Connotation:
A Dash of Sentiment
Beneath the Surface Meaning

Con•notation

"com-" ("together or with") | "notare" ("to mark")

Con•notation

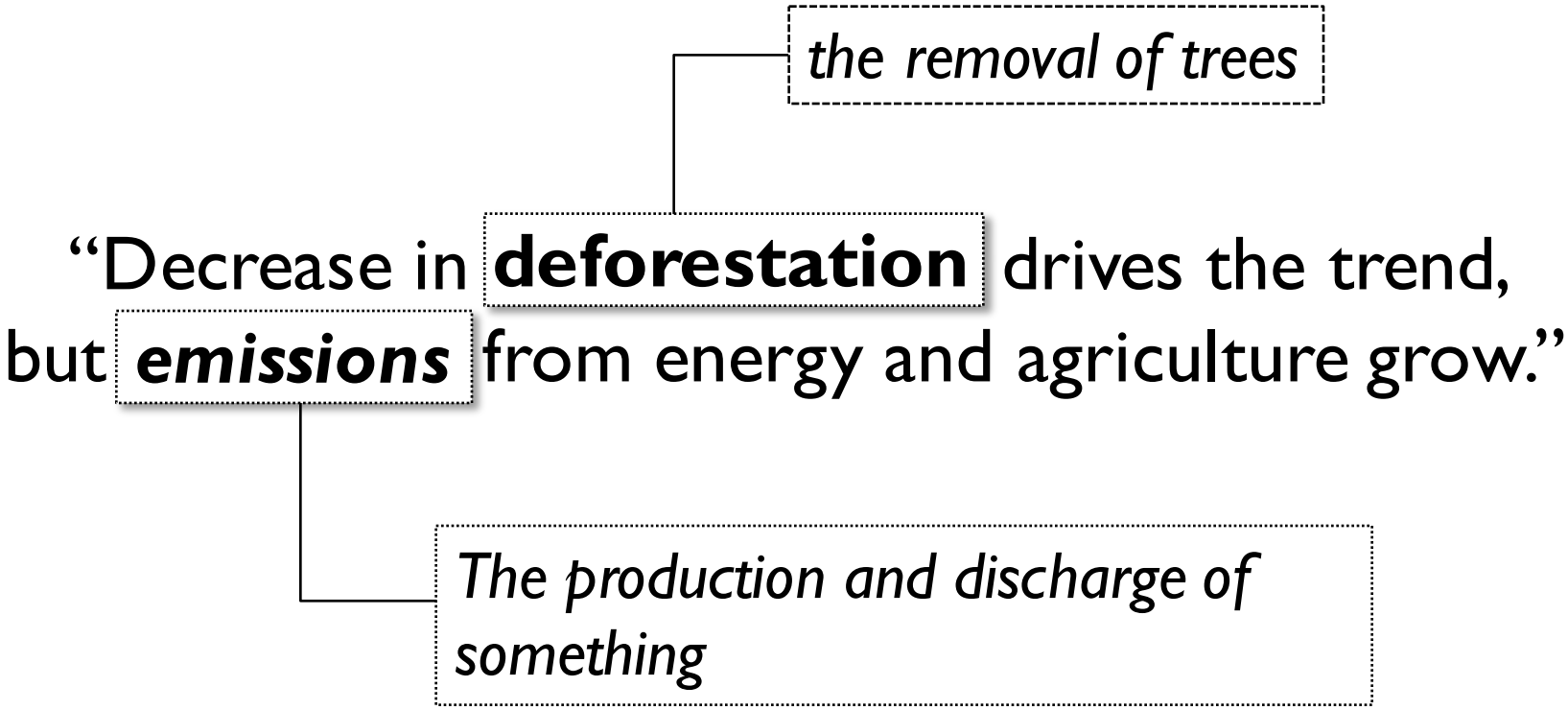
"com-" ("together or with") | "notare" ("to mark")

- Commonly understood cultural or emotional *association* that some word carries, in addition to its explicit or literal meaning (denotation).
- Generally described as positive or negative.

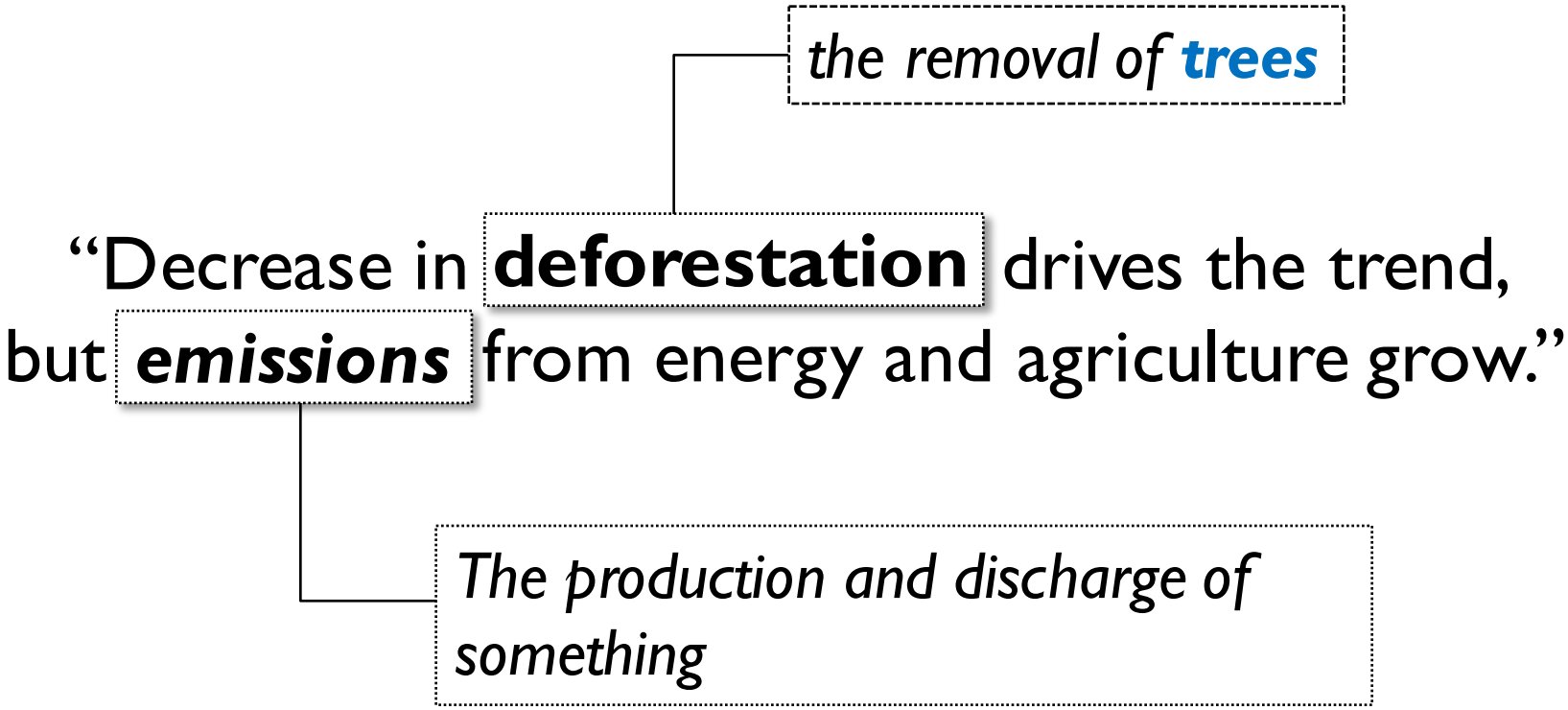
Good News? Bad News?

"Decrease in deforestation drives the trend,
but emissions from energy and agriculture grow."

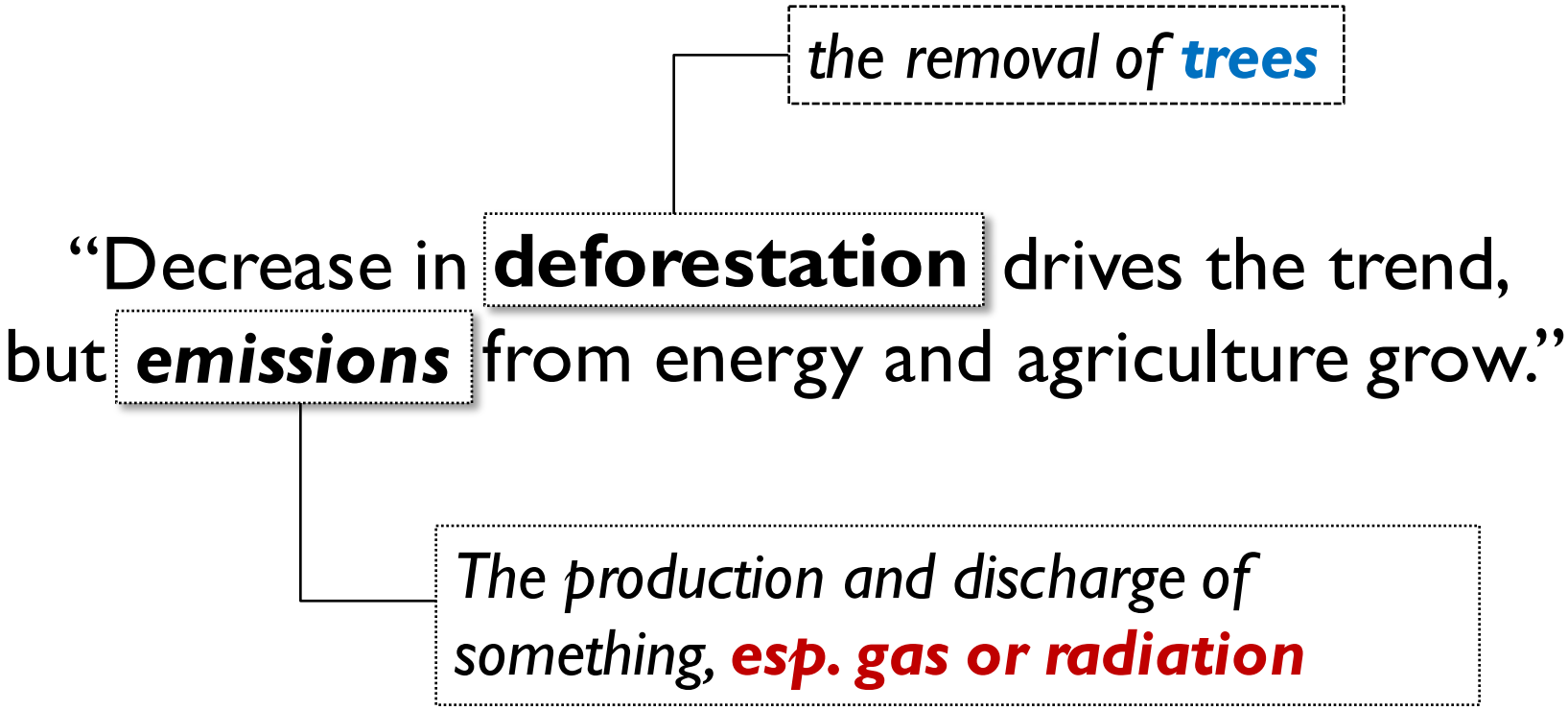
Good News? Bad News?



Good News? Bad News?



Good News? Bad News?



Learning the General Connotation

"Decrease in **deforestation** drives the trend, but ***emissions*** from energy and agriculture grow."

Negatively connotative in general

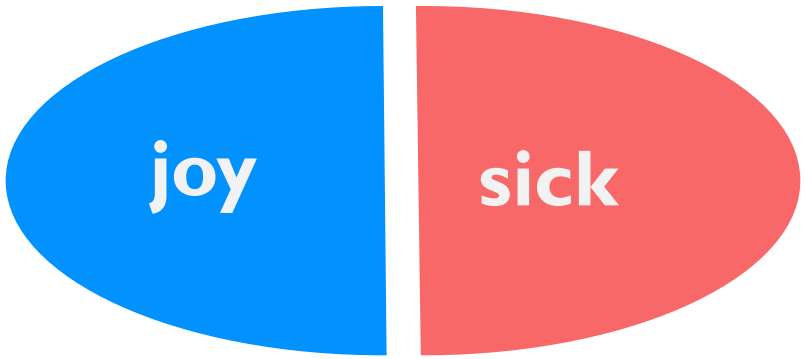
Con•notation

"com-" ("together or with") | "notare" ("to mark")

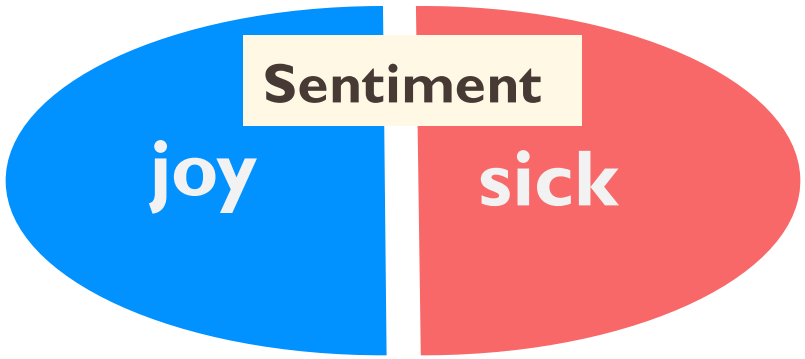
- Commonly understood cultural or emotional *association* that some word carries, in addition to its explicit or literal meaning (denotation).
- Generally described as positive or negative.

Sentiment vs. Connotation

Sentiment vs. Connotation



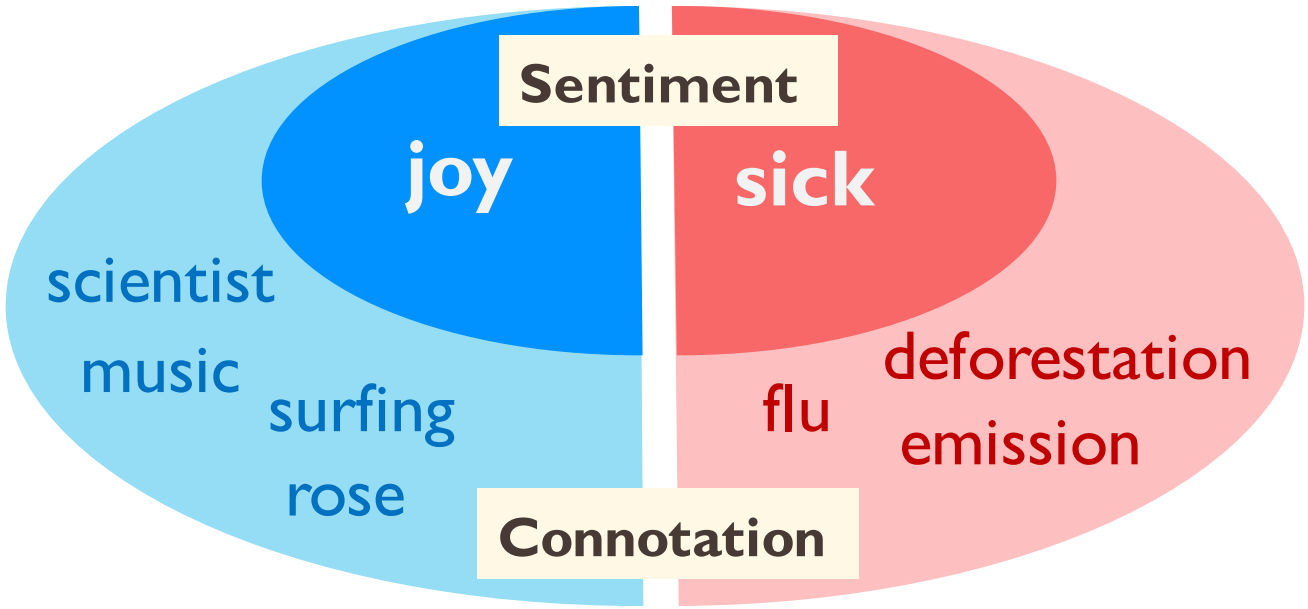
Sentiment vs. Connotation



Neutral

surfing
scientist
grid
rose
header
blister
emission
salt
bedbug
…

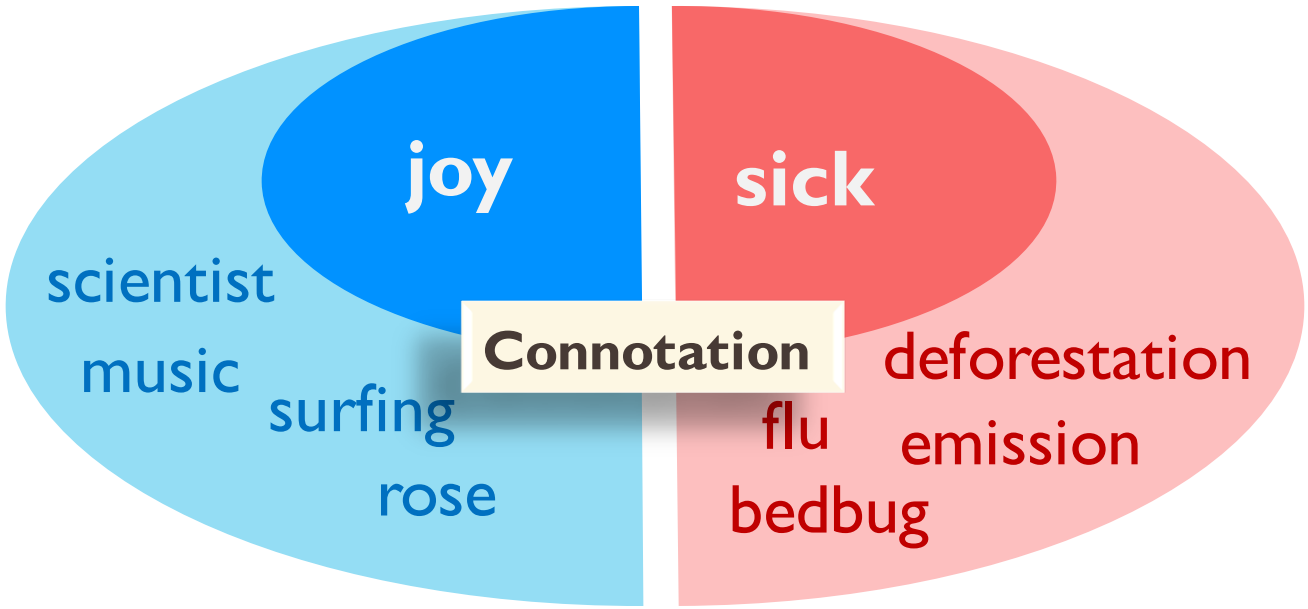
Sentiment vs. Connotation



Neutral

~~surfing~~
~~scientist~~
grid
~~rose~~
header
~~blister~~
~~emission~~
salt
~~bedbug~~
…

Sentiment vs. Connotation



Neutral

~~surfing~~
~~scientist~~
grid
~~rose~~
header
~~blister~~
~~emission~~
salt
~~bedbug~~
…

Learning the General Connotation

- Data
- Linguistic insights
- Graph representation
- Inference algorithms
- Evaluations

Data

- Web-driven data

  Google Web 1T (Brants and Franz (2006))

  - N-grams (1 <= n <= 5)
  - Frequency of occurrences
  - Example: "*prevent financial malware* 4130"

- Dictionary-drive data

  WordNet (George A. Miller (1995))

  - Synsets: synonyms, antonyms

Learning the General Connotation

- Data
- Linguistic insights
- Graph representation
- Inference algorithms
- Evaluations

Diverse Linguistic Insights

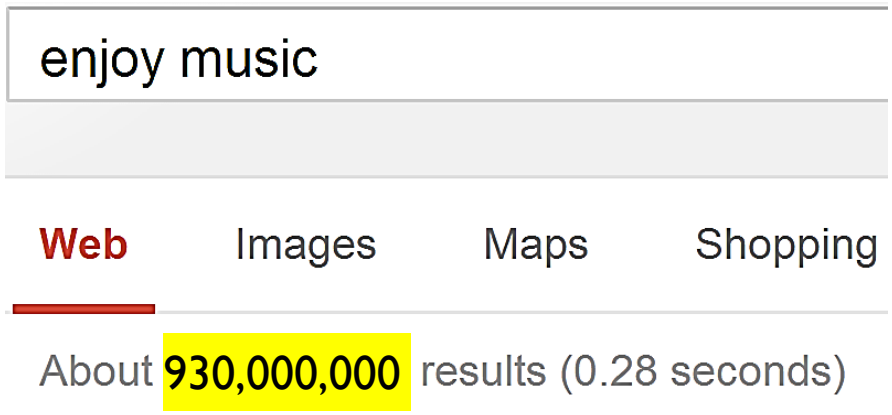
- Semantic prosody
- Semantic parallelism of coordination
- Distributional similarity
- Semantic relations

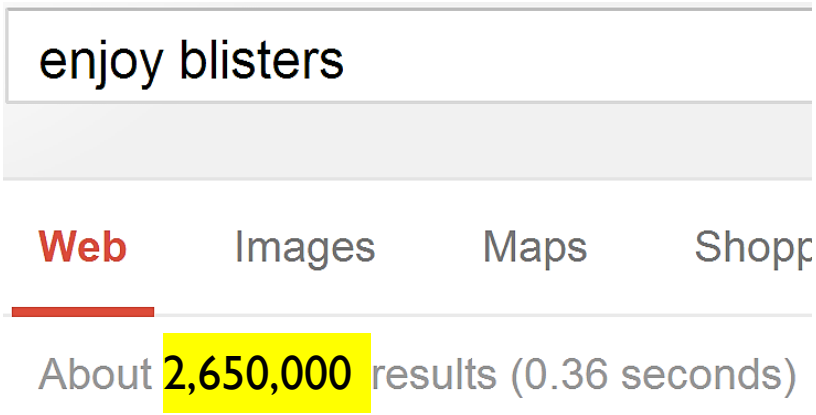
Diverse Linguistic Insights

- Semantic prosody
- Semantic parallelism of coordination
- Distributional similarity
- Semantic relations

Semantic Prosody

Sinclair (1991, 2004), Louw (1993)



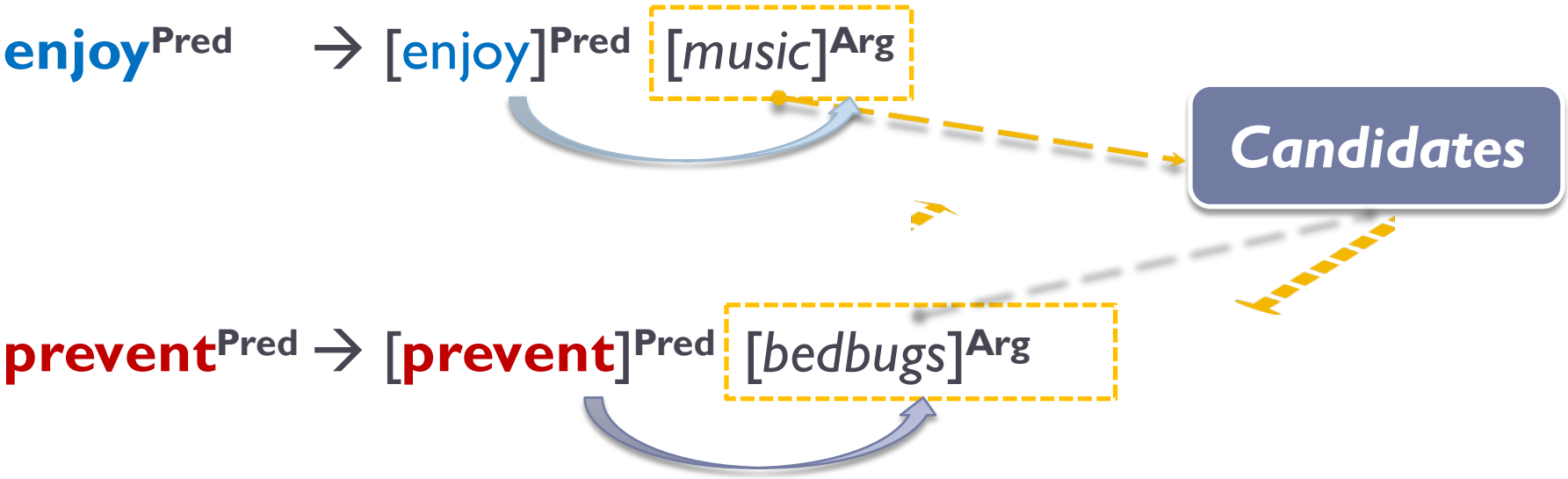


Selectional Preference on Connotation

enjoy$^{\text{Pred}}$ → [enjoy]$^{\text{Pred}}$ [*music*]$^{\text{Arg}}$

prevent$^{\text{Pred}}$ → [**prevent**]$^{\text{Pred}}$ [*bedbugs*]$^{\text{Arg}}$

Selectional Preference on Connotation



Connotative Predicate:

Feng et al. 2011

A predicate that has *selectional preference* on the connotative polarity of some of its semantic arguments.

Connotative Predicate:

Feng et al. 2011

A predicate that has *selectional preference* on the connotative polarity of some of its semantic arguments.

Connotative Predicates	Sentiment of predicate	Preference on arguments	Examples
suffer	negative	negative	"suffering from cough"
cure	positive	negative	"cure backache"
cause	neutral	negative	"caused CO_2 emissions"

Connotative Predicate:

Feng et al. 2011

A predicate that has *selectional preference* on the connotative polarity of some of its semantic arguments.

Connotative Predicates	Sentiment of predicate	Preference on arguments	Examples
suffer	negative	negative	"suffering from cough"
cure	positive	negative	"cure backache"
cause	neutral	negative	"caused CO_2 emissions"

20 Positive Connotative Predicates		20 Negative Connotative Predicates	
Accomplish	Desire	Alleviate	Eradicate
Achieve	Enhance	Accuse	Mitigate
Advance	Enjoy	Avert	Overcome
Advocate	Improve	Avoid	Prevent
Admire	Praise	Cause	Prohibit
Applaud	Promote	Complain	Protest
Appreciate	Respect	Condemn	Refrain
Compliment	Save	Criticize	Suffer
Congratulate	Support	Detect	Tolerate
Develop	Win	Eliminate	Withstand

Feng et al. 2011

Diverse Linguistic Insights

- Semantic prosody [Corpus: GoogleNgram]
  - [enjoy]$^{\text{Pred}}$ [*music*]$^{\text{Arg}}$, [**prevent**]$^{\text{Pred}}$ [*begbugs*]$^{\text{Arg}}$

Diverse Linguistic Insights

- Semantic prosody [Corpus: GoogleNgram]
  - "enjoy * ";"prevent * "
- Semantic parallelism of coordination [Corpus: GoogleNgram]
  - Pattern " * and * ", e.g., "music and wine"

Diverse Linguistic Insights

- Semantic prosody [Corpus: GoogleNgram]
  - "enjoy * ";"prevent * "
- Semantic parallelism of coordination [Corpus: GoogleNgram]
  - Pattern " * and * ", e.g., "music and wine"
- Distributional similarity [Corpus : GoogleNgram]
  - "findings"–"potentials" > "findings"–"modifications"

Diverse Linguistic Insights

- Semantic prosody [Corpus: GoogleNgram]
  - "enjoy * ";"prevent * "
- Semantic parallelism of coordination [Corpus: GoogleNgram]
  - Pattern " * and * ", e.g., "music and wine"
- Distributional similarity [Corpus : GoogleNgram]
  - "findings"–"potentials" > "findings"–"modifications"
- Semantic relations [Corpus: WordNet]
  - Synonyms
  - Antonyms

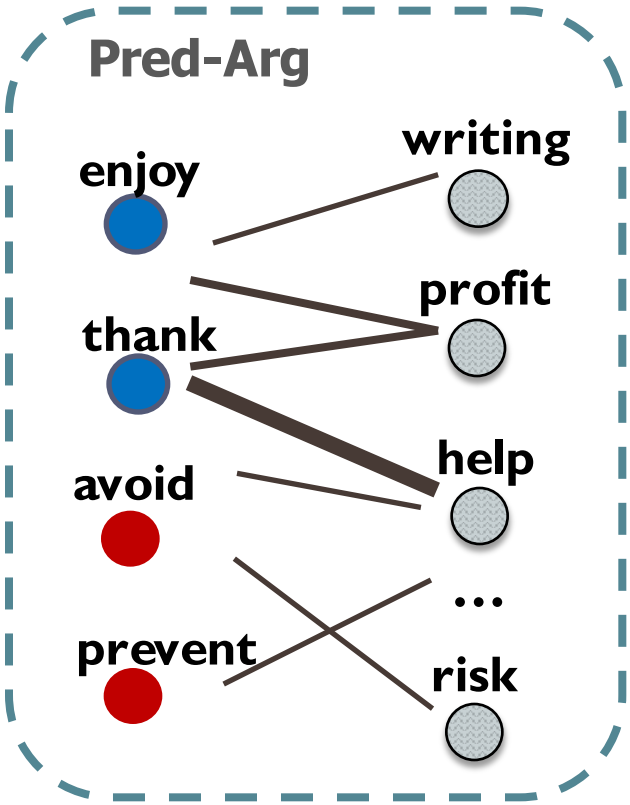
Learning the General Connotation

- Data
- Linguistic insights
- Graph representation
- Inference algorithms
- Evaluations

Graph Representation

- Graph $\mathbf{G = (V, E)}$
  - $\mathbf{V = \{Pred\} \cup \{Arg\}}$
  - $\mathbf{E1}$: Pred – Arg

$$PMI(p,a) = \log_2 \frac{P(p,a)}{P(p)P(a)}$$



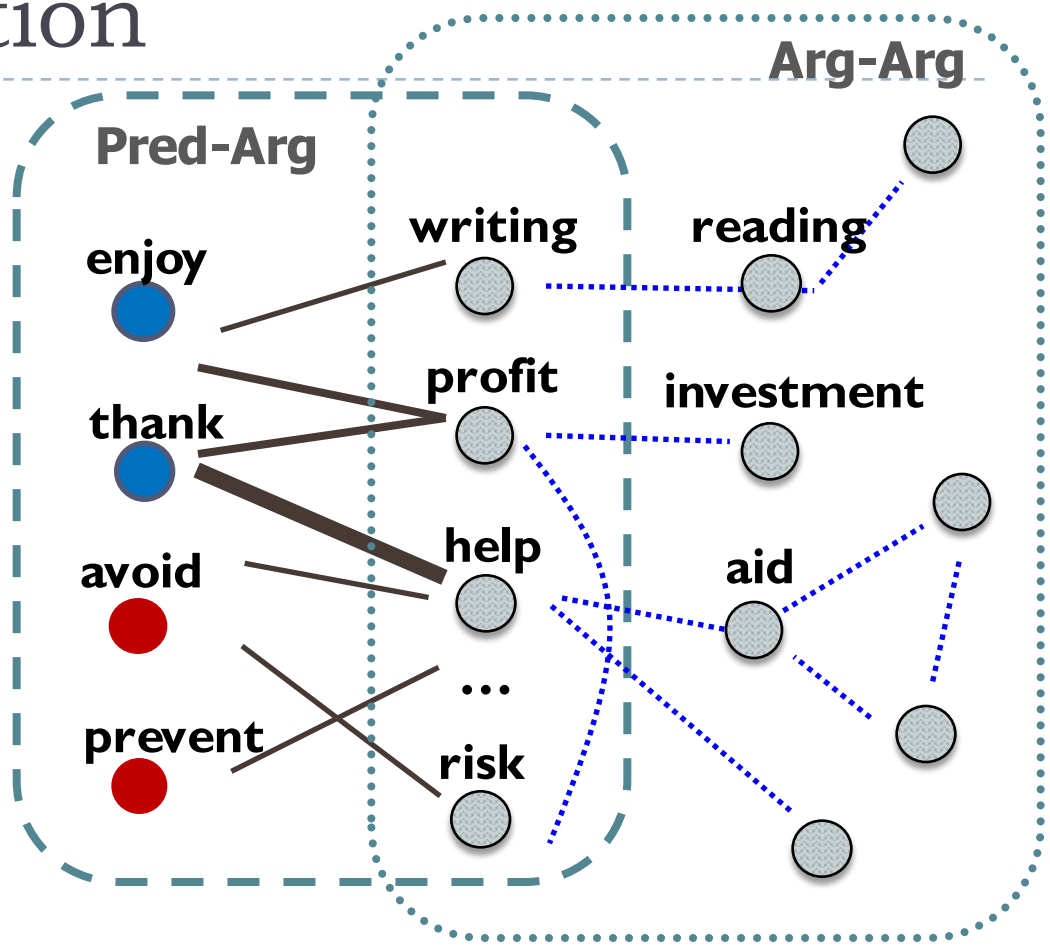
Graph Representation

- Graph **G = (V, E)**
  - **V = {Pred} ∪ {Arg}**
  - **E1:** Pred – Arg

$$PMI(p,a) = \log_2 \frac{P(p,a)}{P(p)P(a)}$$

  - **E2:** Arg – Arg

$$CosineSim(\overrightarrow{a_1}, \overrightarrow{a_2}) = \frac{\overrightarrow{a_1} \cdot \overrightarrow{a_2}}{||\overrightarrow{a_1}||\ ||\overrightarrow{a_2}||}$$



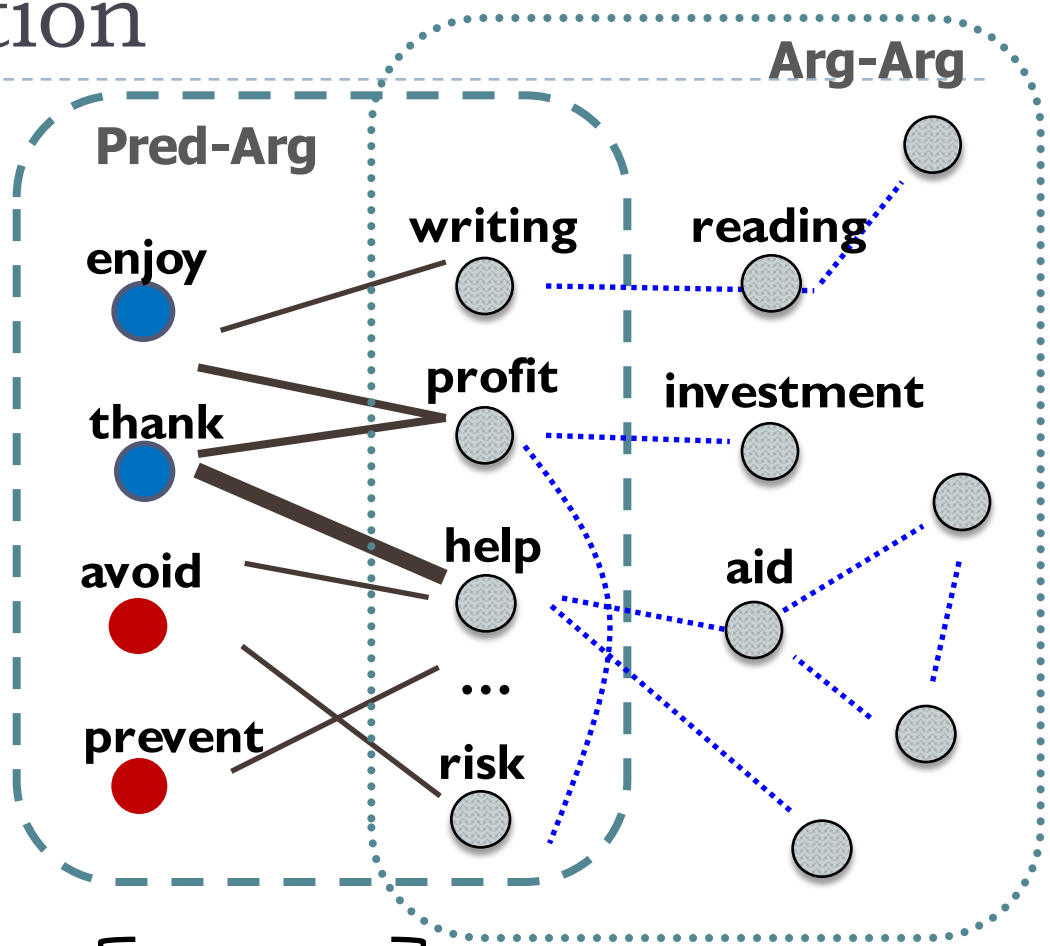
Graph Representation

- Graph **G = (V, E)**
  - **V = {Pred} ∪ {Arg}**
  - **E1: Pred – Arg**

$$PMI(p,a) = \log_2 \frac{P(p,a)}{P(p)P(a)}$$

  - **E2: Arg – Arg**

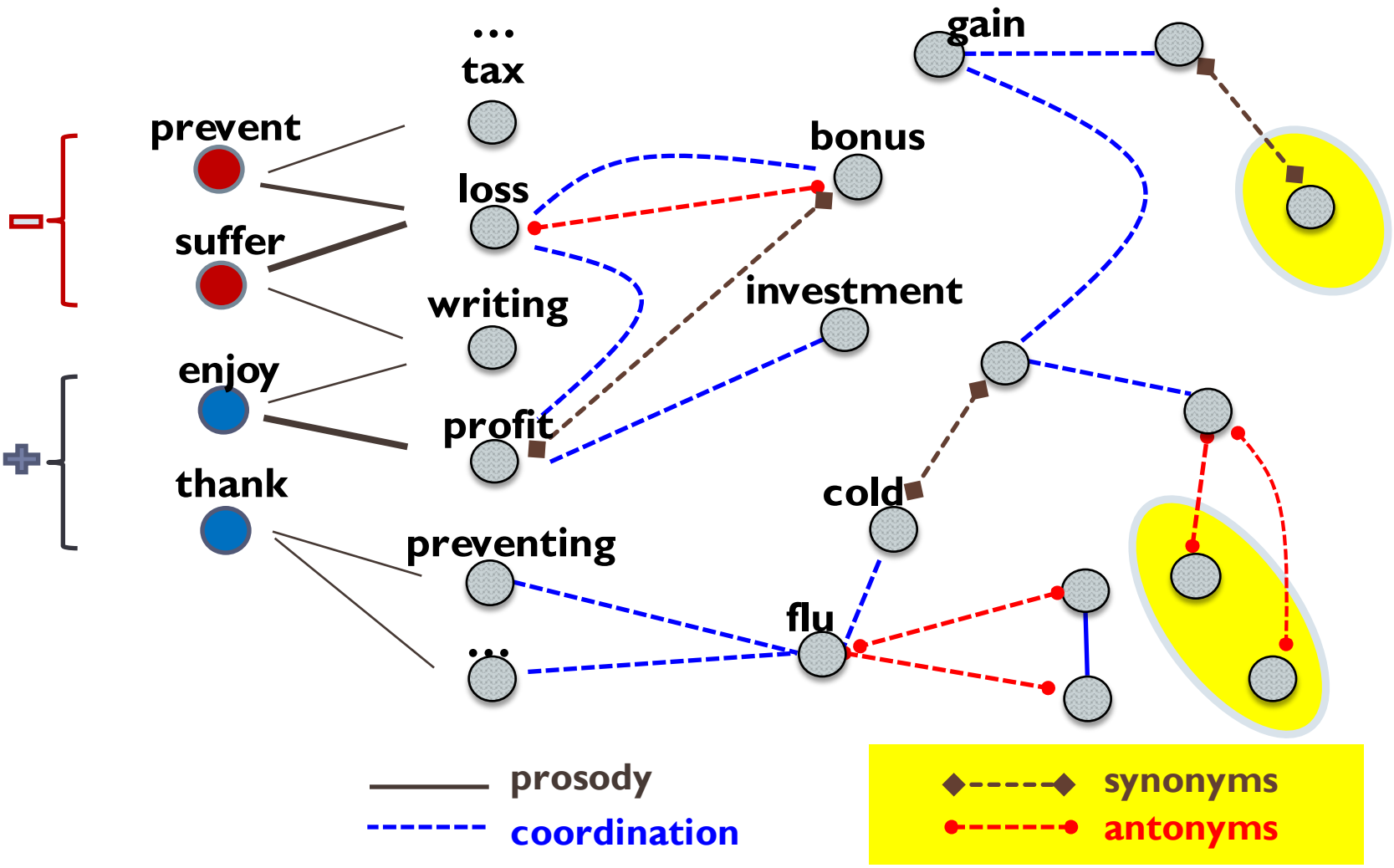
$$CosineSim(\vec{a_1}, \vec{a_2}) = \frac{\vec{a_1} \cdot \vec{a_2}}{||\vec{a_1}||\ ||\vec{a_2}||}$$



"a_1 and w_1" → PMI(a_1, w_1)
"a_1 and w_2" → PMI(a_1, w_2)
…
"a_1 and w_3" → PMI(a_1, w_n)

$$\vec{a_1} := \begin{bmatrix} PMI(a_1, w_1) \\ PMI(a_1, w_2) \\ \dots \\ PMI(a_1, w_n) \end{bmatrix}$$

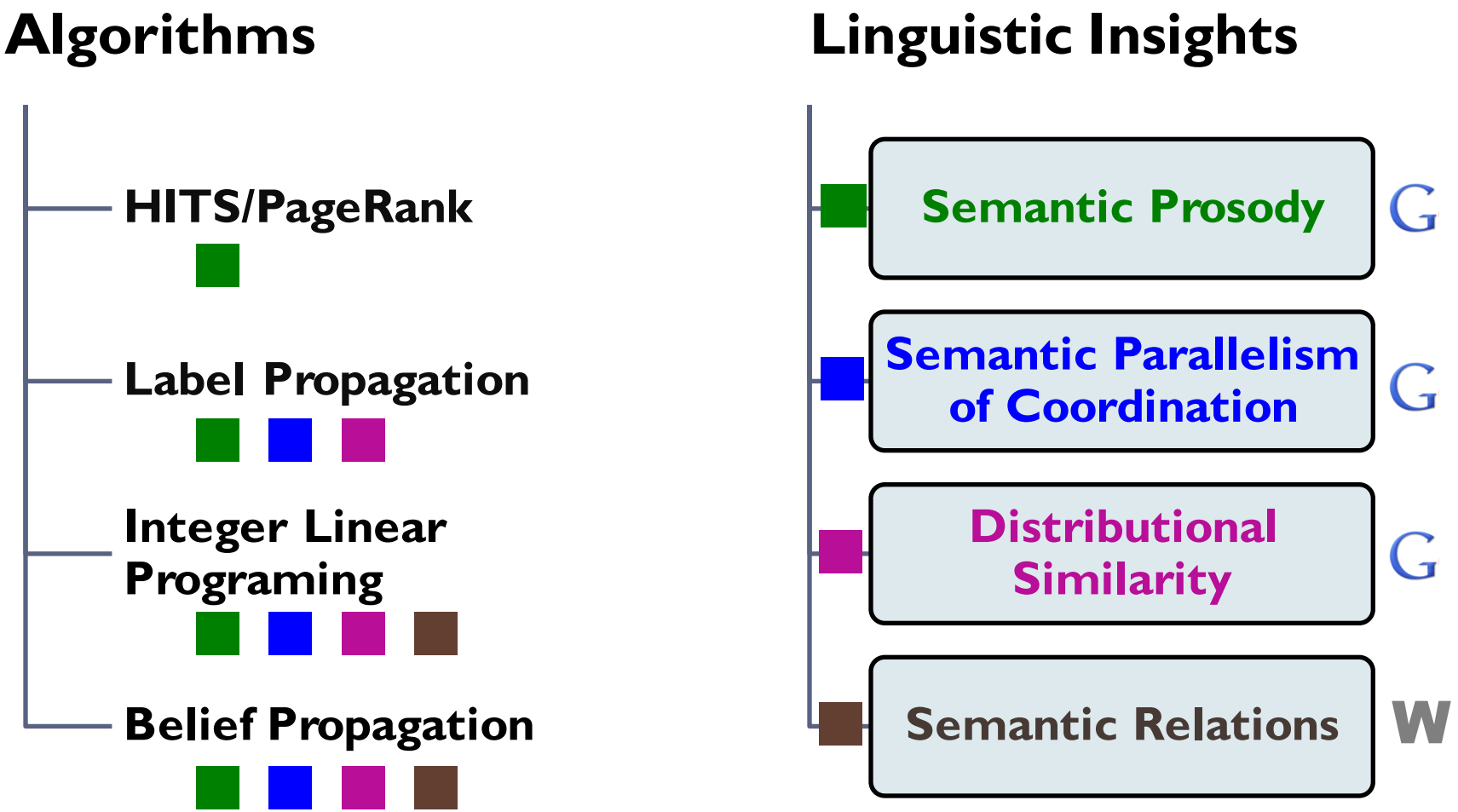
Graph Representation



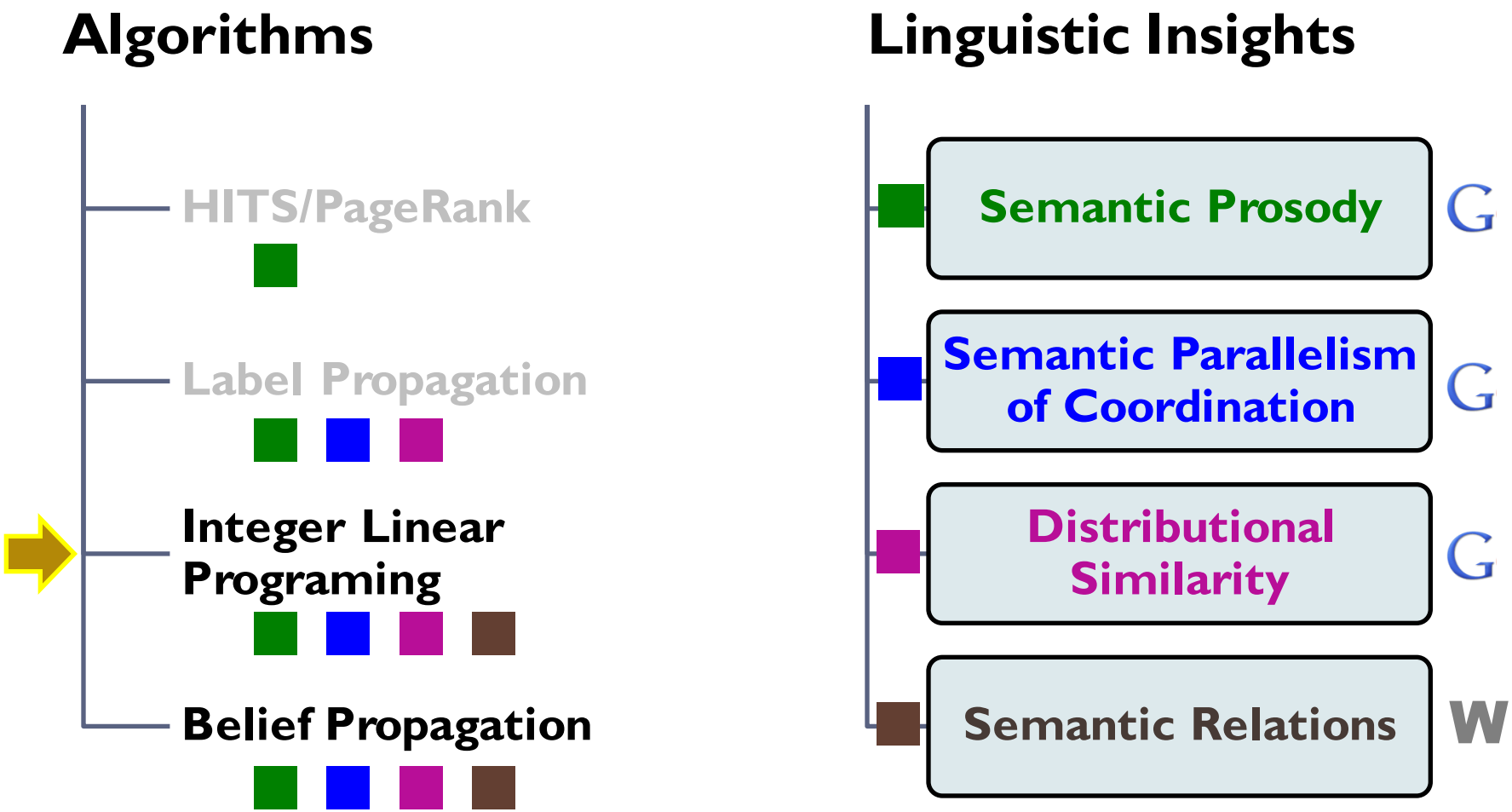
Learning the General Connotation

- Data
- Linguistic insights
- Graph representation
- Inference algorithms
- Evaluations

Inference Algorithm



Inference Algorithm



ILP: Problem Formulation

- For each unlabeled word i ,

  solve $x_i, y_i, z_i \in \{0, 1\}$,

  x_i → positive,

  y_i → negative,

  z_i → neutral;

  $x_i + y_i + z_i = 1$.

ILP: Problem Formulation

- For each unlabeled word i ,

  solve $x_i, y_i, z_i \in \{0, 1\}$,

  $x_i \rightarrow$ positive,

  $y_i \rightarrow$ negative,

  $z_i \rightarrow$ neutral;

  $x_i + y_i + z_i = 1.$

- Initialization (Hard constraints)
  - Positive seed predicates (e.g. "*achieve*") $\rightarrow x_i = 1$
  - Negative seed predicates (e.g. "*prevent*") $\rightarrow y_i = 1$

ILP: Objective Function

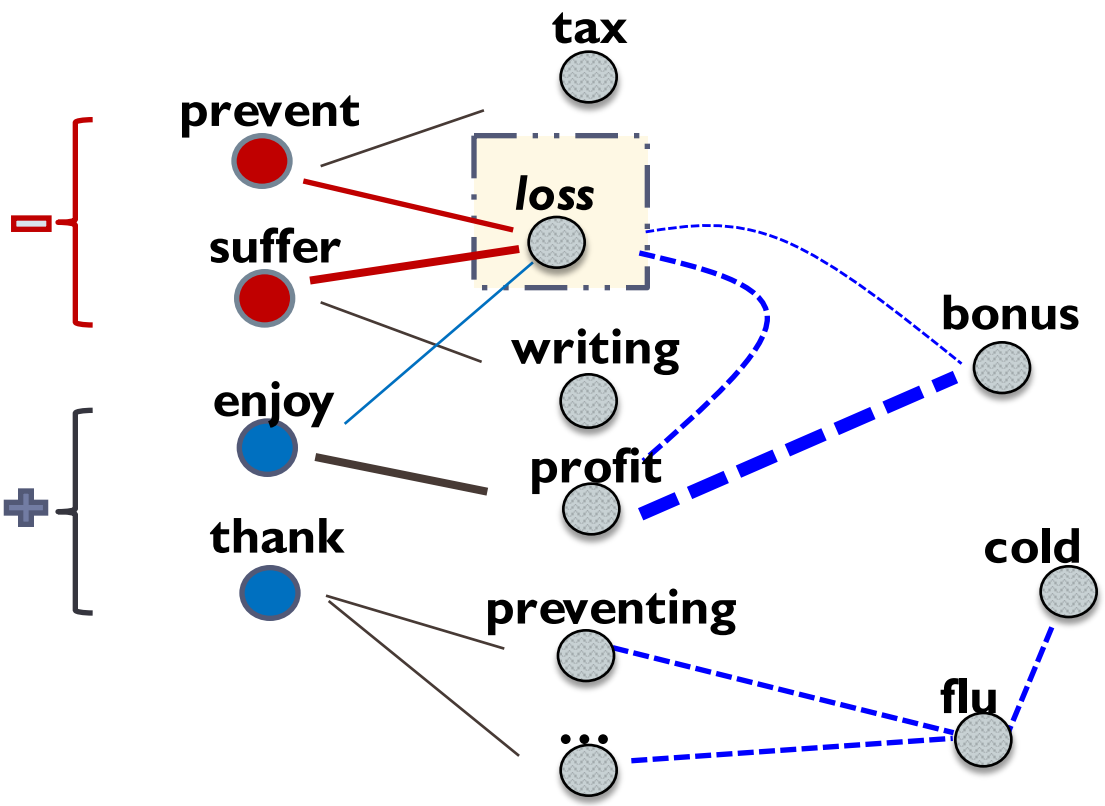
Maximize

$$F = \Phi^{prosody} + \Phi^{coord} + \Phi^{neu}$$

ILP: Objective Function

Maximize

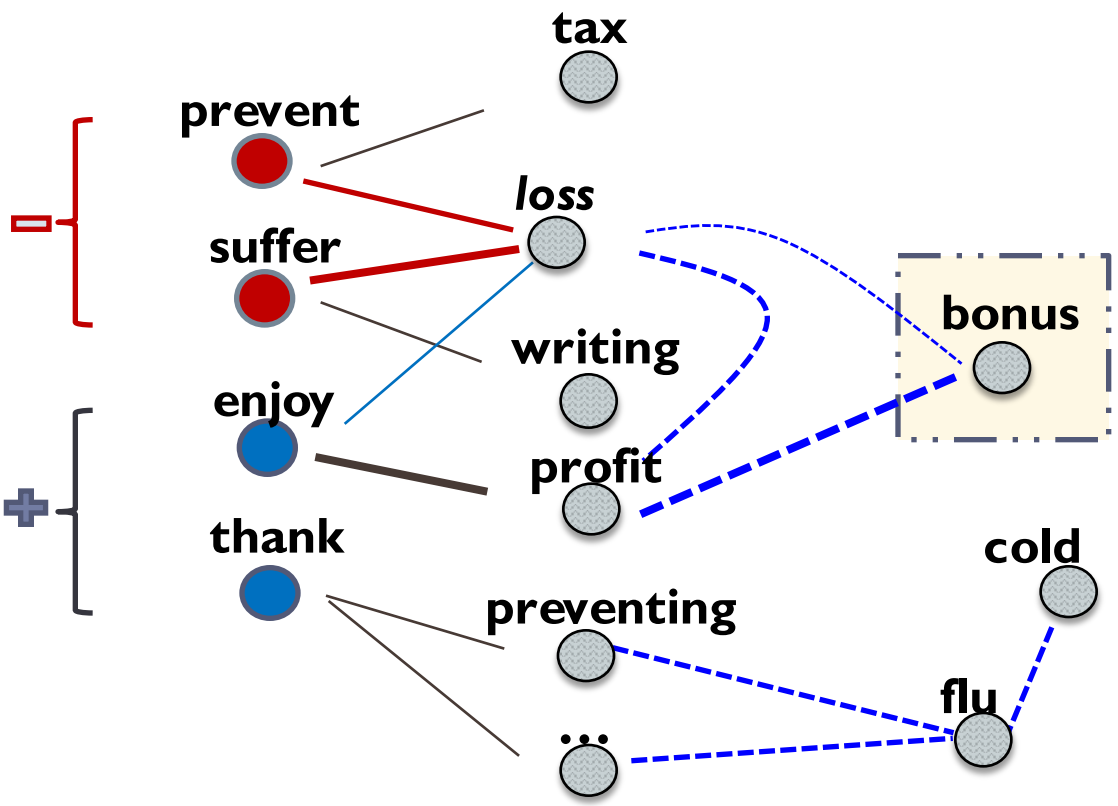
$$F = \Phi^{prosody} + \Phi^{coord} + \Phi^{neu}$$



ILP: Objective Function

Maximize

$$F = \Phi^{prosody} + \Phi^{coord} + \Phi^{neu}$$



ILP: Objective Function

$$\boxed{F = \Phi^{prosody} + \Phi^{coord} + \Phi^{neu}}$$

$$\Phi^{prosody} = \sum_{i,j}^{\mathcal{R}^{pred}} w_{i,j}^{pred} \left(d_{i,j}^{++} + d_{i,j}^{--} - d_{i,j}^{+-} - d_{i,j}^{-+}\right)$$

$$\Phi^{coord} = \sum_{i,j}^{\mathcal{R}^{coord}} w_{i,j}^{coord} \left(d_{i,j}^{++} + d_{i,j}^{--} + d_{i,j}^{00}\right)$$

$$\Phi^{neu} = \alpha \sum_{i,j}^{\mathcal{R}^{pred}} w_{i,j}^{pred} \cdot z_j$$

ILP: Objective Function

$$\boxed{F = \Phi^{prosody} + \Phi^{coord} + \Phi^{neu}}$$

$$\Phi^{prosody} = \sum_{i,j}^{\mathcal{R}^{pred}} w_{i,j}^{pred} \left(d_{i,j}^{++} + d_{i,j}^{--} - d_{i,j}^{+-} - d_{i,j}^{-+}\right)$$

$$\Phi^{coord} = \sum_{i,j}^{\mathcal{R}^{coord}} w_{i,j}^{coord} \left(d_{i,j}^{++} + d_{i,j}^{--} + d_{i,j}^{00}\right)$$

$$\Phi^{neu} = \alpha \sum_{i,j}^{\mathcal{R}^{pred}} w_{i,j}^{pred} \cdot z_j$$

ILP: Soft Constraints

- Predicate – Argument

$$w^{pred}(p,a) = \frac{freq(p,a)}{\sum_{(p,x)\in\mathcal{R}^{pred}} freq(p,x)}$$

- Argument – Argument

$$w^{coord}(a_1,a_2) = CosSim(\overrightarrow{a_1},\overrightarrow{a_2}) = \frac{\overrightarrow{a_1}\cdot\overrightarrow{a_2}}{||\overrightarrow{a_1}||\,||\overrightarrow{a_2}||}$$

$$\Phi^{prosody} = \sum_{i,j}^{\mathcal{R}^{pred}} w_{i,j}^{pred}(d_{i,j}^{++} + d_{i,j}^{--} - d_{i,j}^{+-} - d_{i,j}^{-+})$$

$$\Phi^{coord} = \sum_{i,j} w_{i,j}^{coord}(d_{i,j}^{++} + d_{i,j}^{--} + d_{i,j}^{00})$$

ILP: Hard Constraints

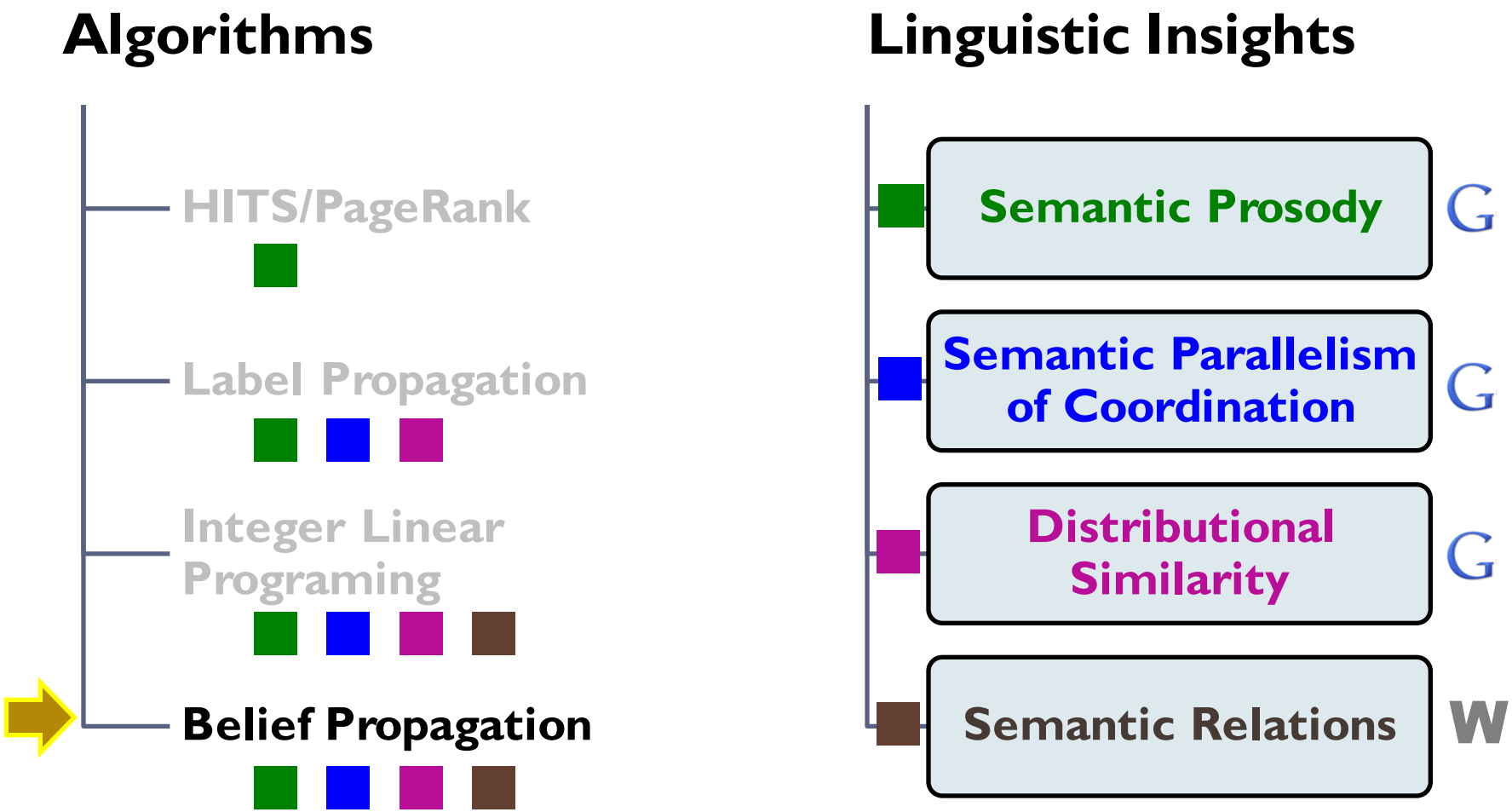
- Semantic relations
  - Antonym pairs will not have the same positive or negative polarity.

$$\forall (i,j) \in \mathcal{R}^{ant},\ x_i + x_j \leq 1,\ y_i + y_j \leq 1$$

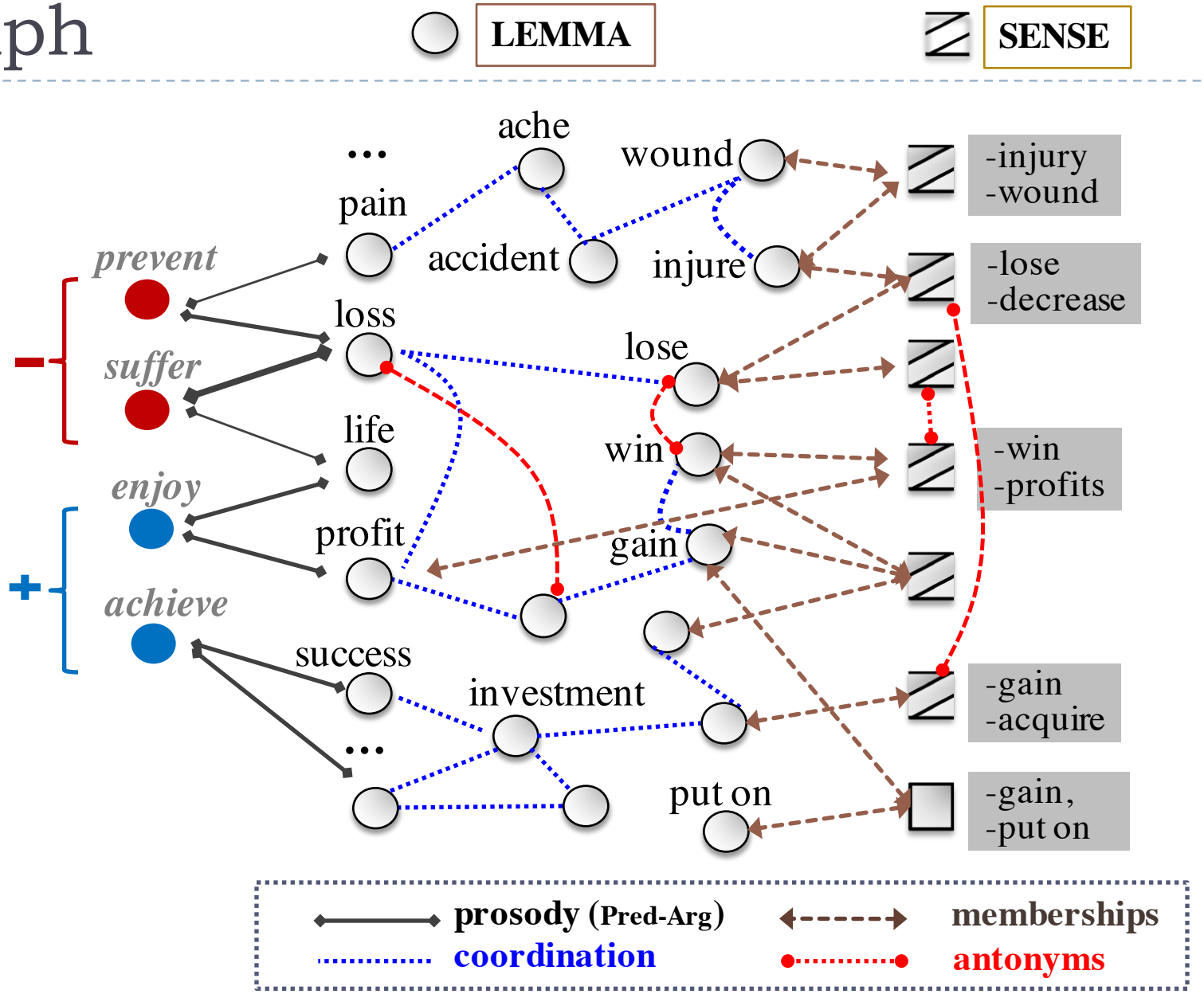
  - Synonym pairs will not have the opposite polarity.

$$\forall (i,j) \in \mathcal{R}^{syn},\ x_i + y_j \leq 1,\ x_j + y_i \leq 1$$

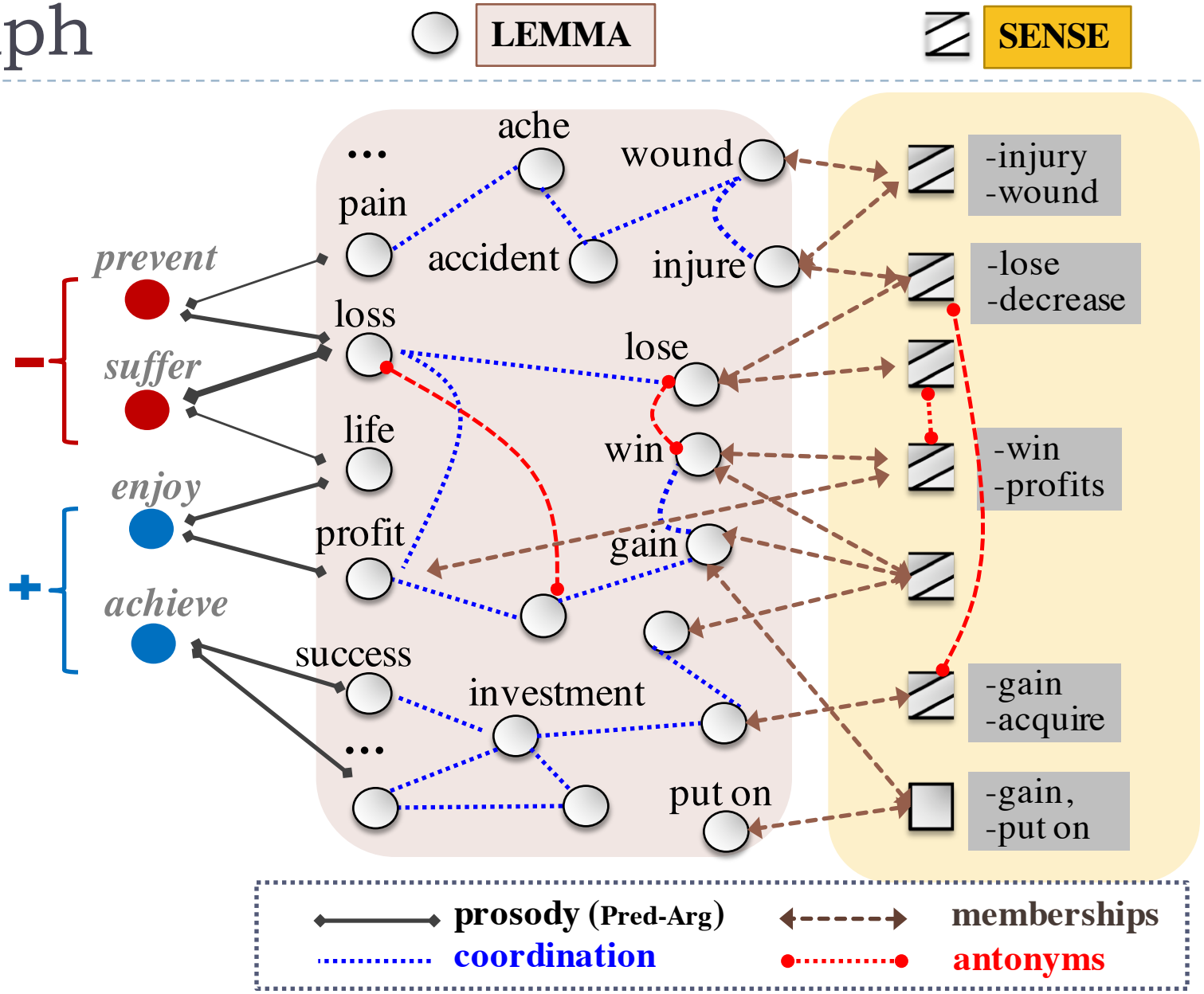
Inference Algorithm



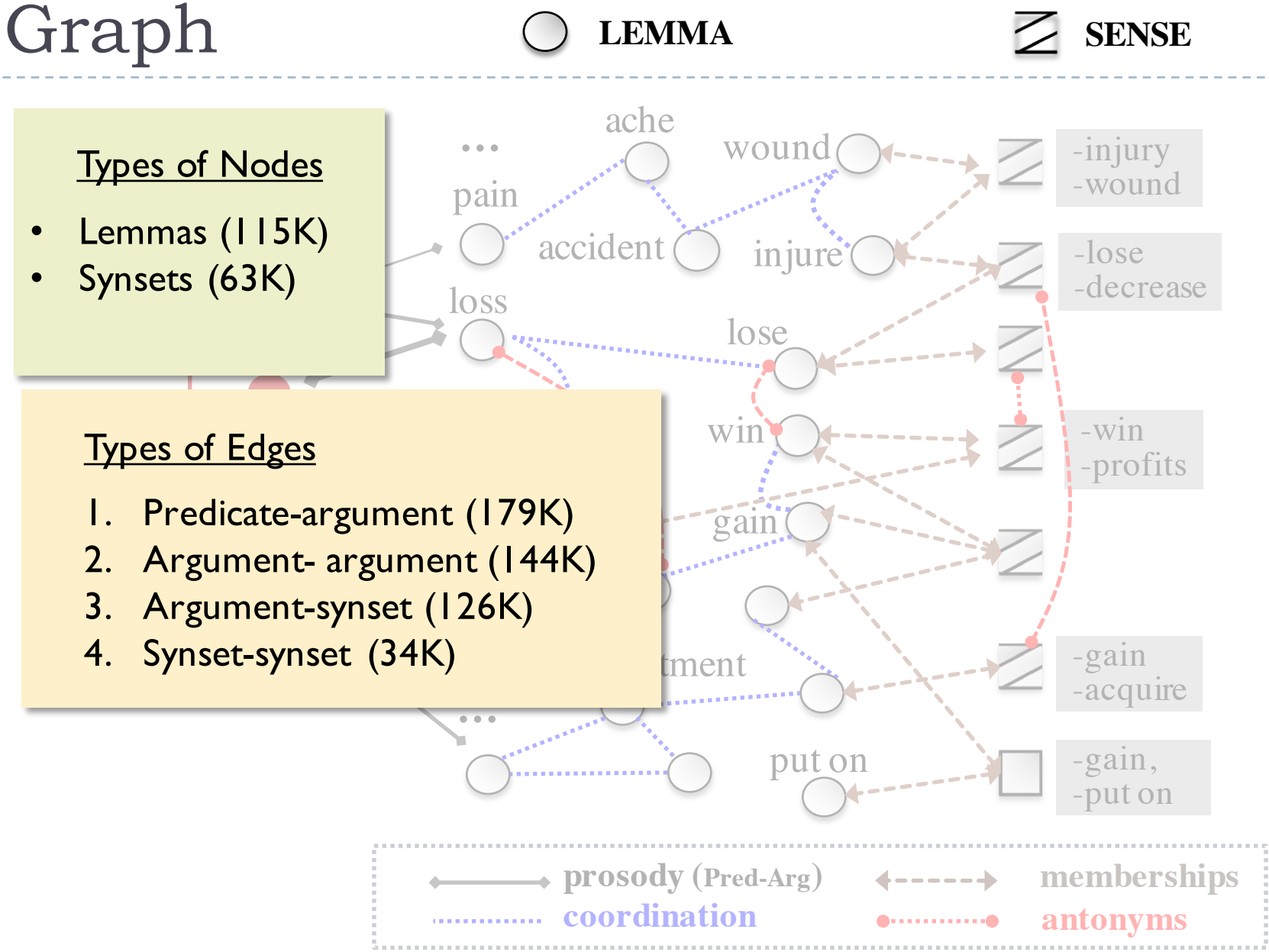
Graph


LEMMA
SENSE
ache
wound
pain
accident
injure
prevent
loss
lose
suffer
life
win
enjoy
profit
gain
achieve
success
investment
put on
-injury
-wound
-lose
-decrease
-win
-profits
-gain
-acquire
-gain,
-put on
prosody (Pred-Arg)
memberships
coordination
antonyms

Graph


LEMMA
SENSE
ache
wound
pain
accident
injure
prevent
loss
lose
suffer
life
win
enjoy
profit
gain
achieve
success
investment
put on
-injury
-wound
-lose
-decrease
-win
-profits
-gain
-acquire
-gain,
-put on
prosody (Pred-Arg)
coordination
memberships
antonyms

Graph


LEMMA
SENSE
Types of Nodes
• Lemmas (115K)
• Synsets (63K)
Types of Edges
1. Predicate-argument (179K)
2. Argument- argument (144K)
3. Argument-synset (126K)
4. Synset-synset (34K)
ache
wound
pain
accident
injure
loss
lose
win
gain
put on
-injury
-wound
-lose
-decrease
-win
-profits
-gain
-acquire
-gain,
-put on
prosody (Pred-Arg)
memberships
coordination
antonyms

Problem Formulation

- $G^{\text{Lemma + Sense}} = (V, E)$
- Nodes (random variables)
  - $V = \{v_1, v_2, \ldots, v_n\}$
  - Unobserved variables: $\Upsilon = \{Y_1, Y_2, ..., Y_n\}, y_i$
- Typed edges
  - $E = \{e(v_i, v_j, v_k)\}$ where $v_i, v_j \in V$, $T_k \in T$
  - $T = \{\text{pre-arg}, \text{arg-arg}, \text{arg-syn}, \text{syn-syn}\}$
- Neighborhood function
  - $N_v = \{u \mid e(u, v) \in E\}$
- Labels
  - $L = \{+, -\}$, y_i denotes the label of Y_i.

Pairwise Markov Random Fields

Objective Function

$$P(y \mid x) = \frac{1}{Z(x)} \prod_{Yi \in \Upsilon} \psi_i(y_i) \prod \psi_{ij}^t(y_i, y_j)$$

Objective Function

An assignment to all the unobserved variables

$$P(y \mid x) = \frac{1}{Z(x)} \prod_{Yi \in \Upsilon} \psi_i(y_i) \prod \psi_{ij}^t(y_i, y_j)$$

Objective Function

An assignment to all the unobserved variables

$$P(y \mid x) = \frac{1}{Z(x)} \prod_{Yi \in \Upsilon} \psi_i(y_i) \prod \psi_{ij}^t(y_i, y_j)$$

Variables with known labels

Objective Function

$$P(y \mid x) = \frac{1}{Z(x)} \prod_{Yi \in \Upsilon} \psi_i(y_i) \prod \psi_{ij}^{t}(y_i, y_j)$$

Prior mapping:
y_i refers to Y_i's label
ψ_i is prior mapping.

Objective Function

$$P(y \mid x) = \frac{1}{Z(x)} \prod_{Yi \in \Upsilon} \psi_i(y_i) \prod \psi_{ij}^t(y_i, y_j)$$

Prior mapping:
$L \rightarrow \mathbb{R}_{\geq 0}$

Compatibility mapping:
$L \times L \rightarrow \mathbb{R}_{\geq 0}$

Objective Function

An assignment to all the unobserved variables

Loopy Belief Propagation

- Message passing

$$m_{i\to j}(y_i) = \alpha \sum_{y_i \in L} (\psi_{ij}^t(y_i, y_j)\psi_i(y_i) \prod_{Y_k \in Ni \cap \Upsilon \setminus Y_i} m_{k\to i}(y_i)), \forall y_j \in L$$

- Belief

$$b_i(y_i) = \beta \psi_i(y_i) \prod_{Y_j \in Ni \cap \Upsilon} m_{j\to i}(y_i), \forall y_i \in L$$

Loopy Belief Propagation

- Initialize "message" between all node pairs connected by an edge.
- Initialize priors for all nodes
- Integrative message passing until all messages stop changing

$$m_{i\to j}(y_i) = \alpha \sum_{y_i \in L} (\psi_{ij}^t(y_i, y_j)\psi_i(y_i) \prod_{Y_k \in Ni \cap \Upsilon \setminus Y_i} m_{k\to i}(y_i)), \forall y_j \in L$$

- Compute beliefs.

$$b_i(y_i) = \beta \psi_i(y_i) \prod_{Y_j \in Ni \cap \Upsilon} m_{j\to i}(y_i), \forall y_i \in L$$

- Assign the label

$$L_i \leftarrow \max_i b_i(y_i)$$