CCGs

recitation 12/4

Austin NP

Miguel NP

Vivian NP

students NP

lectures NP

homework NP

slides N

some NP/N

graded (S\NP)/NP

made (S\NP)/NP

presented (S\NP)/NP

gave (S\NP)/NP/NP

and CONJ

but CONJ

CCG Rules

Forward application: A/B + B = A

Backward application: B + A\B = A

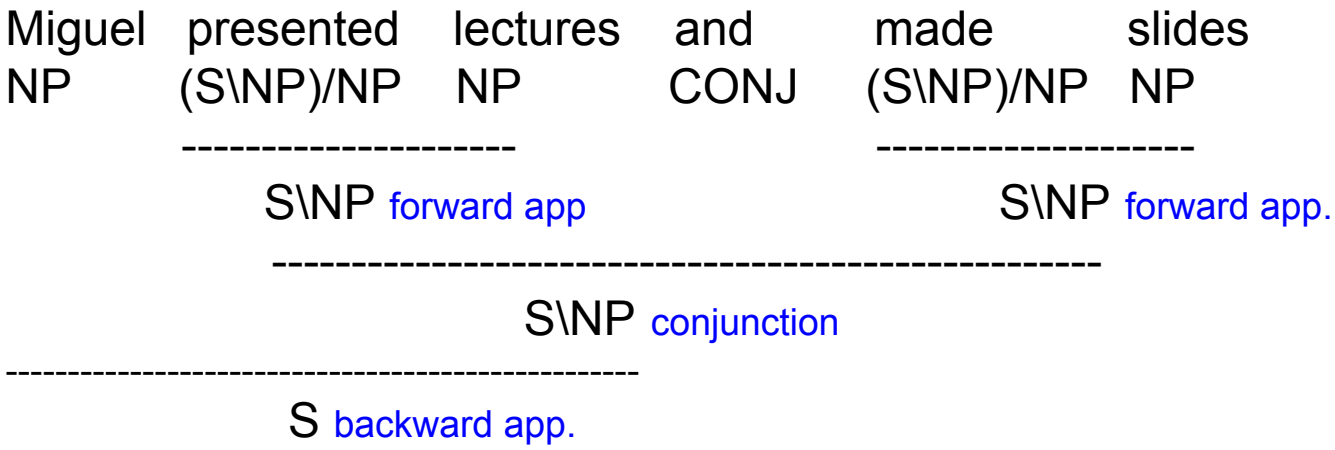
Composition: A/B + B/C = A/C

Composition: B\A + C\B = C\A

Conjunction: A CONJ A' = A''

Type raising: A = X/(X\A)

Type raising: A = X\(X/A)


Miguel presented lectures and made slides
NP (S\NP)/NP NP CONJ (S\NP)/NP NP
S\NP forward app
S\NP forward app.
S\NP conjunction
S backward app.

Miguel	made	and	Vivian	and	Austin	graded	homework
NP	(S\NP)/NP	CONJ	NP	CONJ	NP	(S\NP)/NP	NP

S

Miguel	made	and	Vivian	and	Austin	graded	homework
NP	(S\NP)/NP	CONJ	NP	CONJ	NP	(S\NP)/NP	NP

NP conjunction

Miguel	made	and	Vivian	and	Austin	graded	homework
NP	(S\NP)/NP	CONJ	NP	CONJ	NP	(S\NP)/NP	NP

NP conjunction

We want "Miguel made" and "Vivian and Austin grade" to both relate to "homework" in the same way

Miguel made and Vivian and Austin graded homework

NP (S\NP)/NP CONJ NP CONJ NP (S\NP)/NP NP

NP conjunction

S/NP ?

S/NP ?

S/NP conjunction (because these pieces are parallel)

S foward app.

(easiest way to join a final primitive, let's hope it works)

Miguel made and Vivian and Austin graded homework

NP (S\NP)/NP CONJ NP CONJ NP (S\NP)/NP NP

NP conjunction

Type raising! Type raising!

S/NP ?

S/NP ?

S/NP conjunction

S foward app.

CCG Rules

Forward application: A/B + B = A

Backward application: B + A\B = A

Composition: A/B + B/C = A/C

Composition: B\A + C\B = C\A

Conjunction: A CONJ A' = A''

Type raising: A = X/(X\A)

Type raising: A = X\(X/A)

in: NP (S\NP)/NP

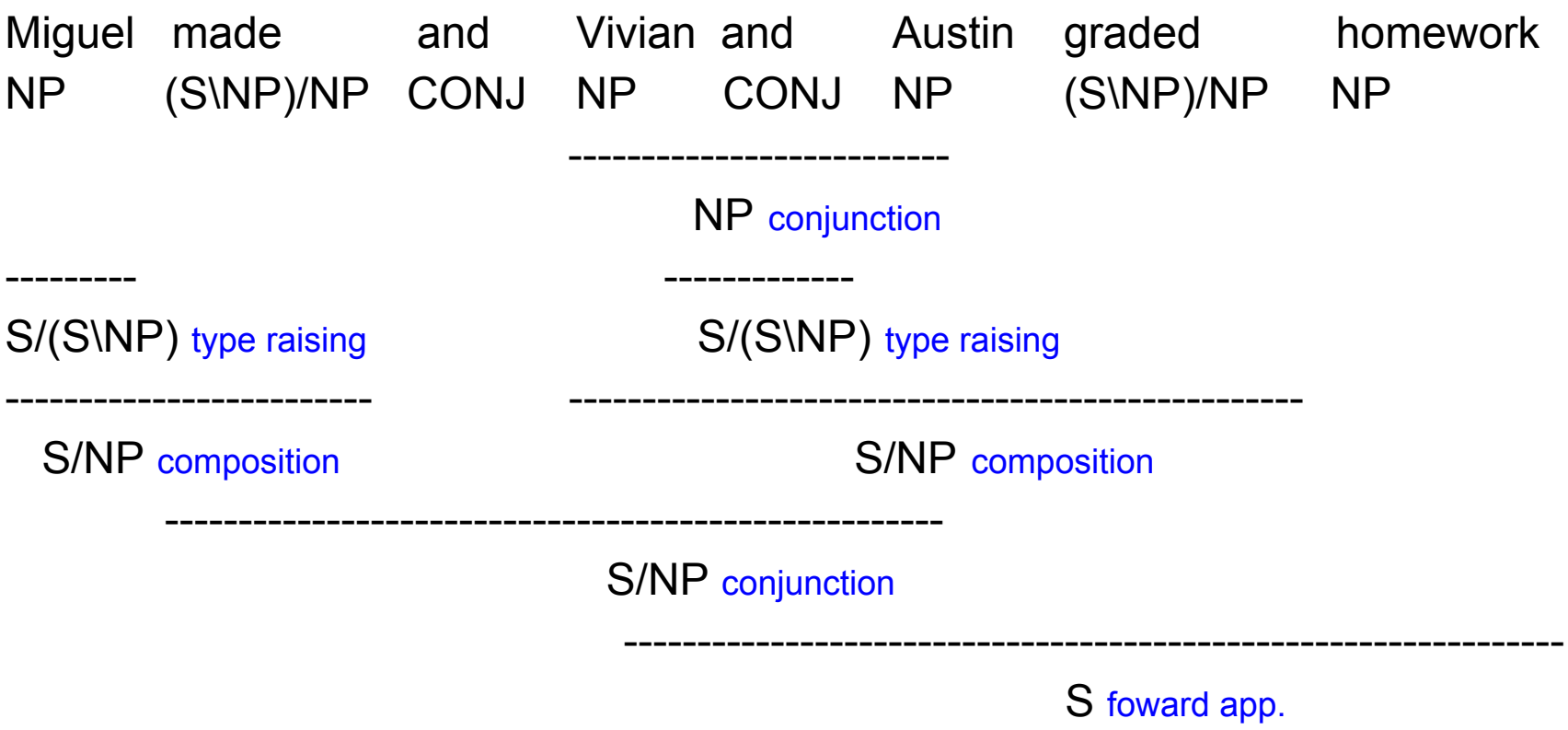
out: S/NP

make (S\NP) disappear, and create a S by itself

A = X/(X\A)

NP = S/(S\NP)

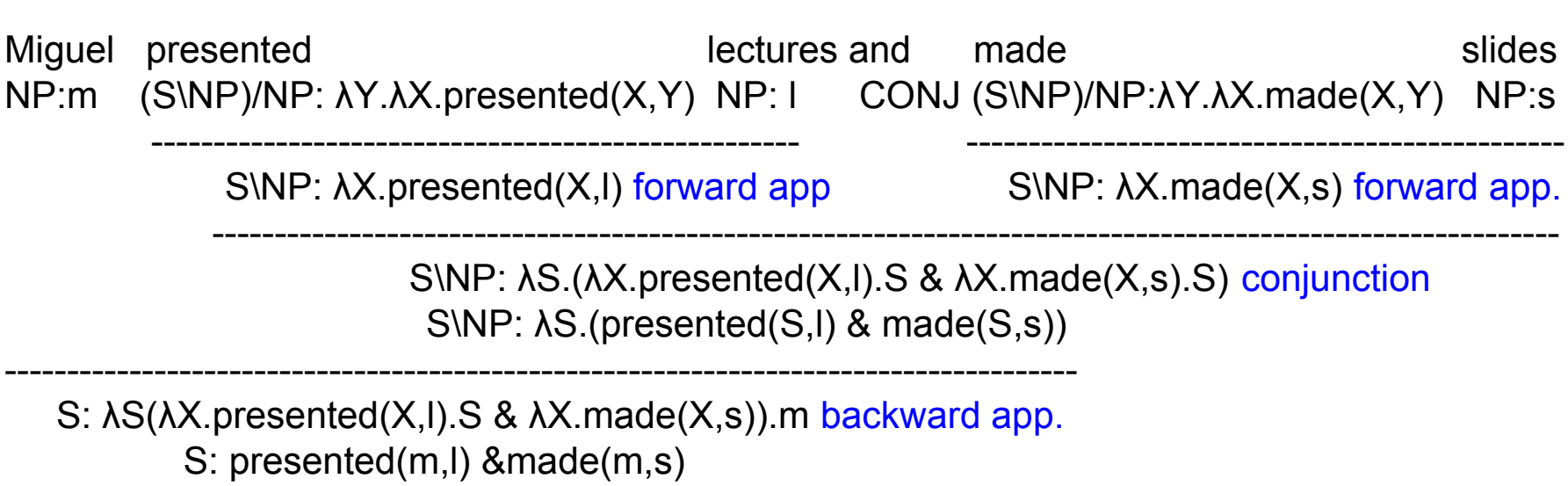
↑ this can now compose with the verbs to consume the (S\NP)


Miguel made and Vivian and Austin graded homework
NP (S\NP)/NP CONJ NP CONJ NP (S\NP)/NP NP
NP conjunction
S/(S\NP) type raising
S/(S\NP) type raising
S/NP composition
S/NP composition
S/NP conjunction
S foward app.

Wait, why could we do that?

- Type raising turns arguments into "functions over functions-over-such-arguments"
- Allows arguments to compose
  - → I like, and Mary dislikes, musicals

http://repository.upenn.edu/cgi/viewcontent.cgi?article=1411&context=cis_reports


Miguel presented lectures and made slides
NP:m (S\NP)/NP: λY.λX.presented(X,Y) NP: l CONJ (S\NP)/NP:λY.λX.made(X,Y) NP:s
S\NP: λX.presented(X,l) forward app
S\NP: λX.made(X,s) forward app.
S\NP: λS.(λX.presented(X,l).S & λX.made(X,s).S) conjunction
S\NP: λS.(presented(S,l) & made(S,s))
S: λS(λX.presented(X,l).S & λX.made(X,s)).m backward app.
S: presented(m,l) &made(m,s)

Miguel gave students homework

NP (S\NP)/NP/NP NP NP

(S\NP)/NP forward app.

S\NP forward app.

S backward app.