

λ -calcolo per costruire rappresentazioni semantiche

Lorenzo Ferrone

July 8, 2013

Outline

- 1 Introduzione
- 2 Il λ -calcolo per la semantica
 - Rappresentazioni semantiche
 - λ -calcolo
 - Lessico
 - Ambiguità
 - Conclusioni

Alonzo Church, 1936

- Inventato da *Alonzo Church* nel 1936 come fondamento del concetto di *computabilità*
- Si basa sui due concetti di *astrazione* e di *applicazione*

Astrazione: $\lambda x.t$

$\lambda x.t$ è la definizione di una funzione che mappa x in t
Per esempio la scrittura $\lambda x.x$ indica la funzione $f(x) = x$ cioè
nient'altro che la funzione identica

Applicazione: $(t s)$

$(t s)$ è l'applicazione della funzione t all'argomento s
Ad esempio $(\lambda x.x 5)$ è la funzione identica applicata al valore "5"

Alonzo Church, 1936

- Inventato da *Alonzo Church* nel 1936 come fondamento del concetto di *computabilità*
- Si basa sui due concetti di *astrazione* e di *applicazione*

Astrazione: $\lambda x.t$

$\lambda x.t$ è la definizione di una funzione che mappa x in t
Per esempio la scrittura $\lambda x.x$ indica la funzione $f(x) = x$ cioè
nient'altro che la funzione identica

Applicazione: $(t s)$

$(t s)$ è l'applicazione della funzione t all'argomento s
Ad esempio $(\lambda x.x 5)$ è la funzione identica applicata al valore "5"

Alonzo Church, 1936

- Inventato da *Alonzo Church* nel 1936 come fondamento del concetto di *computabilità*
- Si basa sui due concetti di *astrazione* e di *applicazione*

Astrazione: $\lambda x.t$

$\lambda x.t$ è la definizione di una funzione che mappa x in t
Per esempio la scrittura $\lambda x.x$ indica la funzione $f(x) = x$ cioè
nient'altro che la funzione identica

Applicazione: $(t s)$

$(t s)$ è l'applicazione della funzione t all'argomento s
Ad esempio $(\lambda x.x 5)$ è la funzione identica applicata al valore "5"

Riduzione

β -riduzione

- Il processo di applicazione di una funzione ad un argomento si chiama *β -riduzione*
 - 1 Si parte dall'espressione $(\lambda x.t N)$
 - 2 Leviamo il prefisso iniziale λx .
 - 3 Sostituiamo in t tutte le occorrenze di x con N
 - 4 Continuiamo fino a che non possiamo più fare ulteriori riduzioni

Riduzione

β -riduzione

- Il processo di applicazione di una funzione ad un argomento si chiama *β -riduzione*
 - 1 Si parte dall'espressione $(\lambda x.t N)$
 - 2 Leviamo il prefisso iniziale λx .
 - 3 Sostituiamo in t tutte le occorrenze di x con N
 - 4 Continuiamo fino a che non possiamo più fare ulteriori riduzioni

Riduzione

β -riduzione

- Il processo di applicazione di una funzione ad un argomento si chiama *β -riduzione*
 - 1 Si parte dall'espressione $(\lambda x.t N)$
 - 2 Leviamo il prefisso iniziale λx .
 - 3 Sostituiamo in t tutte le occorrenze di x con N
 - 4 Continuiamo fino a che non possiamo più fare ulteriori riduzioni

Riduzione

β -riduzione

- Il processo di applicazione di una funzione ad un argomento si chiama *β -riduzione*
 - 1 Si parte dall'espressione $(\lambda x.t N)$
 - 2 Leviamo il prefisso iniziale λx .
 - 3 Sostituiamo in t tutte le occorrenze di x con N
 - 4 Continuiamo fino a che non possiamo più fare ulteriori riduzioni

Riduzione

β -riduzione

- Il processo di applicazione di una funzione ad un argomento si chiama *β -riduzione*
 - 1 Si parte dall'espressione $(\lambda x.t N)$
 - 2 Leviamo il prefisso iniziale λx .
 - 3 Sostituiamo in t tutte le occorrenze di x con N
 - 4 Continuiamo fino a che non possiamo più fare ulteriori riduzioni

Esempio

① Espressione di partenza:

- $(\lambda x.x\ 5)$

② Leviamo λx .

- $x\ 5$

③ Sostituiamo "5" a x :

- 5

④ Non possiamo fare altro quindi "5" è il risultato finale

Esempio

- 1 Espressione di partenza:
 - $(\lambda x.x\ 5)$
- 2 Leviamo λx .
 - $x\ 5$
- 3 Sostituiamo "5" a x :
 - 5
- 4 Non possiamo fare altro quindi "5" è il risultato finale

Esempio

- 1 Espressione di partenza:
 - $(\lambda x.x\ 5)$
- 2 Leviamo λx .
 - $x\ 5$
- 3 Sostituiamo "5" a x :
 - 5
- 4 Non possiamo fare altro quindi "5" è il risultato finale

Esempio

- 1 Espressione di partenza:
 - $(\lambda x.x\ 5)$
- 2 Leviamo λx .
 - $x\ 5$
- 3 Sostituiamo "5" a x :
 - 5
- 4 Non possiamo fare altro quindi "5" è il risultato finale

Outline

- 1 Introduzione
- 2 Il λ -calcolo per la semantica
 - Rappresentazioni semantiche
 - λ -calcolo
 - Lessico
 - Ambiguità
 - Conclusioni

Rappresentazioni semantiche

Rappresentazione semantica

Data una frase restituirne una rappresentazione simbolica del significato

Logica del primo ordine (FOL)

- *Mario corre* \rightarrow $\text{corre}(\text{Mario})$
- *Giovanni ama Maria* \rightarrow $\text{ama}(\text{Giovanni}, \text{Maria})$
- *Ogni uomo ama Maria* \rightarrow $\forall x : (\text{uomo}(x) \Rightarrow \text{ama}(x, \text{Maria}))$

Noi useremo:

- FOL per la rappresentazione finale
- λ -calcolo per gli stadi intermedi

Rappresentazioni semantiche

Rappresentazione semantica

Data una frase restituirne una rappresentazione simbolica del significato

Logica del primo ordine (FOL)

- *Mario corre* \rightarrow $\text{corre}(\text{Mario})$
- *Giovanni ama Maria* \rightarrow $\text{ama}(\text{Giovanni}, \text{Maria})$
- *Ogni uomo ama Maria* \rightarrow $\forall x : (\text{uomo}(x) \Rightarrow \text{ama}(x, \text{Maria}))$

Noi useremo:

- FOL per la rappresentazione finale
- λ -calcolo per gli stadi intermedi

Rappresentazioni semantiche

Rappresentazione semantica

Data una frase restituirne una rappresentazione simbolica del significato

Logica del primo ordine (FOL)

- *Mario corre* \rightarrow $\text{corre}(\text{Mario})$
- *Giovanni ama Maria* \rightarrow $\text{ama}(\text{Giovanni}, \text{Maria})$
- *Ogni uomo ama Maria* \rightarrow $\forall x : (\text{uomo}(x) \Rightarrow \text{ama}(x, \text{Maria}))$

Noi useremo:

- FOL per la rappresentazione finale
- λ -calcolo per gli stadi intermedi

Logica del primo ordine

- È composta da:

Simboli

- Costanti: Gianni, Maria, ...
- Predicati e relazioni: cammina, corre, ama, ...
- Variabili: x, y, \dots
- Connettivi logici: $\wedge, \vee, \Rightarrow, \neg$
- *Quantificatori*: \forall, \exists

Formule

- Atomiche: $\text{ama}(\text{Gianni}, \text{Maria})$
- Due formule unite: $\text{ama}(\text{Gianni}, \text{Maria}) \wedge \text{ama}(\text{Maria}, \text{Gianni})$
- Quantificate: $\exists x : \text{uomo}(x)$

Logica del primo ordine

- È composta da:

Simboli

- Costanti: Gianni, Maria, ...
- Predicati e relazioni: cammina, corre, ama, ...
- Variabili: x, y, \dots
- Connettivi logici: $\wedge, \vee, \Rightarrow, \neg$
- **Quantificatori**: \forall, \exists

Formule

- Atomiche: `ama(Gianni, Maria)`
- Due formule unite: `ama(Gianni, Maria) \wedge ama(Maria, Gianni)`
- Quantificate: `$\exists x : \text{uomo}(x)$`

Logica del primo ordine

- È composta da:

Simboli

- Costanti: Gianni, Maria, ...
- Predicati e relazioni: cammina, corre, ama, ...
- Variabili: x, y, \dots
- Connettivi logici: $\wedge, \vee, \Rightarrow, \neg$
- **Quantificatori**: \forall, \exists

Formule

- Atomiche: $\text{ama}(\text{Gianni}, \text{Maria})$
- Due formule unite: $\text{ama}(\text{Gianni}, \text{Maria}) \wedge \text{ama}(\text{Maria}, \text{Gianni})$
- Quantificate: $\exists x : \text{uomo}(x)$

- Come costruisco queste rappresentazioni semantiche a partire dalla nostra frase?
 - Aumentiamo ogni parola del nostro lessico con una parte semantica (una formula FOL)
 - Componiamo seguendo la struttura sintattica della frase, partendo dal basso

- Come costruisco queste rappresentazioni semantiche a partire dalla nostra frase?
- Aumentiamo ogni parola del nostro lessico con una parte semantica (una formula FOL)
- Componiamo seguendo la struttura sintattica della frase, partendo dal basso

- Come costruisco queste rappresentazioni semantiche a partire dalla nostra frase?
- Aumentiamo ogni parola del nostro lessico con una parte semantica (una formula FOL)
- Componiamo seguendo la struttura sintattica della frase, partendo dal basso

Struttura sintattica

- Consideriamo la frase *Gianni ama Maria*
- Poniamo:
 - *Gianni* \rightarrow Gianni
 - *ama* \rightarrow ama($?, ?$)
 - *Maria* \rightarrow Maria

Struttura sintattica

- Consideriamo la frase *Gianni ama Maria*
- Poniamo:
 - *Gianni* \rightarrow Gianni
 - *ama* \rightarrow ama(?,?)
 - *Maria* \rightarrow Maria

Struttura sintattica

- Consideriamo la frase *Gianni ama Maria*
- Poniamo:
 - *Gianni* \rightarrow Gianni
 - *ama* \rightarrow ama(?,?)
 - *Maria* \rightarrow Maria

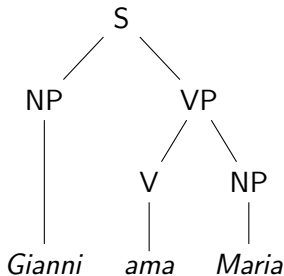
Struttura sintattica

- Consideriamo la frase *Gianni ama Maria*
- Poniamo:
 - *Gianni* \rightarrow Gianni
 - *ama* \rightarrow ama(?,?)
 - *Maria* \rightarrow Maria

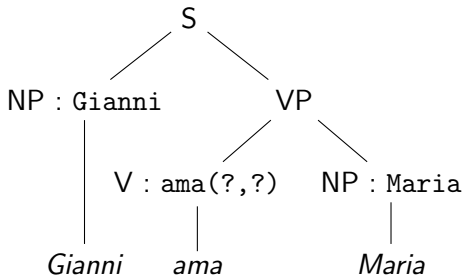
Struttura sintattica

- Consideriamo la frase *Gianni ama Maria*
- Poniamo:
 - *Gianni* \rightarrow Gianni
 - *ama* \rightarrow ama(?,?)
 - *Maria* \rightarrow Maria

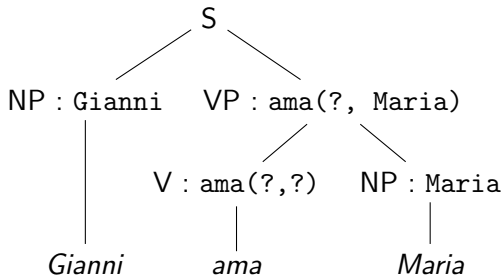
Struttura sintattica



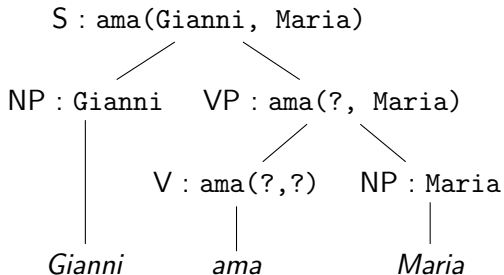
Parte semantica



Composizione



Risultato finale



Limitazioni della FOL

Problemi

- 1 Come possiamo costruire in modo sistematico VP?
 - è $\text{ama}(?,\text{Maria})$ o $\text{ama}(\text{Maria},?)$?
 - Come facciamo a specificare come combinare i vari pezzi?
- 2 Come rappresentiamo gli elementi intermedi?
 - $\text{ama}(?,\text{Maria})$: *non* è una formula nella logica del prim'ordine!
 - $\text{ama}(x,\text{Maria})$: *non* abbiamo controllo sulla variabile libera x

Limitazioni della FOL

Problemi

- 1 Come possiamo costruire in modo sistematico VP?
 - è $\text{ama}(?,\text{Maria})$ o $\text{ama}(\text{Maria},?)$?
 - Come facciamo a specificare come combinare i vari pezzi?
- 2 Come rappresentiamo gli elementi intermedi?
 - $\text{ama}(?,\text{Maria})$: *non* è una formula nella logica del prim'ordine!
 - $\text{ama}(x,\text{Maria})$: *non* abbiamo controllo sulla variabile libera x

Limitazioni della FOL

Problemi

- 1 Come possiamo costruire in modo sistematico VP?
 - è $\text{ama}(?, \text{Maria})$ o $\text{ama}(\text{Maria}, ?)$?
 - Come facciamo a specificare come combinare i vari pezzi?
- 2 Come rappresentiamo gli elementi intermedi?
 - $\text{ama}(?, \text{Maria})$: *non* è una formula nella logica del prim'ordine!
 - $\text{ama}(x, \text{Maria})$: *non* abbiamo controllo sulla variabile libera x

Limitazioni della FOL

Problemi

- 1 Come possiamo costruire in modo sistematico VP?
 - è $\text{ama}(?, \text{Maria})$ o $\text{ama}(\text{Maria}, ?)$?
 - Come facciamo a specificare come combinare i vari pezzi?
- 2 Come rappresentiamo gli elementi intermedi?
 - $\text{ama}(?, \text{Maria})$: *non* è una formula nella logica del prim'ordine!
 - $\text{ama}(x, \text{Maria})$: *non* abbiamo controllo sulla variabile libera x
 - Dovremmo avere qualcosa tipo: $\exists x : \text{ama}(x, \text{Maria})$

Limitazioni della FOL

Problemi

- 1 Come possiamo costruire in modo sistematico VP?
 - è $\text{ama}(?, \text{Maria})$ o $\text{ama}(\text{Maria}, ?)$?
 - Come facciamo a specificare come combinare i vari pezzi?
- 2 Come rappresentiamo gli elementi intermedi?
 - $\text{ama}(?, \text{Maria})$: **non è** una formula nella logica del prim'ordine!
 - $\text{ama}(x, \text{Maria})$: **non** abbiamo controllo sulla variabile libera x
 - Dovremmo avere qualcosa tipo: $\exists x : \text{ama}(x, \text{Maria})$

Limitazioni della FOL

Problemi

- 1 Come possiamo costruire in modo sistematico VP?
 - è $\text{ama}(?, \text{Maria})$ o $\text{ama}(\text{Maria}, ?)$?
 - Come facciamo a specificare come combinare i vari pezzi?
- 2 Come rappresentiamo gli elementi intermedi?
 - $\text{ama}(?, \text{Maria})$: **non è** una formula nella logica del prim'ordine!
 - $\text{ama}(x, \text{Maria})$: **non** abbiamo controllo sulla variabile libera x
 - Dovremmo avere qualcosa tipo: $\exists x : \text{ama}(x, \text{Maria})$

Limitazioni della FOL

Problemi

- 1 Come possiamo costruire in modo sistematico VP?
 - è $\text{ama}(?, \text{Maria})$ o $\text{ama}(\text{Maria}, ?)$?
 - Come facciamo a specificare come combinare i vari pezzi?
- 2 Come rappresentiamo gli elementi intermedi?
 - $\text{ama}(?, \text{Maria})$: **non** è una formula nella logica del prim'ordine!
 - $\text{ama}(x, \text{Maria})$: **non** abbiamo controllo sulla variabile libera x
 - Dovremmo avere qualcosa tipo: $\exists x : \text{ama}(x, \text{Maria})$

Outline

- 1 Introduzione
- 2 Il λ -calcolo per la semantica
 - Rappresentazioni semantiche
 - λ -calcolo
 - Lessico
 - Ambiguità
 - Conclusioni

Il λ -calcolo per la semantica

Soluzione

Usiamo il λ -calcolo per legare le variabili libere.

Definizione:

Una λ -espressione è una formula in cui una variabile è quantificata dall'operatore λ .

Aumentiamo il nostro lessico con λ -espressioni invece che semplici formule del prim'ordine.

Il λ -calcolo per la semantica

Soluzione

Usiamo il λ -calcolo per legare le variabili libere.

Definizione:

Una λ -espressione è una formula in cui una variabile è quantificata dall'operatore λ .

Aumentiamo il nostro lessico con λ -espressioni invece che semplici formule del prim'ordine.

Il λ -calcolo per la semantica

Soluzione

Usiamo il λ -calcolo per legare le variabili libere.

Definizione:

Una λ -espressione è una formula in cui una variabile è quantificata dall'operatore λ .

Aumentiamo il nostro lessico con λ -espressioni invece che semplici formule del prim'ordine.

Composizionalità

- Questo risolve entrambi i nostri problemi!
 - $\lambda x.ama(x, Maria)$ è una formula valida in questo linguaggio
 - Componiamo i costituenti semplicemente applicando la funzione agli argomenti (β -riduzione)

- $\lambda x.ama(x, Maria) (Gianni) \xrightarrow{\beta} ama(Gianni, Maria)$

Composizionalità

- Questo risolve entrambi i nostri problemi!
 - $\lambda x.ama(x, Maria)$ è una formula valida in questo linguaggio
 - Componiamo i costituenti semplicemente applicando la funzione agli argomenti (β -riduzione)

- $\lambda x.ama(x, Maria) (Gianni) \xrightarrow{\beta} ama(Gianni, Maria)$

Composizionalità

- Questo risolve entrambi i nostri problemi!
 - $\lambda x.ama(x, Maria)$ è una formula valida in questo linguaggio
 - Componiamo i costituenti semplicemente applicando la funzione agli argomenti (β -riduzione)

- $\lambda x.ama(x, Maria) (Gianni) \xrightarrow{\beta} ama(Gianni, Maria)$

Composizionalità

- Questo risolve entrambi i nostri problemi!
 - $\lambda x.ama(x, Maria)$ è una formula valida in questo linguaggio
 - Componiamo i costituenti semplicemente applicando la funzione agli argomenti (β -riduzione)

- $\lambda x.ama(x, Maria) (Gianni) \xrightarrow{\beta} ama(Gianni, Maria)$

Composizionalità

- Non è così semplice...

- E se avessimo *Mario corre?*

- *Mario* \rightarrow Mario

- *Corre* \rightarrow λx .corre(x) (questo è un modo di rappresentare la funzione λ -calcolo)

- *Corre* $\rightarrow \lambda x$.corre(x)

- Non funziona più!

- Mario λx .corre(x) *non* è una λ -espressione valida!

- Perché? Perché non è una funzione a cui posso applicare un argomento (a meno di non cambiare l'ordine di applicazione, che non funziona).

Composizionalità

- Non è così semplice...

- E se avessimo *Mario corre*?

- *Mario* \rightarrow Mario

● I nomi propri hanno rappresentazioni atomiche come nella FOL.

- *Corre* $\rightarrow \lambda x. \text{corre}(x)$

- Non funziona più!

- Mario $\lambda x. \text{corre}(x)$ **non** è una λ -espressione valida!

● Mario ora non è una funzione a cui posso applicare un argomento (a meno di non cambiare l'ordine di applicazione, cosa che non vogliamo)

Composizionalità

- Non è così semplice...

- E se avessimo *Mario corre*?

- *Mario* \rightarrow Mario

- I nomi propri hanno rappresentazioni atomiche come nella FOL.

- *Corre* $\rightarrow \lambda x. \text{corre}(x)$

- Non funziona più!

- Mario $\lambda x. \text{corre}(x)$ **non** è una λ -espressione valida!

- Mario ora non è una funzione a cui posso applicare un argomento (a meno di non cambiare l'ordine di applicazione, cosa che non vogliamo)

Composizionalità

- Non è così semplice...

- E se avessimo *Mario corre*?

- *Mario* \rightarrow Mario

- I nomi propri hanno rappresentazioni atomiche come nella FOL.

- *Corre* $\rightarrow \lambda x.corre(x)$

- Non funziona più!

- Mario $\lambda x.corre(x)$ **non** è una λ -espressione valida!

- Mario ora non è una funzione a cui posso applicare un argomento (a meno di non cambiare l'ordine di applicazione, cosa che non vogliamo)

Composizionalità

- Non è così semplice...

- E se avessimo *Mario corre*?

- *Mario* \rightarrow Mario

- I nomi propri hanno rappresentazioni atomiche come nella FOL.

- *Corre* $\rightarrow \lambda x. \text{corre}(x)$

- Non funziona più!

- Mario $\lambda x. \text{corre}(x)$ **non** è una λ -espressione valida!

- Mario ora non è una funzione a cui posso applicare un argomento (a meno di non cambiare l'ordine di applicazione, cosa che non vogliamo)

Composizionalità

- Non è così semplice...
- E se avessimo *Mario corre*?
- *Mario* \rightarrow Mario
 - I nomi propri hanno rappresentazioni atomiche come nella FOL.
- *Corre* $\rightarrow \lambda x. \text{corre}(x)$
- Non funziona più!
- *Mario* $\lambda x. \text{corre}(x)$ **non** è una λ -espressione valida!
 - *Mario* ora non è una funzione a cui posso applicare un argomento (a meno di non cambiare l'ordine di applicazione, cosa che non vogliamo)

Composizionalità

- Non è così semplice...
- E se avessimo *Mario corre*?
- *Mario* \rightarrow Mario
 - I nomi propri hanno rappresentazioni atomiche come nella FOL.
- *Corre* $\rightarrow \lambda x. \text{corre}(x)$
- Non funziona più!
- Mario $\lambda x. \text{corre}(x)$ **non** è una λ -espressione valida!
 - Mario ora non è una funzione a cui posso applicare un argomento (a meno di non cambiare l'ordine di applicazione, cosa che non vogliamo)

Composizionalità

- Non è così semplice...
- E se avessimo *Mario corre*?
- *Mario* \rightarrow Mario
 - I nomi propri hanno rappresentazioni atomiche come nella FOL.
- *Corre* $\rightarrow \lambda x. \text{corre}(x)$
- Non funziona più!
- Mario $\lambda x. \text{corre}(x)$ **non** è una λ -espressione valida!
 - Mario ora non è una funzione a cui posso applicare un argomento (a meno di non cambiare l'ordine di applicazione, cosa che non vogliamo)

Composizionalità

Soluzione

Trasformiamo **tutto** in funzione:

- $Mario \rightarrow \lambda P.P(Mario)$
 - Un nome ora è una funzione che, dato un qualunque *predicato* P restituisce quel predicato applicato al nome

Composizionalità

Soluzione

Trasformiamo **tutto** in funzione:

- $Mario \rightarrow \lambda P.P(\text{Mario})$
 - Un nome ora è una funzione che, dato un qualunque *predicato* P restituisce quel predicato applicato al nome

Composizionalità

Soluzione

Trasformiamo **tutto** in funzione:

- $Mario \rightarrow \lambda P.P(Mario)$
 - Un nome ora è una funzione che, dato un qualunque *predicato* P restituisce quel predicato applicato al nome

Composizionalità

- *Mario corre:*

- $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \xrightarrow{\beta}$
- $\xrightarrow{\beta} \lambda x.\text{corre}(x) (\text{Mario})$
- $\xrightarrow{\beta} \text{corre}(\text{Mario})$

- Ora funziona!

Composizionalità

- *Mario corre:*

- $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \xrightarrow{\beta}$
- $\xrightarrow{\beta} \lambda x.\text{corre}(x) (\text{Mario})$
- $\xrightarrow{\beta} \text{corre}(\text{Mario})$

- Ora funziona!

Composizionalità

- *Mario corre:*

- $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \xrightarrow{\beta}$
- $\xrightarrow{\beta} \lambda x.\text{corre}(x) (\text{Mario})$
- $\xrightarrow{\beta} \text{corre}(\text{Mario})$

- Ora funziona!

Composizionalità

- *Mario corre:*

- $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \xrightarrow{\beta}$
- $\xrightarrow{\beta} \lambda x.\text{corre}(x) (\text{Mario})$
- $\xrightarrow{\beta} \text{corre}(\text{Mario})$

- Ora funziona!

Composizionalità

- *Mario corre:*

- $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \xrightarrow{\beta}$
- $\xrightarrow{\beta} \lambda x.\text{corre}(x) (\text{Mario})$
- $\xrightarrow{\beta} \text{corre}(\text{Mario})$

- Ora funziona!

Outline

- 1 Introduzione
- 2 Il λ -calcolo per la semantica
 - Rappresentazioni semantiche
 - λ -calcolo
 - **Lessico**
 - Ambiguità
 - Conclusioni

Lessico

- Il grosso del lavoro sta nella definizione delle λ -espressioni per ogni parola del lessico
- Vediamo qualche esempio esplicito per diverse parti del discorso

Nomi propri:

Mario $\rightarrow \lambda P.P(\text{Mario})$

Nomi comuni

cane $\rightarrow \lambda x.\text{cane}(x)$

Lessico

- Il grosso del lavoro sta nella definizione delle λ -espressioni per ogni parola del lessico
- Vediamo qualche esempio esplicito per diverse parti del discorso

Nomi propri:

Mario $\rightarrow \lambda P.P(\text{Mario})$

Nomi comuni

cane $\rightarrow \lambda x.\text{cane}(x)$

Lessico

- Il grosso del lavoro sta nella definizione delle λ -espressioni per ogni parola del lessico
- Vediamo qualche esempio esplicito per diverse parti del discorso

Nomi propri:

Mario $\rightarrow \lambda P.P(\text{Mario})$

Nomi comuni

cane $\rightarrow \lambda x.\text{cane}(x)$

Lessico

- Il grosso del lavoro sta nella definizione delle λ -espressioni per ogni parola del lessico
- Vediamo qualche esempio esplicito per diverse parti del discorso

Nomi propri:

Mario $\rightarrow \lambda P.P(\text{Mario})$

Nomi comuni

cane $\rightarrow \lambda x.\text{cane}(x)$

Verbi

Verbi intransitivi

corre $\rightarrow \lambda x. \text{corre}(x)$

Verbi transitivi

- Proviamo con

$\lambda x. \lambda y. \text{corre}(x, y)$

- Poniamo:

$\text{corre} = \lambda x. \lambda y. \text{corre}(x, y)$

Verbi

Verbi intransitivi

corre $\rightarrow \lambda x.corre(x)$

Verbi transitivi

- Proviamo con

- *ama* $\rightarrow \lambda y.\lambda x.ama(x,y)$

- $\lambda P.P(\text{Gianni}) \lambda y.\lambda x.ama(x,y) \lambda Q.Q(\text{Maria})$

- Non si riduce bene! $ama(\text{Gianni}, \lambda Q.Q(\text{Maria})) ??$

- Poniamo:

- *ama* $\rightarrow \lambda R.\lambda x.R(\lambda y.ama(x,y))$

Verbi

Verbi intransitivi

corre $\rightarrow \lambda x.corre(x)$

Verbi transitivi

- Proviamo con

- *ama* $\rightarrow \lambda y.\lambda x.ama(x,y)$

- $\lambda P.P(\text{Gianni}) \lambda y.\lambda x.ama(x,y) \lambda Q.Q(\text{Maria})$

- Non si riduce bene! $ama(\text{Gianni}, \lambda Q.Q(\text{Maria})) ??$

- Poniamo:

- *ama* $\rightarrow \lambda R.\lambda x.R(\lambda y.ama(x,y))$

Verbi

Verbi intransitivi

corre $\rightarrow \lambda x.corre(x)$

Verbi transitivi

- Proviamo con
 - *ama* $\rightarrow \lambda y.\lambda x.ama(x,y)$
 - $\lambda P.P(\text{Gianni}) \quad \lambda y.\lambda x.ama(x,y) \quad \lambda Q.Q(\text{Maria})$
 - Non si riduce bene! $ama(\text{Gianni}, \lambda Q.Q(\text{Maria})) ??$
- Poniamo:
 - *ama* $\rightarrow \lambda R.\lambda x.R(\lambda y.ama(x,y))$

Verbi

Verbi intransitivi

corre $\rightarrow \lambda x.corre(x)$

Verbi transitivi

- Proviamo con
 - *ama* $\rightarrow \lambda y.\lambda x.ama(x,y)$
 - $\lambda P.P(\text{Gianni}) \lambda y.\lambda x.ama(x,y) \lambda Q.Q(\text{Maria})$
 - Non si riduce bene! $ama(\text{Gianni}, \lambda Q.Q(\text{Maria})) ??$

- Poniamo:

ama $\rightarrow \lambda R.\lambda x.R(\lambda y.ama(x,y))$

Verbi

Verbi intransitivi

corre $\rightarrow \lambda x.corre(x)$

Verbi transitivi

- Proviamo con
 - *ama* $\rightarrow \lambda y.\lambda x.ama(x,y)$
 - $\lambda P.P(\text{Gianni}) \quad \lambda y.\lambda x.ama(x,y) \quad \lambda Q.Q(\text{Maria})$
 - Non si riduce bene! $ama(\text{Gianni}, \lambda Q.Q(\text{Maria}))$??
- Poniamo:
 - *ama* $\rightarrow \lambda R.\lambda x.R(\lambda y.ama(x,y))$

Verbi

Verbi intransitivi

corre $\rightarrow \lambda x.corre(x)$

Verbi transitivi

- Proviamo con
 - *ama* $\rightarrow \lambda y.\lambda x.ama(x,y)$
 - $\lambda P.P(\text{Gianni}) \quad \lambda y.\lambda x.ama(x,y) \quad \lambda Q.Q(\text{Maria})$
 - Non si riduce bene! $ama(\text{Gianni}, \lambda Q.Q(\text{Maria})) ??$
- Poniamo:
 - *ama* $\rightarrow \lambda R.\lambda x.R(\lambda y.ama(x,y))$

Verbi

Verbi transitivi

- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda R.\lambda x.R(\lambda y.\text{ama}(x,y)) \quad \lambda Q.Q(\text{Maria})}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda Q.Q(\text{Maria}) (\lambda y.\text{ama}(x,y)))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda y.\text{ama}(x,y) (\text{Maria}))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\text{ama}(x,\text{Maria}))}$
- $\underbrace{\lambda x.(\text{ama}(x,\text{Maria}))} (\text{Gianni})$
- $\text{ama}(\text{Gianni},\text{Maria})$

Verbi

Verbi transitivi

- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda R.\lambda x.R(\lambda y.\text{ama}(x,y)) \quad \lambda Q.Q(\text{Maria})}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda Q.Q(\text{Maria}) (\lambda y.\text{ama}(x,y)))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda y.\text{ama}(x,y) (\text{Maria}))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\text{ama}(x,\text{Maria}))}$
- $\underbrace{\lambda x.(\text{ama}(x,\text{Maria}))} (\text{Gianni})$
- $\text{ama}(\text{Gianni},\text{Maria})$

Verbi

Verbi transitivi

- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda R.\lambda x.R(\lambda y.\text{ama}(x,y)) \quad \lambda Q.Q(\text{Maria})}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda Q.Q(\text{Maria}) (\lambda y.\text{ama}(x,y)))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda y.\text{ama}(x,y) (\text{Maria}))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\text{ama}(x,\text{Maria}))}$
- $\underbrace{\lambda x.(\text{ama}(x,\text{Maria}))} (\text{Gianni})$
- $\text{ama}(\text{Gianni},\text{Maria})$

Verbi

Verbi transitivi

- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda R.\lambda x.R(\lambda y.\text{ama}(x,y)) \quad \lambda Q.Q(\text{Maria})}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda Q.Q(\text{Maria}) (\lambda y.\text{ama}(x,y)))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda y.\text{ama}(x,y) (\text{Maria}))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\text{ama}(x,\text{Maria}))}$
- $\underbrace{\lambda x.(\text{ama}(x,\text{Maria}))} (\text{Gianni})$
- $\text{ama}(\text{Gianni},\text{Maria})$

Verbi

Verbi transitivi

- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda R.\lambda x.R(\lambda y.\text{ama}(x,y)) \quad \lambda Q.Q(\text{Maria})}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda Q.Q(\text{Maria}) (\lambda y.\text{ama}(x,y)))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda y.\text{ama}(x,y) (\text{Maria}))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\text{ama}(x,\text{Maria}))}$
- $\underbrace{\lambda x.(\text{ama}(x,\text{Maria}))} (\text{Gianni})$
- $\text{ama}(\text{Gianni},\text{Maria})$

Verbi

Verbi transitivi

- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda R.\lambda x.R(\lambda y.\text{ama}(x,y)) \quad \lambda Q.Q(\text{Maria})}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda Q.Q(\text{Maria}) (\lambda y.\text{ama}(x,y)))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\lambda y.\text{ama}(x,y) (\text{Maria}))}$
- $\lambda P.P(\text{Gianni}) \quad \underbrace{\lambda x.(\text{ama}(x,\text{Maria}))}$
- $\underbrace{\lambda x.(\text{ama}(x,\text{Maria}))} (\text{Gianni})$
- $\text{ama}(\text{Gianni},\text{Maria})$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo *e* $\rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$
- In questo modo:
 - $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo *e* $\rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$
- In questo modo:
 - $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo *e* $\rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$

- In questo modo:

- $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
- $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
- $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
- $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
- $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo *e* $\rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$
- In questo modo:
 - $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo $e \rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$
- In questo modo:
 - $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo $e \rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$
- In questo modo:
 - $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo *e* $\rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$
- In questo modo:
 - $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo *e* $\rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$
- In questo modo:
 - $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- Come rappresento la coordinazione di due elementi?
 - *Gianni e Maria corrono* \rightarrow $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$
- Poniamo $e \rightarrow \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R))$
- In questo modo:
 - $\lambda P.P(\text{Gianni}) \quad \lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda X.\lambda Y.\lambda R.(R(X) \wedge Y(R)) \quad (\text{Gianni}) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda Y.\lambda R.(R(\text{Gianni}) \wedge Y(R)) \quad \lambda Q.Q(\text{Maria})$
 - $\lambda R.(R(\text{Gianni}) \wedge \lambda Q.Q(\text{Maria}) (R))$
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria}))$

Congiunzioni

Congiunzioni

- E infine aggiungiamo anche il verbo:
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria})) \lambda x.\text{corre}(x)$
 - $(\lambda x.\text{corre}(x) (\text{Gianni}) \wedge \lambda x.\text{corre}(x) (\text{Maria}))$
 - $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$

Congiunzioni

Congiunzioni

- E infine aggiungiamo anche il verbo:
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria})) \lambda x.\text{corre}(x)$
 - $(\lambda x.\text{corre}(x) (\text{Gianni}) \wedge \lambda x.\text{corre}(x) (\text{Maria}))$
 - $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$

Congiunzioni

Congiunzioni

- E infine aggiungiamo anche il verbo:
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria})) \lambda x.\text{corre}(x)$
 - $(\lambda x.\text{corre}(x) (\text{Gianni}) \wedge \lambda x.\text{corre}(x) (\text{Maria}))$
 - $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$

Congiunzioni

Congiunzioni

- E infine aggiungiamo anche il verbo:
 - $\lambda R.(R(\text{Gianni}) \wedge R(\text{Maria})) \lambda x.\text{corre}(x)$
 - $(\lambda x.\text{corre}(x) (\text{Gianni}) \wedge \lambda x.\text{corre}(x) (\text{Maria}))$
 - $\text{corre}(\text{Gianni}) \wedge \text{corre}(\text{Maria})$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:

• *Mario corre e salta* \rightarrow $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

- Poniamo $e \rightarrow \lambda R. \wedge R$

• $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.\text{salta}(x)}$

• $\lambda P.P(\text{Mario}) \quad \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)}$

• $\underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)} \quad (\text{Mario})$

• $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:

• *Mario corre e salta* \rightarrow $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

- Poniamo $e \rightarrow \lambda R. \wedge R$

• $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.\text{salta}(x)}$

• $\lambda P.P(\text{Mario}) \quad \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)}$

• $\underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)} \quad (\text{Mario})$

• $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:

• *Mario corre e salta* \rightarrow $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

- Poniamo $e \rightarrow \lambda R. \wedge R$

• $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.\text{salta}(x)}$

• $\lambda P.P(\text{Mario}) \quad \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)}$

• $\underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)} \quad (\text{Mario})$

• $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:

• *Mario corre e salta* \rightarrow $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

- Poniamo $e \rightarrow \lambda R. \wedge R$

• $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.\text{salta}(x)}$

• $\lambda P.P(\text{Mario}) \quad \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)}$

• $\underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)} \quad (\text{Mario})$

• $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:
 - *Mario corre e salta* \rightarrow $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

- Poniamo $e \rightarrow \lambda R. \wedge R$

$$\bullet \lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.\text{salta}(x)}$$

$$\bullet \lambda P.P(\text{Mario}) \quad \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)}$$

$$\bullet \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)} (\text{Mario})$$

$$\bullet \text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:
 - *Mario corre e salta* \rightarrow $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

- Poniamo $e \rightarrow \lambda R. \wedge R$

- $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.\text{salta}(x)}$
- $\lambda P.P(\text{Mario}) \quad \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)}$
- $\underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)} \quad (\text{Mario})$
- $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:
 - *Mario corre e salta* \rightarrow $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

- Poniamo $e \rightarrow \lambda R. \wedge R$

- $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.\text{salta}(x)}$

- $\lambda P.P(\text{Mario}) \quad \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)}$

- $\underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)} \quad (\text{Mario})$

- $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:
 - *Mario corre e salta* \rightarrow $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

- Poniamo $e \rightarrow \lambda R. \wedge R$

- $\lambda P.P(\text{Mario}) \quad \lambda x.\text{corre}(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.\text{salta}(x)}$

- $\lambda P.P(\text{Mario}) \quad \underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)}$

- $\underbrace{\lambda x.\text{corre}(x) \wedge \lambda x.\text{salta}(x)} \quad (\text{Mario})$

- $\text{corre}(\text{Mario}) \wedge \text{salta}(\text{Mario})$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...

- Vediamo come coordinare due verbi:
 - $Mario\ corre\ e\ salta \rightarrow corre(Mario) \wedge salta(Mario)$
- Poniamo $e \rightarrow \lambda R. \wedge R$
 - $\lambda P.P(Mario) \quad \lambda x.corre(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.salta(x)}$
 - $\lambda P.P(Mario) \quad \underbrace{\lambda x.corre(x) \wedge \lambda x.salta(x)}$
 - $\underbrace{\lambda x.corre(x) \wedge \lambda x.salta(x)} (Mario)$
 - $corre(Mario) \wedge salta(Mario)$

Congiunzioni

Problema!

- Questa era la coordinazione di due nomi....
- Per coordinare altre parti del discorso dobbiamo definire formule diverse...
- Vediamo come coordinare due verbi:
 - $Mario\ corre\ e\ salta \rightarrow corre(Mario) \wedge salta(Mario)$
 - Poniamo $e \rightarrow \lambda R. \wedge R$
 - $\lambda P.P(Mario) \quad \lambda x.corre(x) \quad \underbrace{\lambda R. \wedge R \quad \lambda x.salta(x)}$
 - $\lambda P.P(Mario) \quad \underbrace{\lambda x.corre(x) \wedge \lambda x.salta(x)}$
 - $\underbrace{\lambda x.corre(x) \wedge \lambda x.salta(x)} (Mario)$
 - $corre(Mario) \wedge salta(Mario)$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))}_{\text{un}} \underbrace{\lambda x.\text{uomo}(x)}_{\text{uomo}} \underbrace{\lambda x.\text{corre}(x)}_{\text{corre}}$
 - $\underbrace{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z))}_{\text{un uomo}} \lambda x.\text{corre}(x)$
 - $\underbrace{\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z))}_{\text{un uomo}} \lambda x.\text{corre}(x)$
 - $\underbrace{\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))}_{\text{un uomo corre}}$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))}_{\text{un}} \underbrace{\lambda x.\text{uomo}(x)}_{\text{uomo}} \underbrace{\lambda x.\text{corre}(x)}_{\text{corre}}$
 - $\underbrace{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z))}_{\text{un uomo}} \lambda x.\text{corre}(x)$
 - $\underbrace{\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z))}_{\text{un uomo}} \lambda x.\text{corre}(x)$
 - $\underbrace{\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))}_{\text{un uomo corre}}$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))}_{\text{un}} \underbrace{\lambda x.\text{uomo}(x)}_{\text{uomo}} \underbrace{\lambda x.\text{corre}(x)}_{\text{corre}}$
 - $\underbrace{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z))}_{\text{un uomo}} \lambda x.\text{corre}(x)$
 - $\underbrace{\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z))}_{\text{un uomo}} \lambda x.\text{corre}(x)$
 - $\underbrace{\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))}_{\text{un uomo corre}}$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z)) \quad \lambda x.\text{uomo}(x)} \quad \underbrace{\lambda x.\text{corre}(x)}$
 - $\underbrace{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z))} \quad \lambda x.\text{corre}(x)$
 - $\underbrace{\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z))} \quad \lambda x.\text{corre}(x)$
 - $\underbrace{\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))}$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))}_{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z))} \underbrace{\lambda x.\text{uomo}(x)}_{\lambda x.\text{corre}(x)}$
 - $\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z)) \lambda x.\text{corre}(x)$
 - $\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z)) \lambda x.\text{corre}(x)$
 - $\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z)) \lambda x.\text{uomo}(x)} \quad \underbrace{\lambda x.\text{corre}(x)}$
 - $\underbrace{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z)) \lambda x.\text{corre}(x)}$
 - $\underbrace{\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z)) \lambda x.\text{corre}(x)}$
 - $\underbrace{\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))}$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z)) \lambda x.\text{uomo}(x)} \quad \underbrace{\lambda x.\text{corre}(x)}$
 - $\underbrace{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z))} \quad \lambda x.\text{corre}(x)$
 - $\underbrace{\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z))} \quad \lambda x.\text{corre}(x)$
 - $\underbrace{\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))}$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z)) \lambda x.\text{uomo}(x)} \quad \underbrace{\lambda x.\text{corre}(x)}$
 - $\underbrace{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z)) \lambda x.\text{corre}(x)}$
 - $\underbrace{\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z)) \lambda x.\text{corre}(x)}$
 - $\underbrace{\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))}$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Come vogliamo che sia la rappresentazione di *un uomo*?
 - *un uomo corre* $\rightarrow \exists x : (\text{uomo}(x) \wedge \text{corre}(x))$
- Poniamo *un* $\rightarrow \lambda P.\lambda Q.\exists z (P(z) \wedge Q(z))$
- In questo modo *un uomo corre*:
 - $\underbrace{\lambda P.\lambda Q.\exists z (P(z) \wedge Q(z)) \lambda x.\text{uomo}(x)} \quad \underbrace{\lambda x.\text{corre}(x)}$
 - $\underbrace{\lambda Q.\exists z (\lambda x.\text{uomo}(x) (z) \wedge Q(z)) \lambda x.\text{corre}(x)}$
 - $\underbrace{\lambda Q.\exists z (\text{uomo}(z) \wedge Q(z)) \lambda x.\text{corre}(x)}$
 - $\underbrace{\exists z (\text{uomo}(z) \wedge \lambda x.\text{corre}(x) (z))}$
 - $\exists z (\text{uomo}(z) \wedge \text{corre}(z))$

Determinativi

- Altri determinativi si definiscono in maniera analoga:
 - Ogni uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \text{corre}(z))$
 - *ogni* $\rightarrow \lambda P.\lambda Q.\forall z(P(z) \Rightarrow Q(z))$
 - Nessun uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \neg \text{corre}(z))$
 - *nessun* $\rightarrow \lambda P.\lambda Q.\forall z(P(z) \Rightarrow \neg Q(z))$

Determinativi

- Altri determinativi si definiscono in maniera analoga:
 - Ogni uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \text{corre}(z))$
 - *ogni* $\rightarrow \lambda P.\lambda Q.\forall z (P(z) \Rightarrow Q(z))$
 - Nessun uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \neg \text{corre}(z))$
 - *nessun* $\rightarrow \lambda P.\lambda Q.\forall z (P(z) \Rightarrow \neg Q(z))$

Determinativi

- Altri determinativi si definiscono in maniera analoga:
 - Ogni uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \text{corre}(z))$
 - *ogni* $\rightarrow \lambda P.\lambda Q.\forall z (P(z) \Rightarrow Q(z))$
 - Nessun uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \neg \text{corre}(z))$
 - *nessun* $\rightarrow \lambda P.\lambda Q.\forall z (P(z) \Rightarrow \neg Q(z))$

Determinativi

- Altri determinativi si definiscono in maniera analoga:
 - Ogni uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \text{corre}(z))$
 - *ogni* $\rightarrow \lambda P.\lambda Q.\forall z (P(z) \Rightarrow Q(z))$
 - Nessun uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \neg \text{corre}(z))$
 - *nessun* $\rightarrow \lambda P.\lambda Q.\forall z (P(z) \Rightarrow \neg Q(z))$

Determinativi

- Altri determinativi si definiscono in maniera analoga:
 - Ogni uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \text{corre}(z))$
 - *ogni* $\rightarrow \lambda P.\lambda Q.\forall z (P(z) \Rightarrow Q(z))$
 - Nessun uomo corre: $\forall z : (\text{uomo}(z) \Rightarrow \neg \text{corre}(z))$
 - *nessun* $\rightarrow \lambda P.\lambda Q.\forall z (P(z) \Rightarrow \neg Q(z))$

Outline

- 1 Introduzione
- 2 Il λ -calcolo per la semantica
 - Rappresentazioni semantiche
 - λ -calcolo
 - Lessico
 - **Ambiguità**
 - Conclusioni

Ambiguità

- Cosa succede quando ho ambiguità?

Ambiguità semantica

- Devo disambiguare in qualche modo a priori
- Ogni *senso* avrà la sua rappresentazione:
 - *porta* (sostantivo) $\rightarrow \lambda x. \text{porta}(x)$
 - *porta* (verbo) $\rightarrow \lambda R. \lambda x. R(\lambda y. \text{porta}(x, y))$

Ambiguità sintattica

Produciamo il parsing più probabile e usiamo quello (oppure li produciamo tutti)

Ambiguità

- Cosa succede quando ho ambiguità?

Ambiguità semantica

- Devo disambiguare in qualche modo a priori
- Ogni *senso* avrà la sua rappresentazione:
 - *porta* (sostantivo) $\rightarrow \lambda x. \text{porta}(x)$
 - *porta* (verbo) $\rightarrow \lambda R. \lambda x. R(\lambda y. \text{porta}(x, y))$

Ambiguità sintattica

Produciamo il parsing più probabile e usiamo quello (oppure li produciamo tutti)

Ambiguità

- Cosa succede quando ho ambiguità?

Ambiguità semantica

- Devo disambiguare in qualche modo a priori
- Ogni *senso* avrà la sua rappresentazione:
 - *porta* (sostantivo) $\rightarrow \lambda x. \text{porta}(x)$
 - *porta* (verbo) $\rightarrow \lambda R. \lambda x. R(\lambda y. \text{porta}(x, y))$

Ambiguità sintattica

Produciamo il parsing più probabile e usiamo quello (oppure li produciamo tutti)

Ambiguità

- Cosa succede quando ho ambiguità?

Ambiguità semantica

- Devo disambiguare in qualche modo a priori
- Ogni *sensò* avrà la sua rappresentazione:
 - *porta* (sostantivo) $\rightarrow \lambda x. \text{porta}(x)$
 - *porta* (verbo) $\rightarrow \lambda R. \lambda x. R(\lambda y. \text{porta}(x, y))$

Ambiguità sintattica

Produciamo il parsing più probabile e usiamo quello (oppure li produciamo tutti)

Ambiguità

- Cosa succede quando ho ambiguità?

Ambiguità semantica

- Devo disambiguare in qualche modo a priori
- Ogni *sensò* avrà la sua rappresentazione:
 - *porta* (sostantivo) $\rightarrow \lambda x. \text{porta}(x)$
 - *porta* (verbo) $\rightarrow \lambda R. \lambda x. R(\lambda y. \text{porta}(x, y))$

Ambiguità sintattica

Produciamo il parsing più probabile e usiamo quello (oppure li produciamo tutti)

Ambiguità

Scope dei quantificatori

Ogni uomo ama una donna....

- una donna ciascuno:

$$\forall z (\text{uomo}(z) \Rightarrow \exists x : \text{donna}(x) \wedge \text{ama}(z, x))$$

- la stessa donna per tutti:

$$\exists x : (\text{donna}(x) \wedge (\forall z (\text{uomo}(z) \Rightarrow \text{ama}(z, x))))$$

Ambiguità

Scope dei quantificatori

Ogni uomo ama una donna....

- una donna ciascuno:

$$\forall z (\text{uomo}(z) \Rightarrow \exists x : \text{donna}(x) \wedge \text{ama}(z, x))$$

- la stessa donna per tutti:

$$\exists x : (\text{donna}(x) \wedge (\forall z (\text{uomo}(z) \Rightarrow \text{ama}(z, x))))$$

Ambiguità

Scope dei quantificatori

Ogni uomo ama una donna....

- una donna ciascuno:

$$\forall z (\text{uomo}(z) \Rightarrow \exists x : \text{donna}(x) \wedge \text{ama}(z, x))$$

- la stessa donna per tutti:

$$\exists x : (\text{donna}(x) \wedge (\forall z (\text{uomo}(z) \Rightarrow \text{ama}(z, x))))$$

Ambiguità

- Vediamo il nostro metodo cosa ci da...

- *Ogni uomo*

- $\lambda P.\lambda Q.\forall z (P(z) \Rightarrow Q(z))$ $\lambda x.uomo(x)$
- $\lambda Q.\forall z (uomo(z) \Rightarrow Q(z))$

- *Una donna*

- $\lambda S.\lambda T.\exists z : (S(z) \wedge T(z))$ $\lambda x.donna(x)$
- $\lambda T.\exists z : (donna(z) \wedge T(z))$

- *ama una donna*

- $\lambda R.\lambda u.R(\lambda v.ama(v, u))$ $\lambda T.\exists w : (donna(w) \wedge T(w))$
- $\lambda u.\exists w : (donna(w) \wedge ama(u, w))$

Ambiguità

- Vediamo il nostro metodo cosa ci da...

- *Ogni uomo*

- $\lambda P.\lambda Q.\forall z (P(z) \Rightarrow Q(z))$ $\lambda x.\text{uomo}(x)$
- $\lambda Q.\forall z (\text{uomo}(z) \Rightarrow Q(z))$

- *Una donna*

- $\lambda S.\lambda T.\exists z : (S(z) \wedge T(z))$ $\lambda x.\text{donna}(x)$
- $\lambda T.\exists z : (\text{donna}(z) \wedge T(z))$

- *ama una donna*

- $\lambda R.\lambda u.R(\lambda v.\text{ama}(v, u))$ $\lambda T.\exists w : (\text{donna}(w) \wedge T(w))$
- $\lambda u.\exists w : (\text{donna}(w) \wedge \text{ama}(u, w))$

Ambiguità

- Vediamo il nostro metodo cosa ci da...

- *Ogni uomo*

- $\lambda P.\lambda Q.\forall z (P(z) \Rightarrow Q(z))$ $\lambda x.uomo(x)$
- $\lambda Q.\forall z (uomo(z) \Rightarrow Q(z))$

- *Una donna*

- $\lambda S.\lambda T.\exists z : (S(z) \wedge T(z))$ $\lambda x.donna(x)$
- $\lambda T.\exists z : (donna(z) \wedge T(z))$

- *ama una donna*

- $\lambda R.\lambda u.R(\lambda v.ama(v, u))$ $\lambda T.\exists w : (donna(w) \wedge T(w))$
- $\lambda u.\exists w : (donna(w) \wedge ama(u, w))$

Ambiguità

- Vediamo il nostro metodo cosa ci da...

- *Ogni uomo*

- $\lambda P.\lambda Q.\forall z (P(z) \Rightarrow Q(z))$ $\lambda x.\text{uomo}(x)$
- $\lambda Q.\forall z (\text{uomo}(z) \Rightarrow Q(z))$

- *Una donna*

- $\lambda S.\lambda T.\exists z : (S(z) \wedge T(z))$ $\lambda x.\text{donna}(x)$
- $\lambda T.\exists z : (\text{donna}(z) \wedge T(z))$

- *ama una donna*

- $\lambda R.\lambda u.R(\lambda v.\text{ama}(v, u))$ $\lambda T.\exists w : (\text{donna}(w) \wedge T(w))$
- $\lambda u.\exists w : (\text{donna}(w) \wedge \text{ama}(u, w))$

Riduzione finale

Ogni uomo ama una donna

- $\lambda Q.\forall z(\text{uomo}(z) \Rightarrow Q(z)) \quad \lambda u.\exists w : (\text{donna}(w) \wedge \text{ama}(u, w))$
- $\forall z(\text{uomo}(z) \Rightarrow \lambda u.\exists w : (\text{donna}(w) \wedge \text{ama}(u, w))(z))$
- $\forall z(\text{uomo}(z) \Rightarrow \exists w : (\text{donna}(w) \wedge \text{ama}(z, w)))$

Viene il significato “giusto”!

Riduzione finale

Ogni uomo ama una donna

- $\lambda Q.\forall z(\text{uomo}(z) \Rightarrow Q(z)) \quad \lambda u.\exists w : (\text{donna}(w) \wedge \text{ama}(u, w))$
- $\forall z(\text{uomo}(z) \Rightarrow \lambda u.\exists w : (\text{donna}(w) \wedge \text{ama}(u, w))(z))$
- $\forall z(\text{uomo}(z) \Rightarrow \exists w : (\text{donna}(w) \wedge \text{ama}(z, w)))$

Viene il significato “giusto”!

Riduzione finale

Ogni uomo ama una donna

- $\lambda Q.\forall z(\text{uomo}(z) \Rightarrow Q(z)) \quad \lambda u.\exists w : (\text{donna}(w) \wedge \text{ama}(u, w))$
- $\forall z(\text{uomo}(z) \Rightarrow \lambda u.\exists w : (\text{donna}(w) \wedge \text{ama}(u, w))(z))$
- $\forall z(\text{uomo}(z) \Rightarrow \exists w : (\text{donna}(w) \wedge \text{ama}(z, w)))$

Viene il significato "giusto"!

Riduzione finale

Ogni uomo ama una donna

- $\lambda Q. \forall z (\text{uomo}(z) \Rightarrow Q(z)) \quad \lambda u. \exists w : (\text{donna}(w) \wedge \text{ama}(u, w))$
- $\forall z (\text{uomo}(z) \Rightarrow \lambda u. \exists w : (\text{donna}(w) \wedge \text{ama}(u, w))(z))$
- $\forall z (\text{uomo}(z) \Rightarrow \exists w : (\text{donna}(w) \wedge \text{ama}(z, w)))$

Viene il significato "giusto"!

Riduzione finale

Ogni uomo ama una donna

- $\lambda Q. \forall z (\text{uomo}(z) \Rightarrow Q(z)) \quad \lambda u. \exists w : (\text{donna}(w) \wedge \text{ama}(u, w))$
- $\forall z (\text{uomo}(z) \Rightarrow \lambda u. \exists w : (\text{donna}(w) \wedge \text{ama}(u, w))(z))$
- $\forall z (\text{uomo}(z) \Rightarrow \exists w : (\text{donna}(w) \wedge \text{ama}(z, w)))$

Viene il significato “giusto”!

Outline

- 1 Introduzione
- 2 Il λ -calcolo per la semantica
 - Rappresentazioni semantiche
 - λ -calcolo
 - Lessico
 - Ambiguità
 - Conclusioni

Ricapitolando

Scopo

- Vogliamo ottenere una rappresentazione semantica di una frase
- Vogliamo che questa rappresentazione sia una formula nella logica del prim'ordine

Metodo

- Per ottenere la rappresentazione:
 1. Assegniamo il nostro lessico con λ -termini
 2. Traduciamo la frase per ottenere la struttura sintattica
 3. Costruiamo la rappresentazione semantica

Ricapitolando

Scopo

- Vogliamo ottenere una rappresentazione semantica di una frase
- Vogliamo che questa rappresentazione sia una formula nella logica del prim'ordine

Metodo

- Per ottenere la rappresentazione:
 1. Assegniamo il nostro lessico con λ -termini
 2. Traduciamo la frase per ottenere la struttura sintattica
 3. Costruiamo la rappresentazione semantica

Ricapitolando

Scopo

- Vogliamo ottenere una rappresentazione semantica di una frase
- Vogliamo che questa rappresentazione sia una formula nella logica del prim'ordine

Metodo

- Per ottenere la rappresentazione:

1. Analizziamo il testo lessicalmente (semantica lessicale)
2. Analizziamo il testo sintatticamente (analisi sintattica)
3. Combiniamo i risultati delle due analisi

Ricapitolando

Scopo

- Vogliamo ottenere una rappresentazione semantica di una frase
- Vogliamo che questa rappresentazione sia una formula nella logica del prim'ordine

Metodo

- Per ottenere la rappresentazione:
 - Aumentiamo il nostro lessico con λ -espressioni
 - Parsiamo la frase per ottenerne la struttura sintattica
 - Componiamo le λ -espressioni tramite β -riduzione

Ricapitolando

Scopo

- Vogliamo ottenere una rappresentazione semantica di una frase
- Vogliamo che questa rappresentazione sia una formula nella logica del prim'ordine

Metodo

- Per ottenere la rappresentazione:
 - 1 Aumentiamo il nostro lessico con λ -espressioni
 - 2 Parsiamo la frase per ottenerne la struttura sintattica
 - 3 Componiamo le λ -espressioni tramite β -riduzione

Ricapitolando

Scopo

- Vogliamo ottenere una rappresentazione semantica di una frase
- Vogliamo che questa rappresentazione sia una formula nella logica del prim'ordine

Metodo

- Per ottenere la rappresentazione:
 - 1 Aumentiamo il nostro lessico con λ -espressioni
 - 2 Parsiamo la frase per ottenerne la struttura sintattica
 - 3 Componiamo le λ -espressioni tramite β -riduzione

Ricapitolando

Scopo

- Vogliamo ottenere una rappresentazione semantica di una frase
- Vogliamo che questa rappresentazione sia una formula nella logica del prim'ordine

Metodo

- Per ottenere la rappresentazione:
 - 1 Aumentiamo il nostro lessico con λ -espressioni
 - 2 Parsiamo la frase per ottenerne la struttura sintattica
 - 3 Componiamo le λ -espressioni tramite β -riduzione

Ricapitolando

Scopo

- Vogliamo ottenere una rappresentazione semantica di una frase
- Vogliamo che questa rappresentazione sia una formula nella logica del prim'ordine

Metodo

- Per ottenere la rappresentazione:
 - 1 Aumentiamo il nostro lessico con λ -espressioni
 - 2 Parsiamo la frase per ottenerne la struttura sintattica
 - 3 Componiamo le λ -espressioni tramite β -riduzione

Nel definire le nostre λ -espressioni teniamo conto:

- devono potersi comporre e produrre nuove λ -espressioni
- il risultato finale deve essere una formula del prim'ordine (possibilmente sensata)

Ulteriori dettagli e limitazioni

- Altre parti del discorso da codificare
- Come rappresentare tempi e modi verbali...
- Il tutto può funzionare solo con strutture sintattiche relativamente semplici

Nel definire le nostre λ -espressioni teniamo conto:

- devono potersi comporre e produrre nuove λ -espressioni
- il risultato finale deve essere una formula del prim'ordine (possibilmente sensata)

Ulteriori dettagli e limitazioni

- Altre parti del discorso da codificare
- Come rappresentare tempi e modi verbali...
- Il tutto può funzionare solo con strutture sintattiche relativamente semplici

Nel definire le nostre λ -espressioni teniamo conto:

- devono potersi comporre e produrre nuove λ -espressioni
- il risultato finale deve essere una formula del prim'ordine (possibilmente sensata)

Ulteriori dettagli e limitazioni

- Altre parti del discorso da codificare
- Come rappresentare tempi e modi verbali...
- Il tutto può funzionare solo con strutture sintattiche relativamente semplici

Nel definire le nostre λ -espressioni teniamo conto:

- devono potersi comporre e produrre nuove λ -espressioni
- il risultato finale deve essere una formula del prim'ordine (possibilmente sensata)

Ulteriori dettagli e limitazioni

- Altre parti del discorso da codificare
- Come rappresentare tempi e modi verbali....
- Il tutto può funzionare solo con strutture sintattiche relativamente semplici

Nel definire le nostre λ -espressioni teniamo conto:

- devono potersi comporre e produrre nuove λ -espressioni
- il risultato finale deve essere una formula del prim'ordine (possibilmente sensata)

Ulteriori dettagli e limitazioni

- Altre parti del discorso da codificare
- Come rappresentare tempi e modi verbali....
- Il tutto può funzionare solo con strutture sintattiche relativamente semplici

Nel definire le nostre λ -espressioni teniamo conto:

- devono potersi comporre e produrre nuove λ -espressioni
- il risultato finale deve essere una formula del prim'ordine (possibilmente sensata)

Ulteriori dettagli e limitazioni

- Altre parti del discorso da codificare
- Come rappresentare tempi e modi verbali....
- Il tutto può funzionare solo con strutture sintattiche relativamente semplici

Nel definire le nostre λ -espressioni teniamo conto:

- devono potersi comporre e produrre nuove λ -espressioni
- il risultato finale deve essere una formula del prim'ordine (possibilmente sensata)

Ulteriori dettagli e limitazioni

- Altre parti del discorso da codificare
- Come rappresentare tempi e modi verbali....
- Il tutto può funzionare solo con strutture sintattiche relativamente semplici

Bibliografia

- *Building semantic representation using λ -calculus*, Bill MacCartney
- *Computational semantics: Lambda calculus*, Scott Farrar