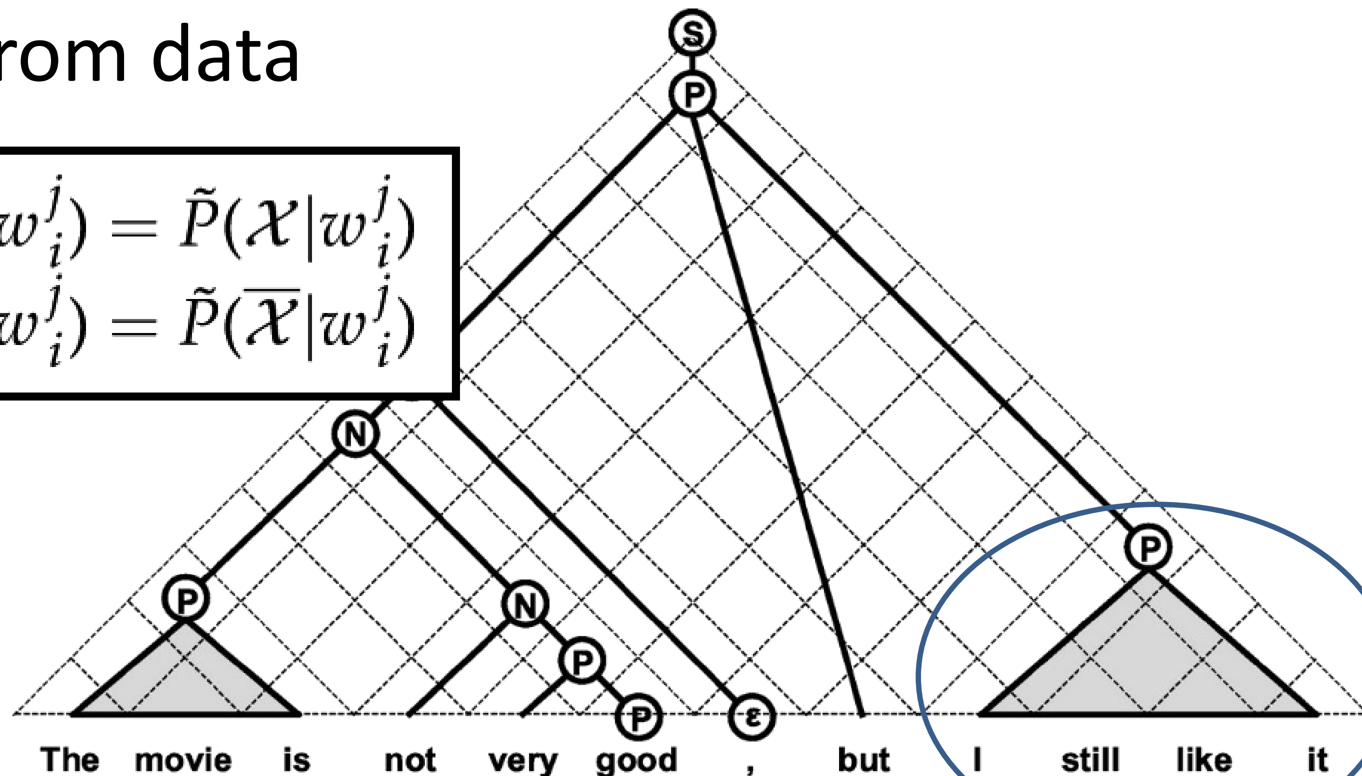
Polarity Model

- Two constraints for polarity strength
  - Non-negative $P(\mathcal{X}|w_i^j) \geq 0, P(\overline{\mathcal{X}}|w_i^j) \geq 0$
  - Normalized to 1 $P(\mathcal{X}|w_i^j) + P(\overline{\mathcal{X}}|w_i^j) = 1$

Polarity Model

- Dictionary rules $X \rightarrow w_0^k$, where $X \in \{N, P\}$, $w_0^k = w_0 \dots w_{k-1}$
  - P -> I still like it
  - P -> good
- Estimated from data

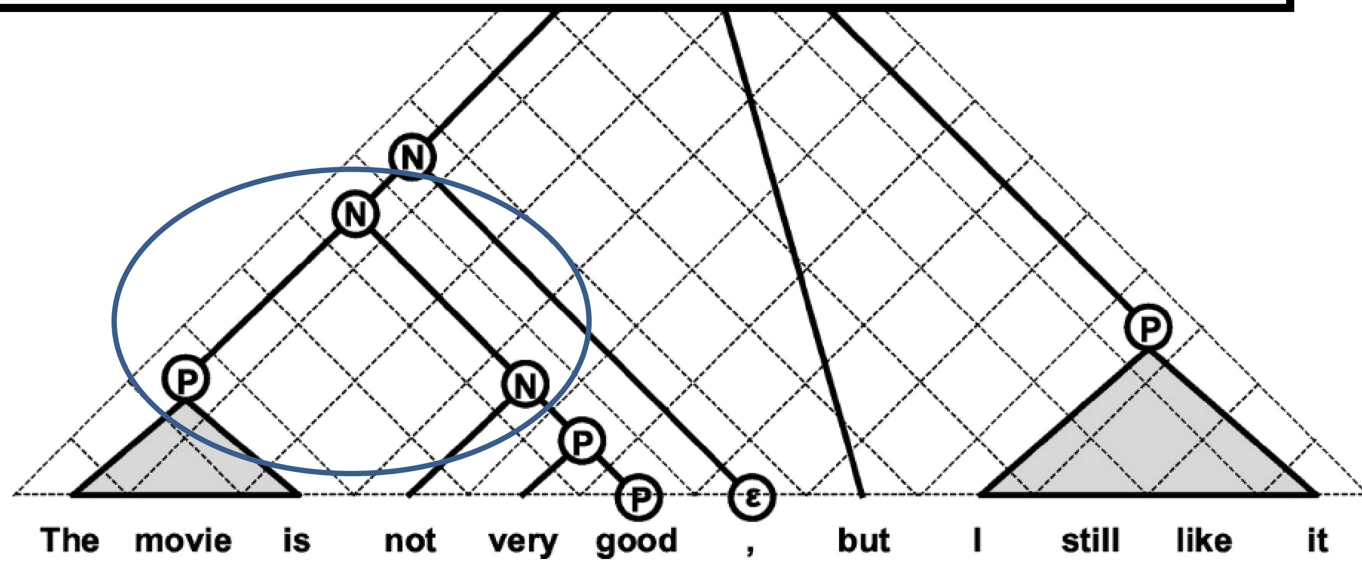
$$\Phi : \begin{cases} P(\mathcal{X}|w_i^j) = \tilde{P}(\mathcal{X}|w_i^j) \\ P(\overline{\mathcal{X}}|w_i^j) = \tilde{P}(\overline{\mathcal{X}}|w_i^j) \end{cases}$$



Polarity Model

- Glue rules $X \to X_1 X_2$, where $X, X_1, X_2 \in \{N, P\}$
  - P->NP
  - P->PN
  - P->PP
  - N->NN

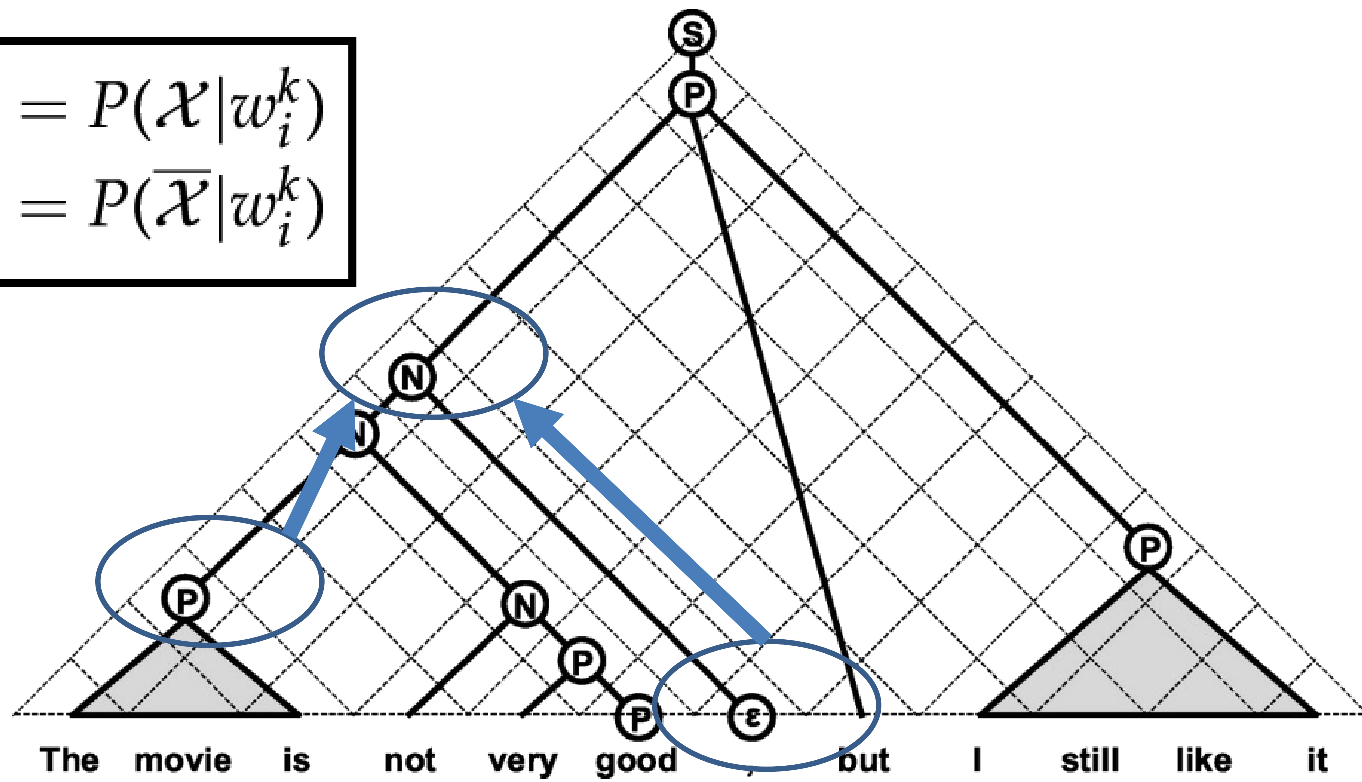
$$\Phi : \begin{cases} P(\mathcal{X}|w_i^j) = \dfrac{P(\mathcal{X}|w_i^k)P(\mathcal{X}|w_k^j)}{P(\mathcal{X}|w_i^k)P(\mathcal{X}|w_k^j)+P(\overline{\mathcal{X}}|w_i^k)P(\overline{\mathcal{X}}|w_k^j)} \\ P(\overline{\mathcal{X}}|w_i^j) = 1 - P(\mathcal{X}|w_i^j) \end{cases}$$



Polarity Model

- Auxiliary rules $X \rightarrow \mathcal{E}X_1, X \rightarrow X_1\mathcal{E}, \text{where } X, X_1 \in \{N, P\}$
  - Combine OOV span and non-terminal
  - OOV span is ignored

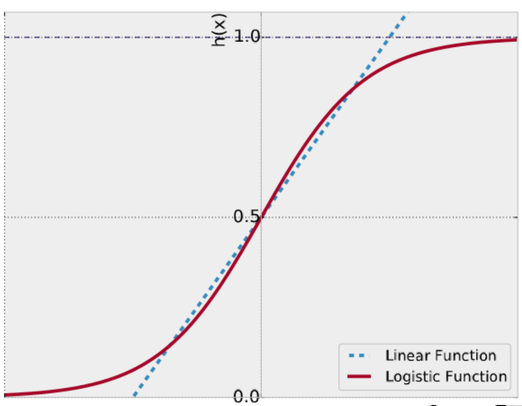
$$\Phi : \begin{cases} P(\mathcal{X}|w_i^j) = P(\mathcal{X}|w_i^k) \\ P(\overline{\mathcal{X}}|w_i^j) = P(\overline{\mathcal{X}}|w_i^k) \end{cases}$$

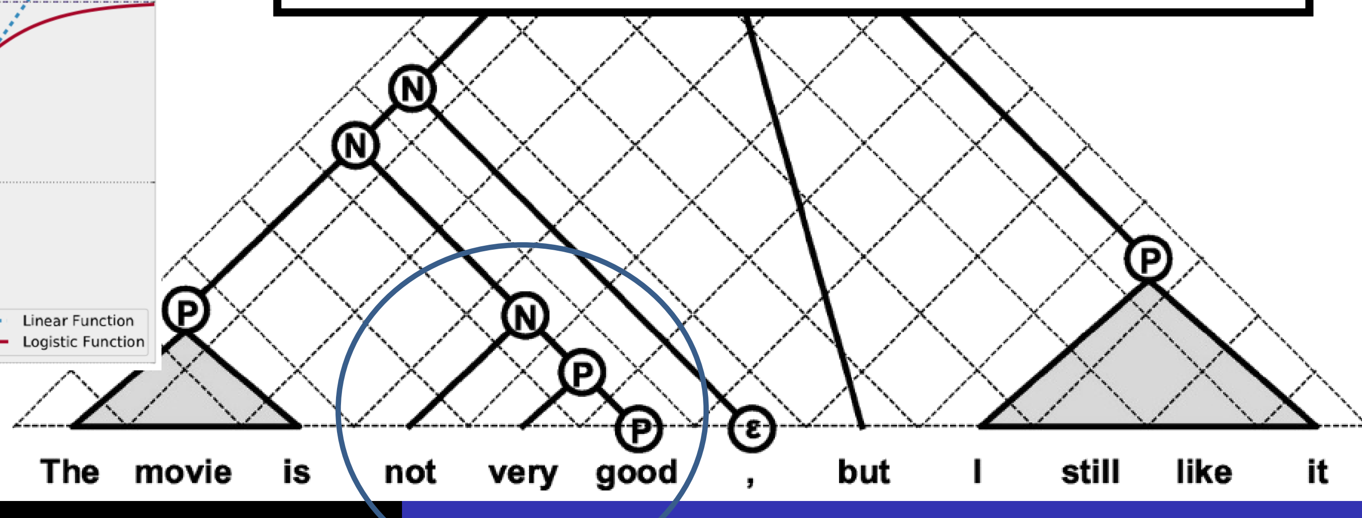


Polarity Model

- Combination rules $X \to c$, where $c \in (V_s \cup \Sigma_s)^+$,
  - P->N but P
  - P->not P
  - P-> very P
- Logistic regression

$$P(\mathcal{X}|w_i^j) = h\left(\theta_0 + \sum_{k=1}^{K} \theta_k P(\mathcal{X}_k|w_{i_k}^{j_k})\right)$$
$$= \frac{1}{1+\exp\left\{-\left(\theta_0 + \sum_{k=1}^{K} \theta_k P(\mathcal{X}_k|w_{i_k}^{j_k})\right)\right\}}$$





Why is logistic regression good?

- Negation (N->not P)
  - Switch negation (Choi and Cardie 2008; Sauri 2008)
    - Simply reverse strength
    - P(Neg|not good) = P(Pos|good)
  - Shift negation (Taboada et al. 2011)
    - Problem: P(Neg|not very good) > P(Neg|not good)
    - Solution: P(Neg|not good) = P(Pos|good) – fixed_value

Parameter	Negation Type $P(\mathcal{X}\vert w_i^j) = h(\theta_0 + \theta_1 P(\overline{\mathcal{X}}\vert w_{i_1}^{j_1}))$	
	Shift	Switch
θ_0 (Shift item)	✓	
θ_1 (Scale item)		✓

Why is logistic regression good?

- Intensification (P->extremely P)
  - Fixed intensification (Polanyi and Zaenen 2006; Kennedy and Inkpen 2006)
    - P(Pos|very good) = P(Pos|good) + fixed_value (>0)
  - Percentage intensification (Taboada et al. 2011)
    - P(Pos|very good) = P(Pos|good) * fixed_value (!=1)

Parameter	Intensification Type $P(\mathcal{X}\|w_i^j) = h(\theta_0 + \theta_1 P(\mathcal{X}\|w_{i_1}^{j_1}))$	
	Percentage	Fixed
θ_0 (Shift item)		✓
θ_1 (Scale item)	✓	

Why is logistic regression good?

- Reasons
  - Smooth polarity strength to (0,1)
  - Can learn various types of negation and intensification
  - Can handle contrast (P->N but P)

Inference Rules (w/ Polarity Model)

P->good
$$\frac{(X \to w_i^j)}{[i,X,j]P(\mathcal{X}|w_i^j) = \tilde{P}(\mathcal{X}|w_i^j)}$$

N->not P
$$\frac{(X \to w_i^{i_1} X_1 w_{j_1}^j) \quad [i_1, X_1, j_1]\Phi_1}{[i,X,j]P(\mathcal{X}|w_i^j) = h(\theta_0 + \theta_1 P(\mathcal{X}_1|w_{i_1}^{j_1}))}$$

P->N but P
$$\frac{(X \to w_i^{i_1} X_1 w_{j_1}^{i_2} X_2 w_{j_2}^j) \quad [i_1, X_1, j_1]\Phi_1 \quad [i_2, X_2, j_2]\Phi_2}{[i,X,j]P(\mathcal{X}|w_i^j) = h(\theta_0 + \theta_1 P(\mathcal{X}_1|w_{i_1}^{j_1}) + \theta_2 P(\mathcal{X}_2|w_{i_2}^{j_2}))}$$

P->NP
$$\frac{(X \to X_1 X_2) \quad [i, X_1, k]\Phi_1 \quad [k, X_2, j]\Phi_2}{[i,X,j]P(\mathcal{X}|w_i^j) = \frac{P(\mathcal{X}|w_i^k)P(\mathcal{X}|w_k^j)}{P(\mathcal{X}|w_i^k)P(\mathcal{X}|w_k^j) + P(\overline{\mathcal{X}}|w_i^k)P(\overline{\mathcal{X}}|w_k^j)}}$$

E->,
$$\frac{(\mathcal{E} \to w_i^j)}{[i,\mathcal{E},j]\circ}$$

P->EP
$$\frac{(X \to \mathcal{E} X_1) \quad [i,\mathcal{E},k]\circ \quad [k, X_1, j]\Phi_1}{[i,X,j]P(\mathcal{X}|w_i^j) = P(\mathcal{X}|w_k^j)}$$

P->PE
$$\frac{(X \to X_1 \mathcal{E}) \quad [i, X_1, k]\Phi_1 \quad [k,\mathcal{E},j]\circ}{[i,X,j]P(\mathcal{X}|w_i^j) = P(\mathcal{X}|w_i^k)}$$

Constraints (in Parsing Model)

- Add side condition C for inference rules

$$\frac{(r) \quad H_1\Phi_1 \quad \ldots \quad H_K\Phi_K}{[i, X, j]\Phi}$$

↓

$$\frac{(r) \quad H_1\Phi_1 \quad \ldots \quad H_K\Phi_K}{[i, X, j]\Phi} C$$

Constraints (in Parsing Model)

$$\frac{(r) \quad H_1\Phi_1 \quad \ldots \quad H_K\Phi_K}{[i, X, j]\Phi} C$$

- Polarity should be consistent with non-terminal

$$C_1 : P(\mathcal{X}|w_i^j) > P(\overline{\mathcal{X}}|w_i^j)$$

polarity label of non-terminal X as $\mathcal{X}$

- Avoid improperly using combination rules for neutral phrase
  - Do not use P-> a lot of P for P->a lot of people

$$C_2 : P(\mathcal{X}_k|w_{i_k}^{j_k}) > \text{threshold}\ \tau\ (\geq 0.5)$$

Inference Rules (w/ Polarity Model and Constraints)

P->good
$$\frac{(X \rightarrow w_i^j)}{[i,X,j]P(\mathcal{X}|w_i^j) = \tilde{P}(\mathcal{X}|w_i^j)} C_1$$

N->not P
$$\frac{(X \rightarrow w_i^{i_1} X_1 w_{j_1}^j) \quad [i_1, X_1, j_1]\Phi_1}{[i,X,j]P(\mathcal{X}|w_i^j) = h(\theta_0 + \theta_1 P(\mathcal{X}_1|w_{i_1}^{j_1}))} C_1 \wedge C_2$$

P->N but P
$$\frac{(X \rightarrow w_i^{i_1} X_1 w_{j_1}^{i_2} X_2 w_{j_2}^j) \quad [i_1, X_1, j_1]\Phi_1 \quad [i_2, X_2, j_2]\Phi_2}{[i,X,j]P(\mathcal{X}|w_i^j) = h(\theta_0 + \theta_1 P(\mathcal{X}_1|w_{i_1}^{j_1}) + \theta_2 P(\mathcal{X}_2|w_{i_2}^{j_2}))} C_1 \wedge C_2$$

P->NP
$$\frac{(X \rightarrow X_1 X_2) \quad [i, X_1, k]\Phi_1 \quad [k, X_2, j]\Phi_2}{[i,X,j]P(\mathcal{X}|w_i^j) = \frac{P(\mathcal{X}|w_i^k)P(\mathcal{X}|w_k^j)}{P(\mathcal{X}|w_i^k)P(\mathcal{X}|w_k^j) + P(\overline{\mathcal{X}}|w_i^k)P(\overline{\mathcal{X}}|w_k^j)}} C_1$$

E->,
$$\frac{(\mathcal{E} \rightarrow w_i^j)}{[i,\mathcal{E},j]\circ} \circ$$

P->EP
$$\frac{(X \rightarrow \mathcal{E} X_1) \quad [i,\mathcal{E},k]\circ \quad [k, X_1, j]\Phi_1}{[i,X,j]P(\mathcal{X}|w_i^j) = P(\mathcal{X}|w_k^j)} C_1$$

P->PE
$$\frac{(X \rightarrow X_1 \mathcal{E}) \quad [i, X_1, k]\Phi_1 \quad [k,\mathcal{E},j]\circ}{[i,X,j]P(\mathcal{X}|w_i^j) = P(\mathcal{X}|w_i^k)} C_1$$

Ranking Model

- Parsing model generates many candidates T(s)
- Use log-linear model to rank and score T(s)

Weight

$$p(t|s;T,\psi) = \exp\left\{\phi(s,t)^{\mathsf{T}}\psi - \mathbf{A}(\psi;s,T)\right\}$$

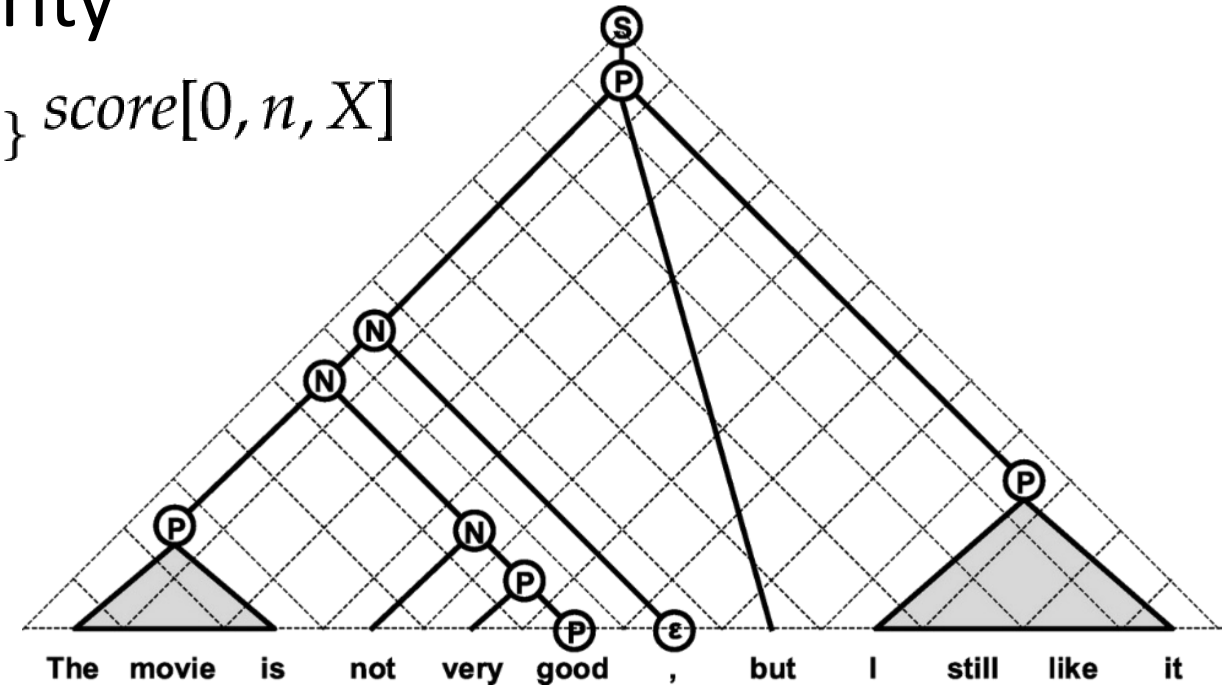
Feature

Log-partition function

Bottom-Up Decoding

- Ranking features decompose along trees
- CYK algorithm can be used to conduct decoding
  - Dynamic programming
- Predicted polarity

$$\arg\max_{\mathcal{X}\in\{\mathcal{N},\mathcal{P}\}} score[0, n, X]$$



Ranking Model Training

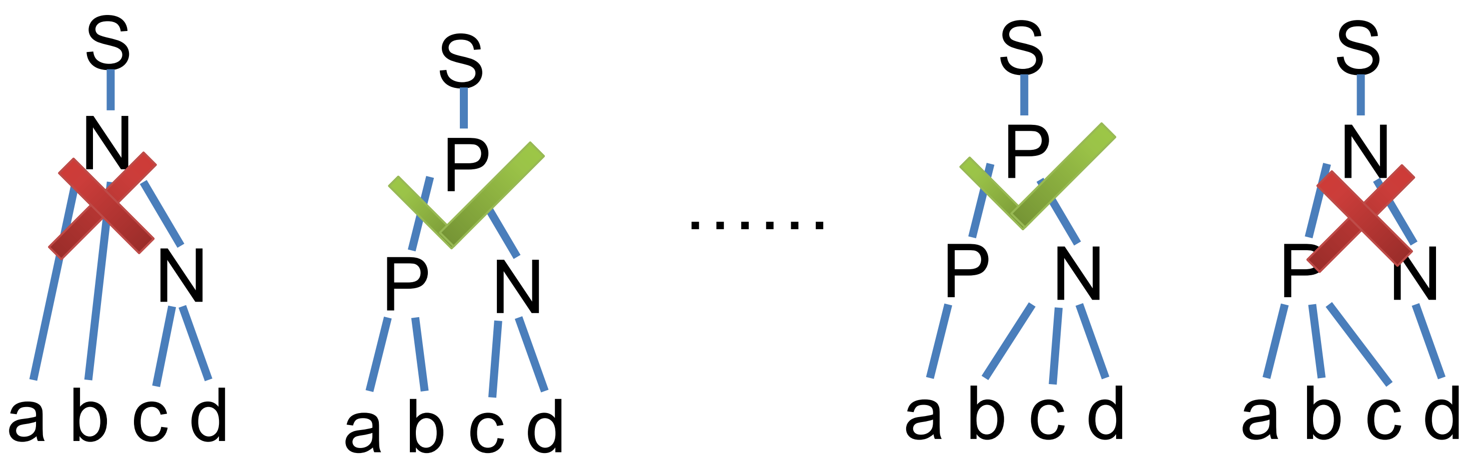
- Maximize probability of decoding correct labels

$$\mathcal{O}(\psi, T) = \sum_{\substack{(s,\mathcal{L}_s)\in\mathcal{D} \\ T^{\mathcal{L}_s}(s)\neq\emptyset}} \log p(\mathcal{L}_s | s; T, \psi) - \frac{\lambda}{2}\|\psi\|_2^2$$

Regularizer

Log-likelihood of trees obtaining the correct polarity label

- Example: (a b c d, P)



Learning Ranking Model

- EM-like training (Liang, Jordan, and Klein 2013)
  - Iteration 1:

    sentence model → beam search and score candidates → K-best trees

    ← update parameters (AdaGrad)

    K-best trees
    - Tree1 ✗
    - Tree2 ✗
    - Tree3 ✓
    - Tree4 ✗
  - ....
  - Iteration N:

    sentence model → beam search and score candidates → K-best trees

    ← update parameters (AdaGrad)

    K-best trees
    - Tree3 ✓
    - Tree5 ✓
    - Tree8 ✓
    - Tree4 ✗

Learning Sentiment Grammar and Polarity Model

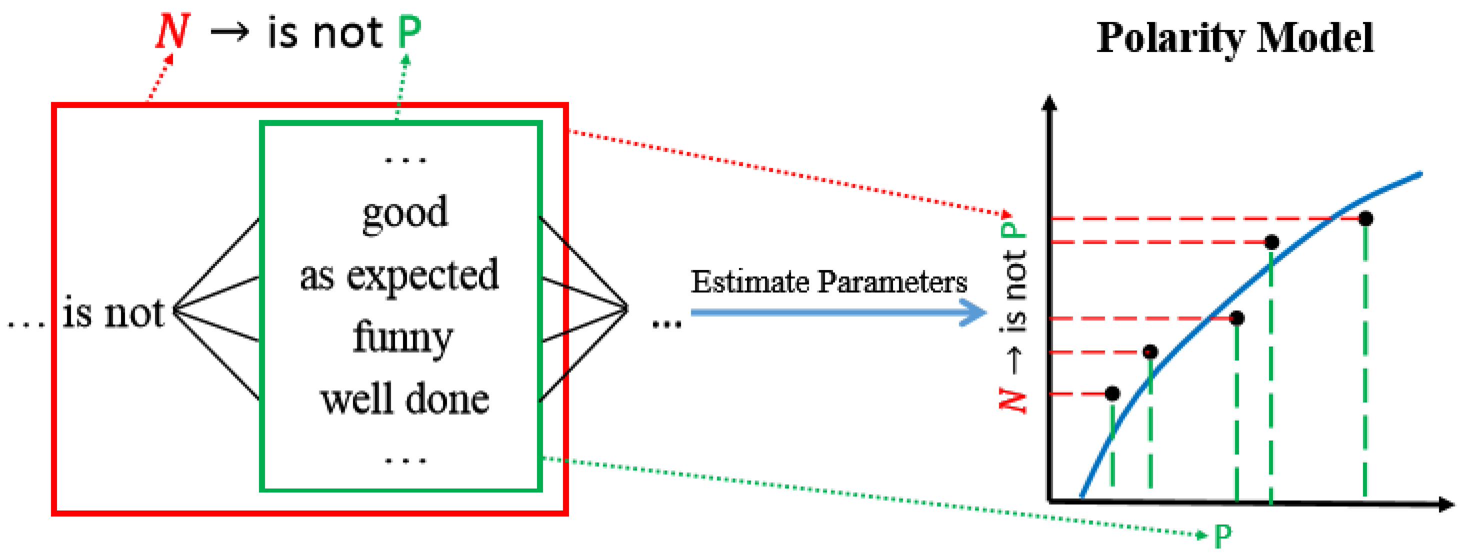
- Dictionary rules (P->good, N->i dislike this)
  - Mine frequent fragments as candidates
  - Prune them using polarity strength

$$P(\mathcal{X}|f) = \frac{\#(f,\mathcal{X}) + 1}{\#(f,\mathcal{N}) + \#(f,\mathcal{P}) + 2}$$

- Problem
  - This is not a **good** movie. (negative)
- Solution
  - Consider <u>negation rules</u> when learning polarity model for dictionary rules

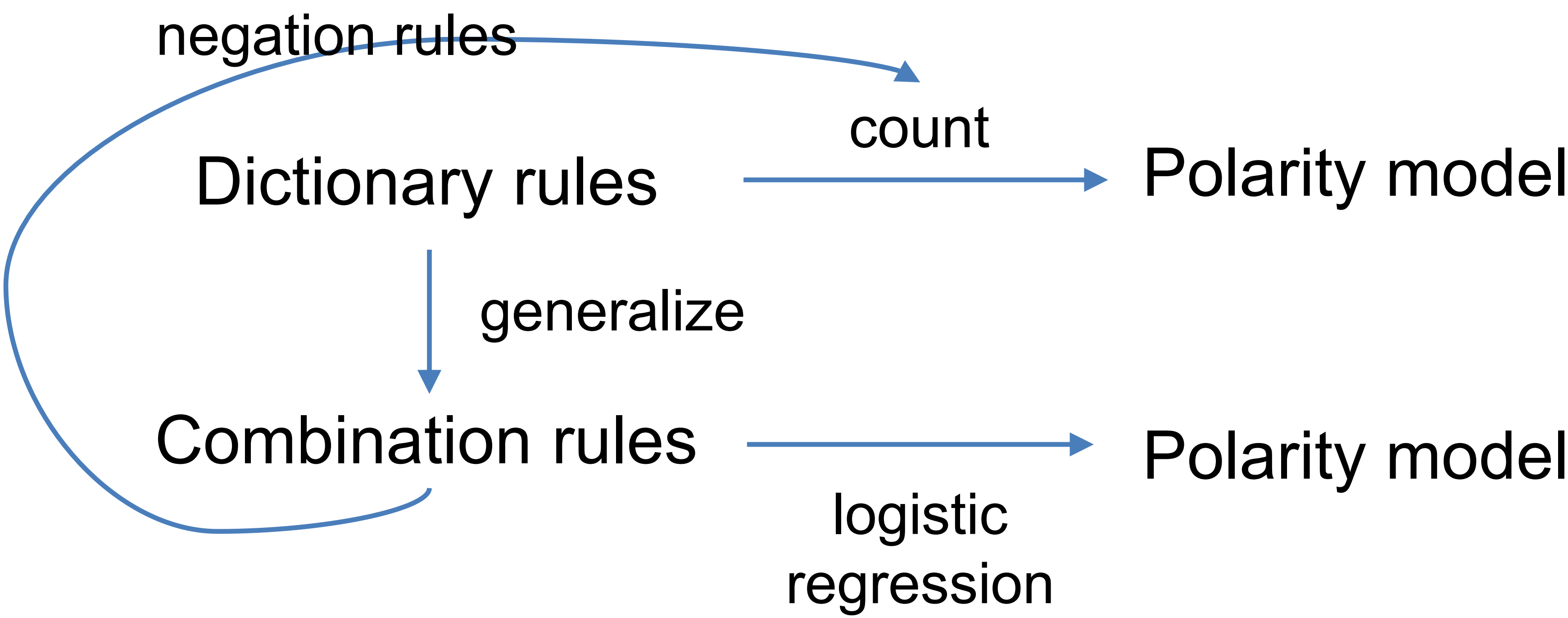
Learning Sentiment Grammar and Polarity Model

- Combination rules (N->not P)
  - Generalize dictionary rules
  - Polarity model: logistic regression



Learning Sentiment Grammar and Polarity Model

- Iteration process

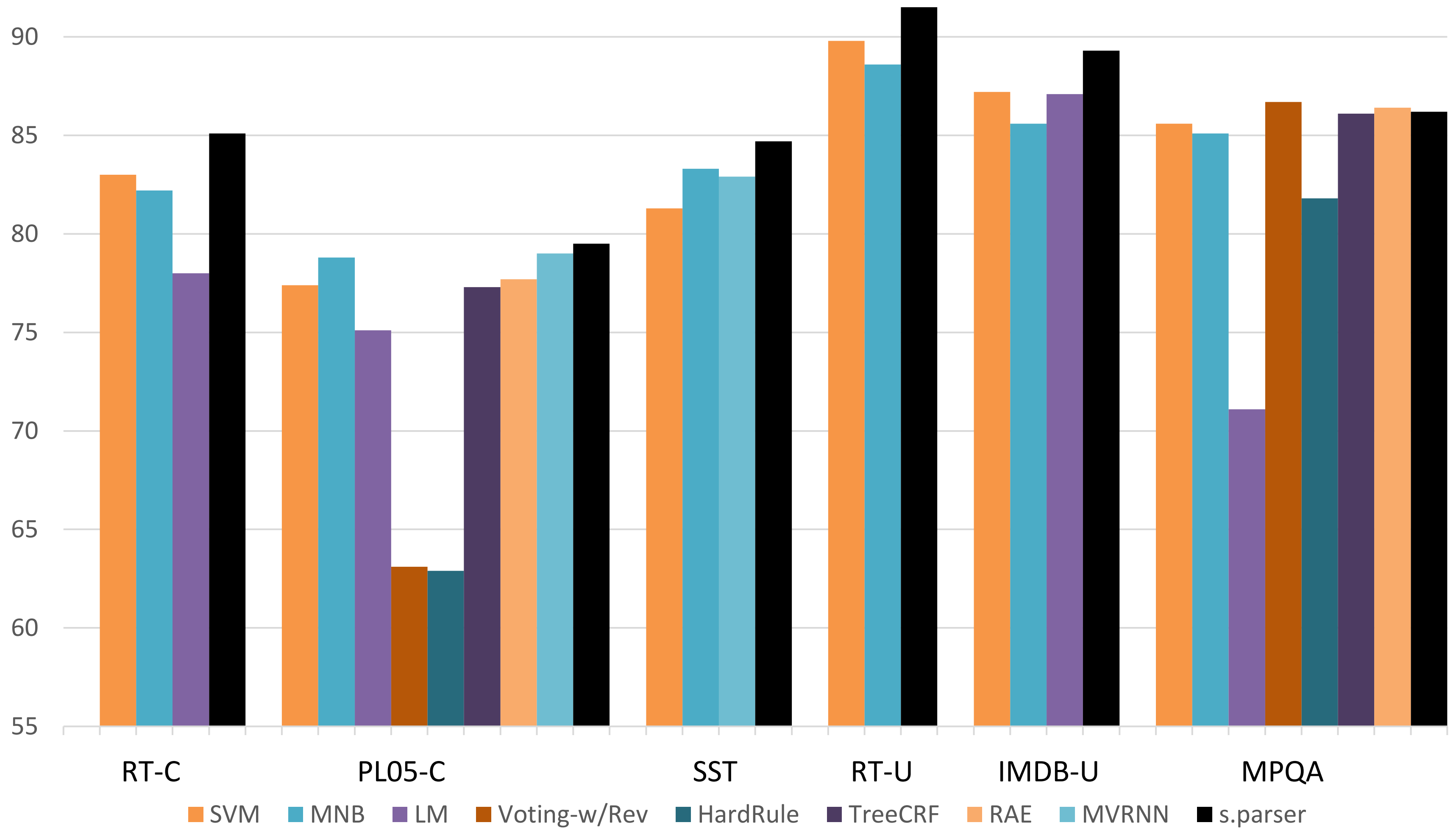


Experiments

- Datasets

	Data Set	Size	#Negative	#Positive	l_{avg}	$\|V\|$
critic reviews	RT-C	436,000	218,000	218,000	23.2	136,006
	PL05-C	10,662	5,331	5,331	21.0	20,263
	SST	98,796	42,608	56,188	7.5	16,372
user reviews	RT-U	737,806	368,903	368,903	15.4	138,815
	IMDB-U	600,000	300,000	300,000	6.6	83,615
	MPQA	10,624	7,308	3,316	3.1	5,992

Experiments

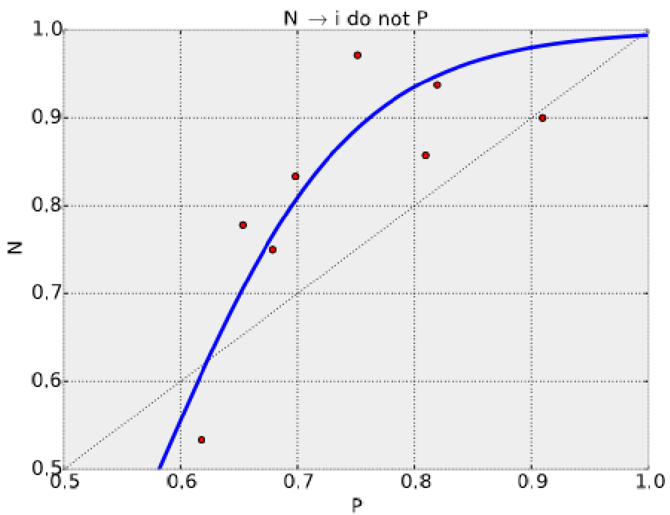


Ablation

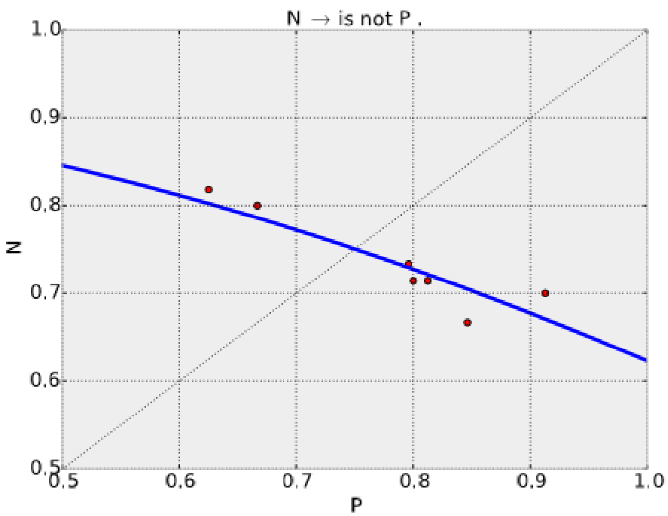
- Heuristic ranking trees
- Without combination rules

Combination Rules

- Switch negation (Choi and Cardie 2008; Sauri 2008)
  - Simply reverse strength
  - P(Neg|i do not like) = P(Pos|like)
- Shift negation (Taboada et al. 2011)
  - P(Neg|is not good) = P(Pos|good) – fixed_value

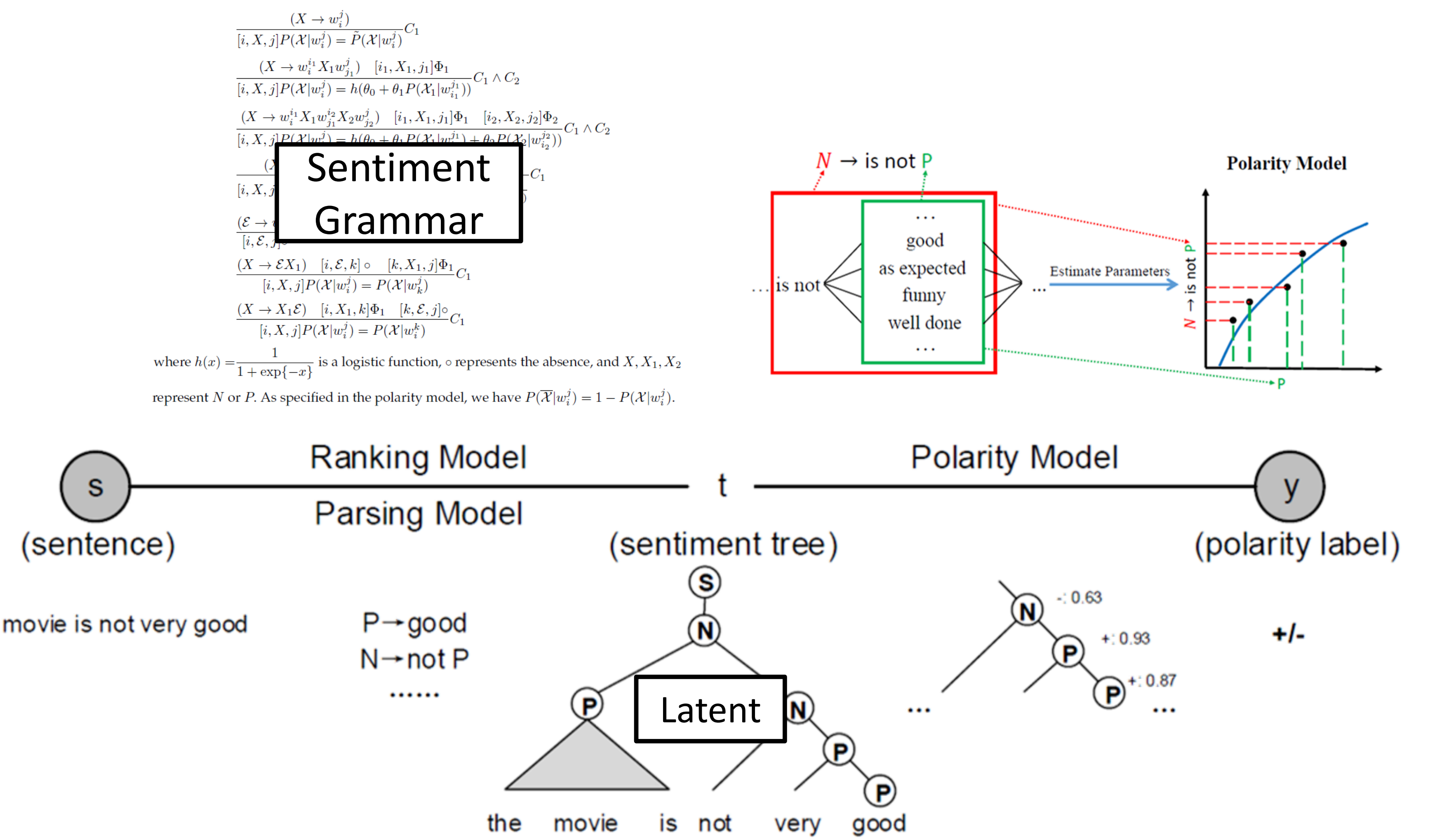


(a) $N \rightarrow$ i do not P



(b) $N \rightarrow$ is not P.

Conclusion



Sentiment Grammar
where $h(x) = \frac{1}{1+\exp\{-x\}}$ is a logistic function, $\circ$ represents the absence, and X, X_1, X_2 represent N or P. As specified in the polarity model, we have $P(\overline{\mathcal{X}}|w_i^j) = 1 - P(\mathcal{X}|w_i^j)$.
N → is not P
... is not
good
as expected
funny
well done
Estimate Parameters
Polarity Model
Ranking Model
Parsing Model
Polarity Model
s
t
y
(sentence)
(sentiment tree)
(polarity label)
the movie is not very good
P→good
N→not P
......
Latent
the movie is not very good
-: 0.63
+: 0.93
+: 0.87
+/-

Many Thanks

P->thanks

P->many P