Funded by the European Union

15. Unrestricted Grammars

This series of tutorials is based upon work from COST Action **Multi3Generation CA18231**, supported by COST (European Cooperation in Science and Technology).

COST (European Cooperation in Science and Technology) is a funding agency for research and innovation networks. Our Actions help connect research initiatives across Europe and enable scientists to grow their ideas by sharing them with their peers. This boosts their research, career and innovation, cf. www.cost.eu

Unrestricted Grammars

- Unrestricted grammars are equivalent to Type 0 Generative Grammars in Chomsky-Schützenberger's hierarchy
- Unrestricted grammars can be more powerful than regular, context-free and context-sensitive grammars.
- They can recognize any recursively enumerated language (therefore, any possible natural language) and can be processed by Turing machines.
- Head-driven Phrase Structure Grammars (HPSG) are equivalent to unrestricted grammars.

Unrestricted Grammars in NooJ

- Unrestricted grammars are similar to context-sensitive grammars; they also contain variables and constraints.
- They can be used to replace matching text (recognized by the grammar input) with the corresponding grammar output
- The grammar output can be anything, *e.g.*, transformation of the text, paraphrases, semantic representations, etc.
- If associated with multilingual dictionaries, unrestricted grammars can produce translations

Unrestricted Grammars in NooJ

- Unrestricted grammars contain variables that can be copied to the output, *e.g.*:

(S1 Joe arrived). (S2 Eva left)/S1 and S2

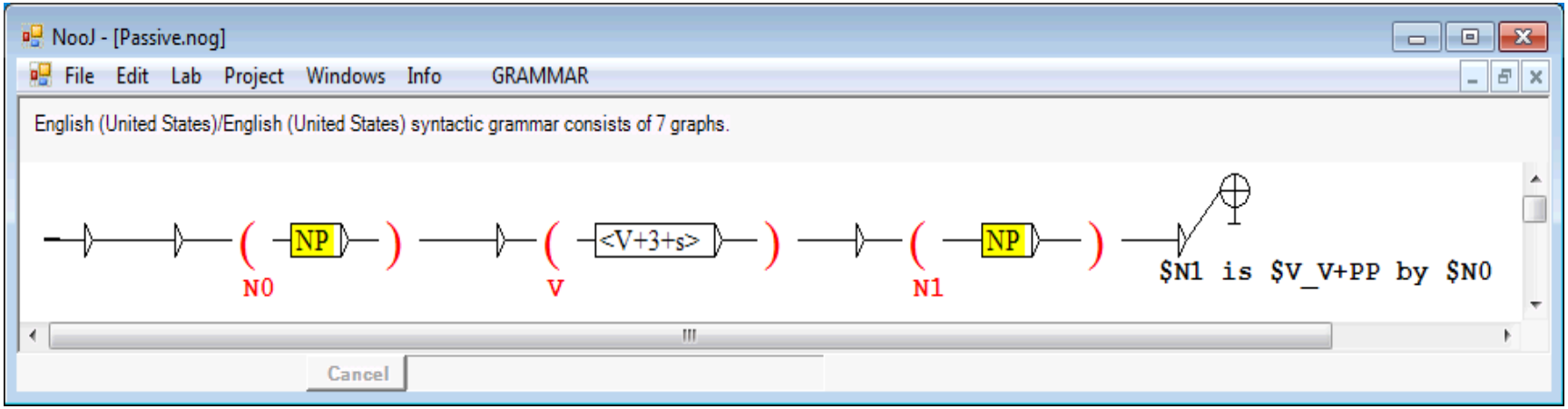
Joe arrived. Eva left. ⟶ Joe arrived and Eva left

- Morphological operators can be applied to variables, *e.g.*:

He (V eats)/They want to $V_

He eats ⟶ They want to eat

Transformations
A simple transformation



Joe sees Mary ⟶ Mary is seen by Joe

- In the dictionary, the entry **`see,V+FLX=SEE`** represents the forms *see* (+STD), *sees* (+PR+3+s), *saw* (+PRET), *seen* (+PP), *seing* (+G)

- If $V = “sees”, then $V_ = “see” (the lexical entry),
- then $V_V represents all the verbal forms associated with see, i.e., “see”, “sees”, “saw”, “seen” and “seing”
- then $V_V+PP = “seen”

Unrestricted Grammars

eSPERTo: System for Paraphrasing in Editing and Revision of Text

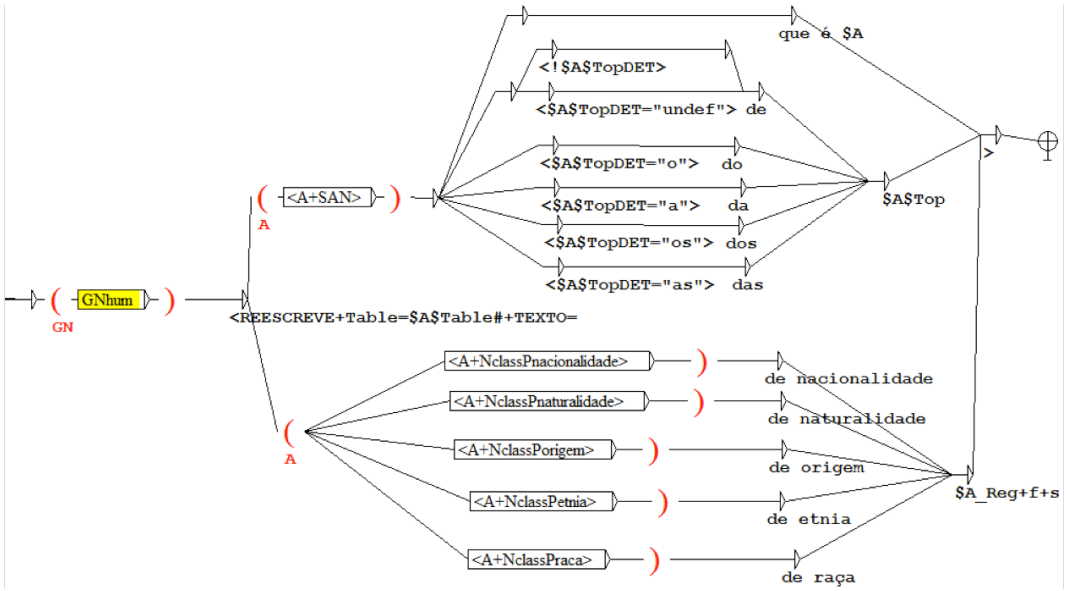
(A. Barreiro, C. Mota, INESC-ID)

SAN																																						
N0					Vcop											Class		Quant		Nclass Pred						Adjectivo	P_Adn		Reestr N_0			(W+E)				Nome
		Nap																		NclassP																		
Nhum	Nap de Nhum	Npabst	Npc	QueF	ser	estar	andar	ficar	permanecer	viver	encontrar-se	sentir-se	mostrar-se	revelar-se	tornar-se	UM Nclas	Um +Modif	Adv Quant	Superlativo	ser de	origem	nacionalidade	naturalidade	etnia	raça		N Adj	Adj N	Nhum Vop A Prep Nap	(de + em) (E+Def) Nap	em Dposs Nap	em Vinf	para com Nhum	CCl	(Seu + Meu) Adj!	
+	-	-	-	-	+	-	-	-	-	-	-	-	-	-	+	+	+	-	-	+	+	+	+	-	-	abissínio	+	-	-	-	-	-	-	-	-	Abissínia
+	-	-	-	-	+	-	-	-	-	-	-	-	-	-	-	+	+	-	-	+	+	-	+	-	-	acadiano	+	-	-	-	-	-	-	-	-	Acádia
+	-	-	-	-	+	-	-	-	-	-	-	-	-	-	-	+	+	-	-	+	+	-	+	-	-	açoriano	+	-	-	-	-	-	-	-	-	Açores
																										…										
+	-	-	-	-	+	-	-	-	-	-	-	-	-	-	+	+	+	-	-	+	+	+	+	-	-	iraquiano	+	-	-	-	-	-	-	-	-	Iraque
+	-	-	-	-	+	-	-	-	-	-	-	-	-	-	+	+	+	-	-	+	+	+	+	-	-	irlandês	+	-	-	-	-	-	-	-	-	Irlanda
+	-	-	-	-	+	-	-	-	-	-	-	-	-	-	+	+	+	-	-	+	+	+	+	-	-	islandês	+	-	-	-	-	-	-	-	-	Islândia

açoriano,A+FLX=ALTO
+Table=SAN+Nhum+Vcopser+UMNclas+UmModif+NclassPserde+NclassPorigem+NclassPnaturalidade+NAdj+DRV=nometop:Açores

…

iraquiano,A+FLX=ALTO+AN+lang+EN=Iraqi
+Table=SAN+Nhum+Vcopser+Vcoptornarse+UMNclas+UmModif+NclassPserde+NclassPorigem+NclassPnacionalidade+NclassPnaturalidade+NAdj+DRV=nometop:Iraque



Unrestricted Grammars

eSPERTo: System for Paraphrasing in Editing and Revision of Text

(A. Barreiro, C. Mota, INESC-ID)

Parameters

Demo mode ☐
Interface idiom English
Resources idiom Portuguese
Dictionary PT-Dict_NEW
Sample text SAN
Paraphrasing Check all Uncheck all

- ☐ Active > Passive
- ☐ Passive > Active
- ☐ Simple adverb > Compound
- ☐ Compound adverb > Simple
- ☐ Nominal/adjectival predicate > Verb
- ☐ Nominal/adjectival predicate > Nominal/adjectival predicate
- ☐ Verb > Nominal/adjectival predicate
- ☐ Relative construct > Adjective
- ☐ Possessive > Relative construct
- ☐ Synonyms
- ☑ Human intransitive adjective
- ☑ Predicate nouns with Vsup fazer
- ☑ Predicate nouns with Vsup ser de
- ☐ PE > PB
- ☐ PB > PE

Input file or text (click to show/hide)

Choose file: Browse file
Insert text in the text box

```
Recentemente, a jornalista iraquiana foi de uma grande ousadia quando fez o protesto à porta da universidade.

Inquestionavelmente, a mulher é parecida com o irmão.
```

Process results

Results (click to show/hide)

Save paraphrased text

Recentemente, [a jornalista [iraquiana][foi de]]uma grande ousadia quando [fez o protesto] à porta da unive[a mulher é parecida com o irmão] .

- houve na jornalista iraquiana
- que é iraquiano
- que é do Iraque
- de origem iraquiana
- de naturalidade iraquiana
- de nacionalidade iraquiana
- iraquiano
- do Iraque
- Suggest your own paraphrase ›

- protestou
- Suggest your own paraphrase ›

- o irmão é parecido com a mulher
- a mulher e o irmão são parecidos
- Suggest your own paraphrase ›

Transformations

How many simple transformations are there?

[Pron-0] John ate an apple = He ate an apple

[Pron-1] John ate an apple = John ate it

[Pron-2] John gave an apple to Marie = John gave her an apple

[Passive] John ate an apple = an apple was eaten by John

[Negation] John ate an apple = John did not eat an apple

[Cleft-0] John ate an apple = it is John who ate an apple

[Cleft-1] John ate an apple = it is an apple that John ate

[Question-0] John ate an apple = Who ate an apple?

[Question-1] John ate an apple = What did John eat?

[Question-V] John ate an apple = What did John do?

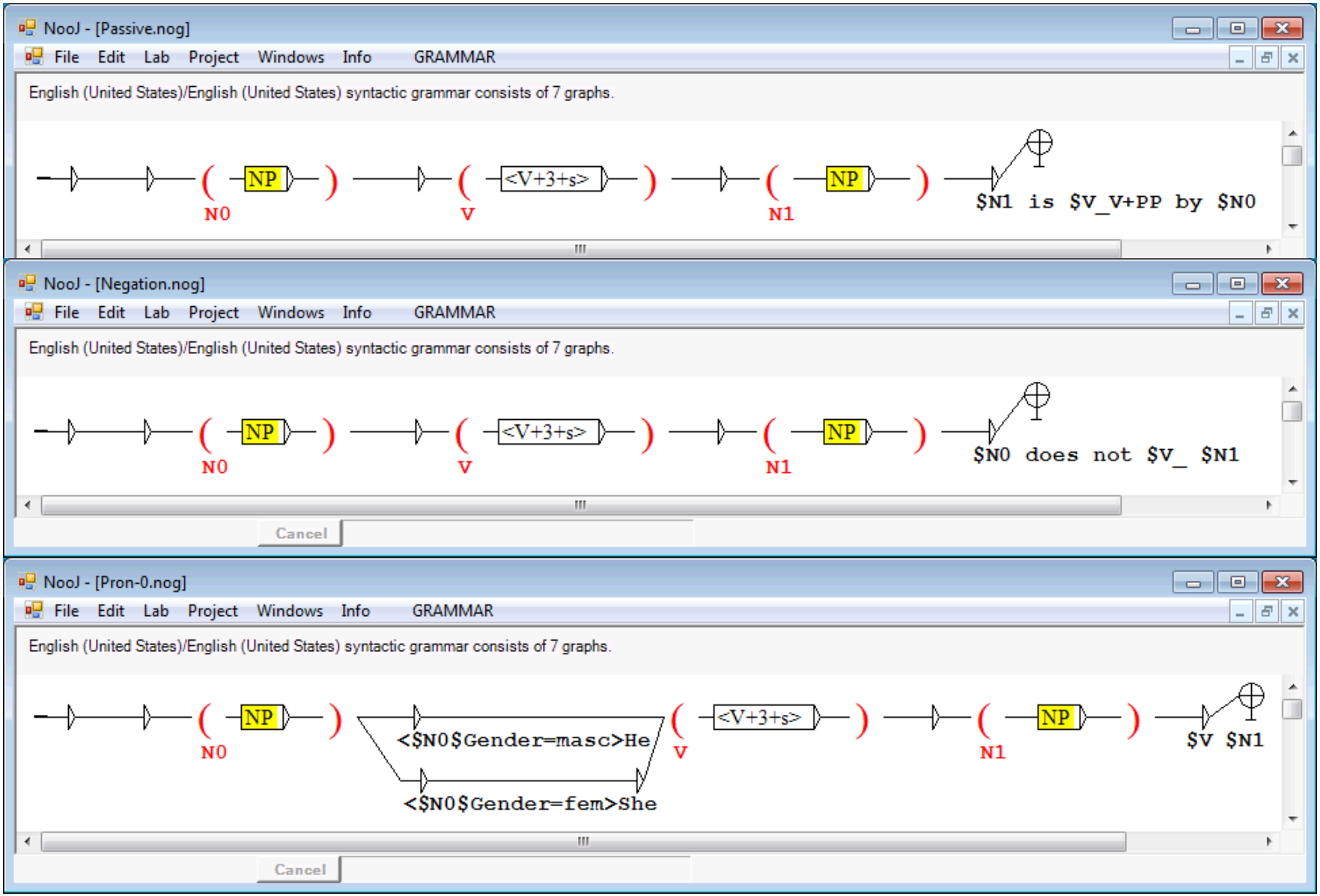
[Nom-0] John loves apples = John is an apple lover

[Nom-V] John gave the card to Mary = The card is John's gift to Mary

…

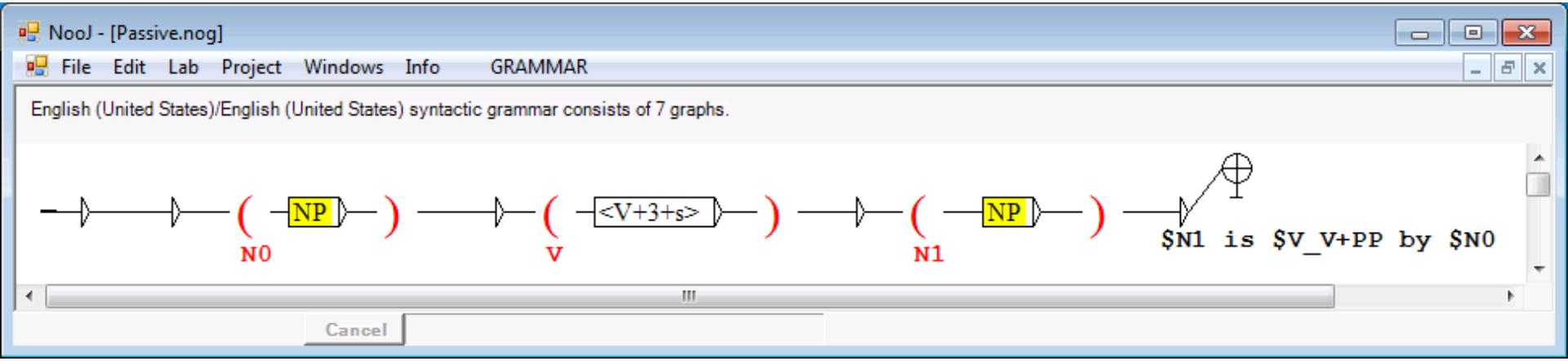
Transformations

On grammar per transformation?

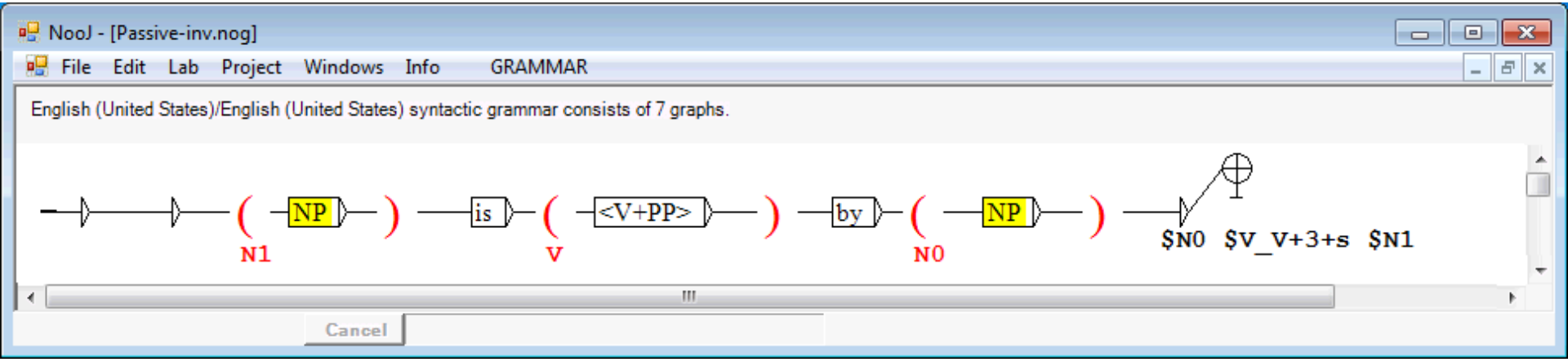


Transformations

Or two?



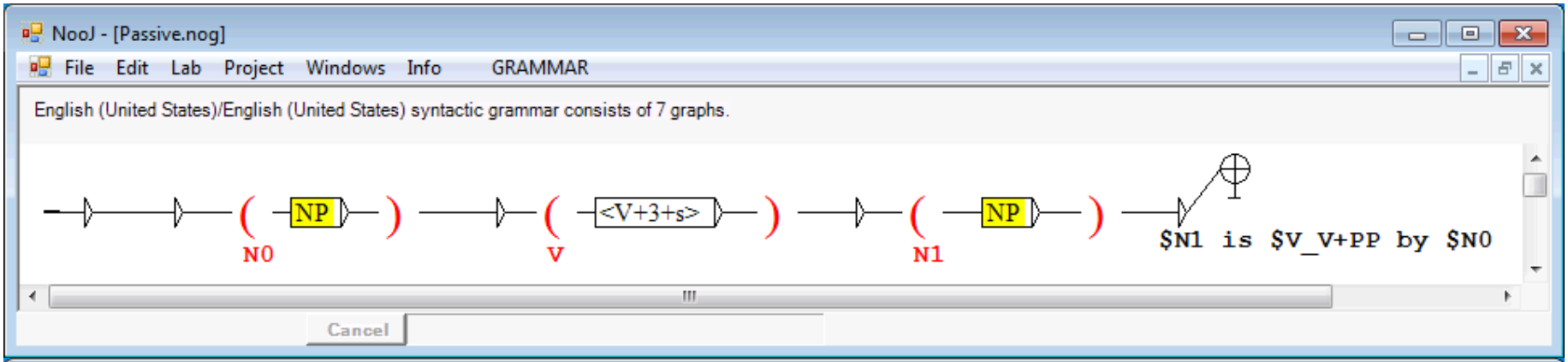
Active to Passive

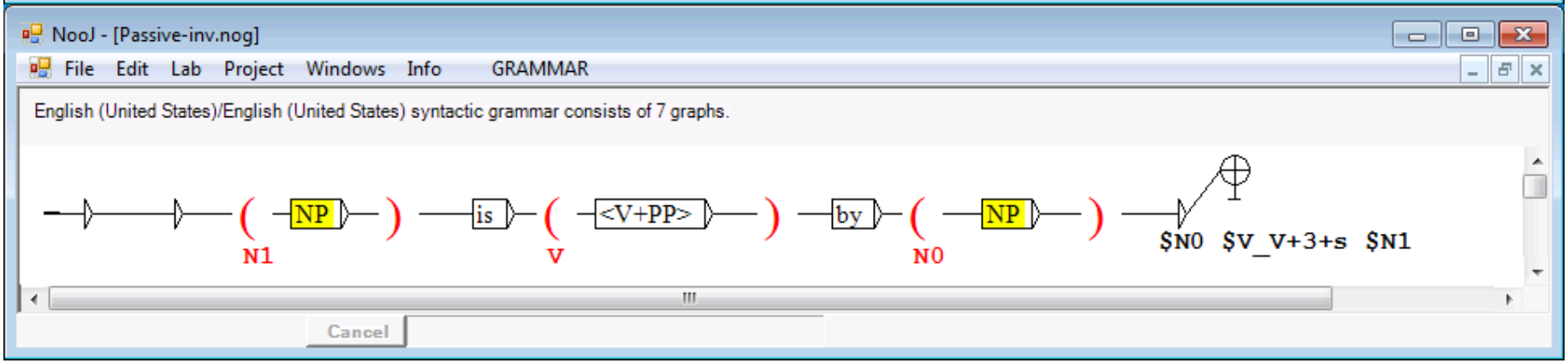


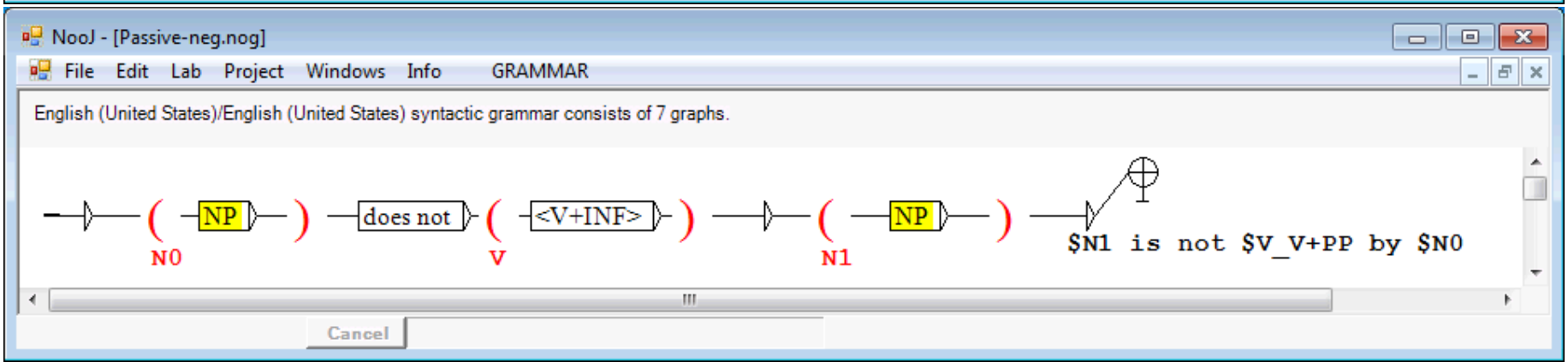
Passive to Active

Transformations

Actually: many more!







Transformations

How many simple transformations are there?

[Pron-0] John ate an apple = He ate an apple ⟶ [Passive1] = An apple was eaten by him

[Pron-2] John gave an apple to Marie = John gave her an apple ⟶ [Passive2] = She was given an apple by John

[Passive] John ate an apple = an apple was eaten by John ⟶ [Passive3]

[Negation] John ate an apple = John did not eat an apple ⟶ [Passive4] = An apple was not eaten by John

[Cleft-0] John ate an apple = it is John who ate an apple ⟶ [Passive5]

[Cleft-1] John ate an apple = it is an apple that John ate ⟶ [Passive6] = It is an apple that was eaten by John

[Question-0] John ate an apple = Who ate an apple? ⟶ [Passive7] = By Whom an apple was eaten?

[Question-1] John ate an apple = What did John eat? ⟶ [Passive8] = What was eaten by John?

[Question-V] John ate an apple = What did John do? ⟶ [Passive9] = What was done by John?

[Nom-0] John loves apples = John is an apple lover ⟶ [Passive10]

[Nom-V] John gave the card to Mary = The card is John's gift to Mary ⟶ [Passive11]

…

Every single transformed sentence may itself be transformed with a [Passive] transformation...

If there are 1,000 elementary transformations, then there are $1{,}000 \times 1{,}000$ combinations of 2 transformations, $1{,}000^3$ combinations of 3 transformations, and $1{,}000^7$ combinations like:

It is not like Joe wanted to stop being her lover

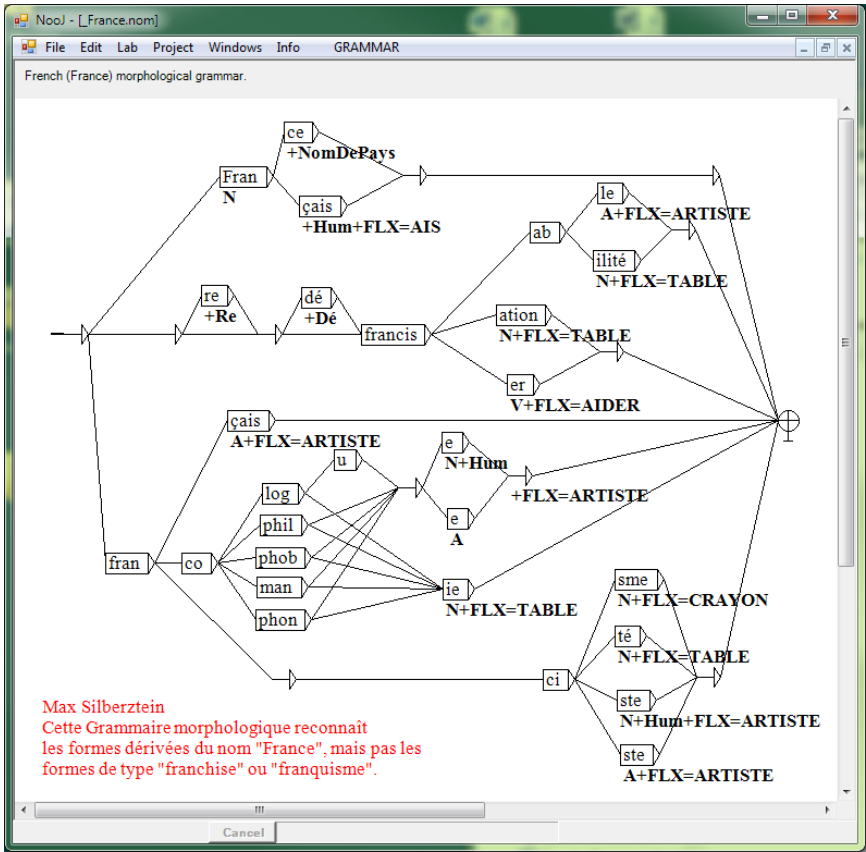
[Cleft-0][Pron-1][Preterit][Negation][Aspect][Modal][Nominalization]

NooJ solution to transformations:
Combine Parsing and Generation

- All NooJ linguistic resources are neutral, i.e., they can be used to parse texts, or to generate texts.
- For example, morphological grammar **France** recognizes all the forms derived and inflected from *France*:

- This grammar can also be used to produce all the forms described in the grammar via NooJ menu item:

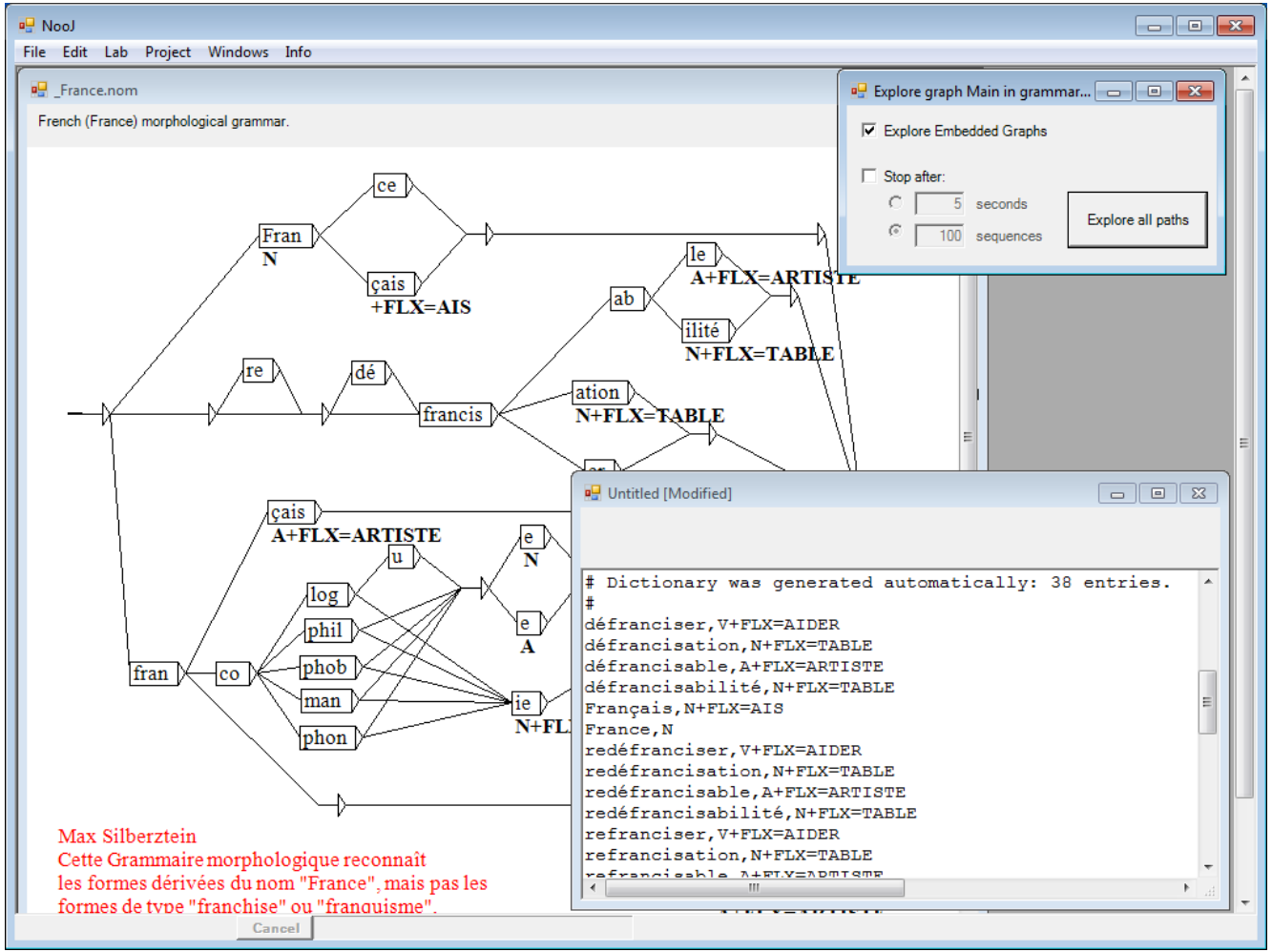
  GRAMMAR > Generate Language



NooJ solution to transformations:
Combine Parsing and Generation

- This grammar can also be used to produce all the forms described in the grammar via NooJ menu item:

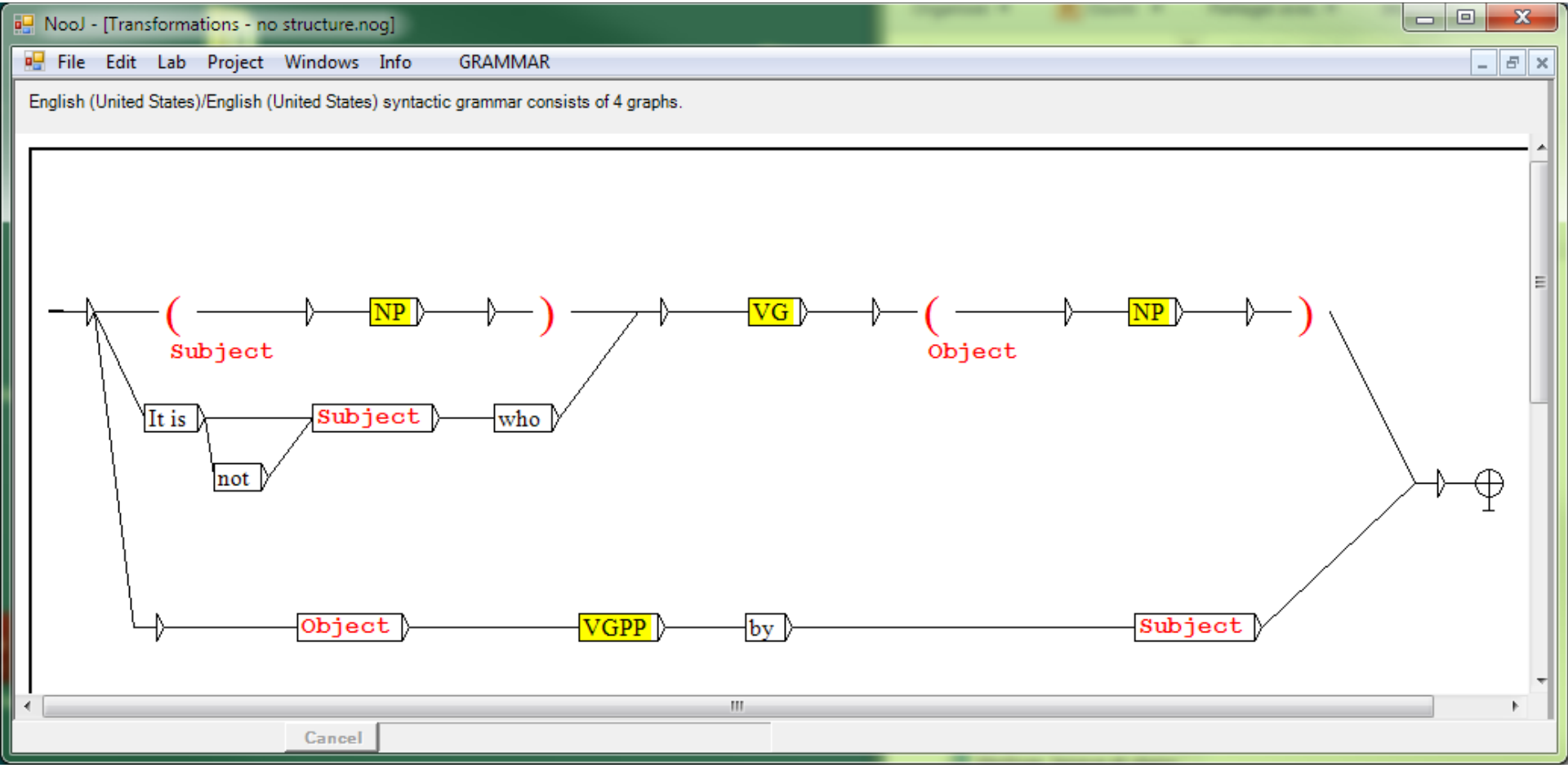
GRAMMAR > Generate Language



NooJ solution to transformations:
Combine Parsing and Generation

- This grammar can also be used to produce all the forms described in the grammar via NooJ menu item:

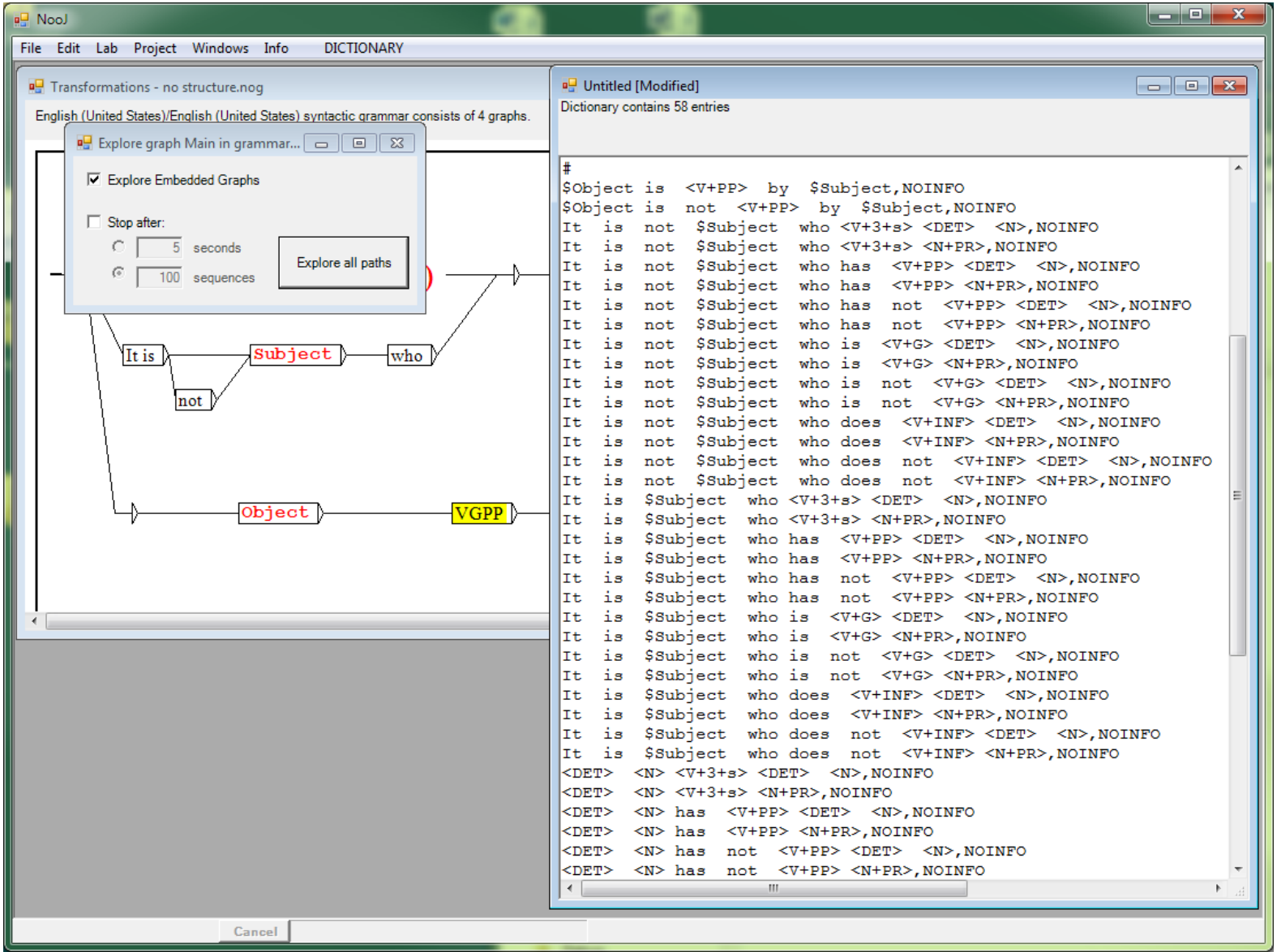
GRAMMAR > Generate Language



NooJ solution to transformations:

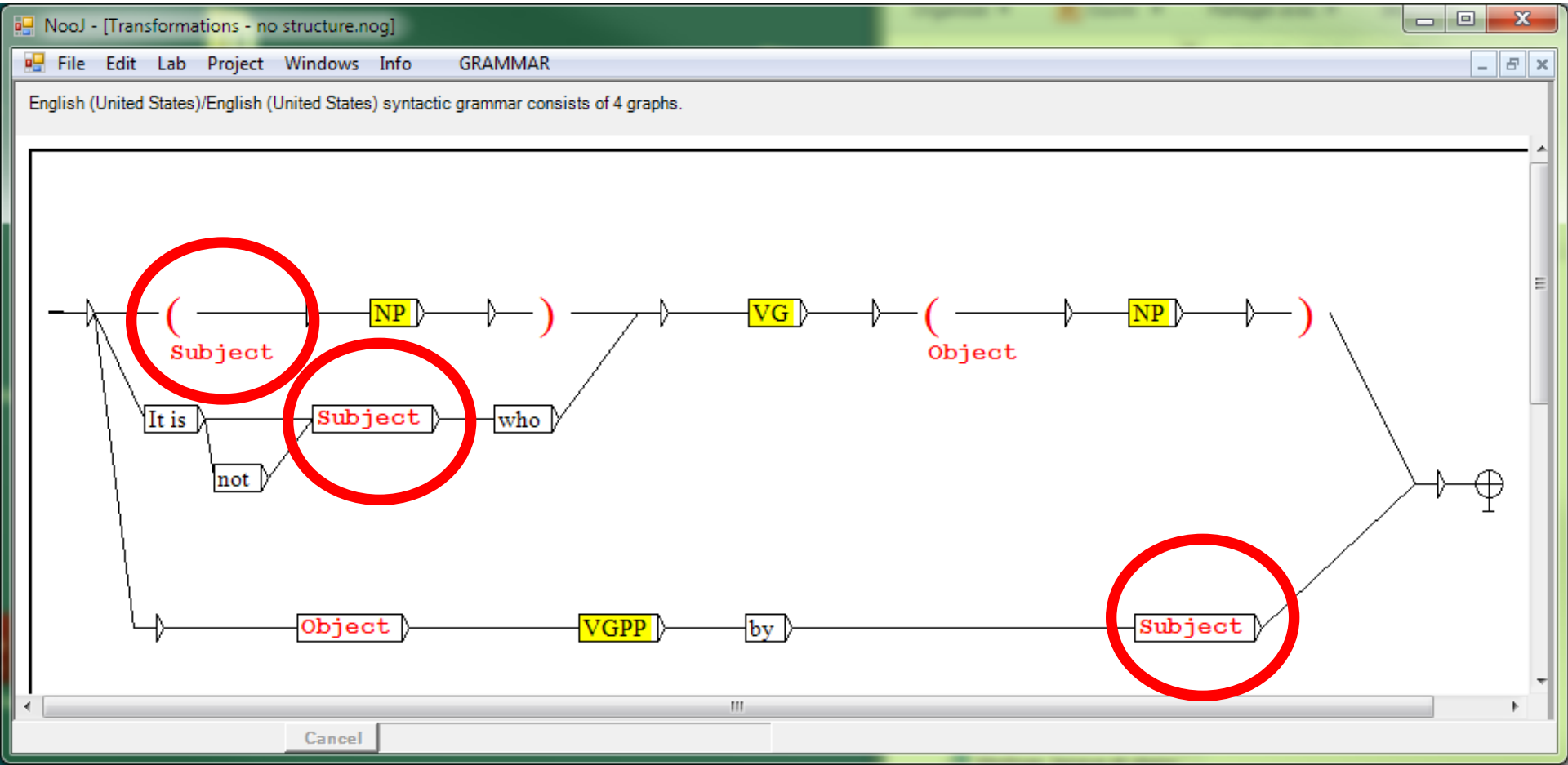
Combine Parsing and Generation

GRAMMAR > Generate Language



NooJ solution to transformations:
Combine Parsing and Generation

A variable stores an ALU; all its references are linked

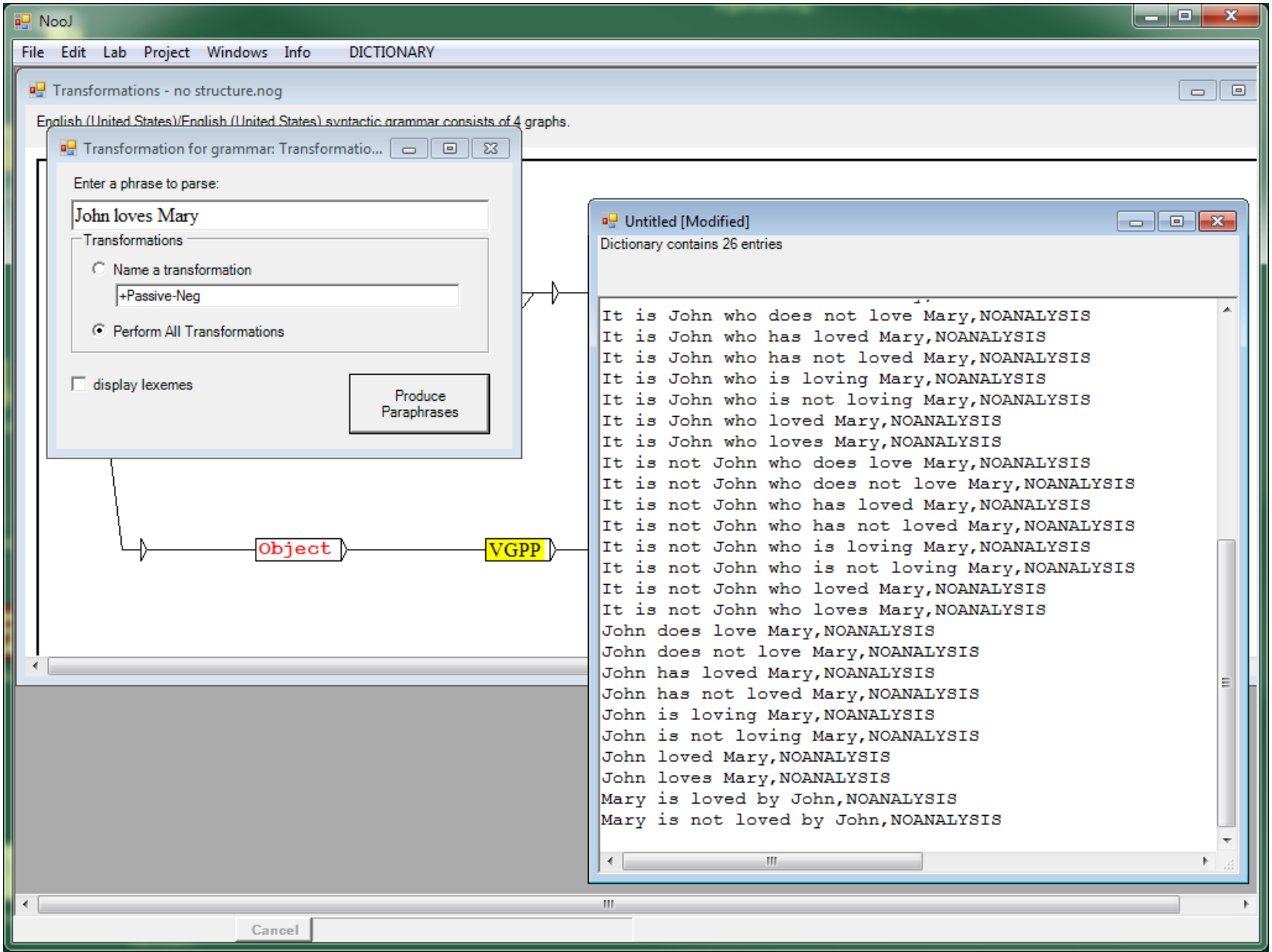


NooJ solution to transformations:
Combine Parsing and Generation

All variables references are linked

- Variables are set during the parsing step
- When parsing the text John loves Mary:
  - variable $Subject is set to “John”
  - variable $Verb is set to “loves”
  - variable $Object is set to “Mary”
- When exploring the graph to produce all the sequences recognized by the grammar, NooJ replaces the variables with their value

NooJ solution to transformations:
Combine Parsing and Generation



NooJ solution to transformations:
Combine Parsing and Generation

All variables references are linked

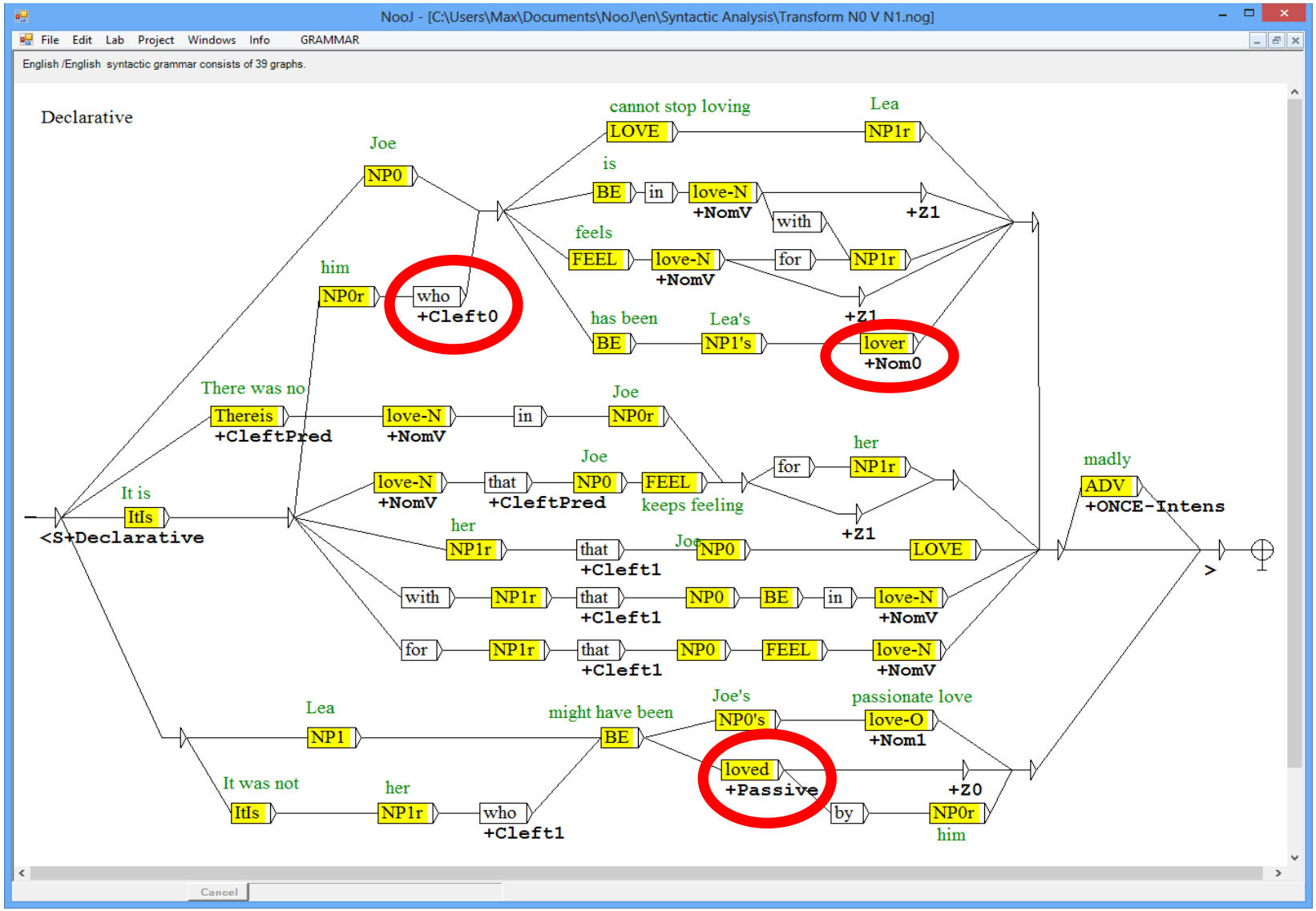
- Variables are set during the parsing step
- When parsing the text John loves Mary:
  - variable $Subject is set to “John”
  - variable $Verb is set to “loves”
  - variable $Object is set to “Mary”
- When exploring the graph to produce all the sequences recognized by the grammar, NooJ replaces the variables with their value

- The grammar may produce properties in its outputs, e.g., the name of the transformations: [Passive], [Pron1], [Aspect]...

NooJ solution to transformations:

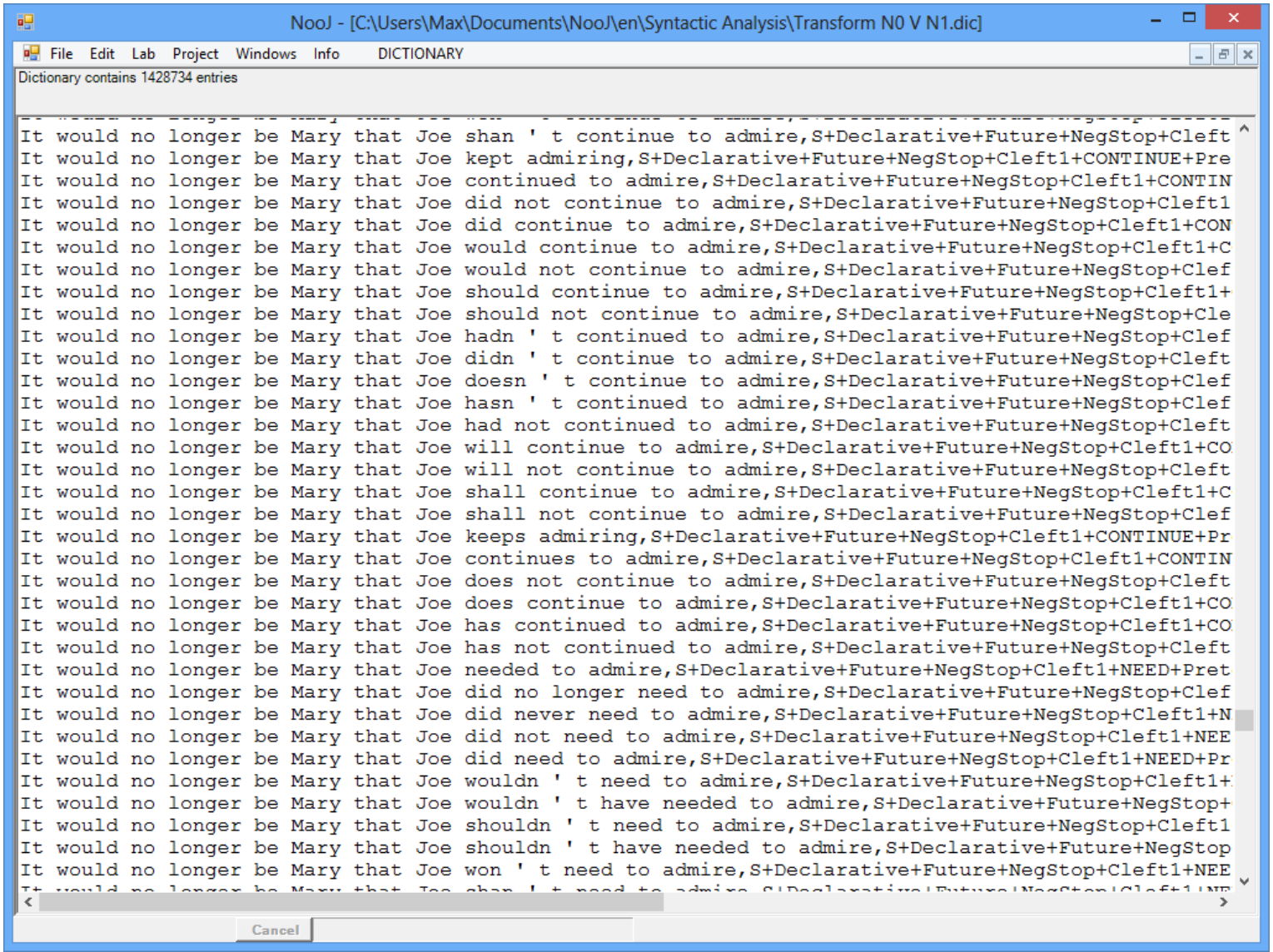
Combine Parsing and Generation

Name the transformations in the outputs



NooJ solution to transformations:
Combine Parsing and Generation

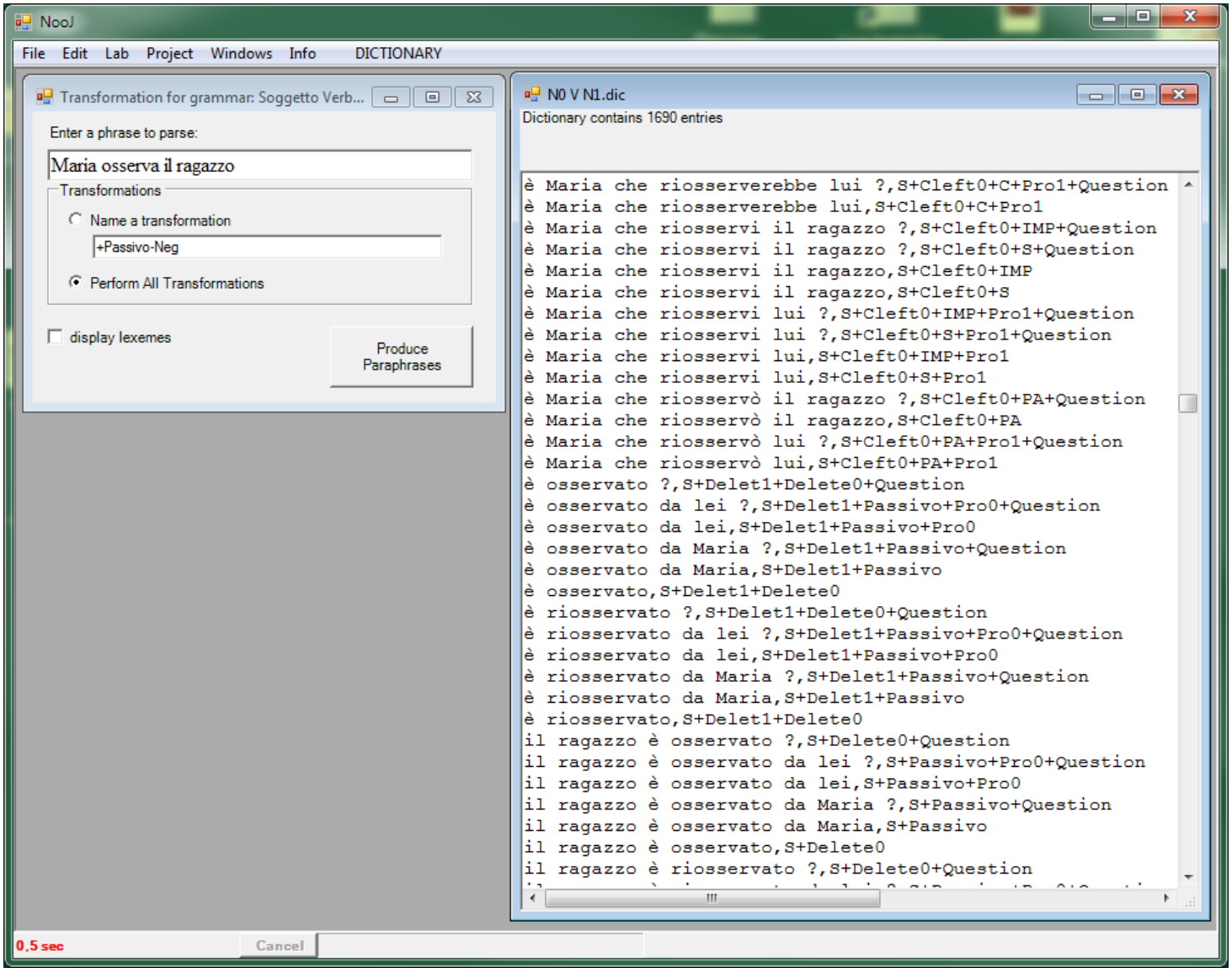
Produce all the transformed sentences from *Joe loves Lea*



NooJ solution to transformations:

In Italian

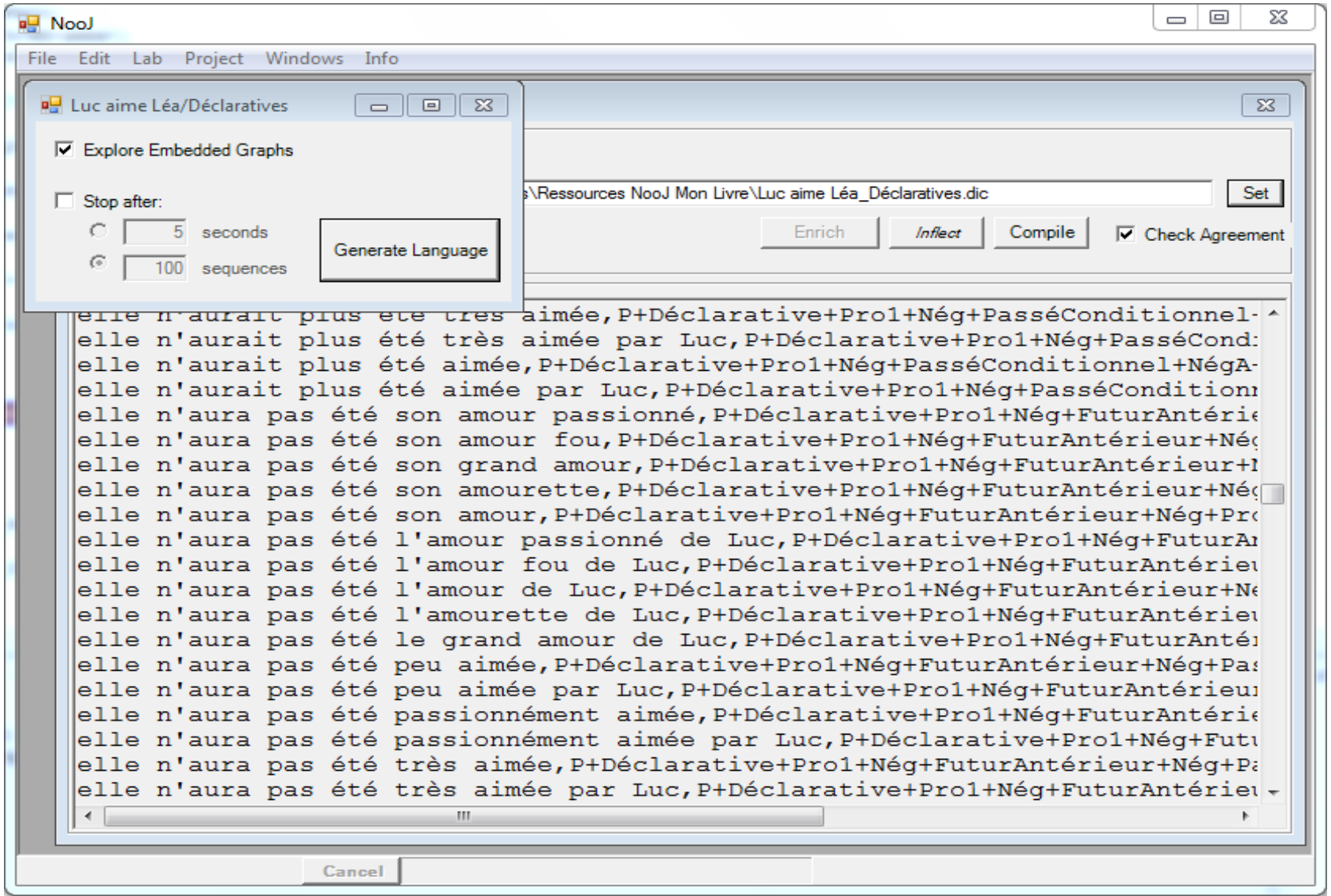
Produce all the transformed sentences from *Maria osserva om ragazzo*



NooJ solution to transformations:

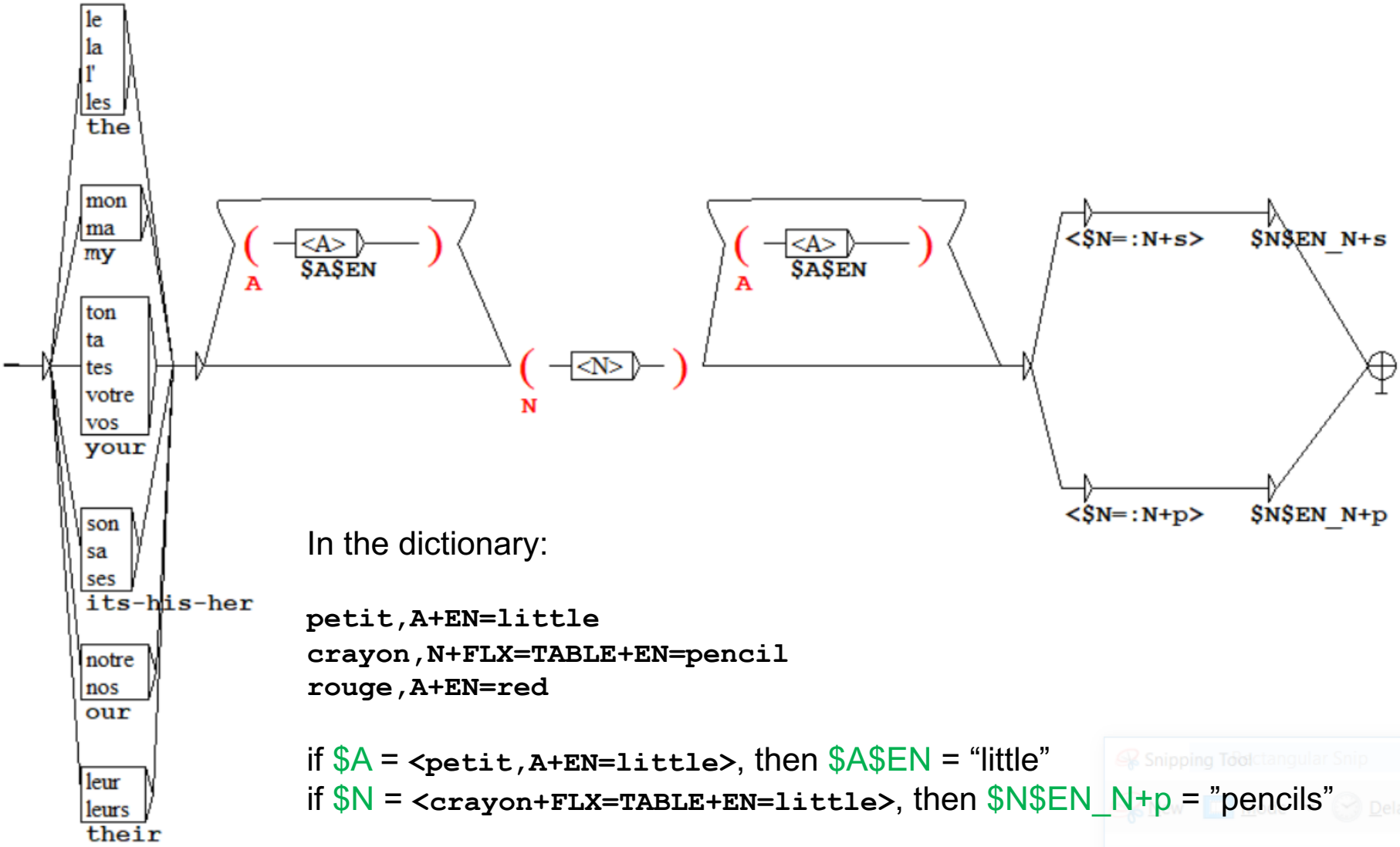
In French

Produce all the transformed sentences from *Luc aime Léa*



Unrestricted Grammars in NooJ

Machine Translation



In the dictionary:

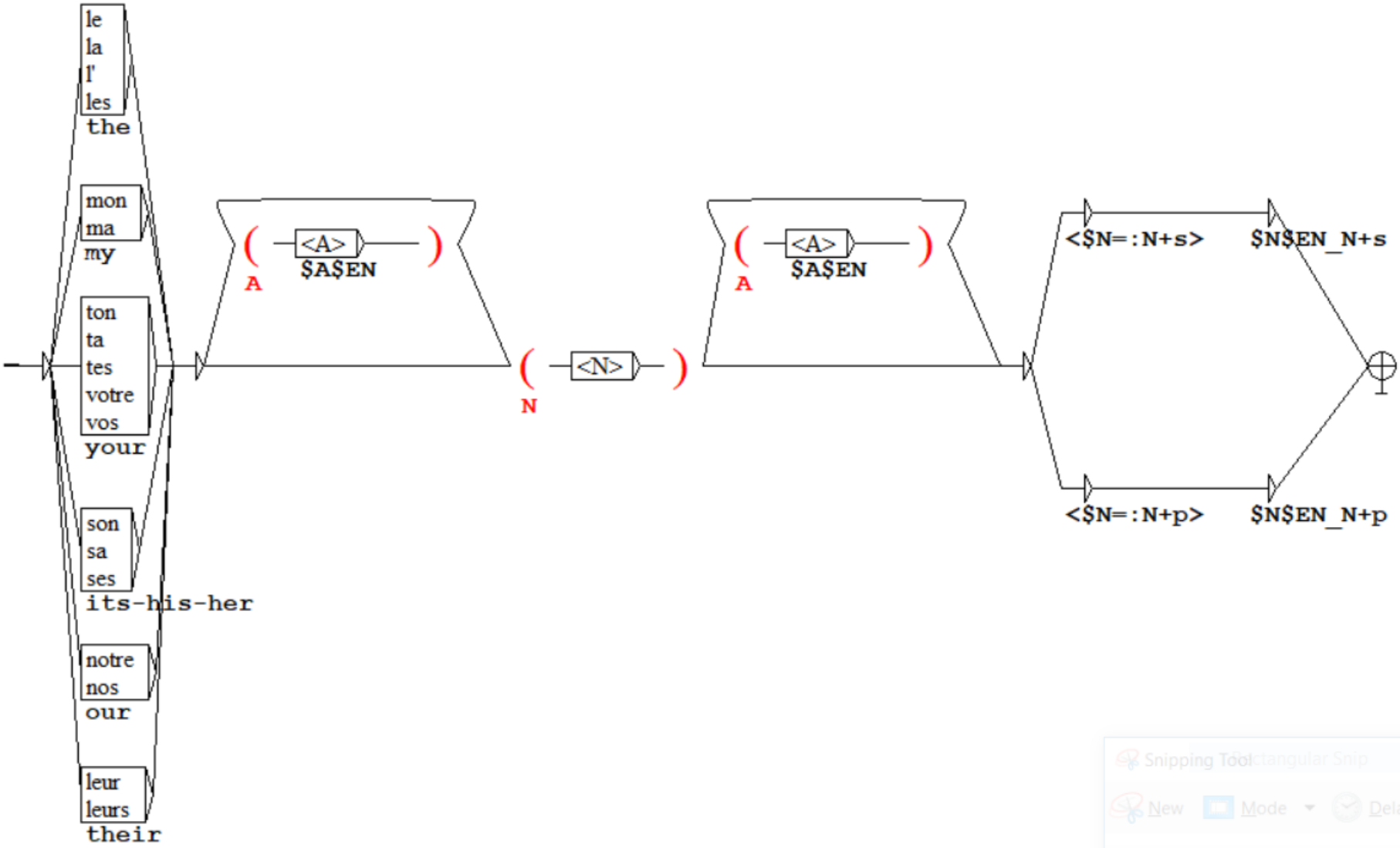
```
petit,A+EN=little
crayon,N+FLX=TABLE+EN=pencil
rouge,A+EN=red
```

if $A = `<petit,A+EN=little>`, then AEN = “little”

if $N = `<crayon+FLX=TABLE+EN=little>`, then NEN_N+p = ”pencils”

Unrestricted Grammars in NooJ

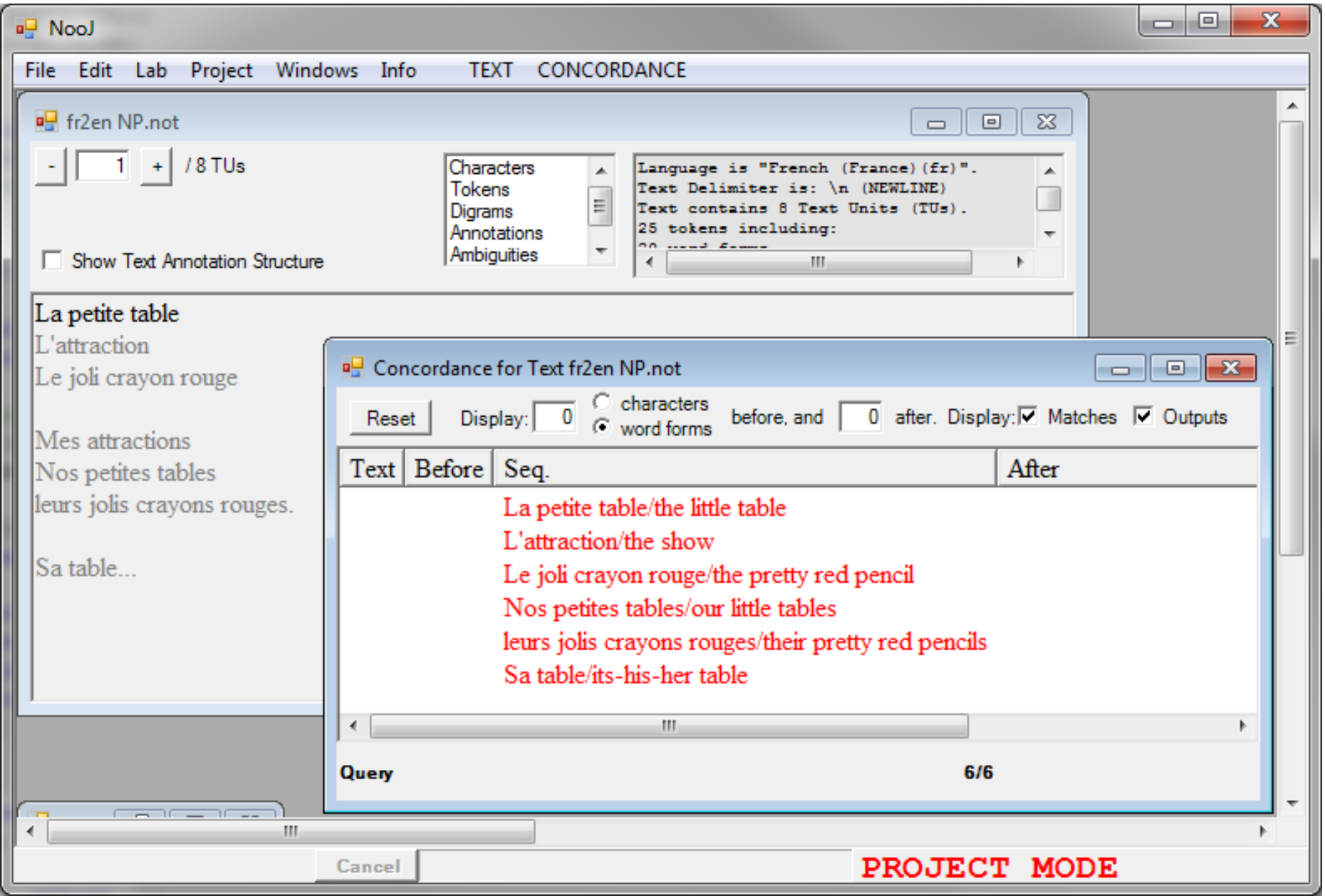
Machine Translation



le petit crayon rouge ⟶ the little red pen

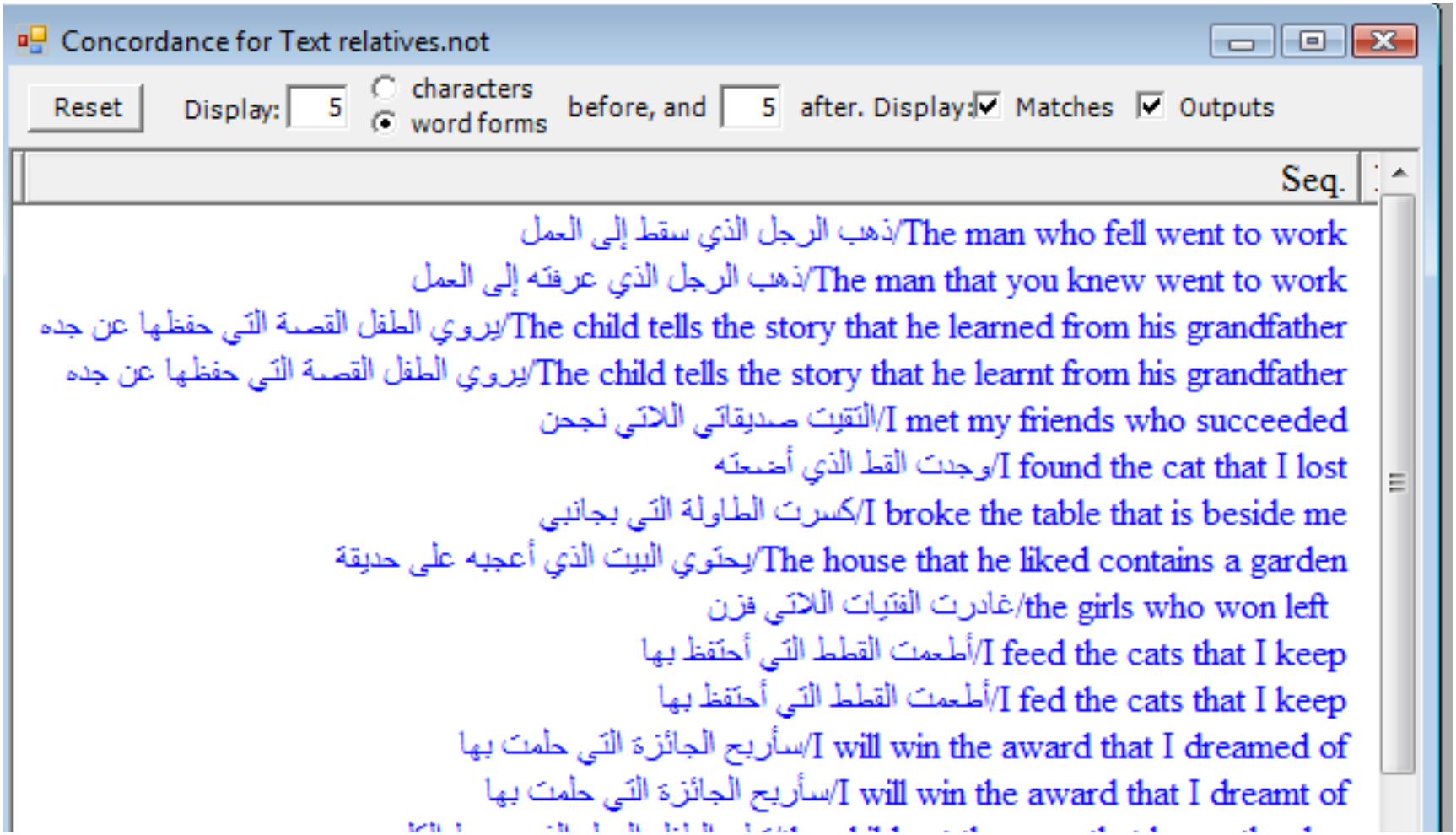
Unrestricted Grammars in NooJ

Machine Translation



Automatic Machine Translation

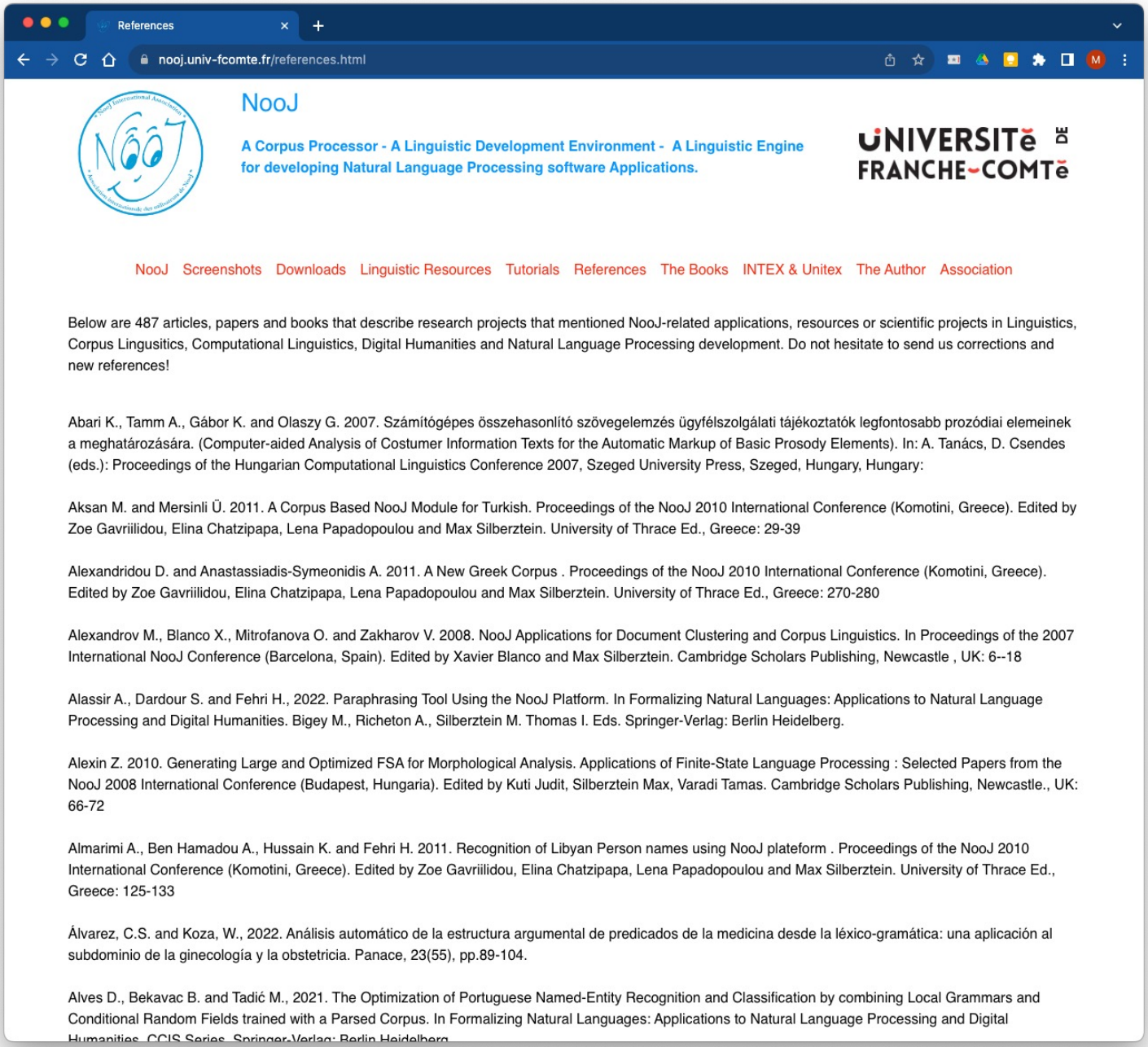
Arabic-French MT (Fehri, 2015-2022), Belarusian-Spanish (Veka, Yakubovich, 2016), Greek-Spanish MT (Papadopoulou, Chadjipapa, 2013), English-Italian MT (Maisto, Guarasci, 2016, Portuguese-English MT (Barreiro, 2008), etc.



Many applications of unrestricted grammars

- Transcribe French⇄Arabic proper names
- Transcribe Phonemic⇄Orthographic notations
- Rewrite a scientific medical text with simpler paraphrases
- Parse a text, and produce its abstract
- Parse a question, *e.g.*, “Who built the Eiffel Tower”, and produce all the potential answers “X built the Eiffel Tower”, “The Eiffel Tower was built by X”, “X has been instrumental in building the Eiffel Tower”, etc. then look for these answer on the Web
- Parse a text, and produce its semantic representation, either in Predicate form GIVE(Joe, apple, Eva), or in XML RDF format
- etc.

Many applications (cf. references)



References

nooj.univ-fcomte.fr/references.html

NooJ

A Corpus Processor - A Linguistic Development Environment - A Linguistic Engine for developing Natural Language Processing software Applications.

UNIVERSITÉ DE FRANCHE-COMTÉ

NooJ Screenshots Downloads Linguistic Resources Tutorials References The Books INTEX & Unitex The Author Association

Below are 487 articles, papers and books that describe research projects that mentioned NooJ-related applications, resources or scientific projects in Linguistics, Corpus Lingusitics, Computational Linguistics, Digital Humanities and Natural Language Processing development. Do not hesitate to send us corrections and new references!

Abari K., Tamm A., Gábor K. and Olaszy G. 2007. Számítógépes összehasonlító szövegelemzés ügyfélszolgálati tájékoztatók legfontosabb prozódiai elemeinek a meghatározására. (Computer-aided Analysis of Costumer Information Texts for the Automatic Markup of Basic Prosody Elements). In: A. Tanács, D. Csendes (eds.): Proceedings of the Hungarian Computational Linguistics Conference 2007, Szeged University Press, Szeged, Hungary, Hungary:

Aksan M. and Mersinli Ü. 2011. A Corpus Based NooJ Module for Turkish. Proceedings of the NooJ 2010 International Conference (Komotini, Greece). Edited by Zoe Gavriilidou, Elina Chatzipapa, Lena Papadopoulou and Max Silberztein. University of Thrace Ed., Greece: 29-39

Alexandridou D. and Anastassiadis-Symeonidis A. 2011. A New Greek Corpus . Proceedings of the NooJ 2010 International Conference (Komotini, Greece). Edited by Zoe Gavriilidou, Elina Chatzipapa, Lena Papadopoulou and Max Silberztein. University of Thrace Ed., Greece: 270-280

Alexandrov M., Blanco X., Mitrofanova O. and Zakharov V. 2008. NooJ Applications for Document Clustering and Corpus Linguistics. In Proceedings of the 2007 International NooJ Conference (Barcelona, Spain). Edited by Xavier Blanco and Max Silberztein. Cambridge Scholars Publishing, Newcastle , UK: 6--18

Alassir A., Dardour S. and Fehri H., 2022. Paraphrasing Tool Using the NooJ Platform. In Formalizing Natural Languages: Applications to Natural Language Processing and Digital Humanities. Bigey M., Richeton A., Silberztein M. Thomas I. Eds. Springer-Verlag: Berlin Heidelberg.

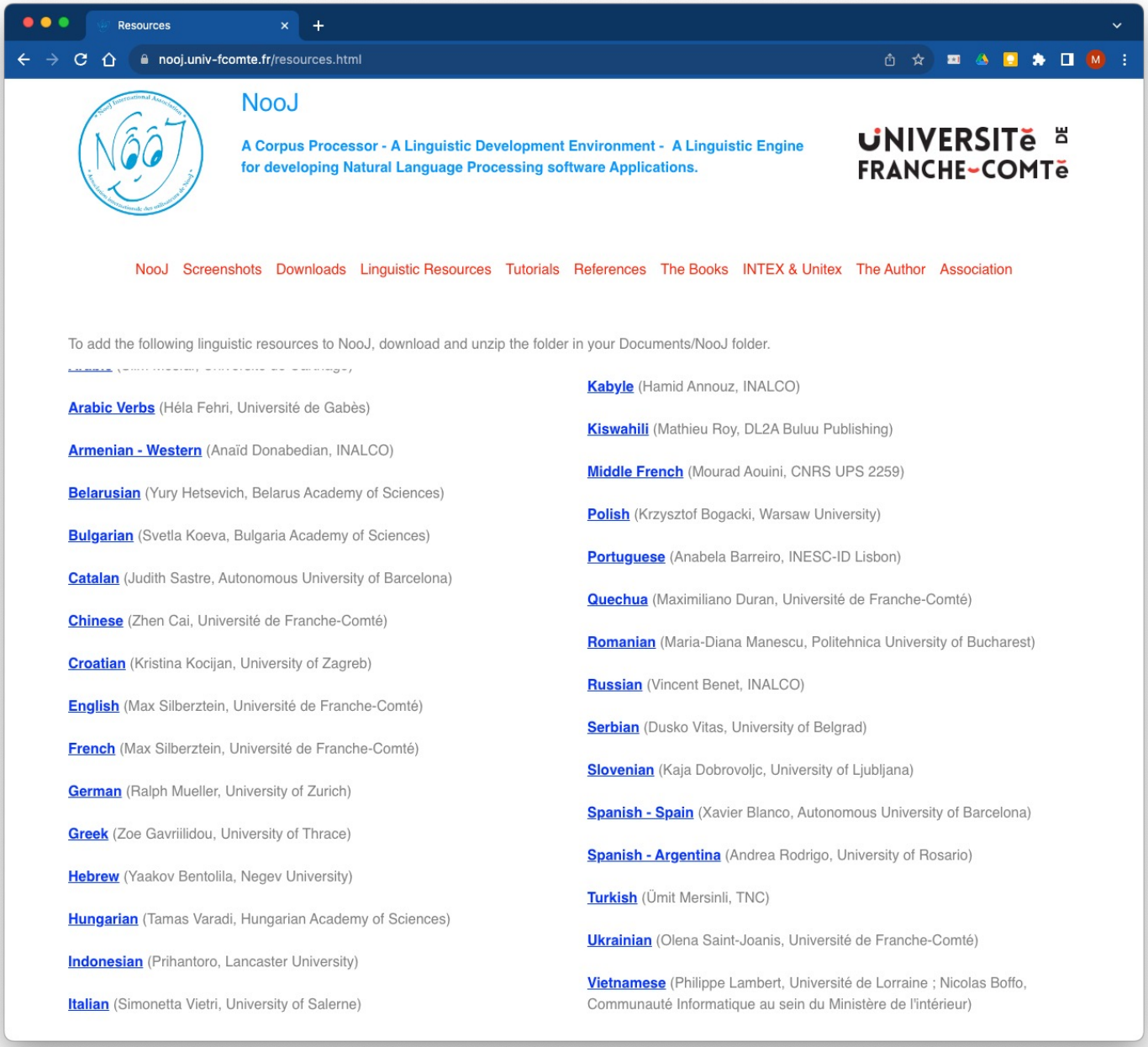
Alexin Z. 2010. Generating Large and Optimized FSA for Morphological Analysis. Applications of Finite-State Language Processing : Selected Papers from the NooJ 2008 International Conference (Budapest, Hungaria). Edited by Kuti Judit, Silberztein Max, Varadi Tamas. Cambridge Scholars Publishing, Newcastle., UK: 66-72

Almarimi A., Ben Hamadou A., Hussain K. and Fehri H. 2011. Recognition of Libyan Person names using NooJ plateform . Proceedings of the NooJ 2010 International Conference (Komotini, Greece). Edited by Zoe Gavriilidou, Elina Chatzipapa, Lena Papadopoulou and Max Silberztein. University of Thrace Ed., Greece: 125-133

Álvarez, C.S. and Koza, W., 2022. Análisis automático de la estructura argumental de predicados de la medicina desde la léxico-gramática: una aplicación al subdominio de la ginecología y la obstetricia. Panace, 23(55), pp.89-104.

Alves D., Bekavac B. and Tadić M., 2021. The Optimization of Portuguese Named-Entity Recognition and Classification by combining Local Grammars and Conditional Random Fields trained with a Parsed Corpus. In Formalizing Natural Languages: Applications to Natural Language Processing and Digital Humanities. CCIS Series. Springer-Verlag: Berlin Heidelberg

Many language resources (30 modules downloadable)



CONGRATULATIONS

You know how to construct unrestricted grammars to perform transformations, compute paraphrases, produce semantic analyses and compute translations.