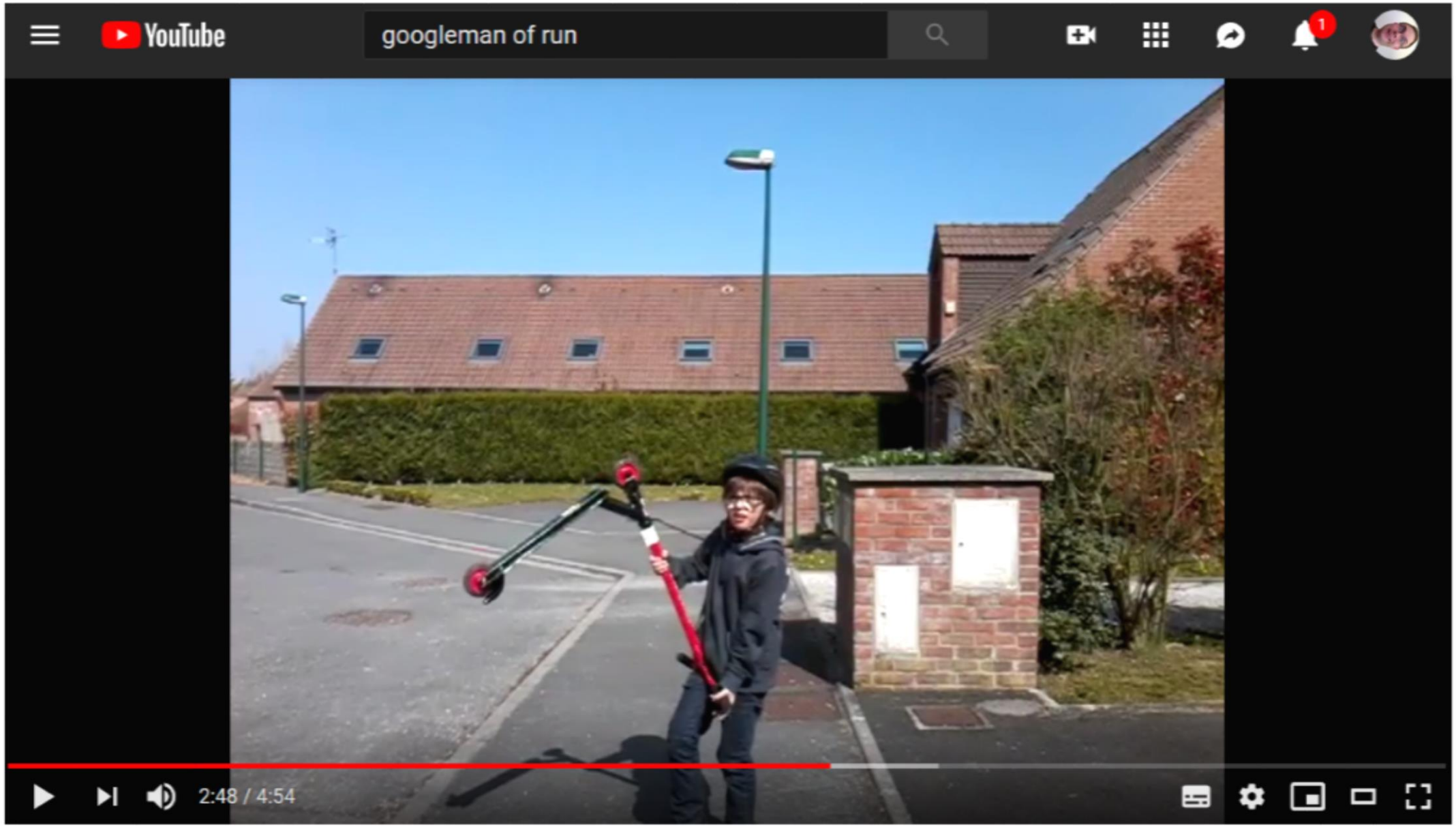
Machine learning techniques for semistructured data

Infer queries and transformations with grammatical inference tools

Aurélien Lemay

ANR DELTA

Université de Lille

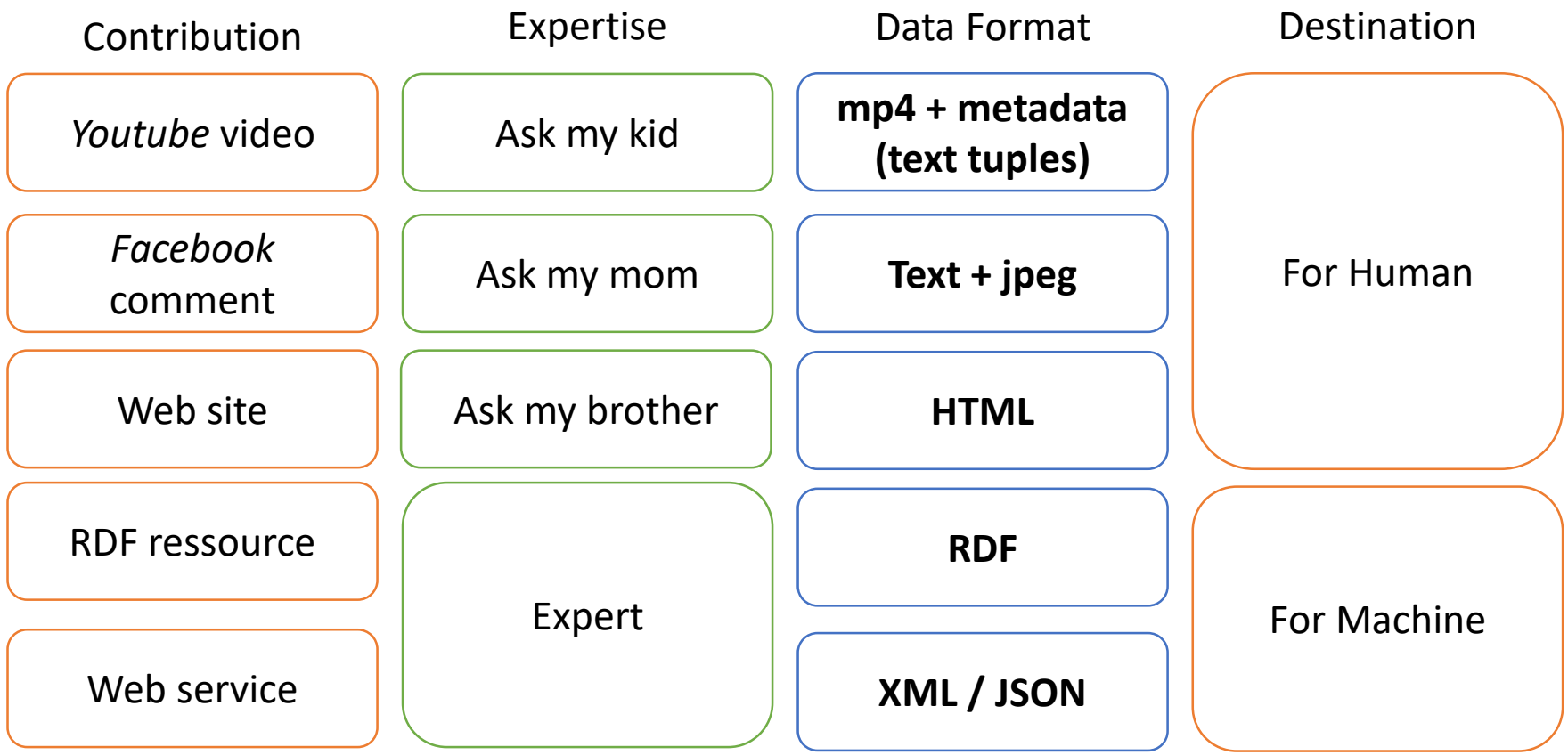


Comment faire un bart spin,briflip et front briflip ?

13 vues

2 1 PARTAGER ENREGISTRER

Personal and professional data: Today, everyone can contribute !



The Formats of Semistructured Data

Structure	Schema Validation	Query	Transformation
Data Tree	DTD XML Schema Schematron …	XPath (1.0, 2.0, 3.0)…	Xquery 2.0, XSLT 3.1, …

The Formats of Semistructured Data

Structure	Schema Validation	Query	Transformation
Data Tree	DTD XML Schema Schematron …	XPath (1.0, 2.0, 3.0)…	Xquery 2.0, XSLT 3.1, …
Data Tree	*Dynamic updates through JavaScript*		

The Formats of Semistructured Data

	Structure	Schema Validation	Query	Transformation
XML, HTML	Data Tree	DTD XML Schema Schematron …	XPath (1.0, 2.0, 3.0)…	Xquery 2.0, XSLT 3.1, …
JSON	Data Tree	*Dynamic updates through JavaScript*		
RDF	Data Graph + ontology	*ShEx*	SPARQL	Data Exchange Tools

The Formats of Semistructured Data

	Structure	Schema Validation	Query	Transformation
XML / HTML	Data Tree	DTD XML Schema Schematron …	XPath (1.0, 2.0, 3.0)…	Xquery 2.0, XSLT 3.1, …
JSON	Data Tree	*Dynamic updates through JavaScript*		
RDF	Data Graph + ontology	*ShEx*	SPARQL	Data Exchange Tools

Many Different Formats, Needs Expertise
Proposition : Use Machine Learning !

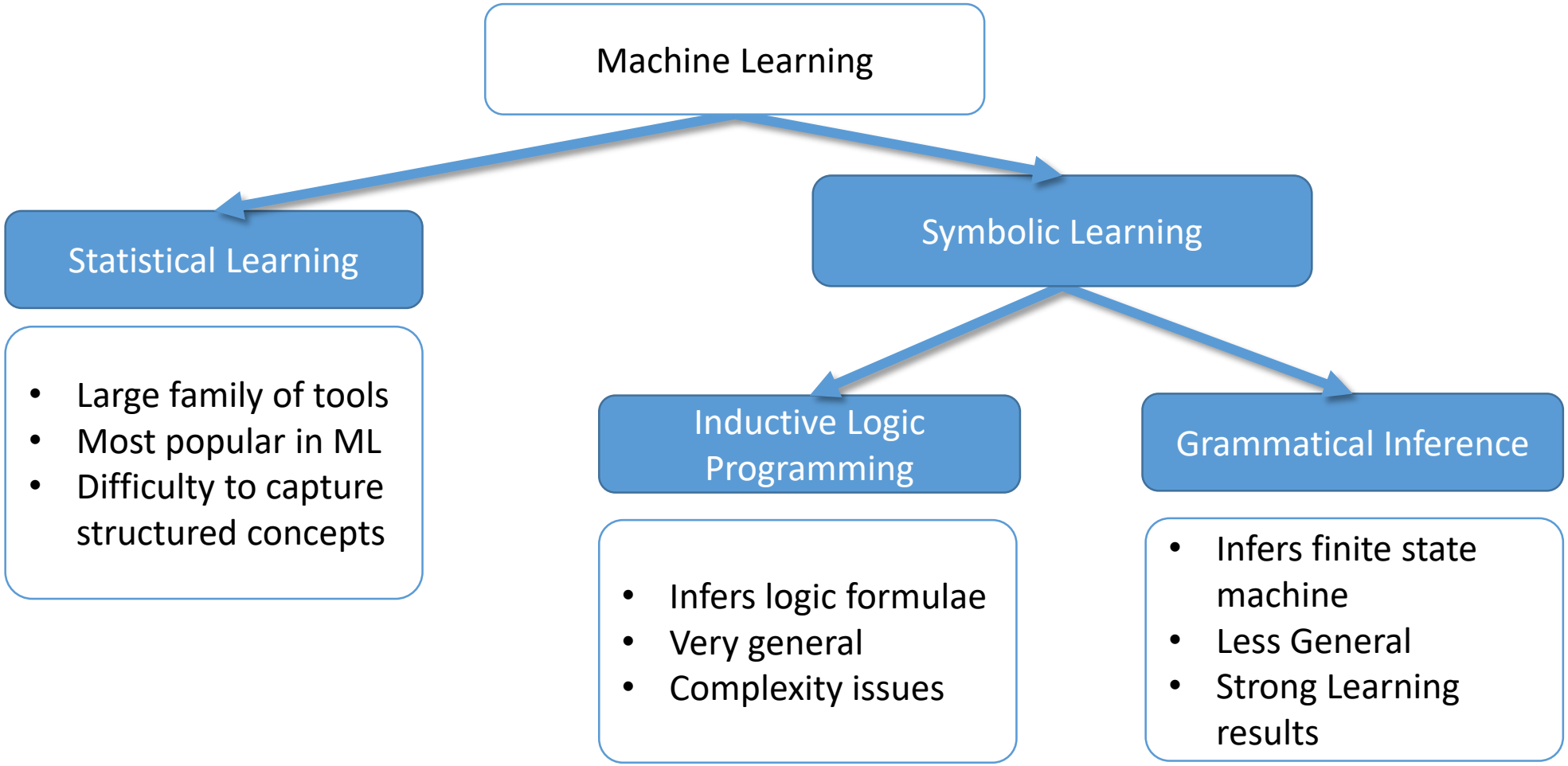
Query Formalisation (navigational aspects)

- **“core” SQL** : First Order Logic [Abiteboul Hull Vianu’95]
- **Core XPath 1.0 / Navigational XPath :** fragment of FO_{tree} [Gottlob Koch Pichler 02, Benedikt Fan Kuper’03]
- **Core XPath 2.0 / Conditional XPath**: FO_{tree} [Marx’05]
- **Regular XPath** : FO^*_{tree} [ten Cate Marx’07]
  - Included in Monadic Second Order Logic [ten Cate Segoufin’08]

Monadic Second Order Logic

- Extends First Order logic (FO)
- Allows recursion
- Strong link with finite state machines

Learning Queries From Examples



Outline

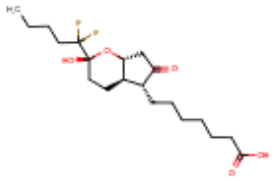
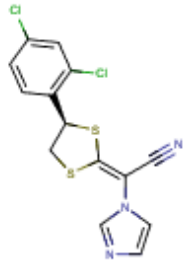
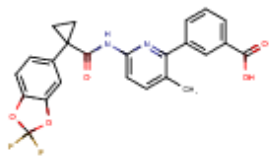
I – Learning Tree Queries

II – Learning Tree Transformations

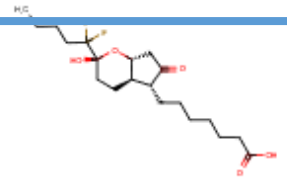
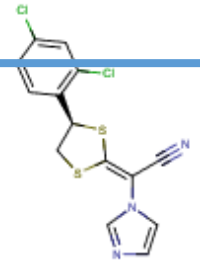
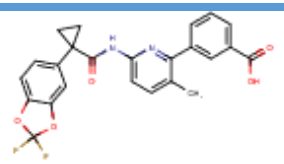
III – Future Works

Part 1
Learning Tree Queries

Displaying drugs **1376 - 1400** of **2526** in total

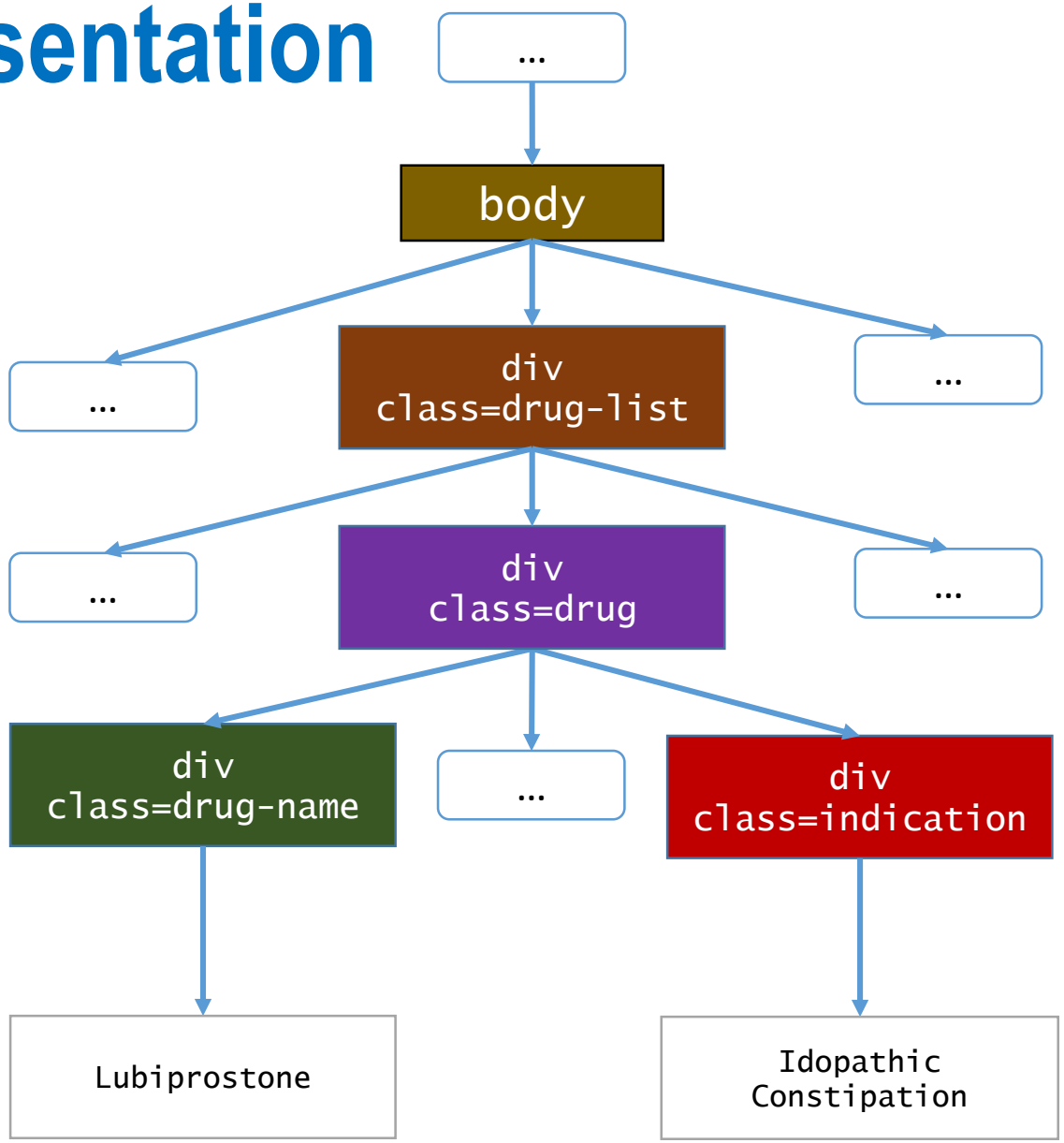
NAME	WEIGHT	STRUCTURE	THERAPEUTIC INDICATION	CATEGORIES
Lubiprostone	390.468 $C_{20}H_{32}F_2O_5$		For the treatment of chronic idiopathic constipation in the adult population. Also used for the treatment of irritable bowel syndrome with constipa...	Alprostadil / Chloride Channel Agonists
Luliconazole	354.27 $C_{14}H_9Cl_2N_3S_2$		Luliconazole is indicated in adults aged 18 years and older for the topical treatment of fungal infections caused by Trichophyton rubrum and Epider...	Imidazole and Triazole Derivatives
Lumacaftor	452.414 $C_{24}H_{18}F_2N_2O_5$		When given in combination with [DB08820] as the fixed dose combination product Orkambi, lumacaftor is indicated for the treatment of cystic fibrosi...	Cystic Fibrosis Transmembrane Conductance Regulator

Displaying drugs **1376 - 1400** of **2526** in total

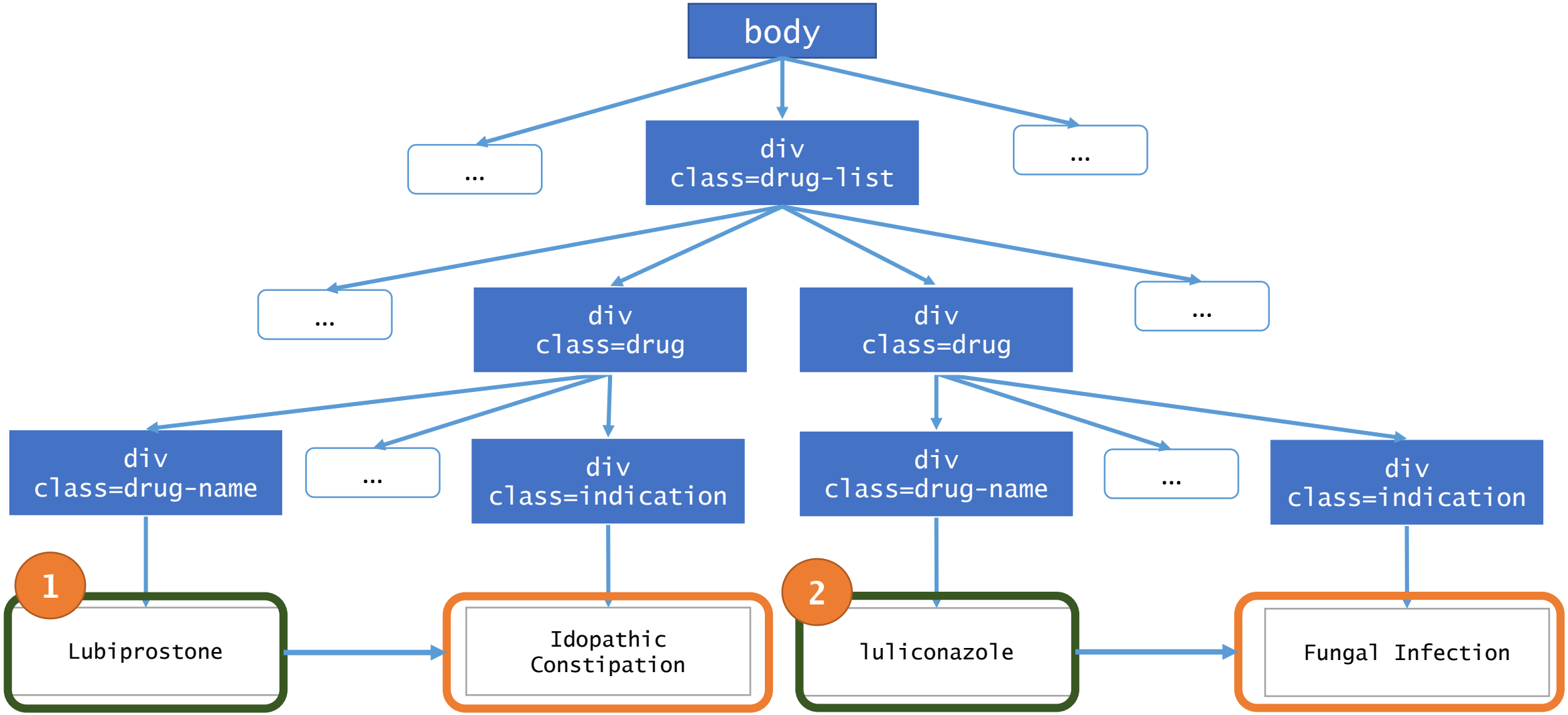
NAME	WEIGHT	STRUCTURE	THERAPEUTIC INDICATION	CATEGORIES
1 **Lubiprostone**	390.468 $C_{20}H_{32}F_2O_5$		For the treatment of chronic idiopathic constipation in the adult population. Also used for the treatment of irritable bowel syndrome with constipa...	Alprostadil / Chloride Channel Agonists
2 **Luliconazole**	354.27 $C_{14}H_9Cl_2N_3S_2$		Luliconazole is indicated in adults aged 18 years and older for the topical treatment of fungal infections caused by Trichophyton rubrum and Epider...	Imidazole and Triazole Derivatives
3 **Lumacaftor**	452.414 $C_{24}H_{18}F_2N_2O_5$		When given in combination with [DB08820] as the fixed dose combination product Orkambi, lumacaftor is indicated for the treatment of cystic fibrosi...	Cystic Fibrosis Transmembrane Conductance Regulator

Tree Representation

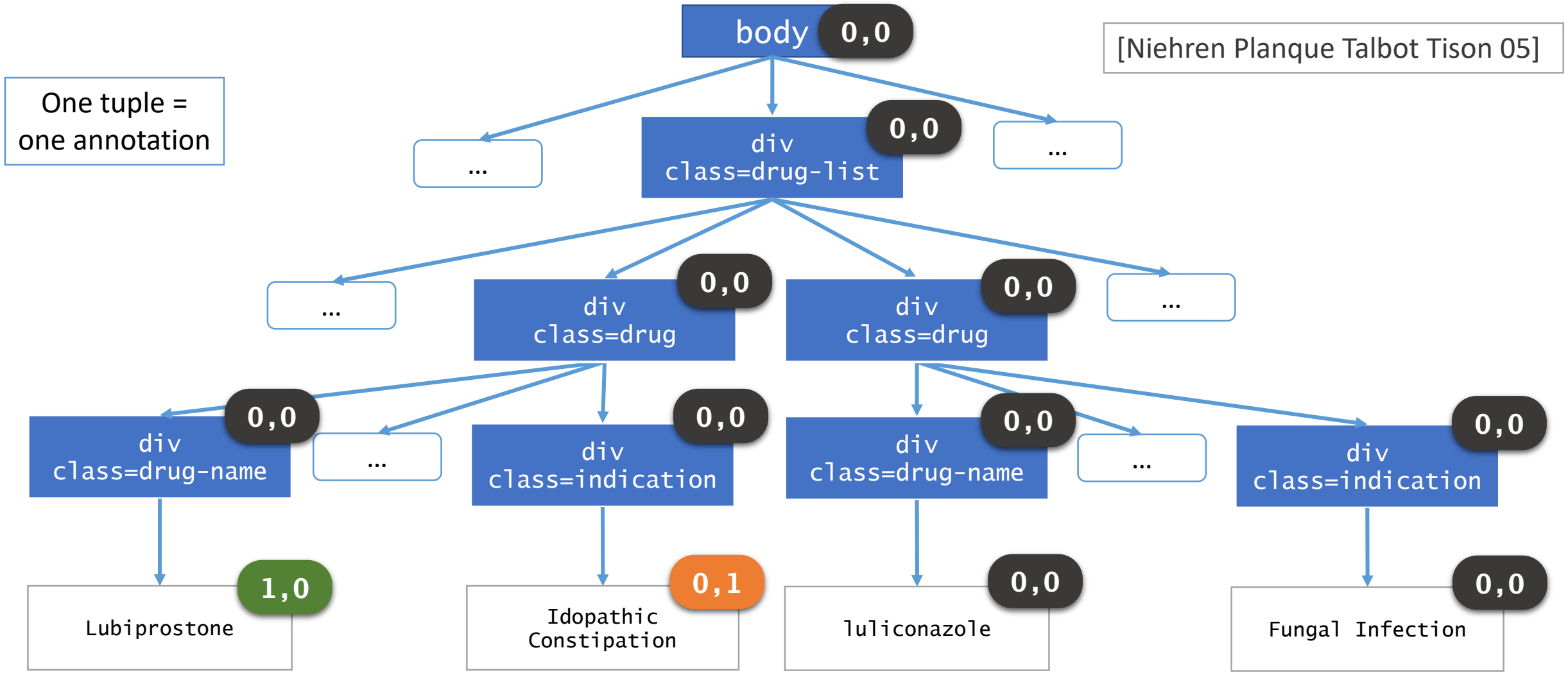
```
<html>
<head> … </head>
<body>
…
<div class='drug-list'>
 <div class='drug'>
  <div class='drug-name'>
   <a href='http://…'>Lubiprostone</a>
  </div>
  …
  <div class='indication'>
    Idiopathic Constipation
  </div>
 </div>
 …
</div>
…
</body>
</html>
```



Tree Query

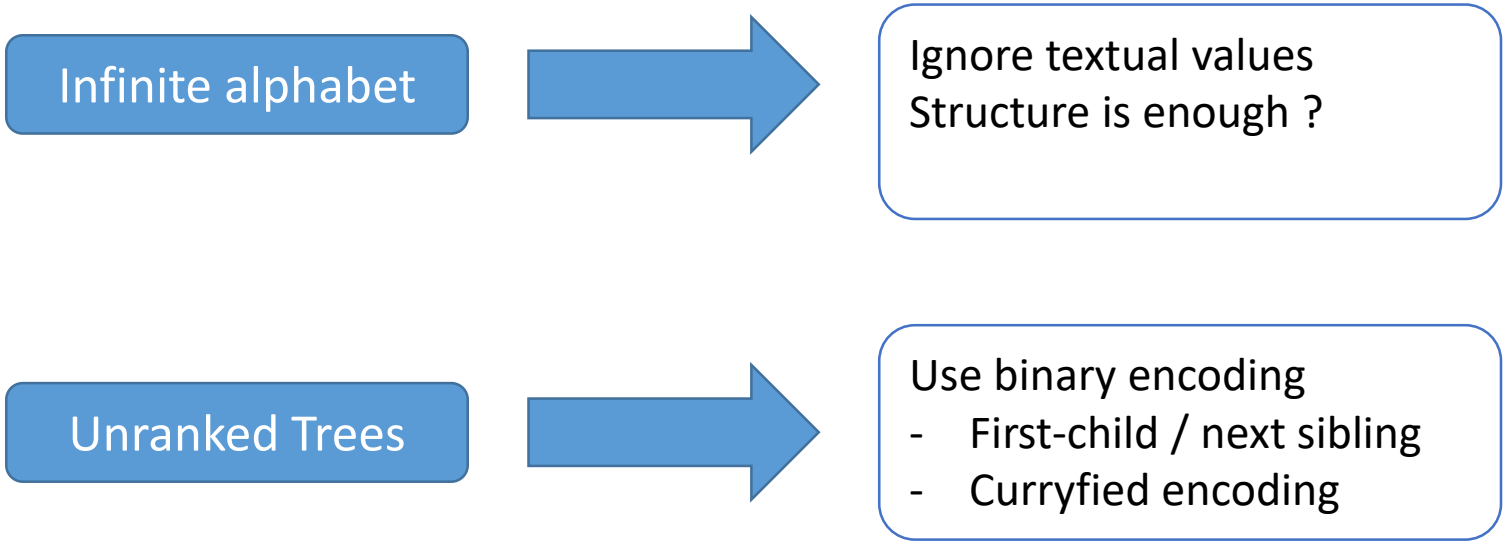

body
...
div
class=drug-list
...
...
div
class=drug
div
class=drug
...
div
class=drug-name
...
div
class=indication
div
class=drug-name
...
div
class=indication
1
Lubiprostone
Idopathic
Constipation
2
luliconazole
Fungal Infection

Query as Tree Language : *extract drug names / indication*



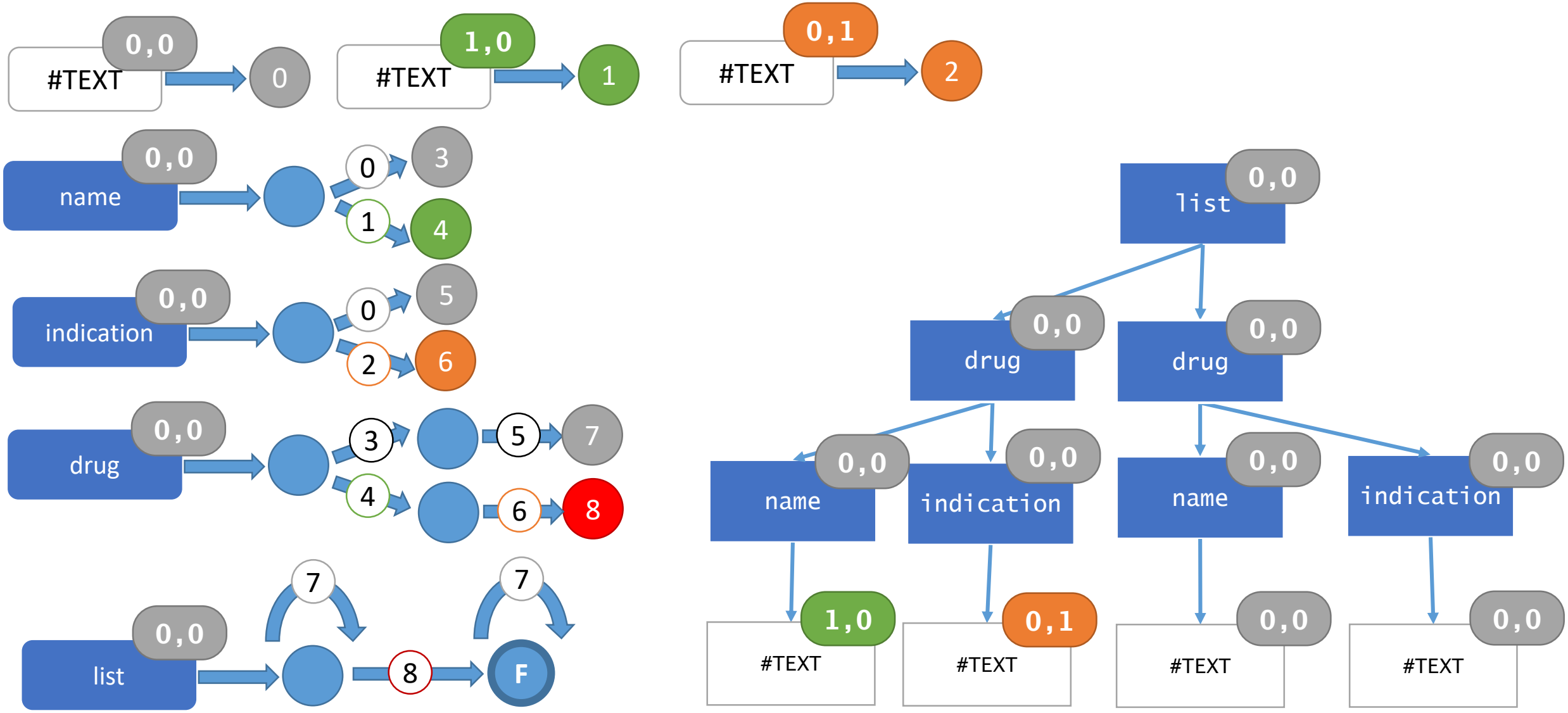
Representing Tree Queries with Tree Automata

Two problems :

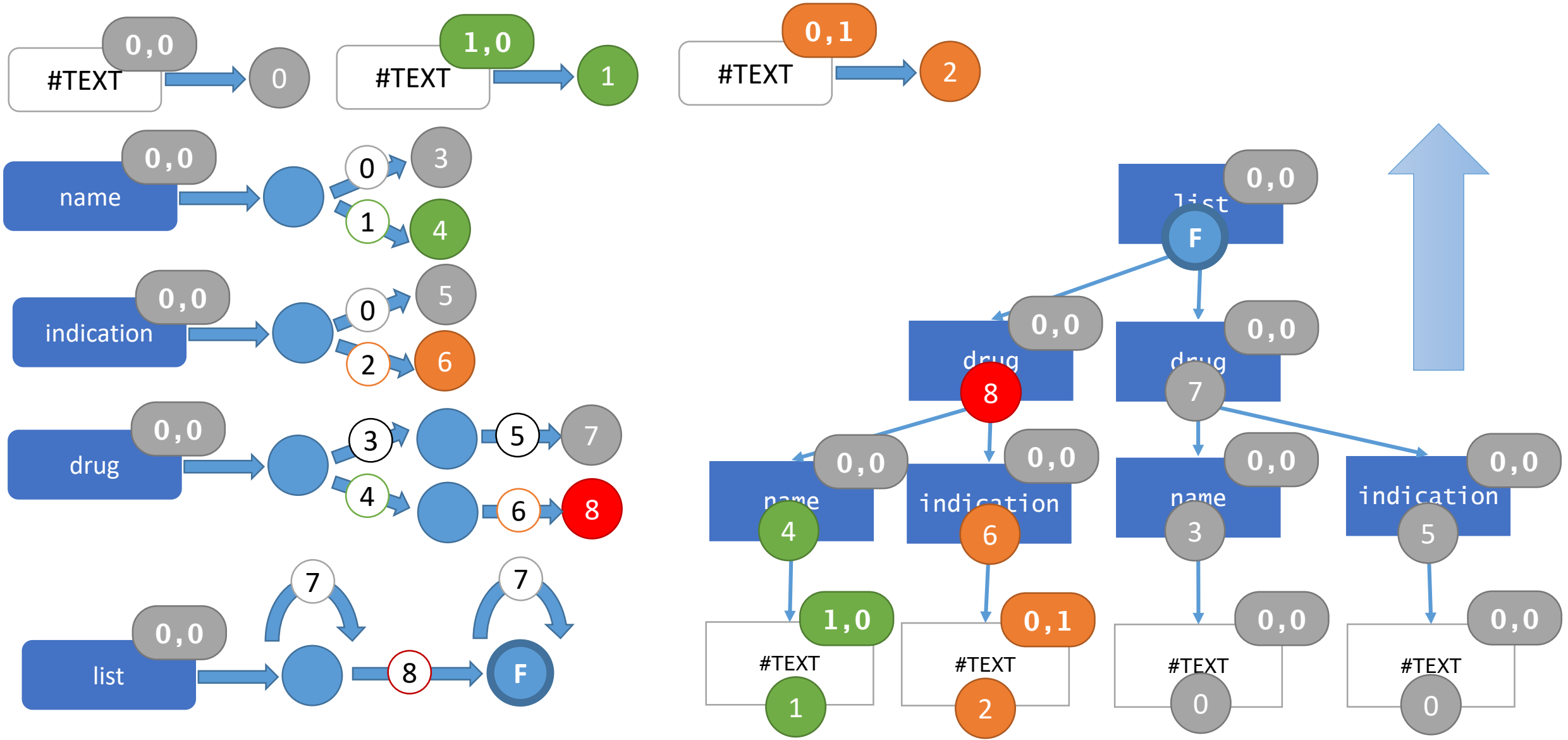


Node Selecting Tree Transducers :
Tree automata that use **Curryfied encoding** and recognizes **annotated tree** languages

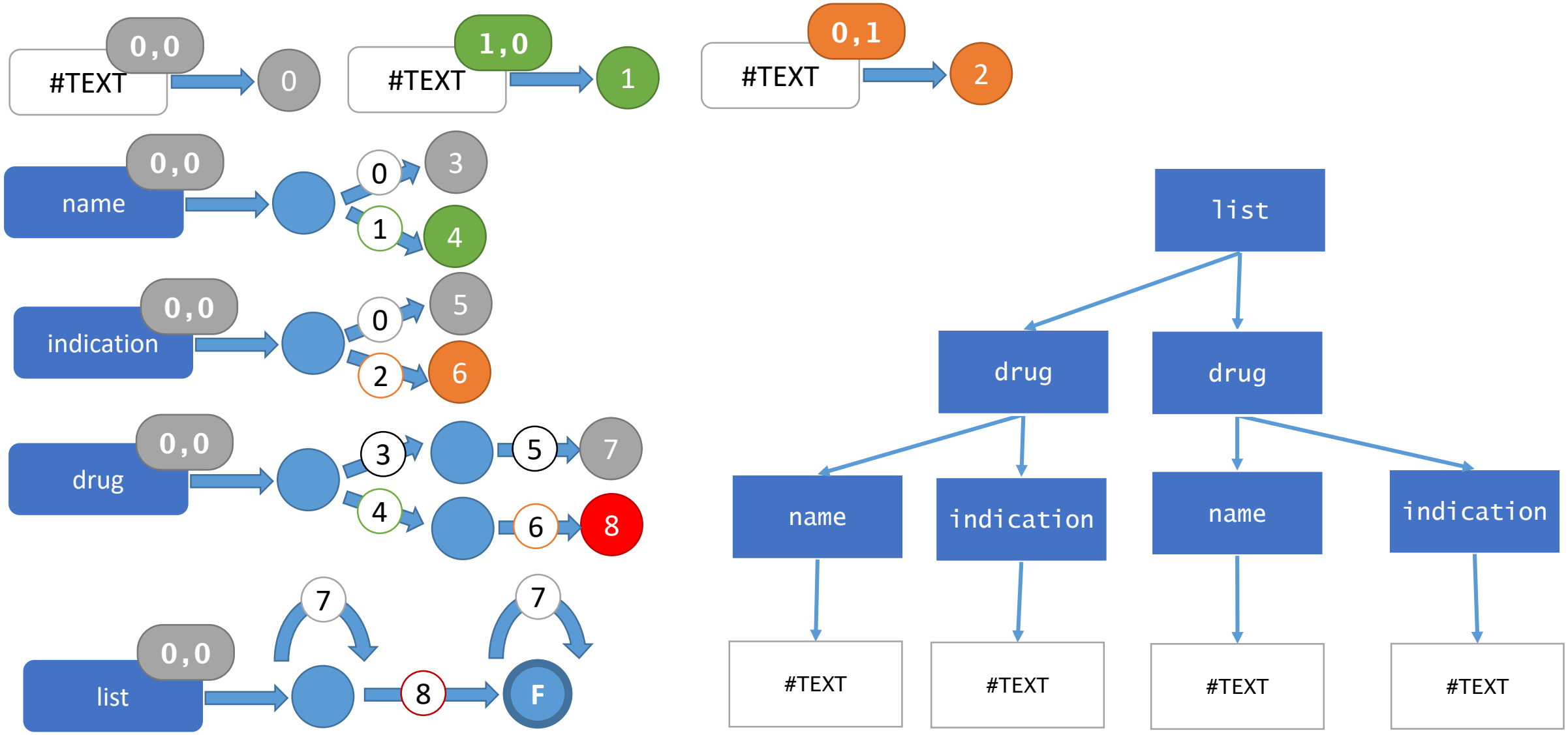
Node Selecting Tree Transducers (NSTT)



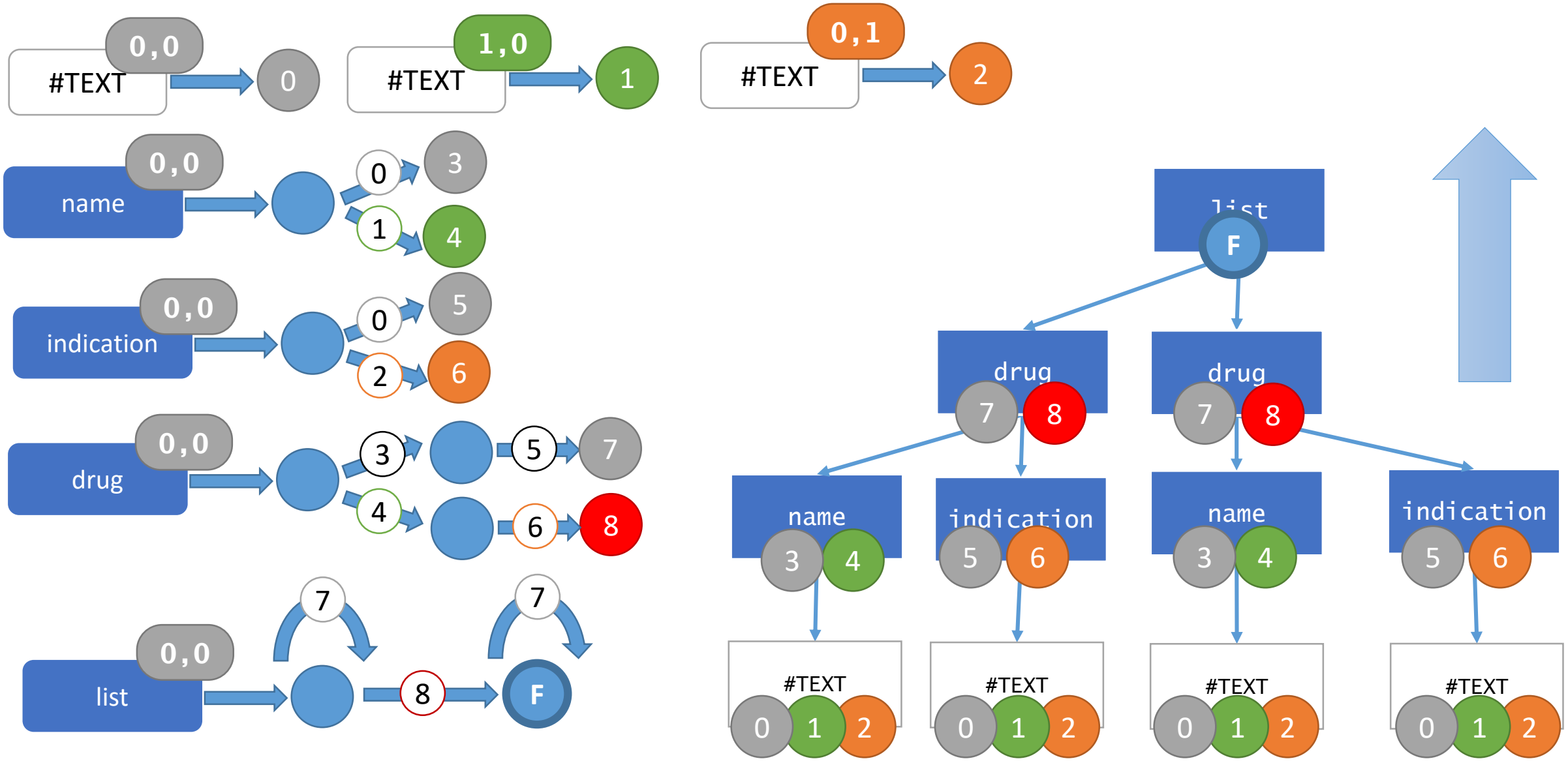
Node Selecting Tree Transducers (NSTT)



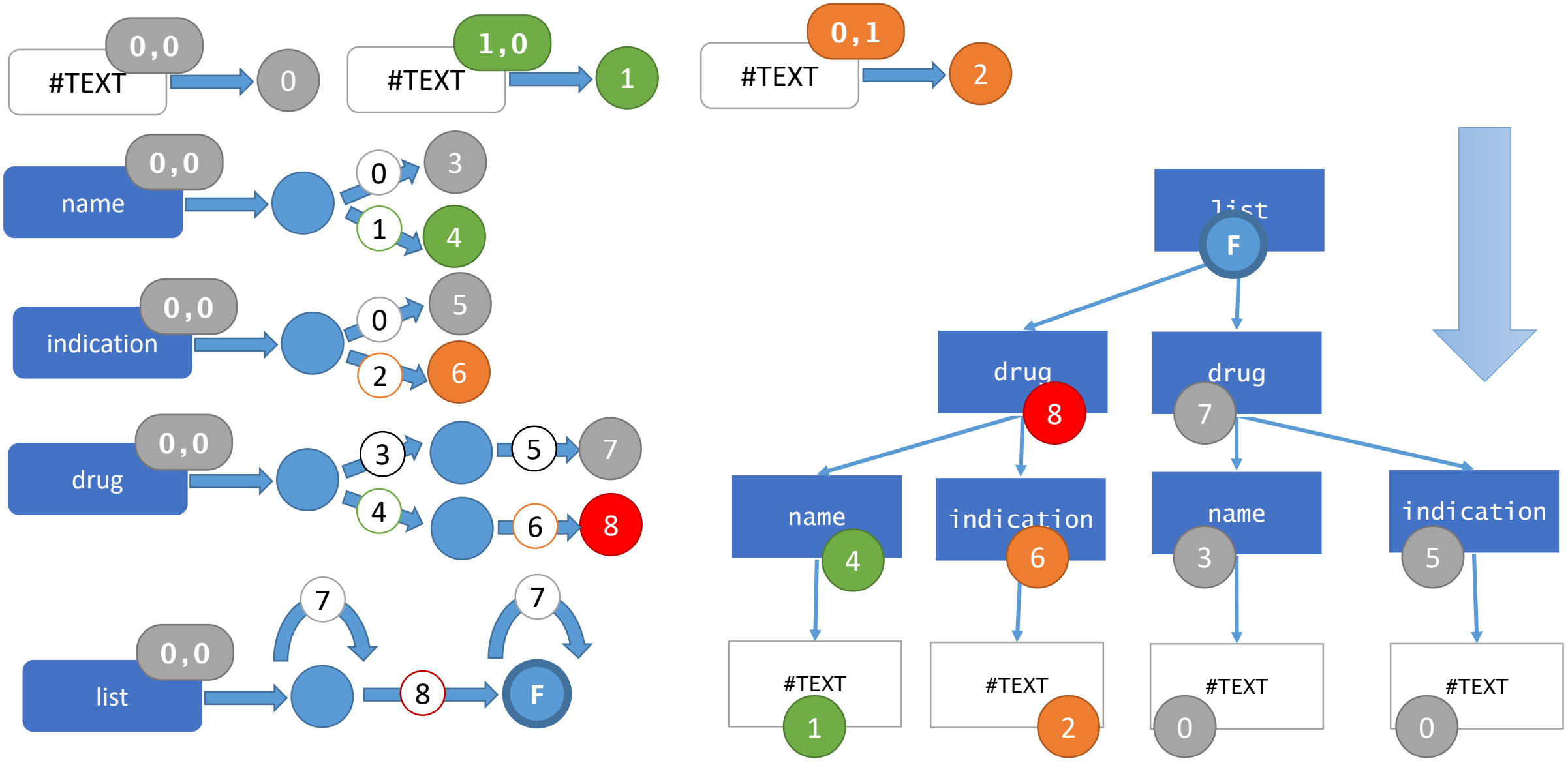
Node Selecting Tree Transducers (NSTT)



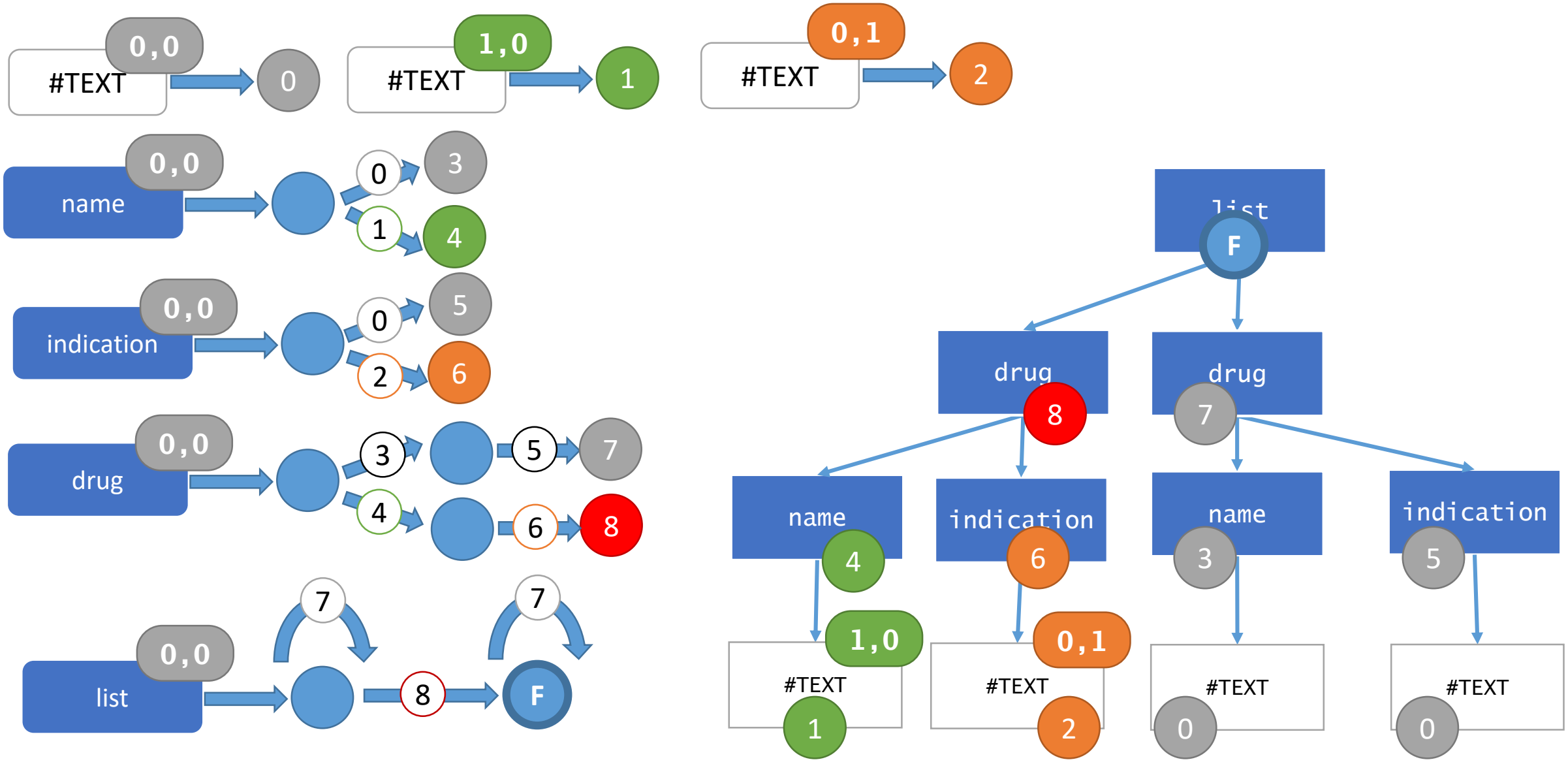
Node Selecting Tree Transducers (NSTT)



Node Selecting Tree Transducers (NSTT)



Node Selecting Tree Transducers (NSTT)



Node Selecting Tree Transducers

[Carme Gilleron Lemay Niehren'05]

Equivalent to :

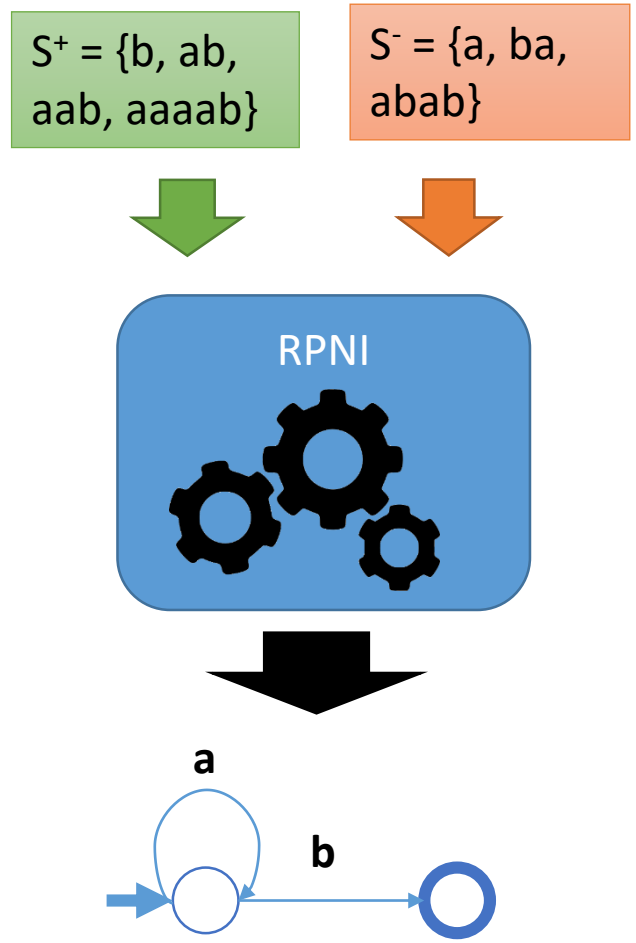
- **deterministic tree automata** for the tree language
  - One annotated trees per tuple
- **non-deterministic transducer** (*relabeling*)
  - one run per tuple

- Operates on **unranked trees (Curryfied encoding)**
- Equivalent to **MSO** queries on trees
- **Ignore textual values** in leaves

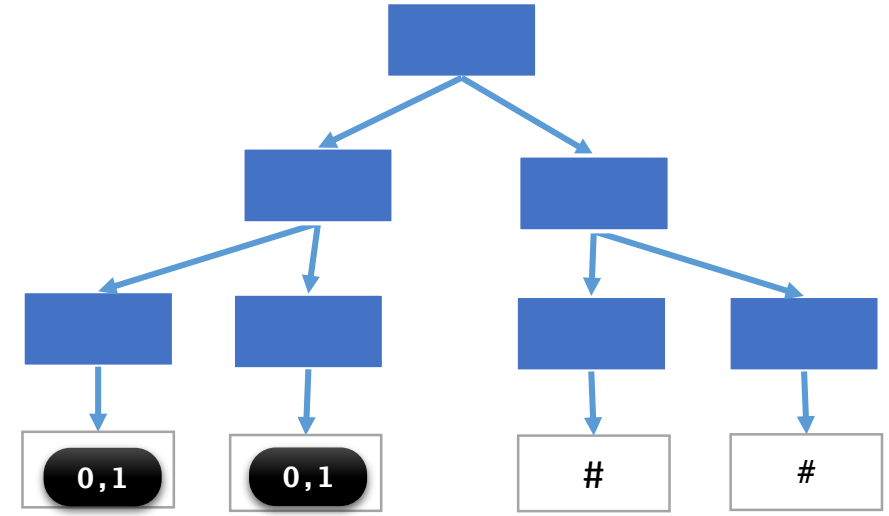
RPNI Learning Algorithm

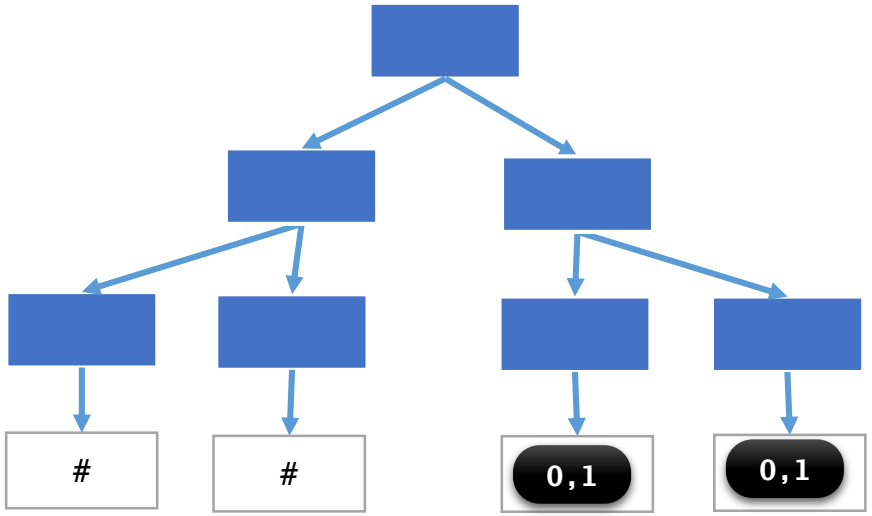
- Learning algorithm for Regular Languages
  - Word Languages [Oncina Garcia'92,Lang'92] (DFA)
  - Tree Languages [Oncina Garcia'93]
  - **State Merging** Algorithm

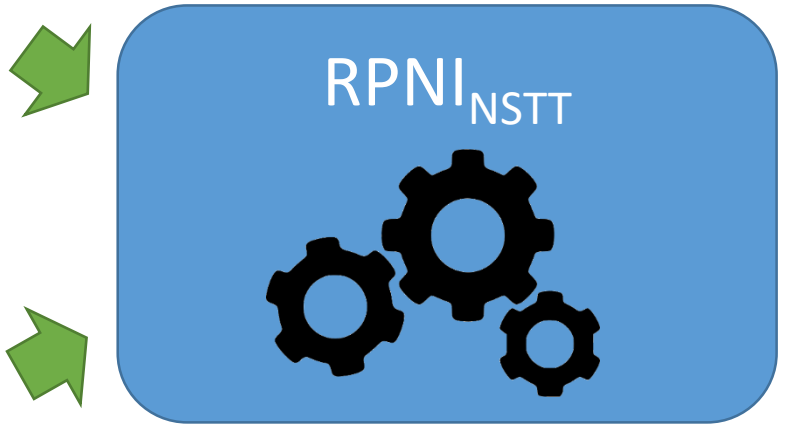
- Learns From **Examples** and **Counter-examples**
  - learnable in **polynomial time and data** [Gold'78]
  - Needs Polynomial Time
  - Requires a sample of Polynomial Size



Learning NSTT







A Node Selecting Tree Transducer

- No need of **explicit counter-example**
- unseen tuples are **implicit counter-example**

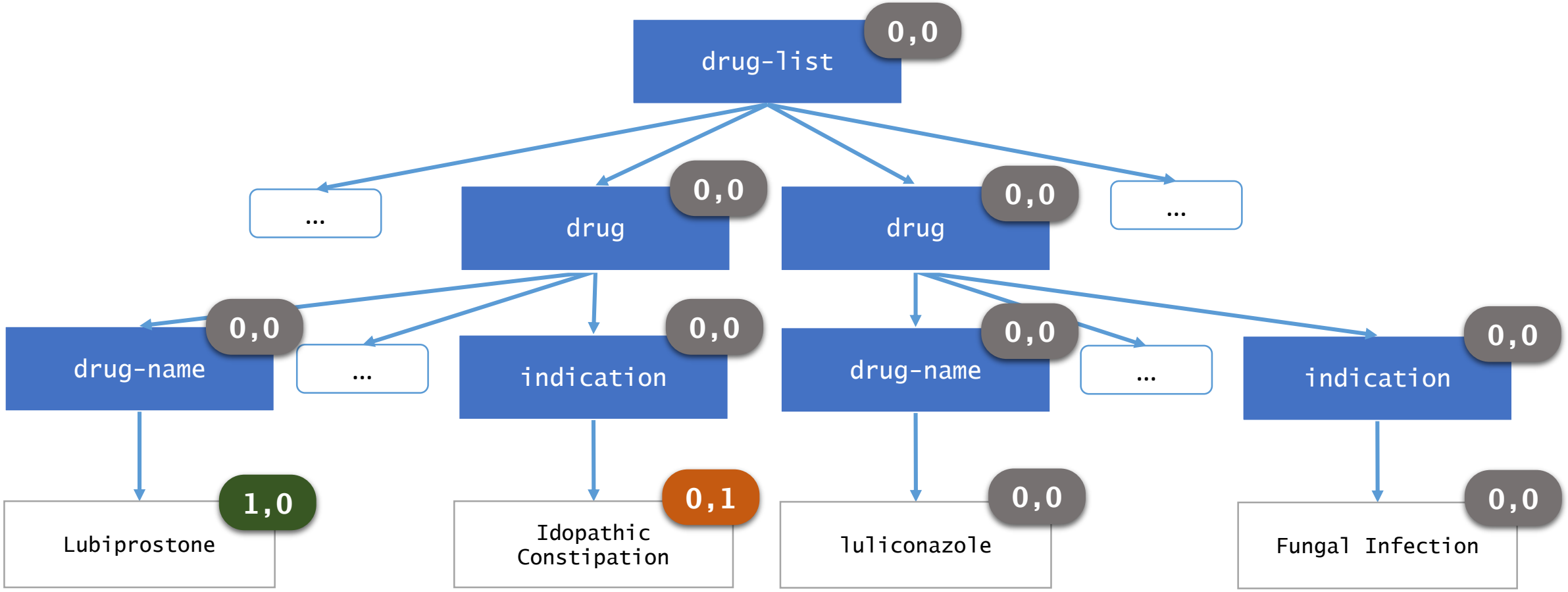
Learning Result

Theorem [Carme Lemay Niehren'04]

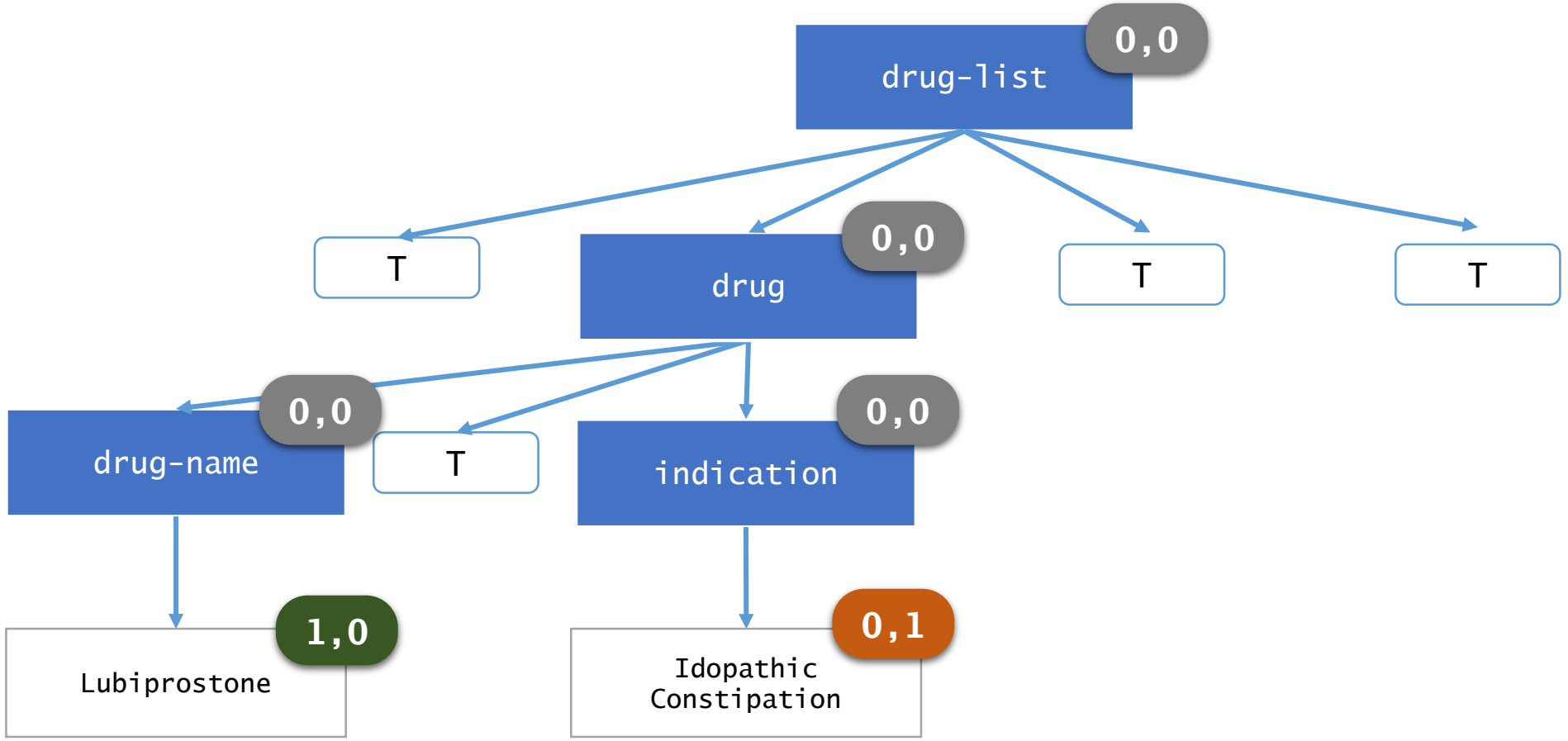
MSO Queries represented by **NSTT** are learnable from **annotated examples** in polynomial Time and Data

- Nice Theoretical Result
- But Fails in practice !
  - Require **complete** annotation
  - The query models the **whole** document

Pruning Trees [Carme Gilleron Lemay Niehren'07]



Pruning Trees [Carme Gilleron Lemay Niehren'07]

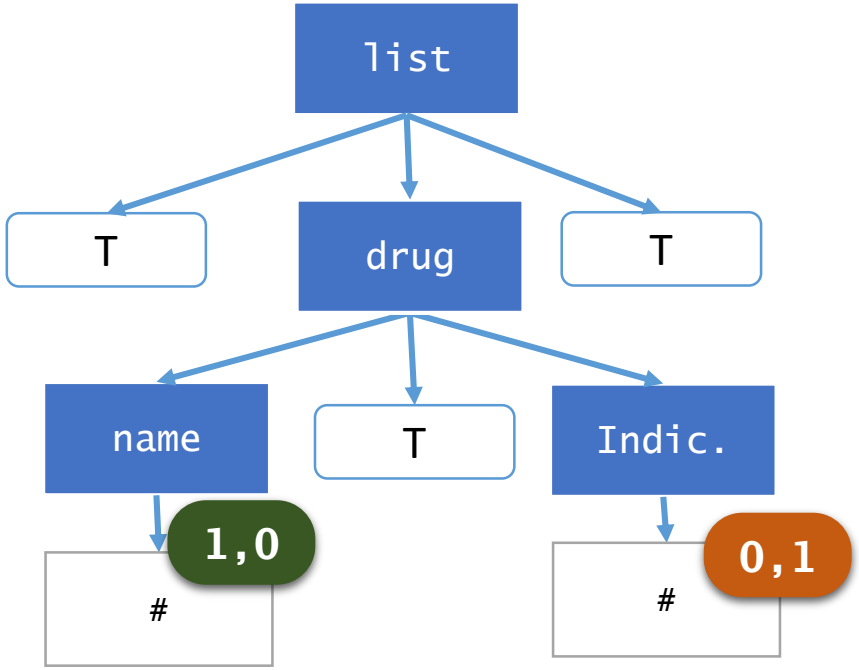


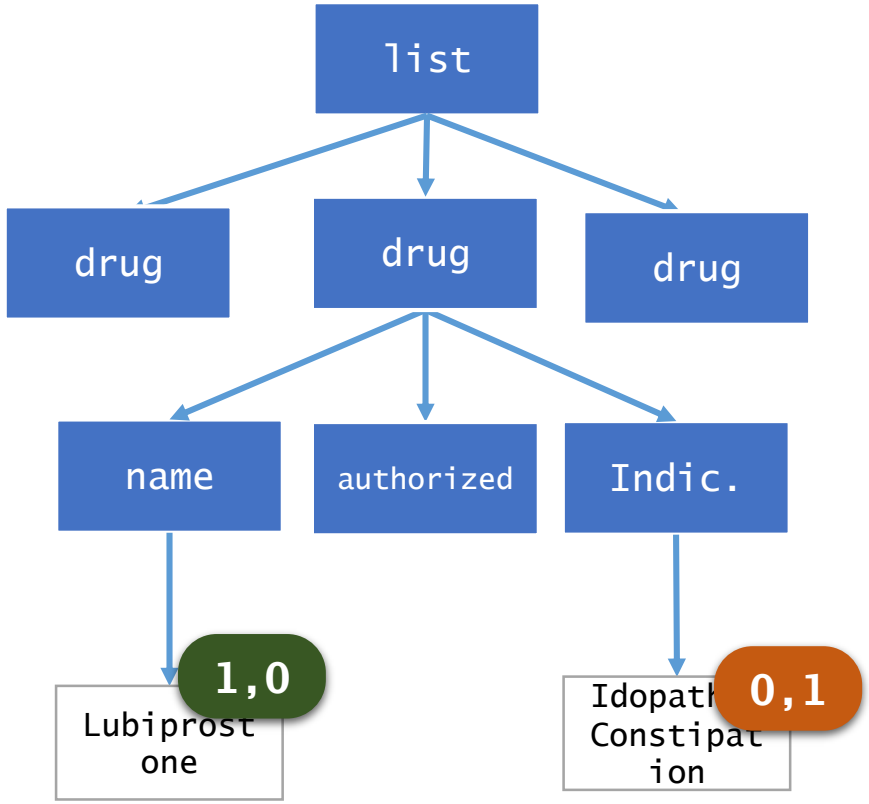
Pruning Functions

- Pruning Function
  - **PathOnly**: keep only nodes from the root to annotated nodes
  - **PathExtended**: also keep siblings of those nodes
  - …
- Replacement Node
  - **Simple** : replace node with a generic symbol (T)
  - **State of Automaton** : Replace with the state of a bottom-up automaton
    - Use Schema ! [Champavère Gilleron Lemay Niehren'08]
  - …

Query as Pruned Tree Language

- Extract all tuples that matches a pruning
- Pruning restricts expressiveness !
  - Extract authorized drugs only ?





Pruned Tree Queries

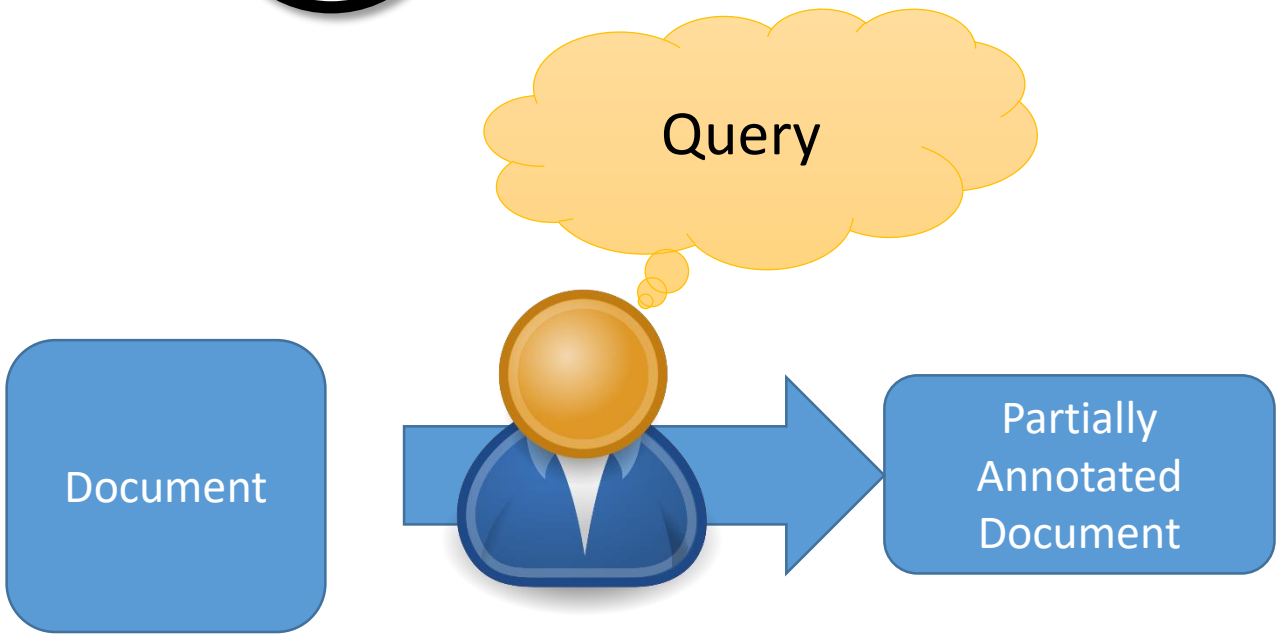
Theorem [Carme Gilleron Lemay Niehren'07]

For a **pruning function** *prune*, MSO Queries **stable with *prune*** represented by pruned NSTT are learnable from **partially annotated** examples in polynomial Time and Data

- Pruning function restricts the class
- But works well in practice ! [Carme Gilleron Lemay Niehren'07]

1

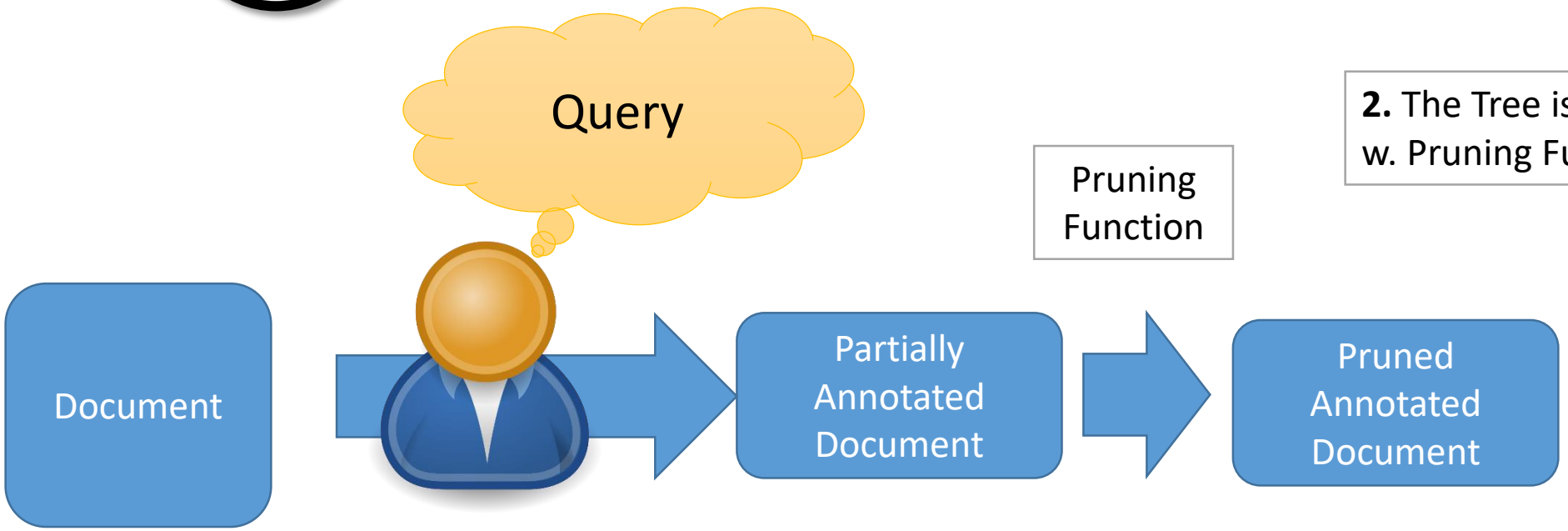
Interactive Setting



1. The user annotates **partially** a document

2

Interactive Setting

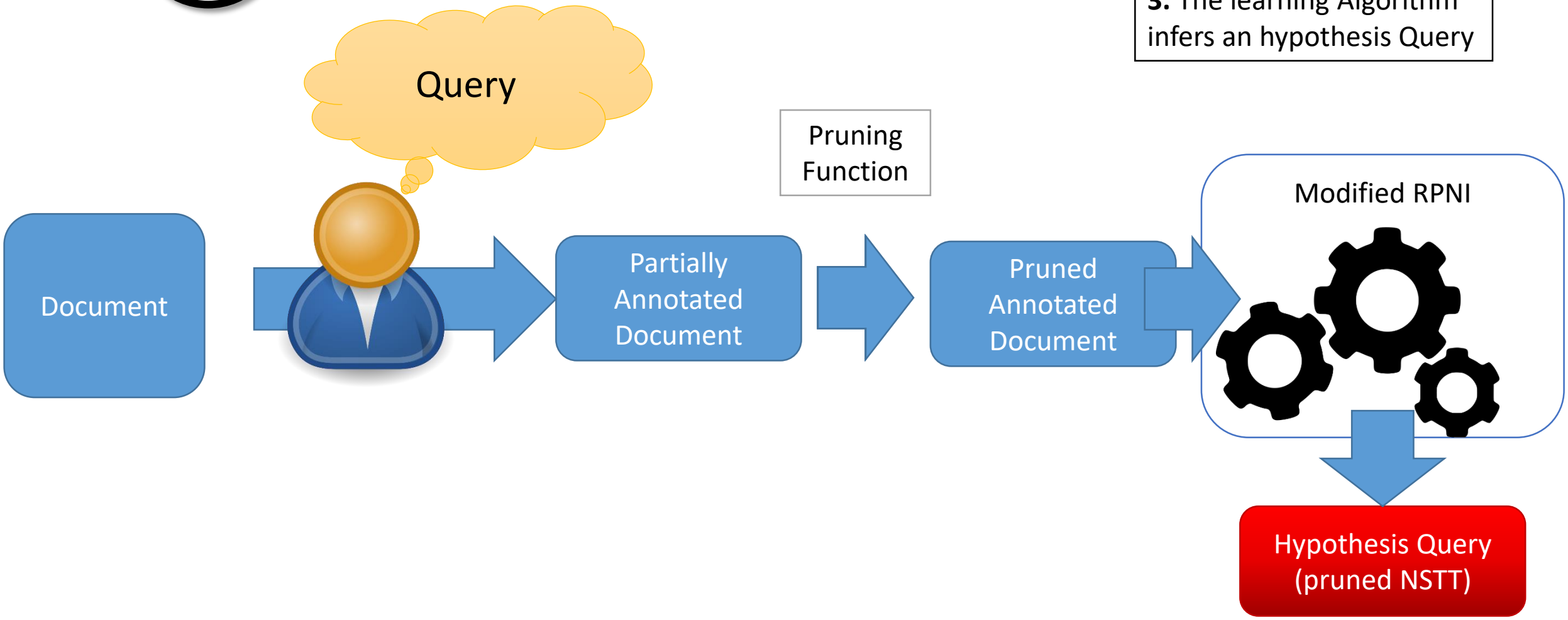


2. The Tree is pruned w. Pruning Function

3

Interactive Setting

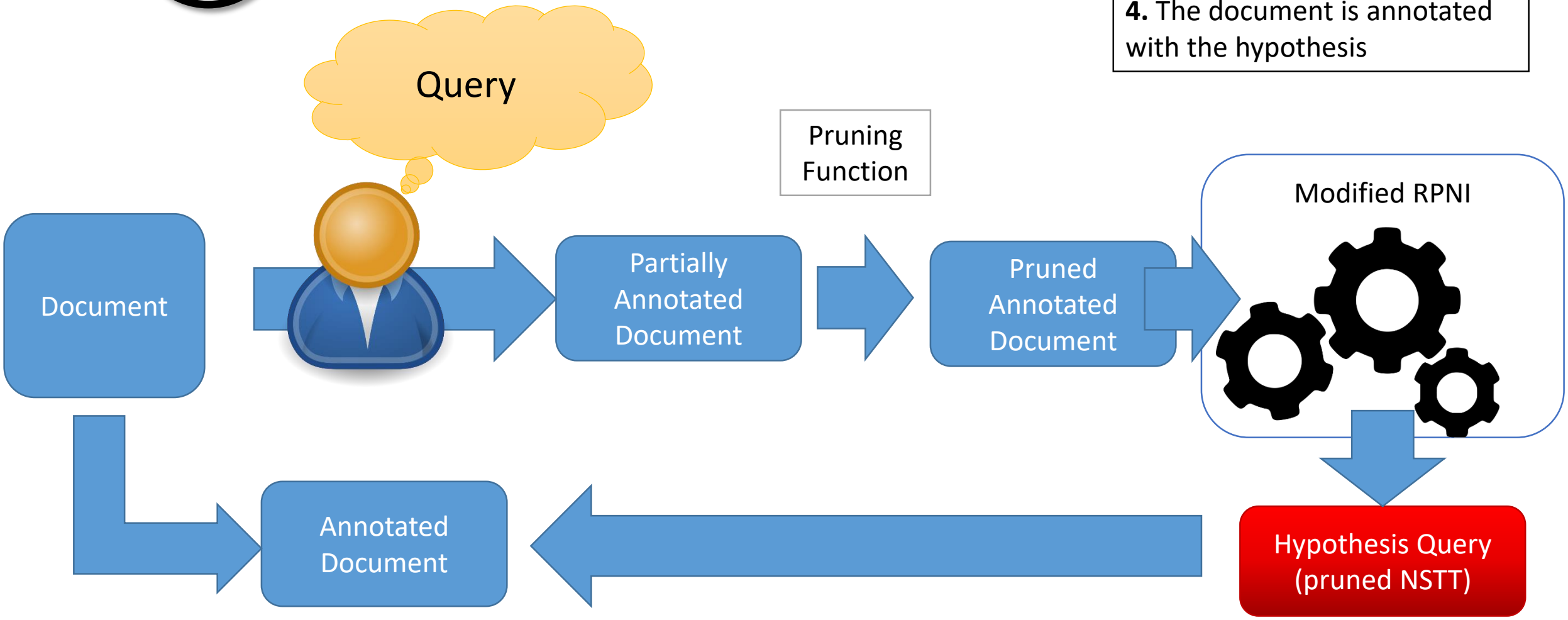
3. The learning Algorithm infers an hypothesis Query



4

Interactive Setting

4. The document is annotated with the hypothesis

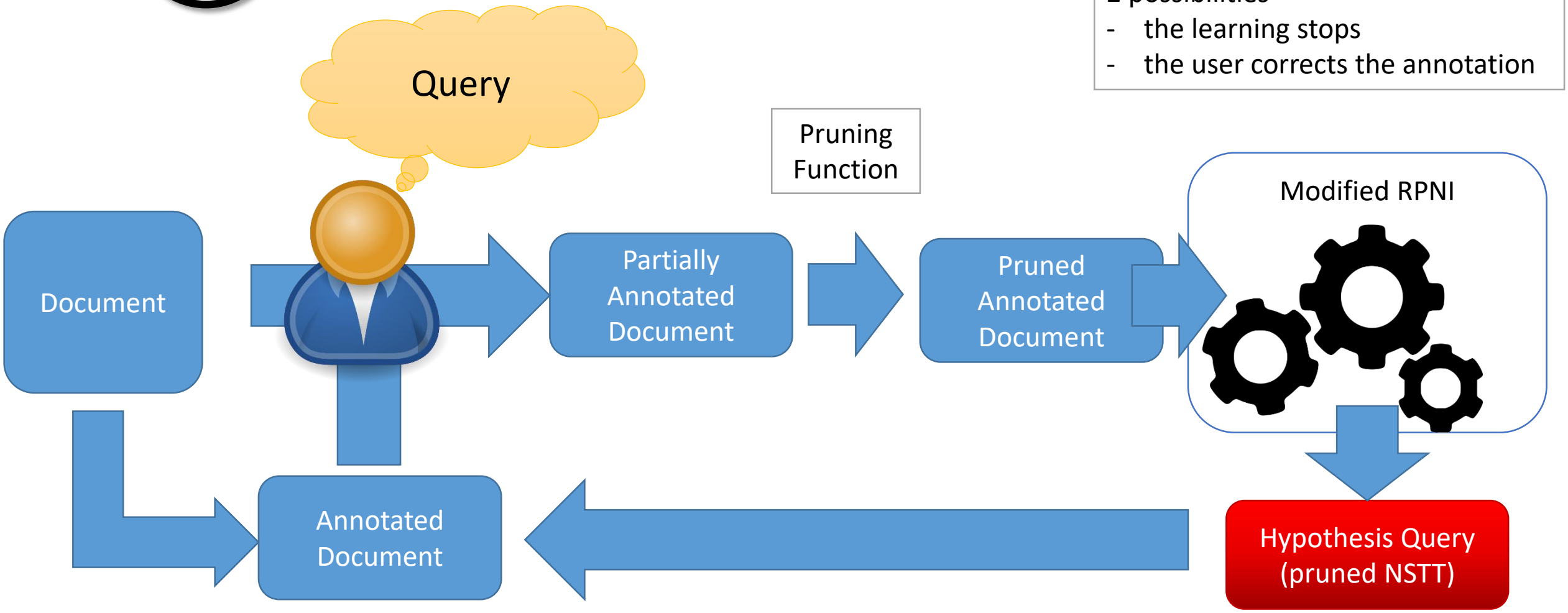


5

Interactive Setting

5. The annotated is presented to the user.

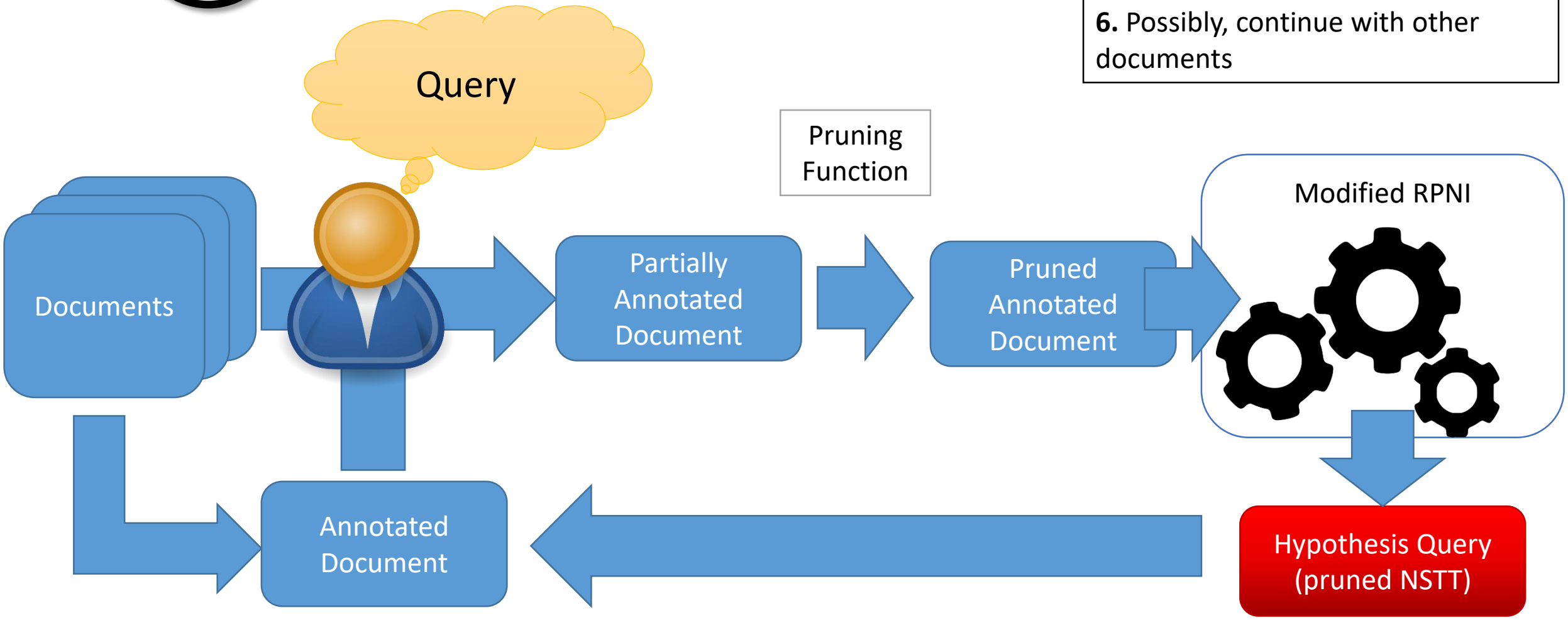
2 possibilities
- the learning stops
- the user corrects the annotation



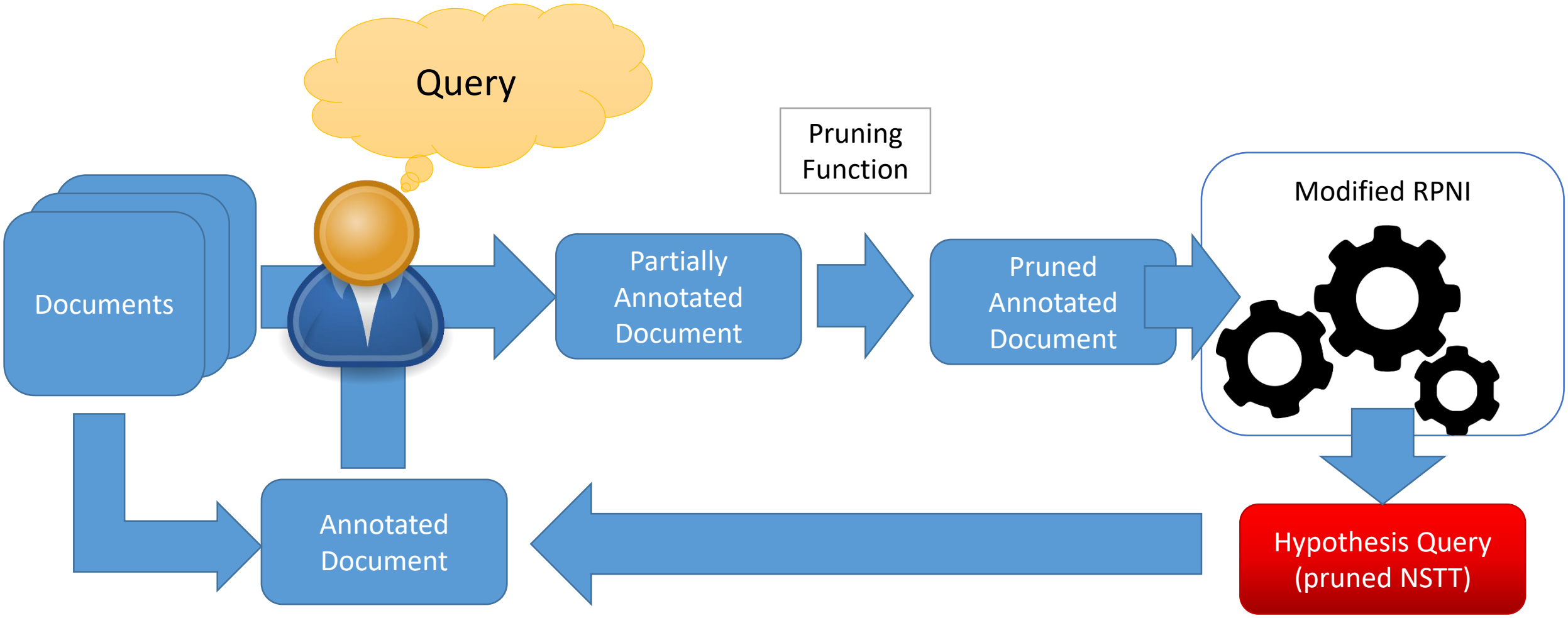
6

Interactive Setting

6. Possibly, continue with other documents



Interactive Setting



Learning Tree Queries - Results

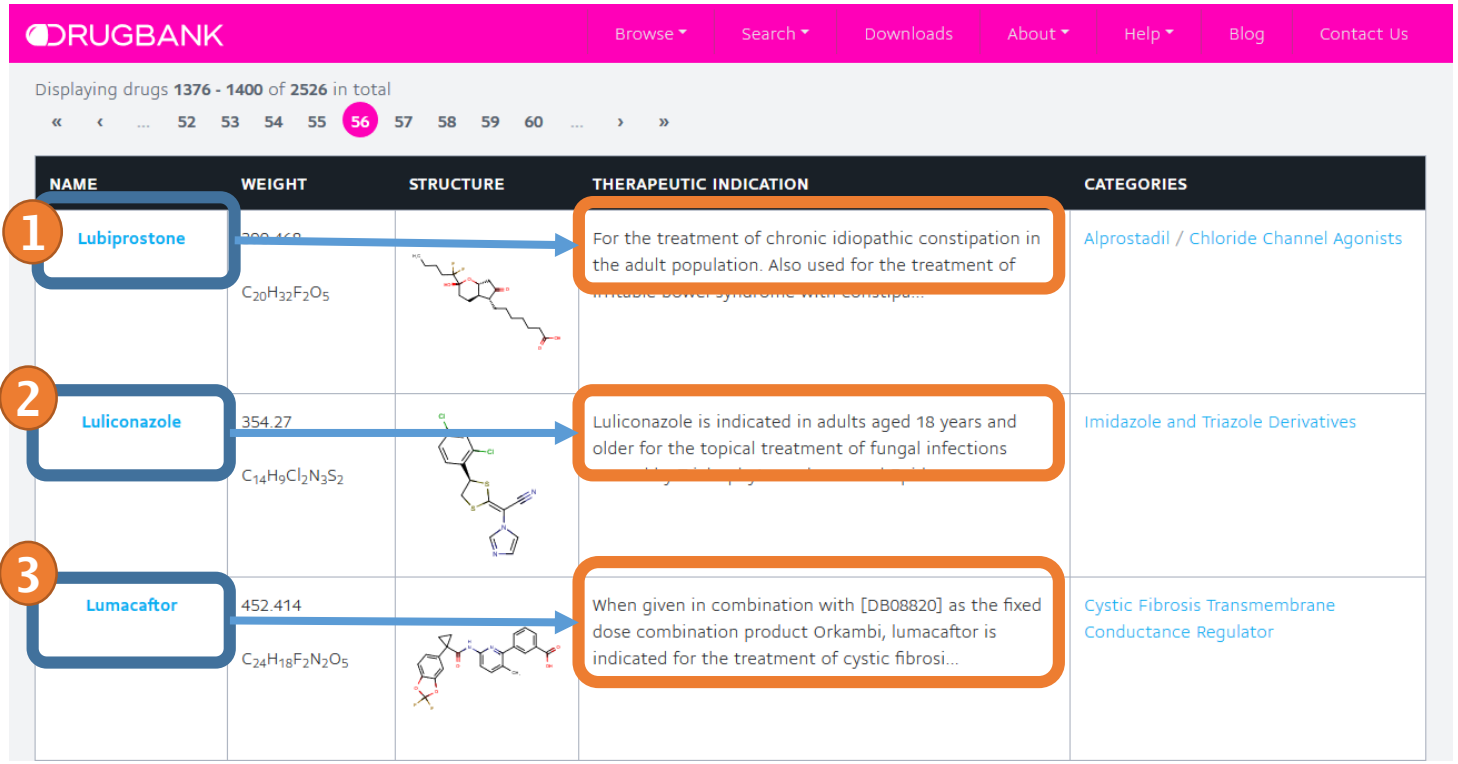
- NSTT : nice theoretical result, but not usable in practice
- Pruned NSTT : more adapted to practical cases

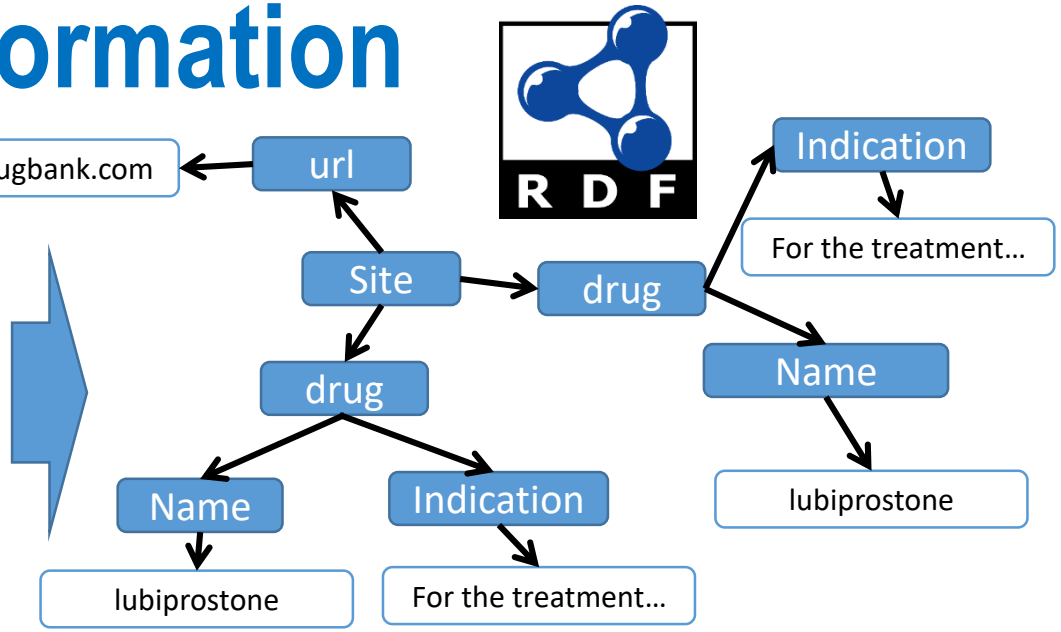
- Intuitive Interactive learning setting with ***partial*** annotation

  [Carme Gilleron Lemay Niehren'07]
- Good Experimental Results [Carme Gilleron Lemay Niehren'07, Champavère Gilleron Lemay Niehren'08]

Part 2
Learning Tree Transformations

Document Transformation






```
…

<#lubiprostone>
   drug:name "Lubiprostone" ;
   drug:indication "For the
treatment …".

<#lumacaftor>
   drug:name "Lumacaftor" ;
   drug:indication "When given in
combination …".
```

MSO Tree Transformations

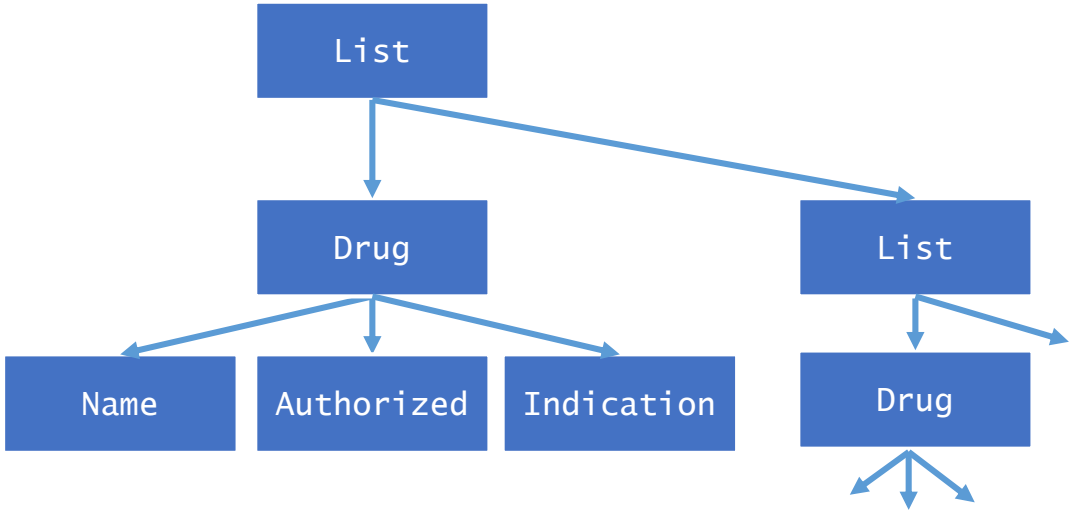
- Monadic Second Order Tree Transformations [Courcelle'94]
  - Tree Restriction of MSO transformations on graphs Tree [Courcelle'94]

  - Captures **First Order Logic**
  - Captures **Recursion**
  - Closed under **composition**
  - Evaluation in **linear** time on trees

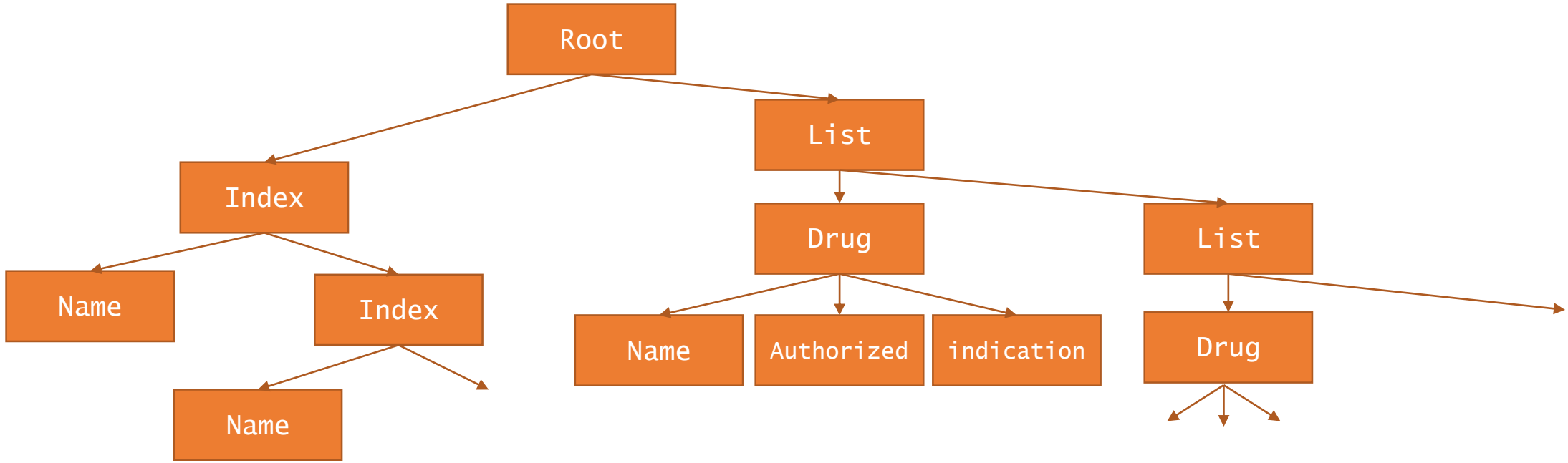
- MSO Tree Transformations ≡ MTT^{R}_{fc} [Engelfriet Maneth'03]
  - **MTT** : Macro Tree Transducers
  - **R** : Regular look-ahead
  - **fc** : finite copy

Deterministic Top Down Tree Transducers

[Thatcher Wright'68,Engelfriet'75]



Example :
Add an index and copy

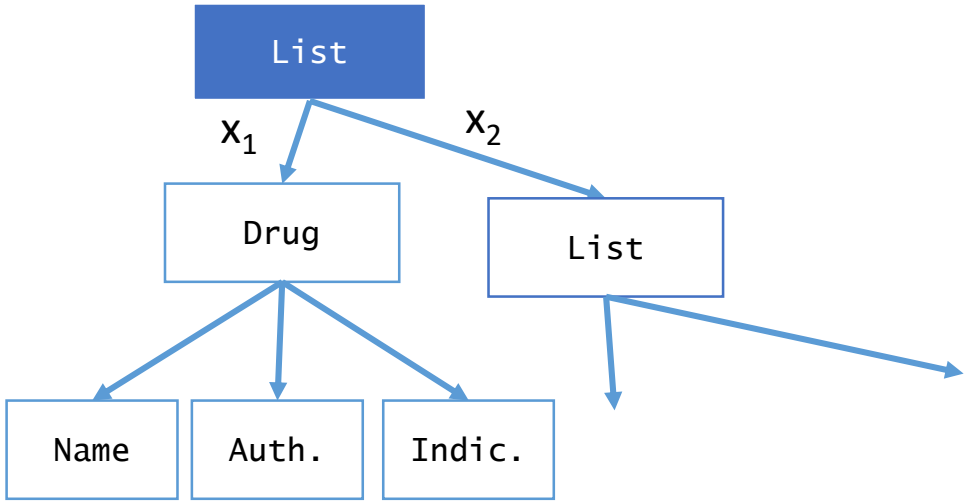


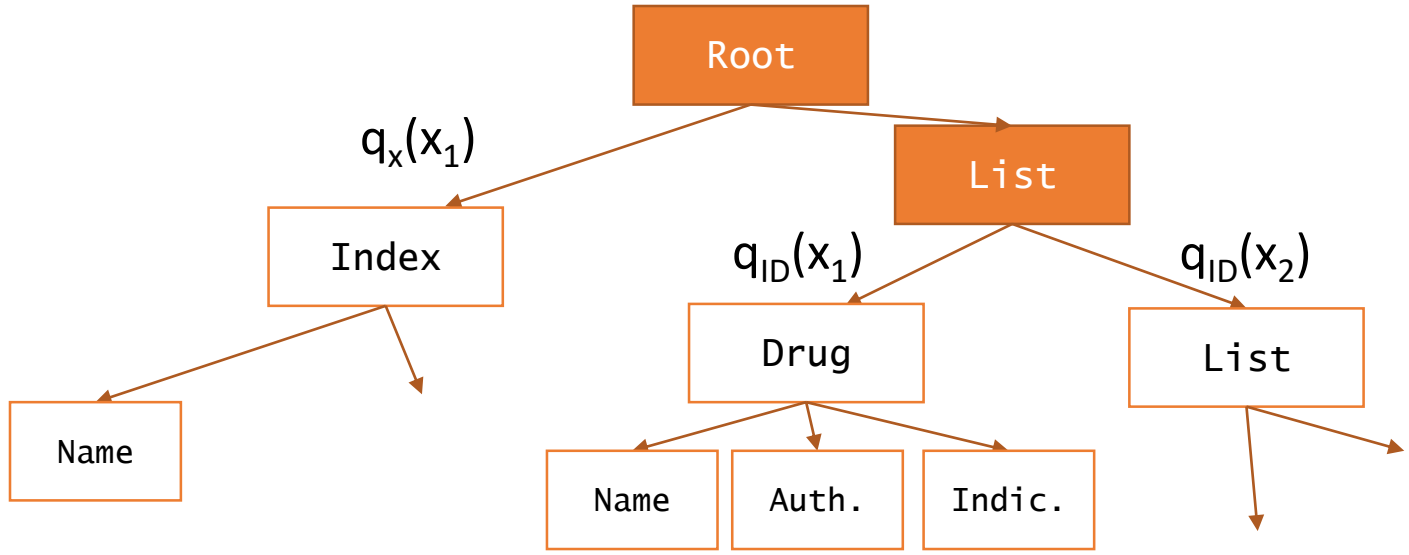
Deterministic Top Down Tree Transducer (DTOP)

Axiom : $q_0(x_0)$

$q_0(\text{List}(x_1,x_2)) \rightarrow \text{Root}(q_X(x_1), \text{List}(q_{ID}(x_1), q_{ID}(x_2))$

...





Earliest Normal Form [Engelfriet Maneth Seidl'09]

Axiom : $q_0(x_0)$

$q_0(\text{List}(x_1,x_2)) \rightarrow \text{Root}(q_X(x_1), \text{List}(q_{ID}(x_1), q_{ID}(x_2))$

…

Axiom : **Root**($q_x(x_0)$, $q_L(x_0)$)

$q_x(\text{List}(x_1,x_2)) \rightarrow \text{Index}(q_x(x_1), q_x(x_2))$

$q_L(\text{List}(x_1,x_2)) \rightarrow \text{List}(\text{Drug}(\ldots), q_L(x_2))$

- Earliest normal form : produces *as soon as* possible
- **Unique** minimal earliest normal form

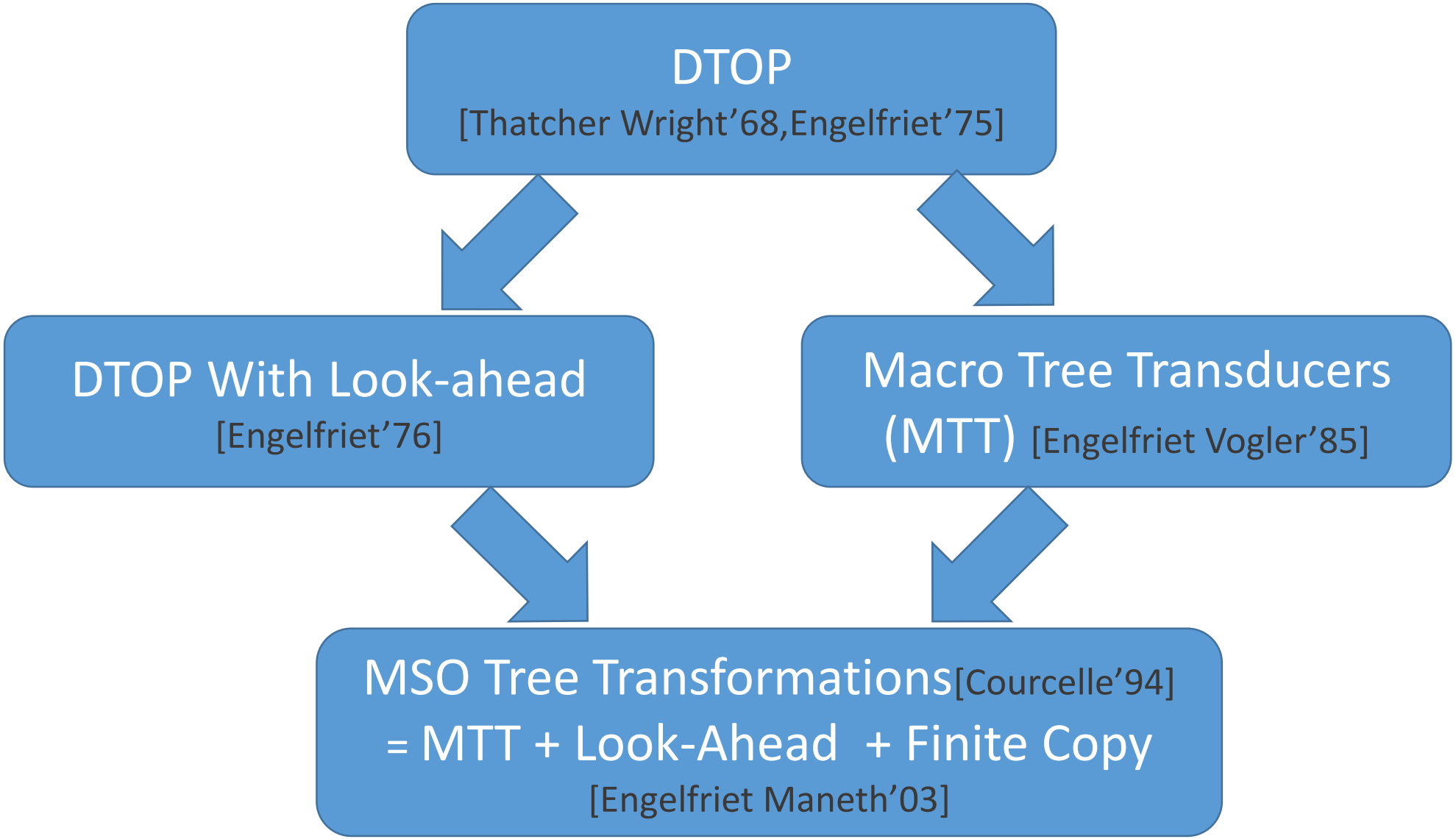
DTOP - Results

Theorem [Lemay Maneth Niehren'10]

Tree transformations represented by **DTOP** are **learnable** from pairs of input and output trees in **polynomial time and data**

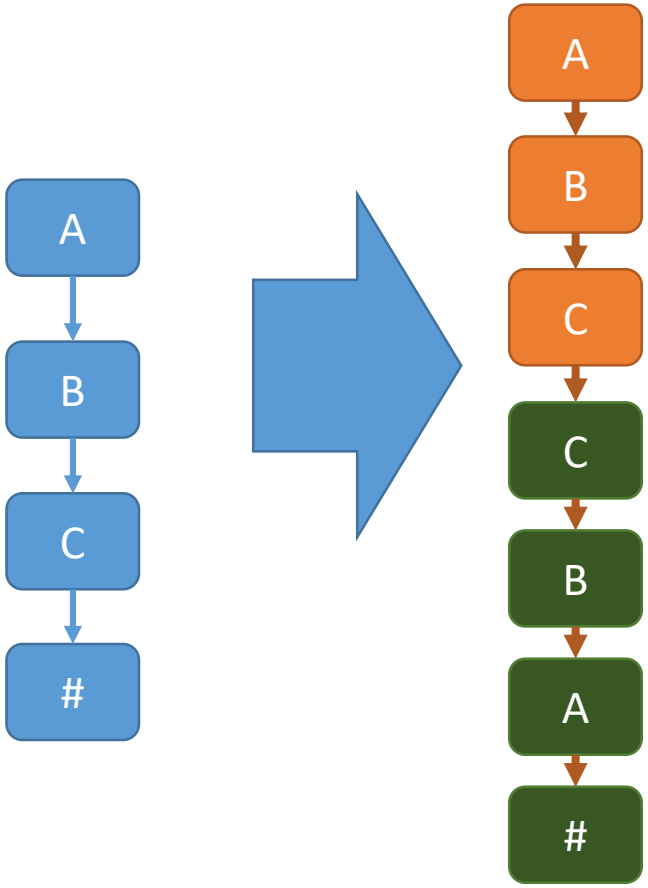
- **Myhill-Nerode** theorem for DTOP
- based on the notion of « **canonical origin** »
- Important « base » result
- But limited expressiveness

From DTOP to MSO Tree Transformations ?



Macro Tree Transducers [Engelfriet Vogler'85]

- Macro tree Transducers :
  - DTOP + states can carry 'macro' that they produces later



Axiom : $q(x_0) < \# >$

$q(A(x_1))\langle y_1 \rangle \rightarrow \mathbf{A}(q(x_1) < A\ (y_1) >)$

$q(B(x_1))\langle y_1 \rangle \rightarrow \mathbf{B}(q(x_1) < B\ (y_1) >)$

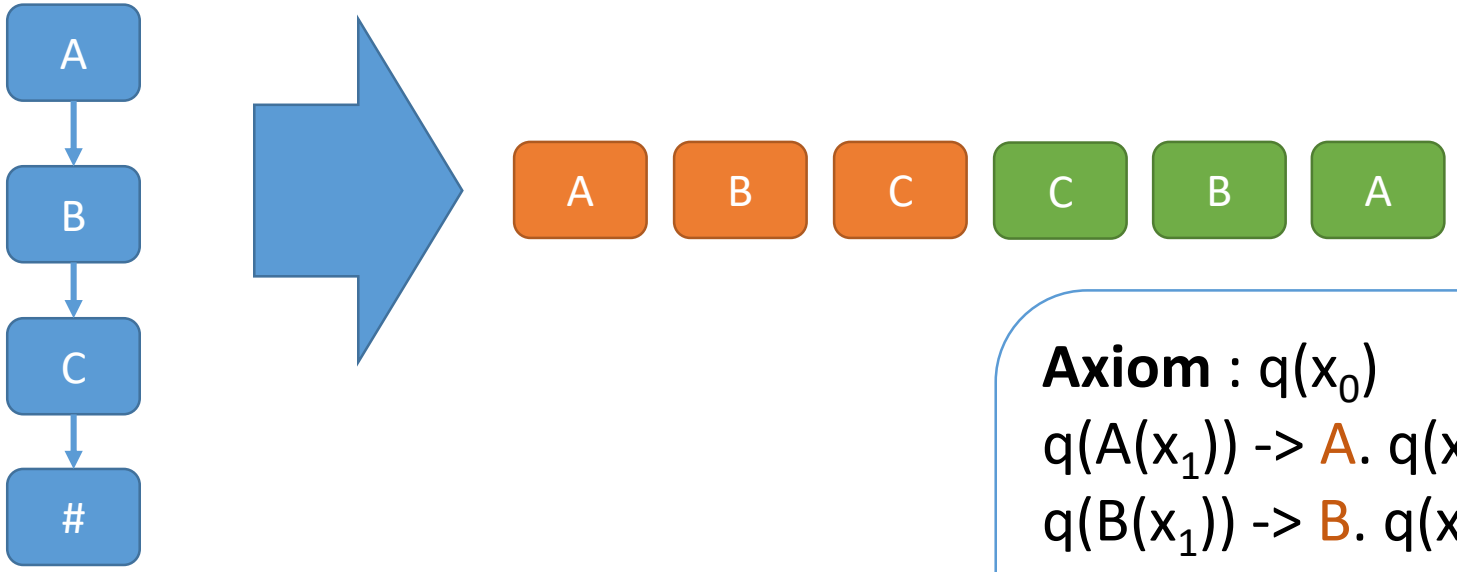
$q(C(x_1))\langle y_1 \rangle \rightarrow \mathbf{C}(q(x_1) < C\ (y_1) >)$

$q(\ \#\ )\langle y_1 \rangle \rightarrow y_1$

Tree to String Transducers

[Laurence Lemay Niehren Staworko Tommasi'10]

- Transducers that produces a string. Restriction of MTT



Axiom : $q(x_0)$

$q(A(x_1))$ -> A. $q(x_1)$ .A

$q(B(x_1))$ -> B. $q(x_1)$ .B

$q(C(x_1))$ -> C. $q(x_1)$ .C

q(#) -> .

Tree to String Transducers - Results

- Sequential Tree to String Transducers
  - **Linear** and **Preserve order**
  - **Myhill-Nerode** Theorem
  - Normal Form
  - Learning algorithm

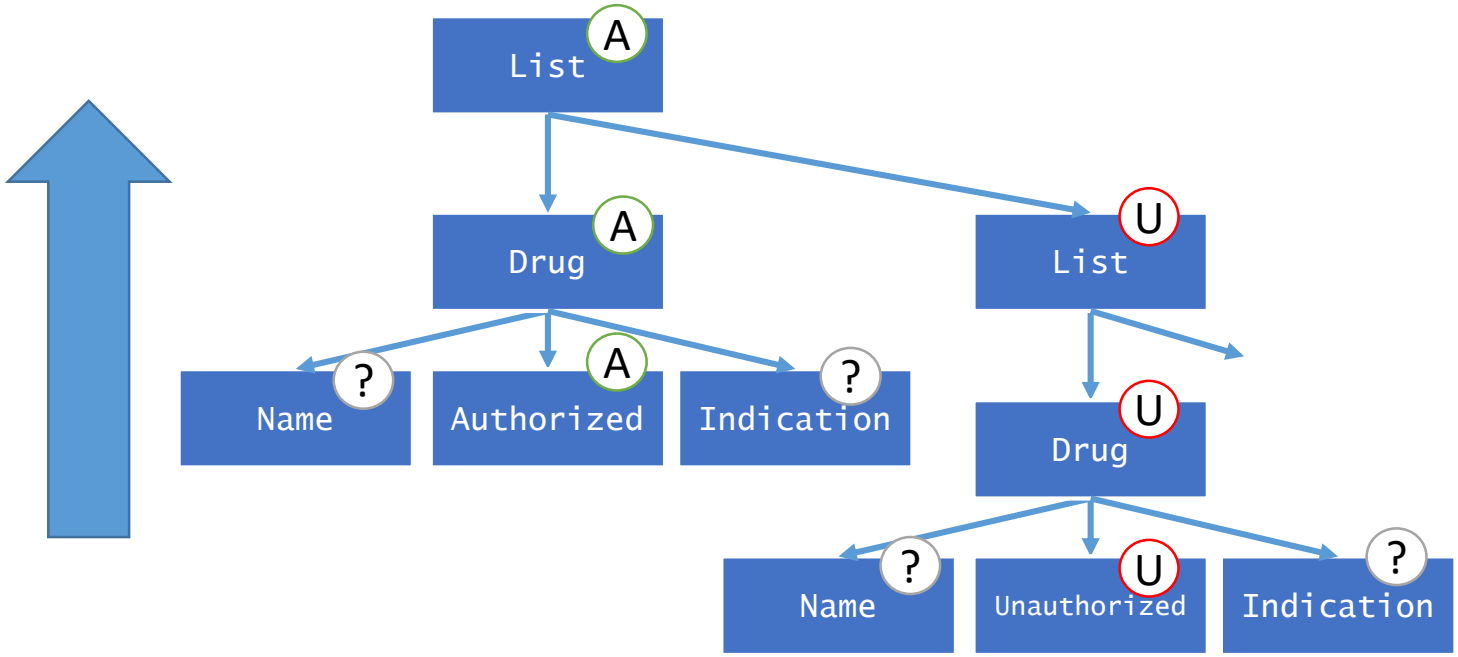
Theorem [Laurence Lemay Niehren Staworko Tommasi'10]

Tree-to-string transformations represented by **Deterministic Sequential Tree-to-string transducers** are learnable in polynomial time and data ***with abstain***

- Results extended by [Boiret Palenta'16] to linear tree to string transducers
- Seems hard to extend further…

Top-Down Tree Transducers with Look-Ahead
[Engelfriet'76]

- ***Example*** : Extract list of *authorized* drugs only
- **Look-ahead** : Bottom up tree automaton that checks whether the drug is authorized or not



Word Rational Functions

- Sequential Transducers [Schützenberger'77,Choffrut'03] with regular look-ahead
- Normal found defined by [Reutenauer Schützenberger'91]
  - On Bimachines

Theorem [Boiret Lemay Niehren'12]

Rational Functions represented by **sequential transducers with look-ahead** are learnable in polynomial time and data

- Extension to the tree case : open

Modelisation of Linux Scripts (ANR Colis)

- Modelisation of Install / Uninstall Scripts
  - Install o Uninstall = ID ?
  - Install1 o Install2 = Install2 o Install1 ?

- DTOP :
  - Composition and equivalence test ok, but not expressive enough

- Definition of High Order Tree Transducers [Paul Gallot Thesis]
  - Captures MSO Tree Transformations
  - Promising target for learning ?

High-Order Tree Transducers

- DTOP
  - Rules : $q(f(x_1,x_2)) \rightarrow G(q_1(x_1), q_2(x_2))$
  - States : functions from trees to trees (order 0)
- Macro Tree Transducers
  - Rules : $q(A(x_1))\langle y_1\rangle \rightarrow \mathbf{A}(q(x_1) \langle A\ (y_1)\rangle)$
    - $q(A(x_1)) \rightarrow \lambda y_1\ \mathbf{A}(q(x_1)\ (A\ (y_1))$
  - States : functions from trees to context (order 1)
- High Order Tree Transformation
  - Rules : $q(A(x_1)) \rightarrow \lambda y_1\ \mathbf{A}(q(x_1)\ (y_1(B)))\ \rangle$
  - States : functions from trees to unrestricted single-type λ-terms

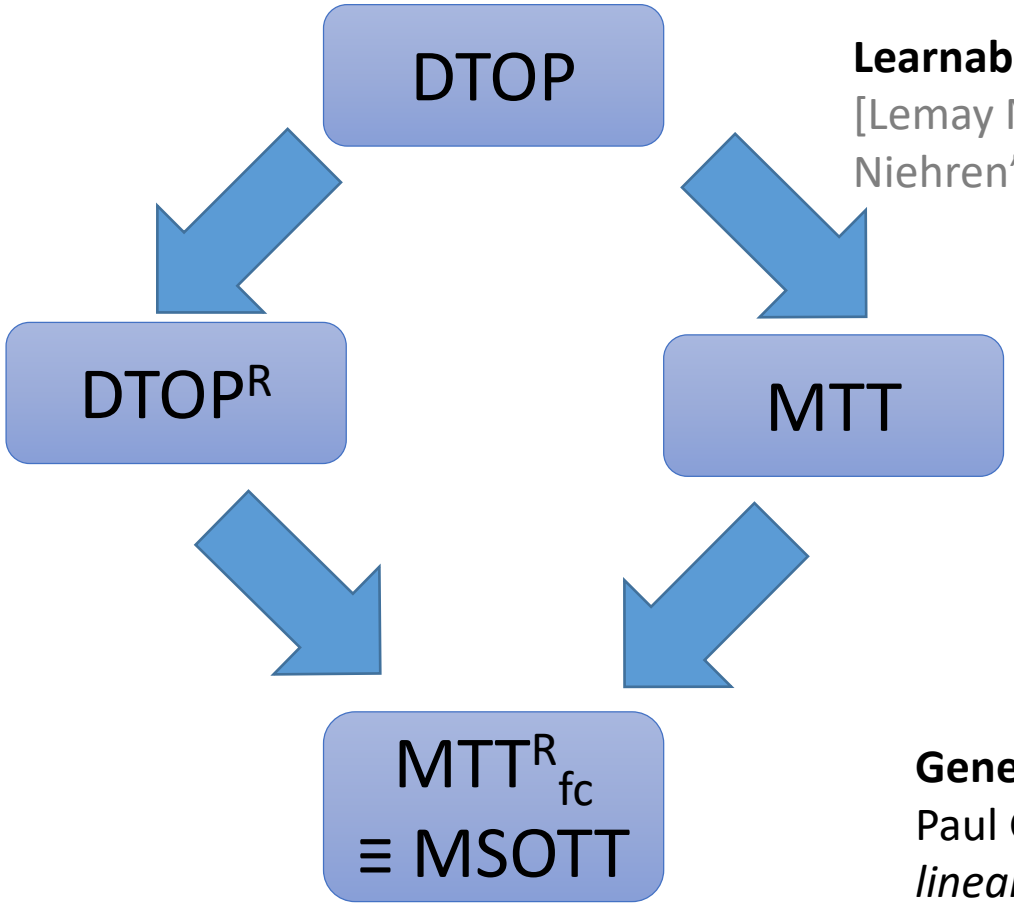
Properties of High-Order Tree Transducers [Work in progress]

- HOTT generalizes DTOP and MTT
- HOTT are closed by composition (but order grows)

- Linear HOTT ≡ MTT^{R}_{fc} ≡ MSOTT
  - Linear condition corresponds to finite copying
- Linear HOTT : closed by composition
  - Linear HOTT : order bounded !

- Normal form ? With origin information ?
- Promising target for ML ?

Towards Learning MSO Tree Transformations ?



Learnable
[Lemay Maneth Niehren'10]

Word case :
[Boiret Lemay Niehren'16,Boiret'16]
Tree case : Open

Sequential Tree to String : [Laurence Lemay Niehren Staworko Tommasi'14,Laurence'14]
General MTT : Open

General problem : Open
Paul Gallot thesis on
linear high order tree transducers ?

Part 3 - Future Works

Axes of Research

Learning Tree Queries	Learning Tree Transformations	Learning on Graph
- **NSTT** [CLN04,LNG06] - **pruned NSTT** [CGLN08,NCGL12,NCGL13] - **Interactive Learning** Setting for regular queries [CGLN05]	- Learning **DTOP** [LMN10,BLN16] - Learning **Tree to String** Transducers [SLLN09,LLNST11,LLNST14] - Learning Transducers with **Look-ahead** [BLN12]	- Learning **Regular Path Queries** [BCL15]
- Use **Data Values**	- **Look-ahead** on Trees - **Macro Tree Transducers ?** - **High Order Tree Transducers**	- Use **Data Values** - Regular **Tree Pattern** Queries - Learning Graph **Transformations**


thank you
danke
謝謝
ngiyabonga
teşekkür ederim
gracias
спасибо
obrigado
bedankt
dziękuję
merci
go raibh maith agat
mochchakkeram
tapadh leat
dakujem
arigatô
grazie
sukriya
kop khun krap
terima kasih
감사합니다
ευχαριστώ
рахмат
Баярлалаа
faafetai lava
vinaka
спасибі
blagodaram
mersi
barka
welalin
tack
spas
dank je
misaotra
matondo
paldies
grazzi
mahalo
kiitos
dankie
dhanyavad
nanni
nandri
bayarlaa
enkosi
gràçie
hvala
maruuru
köszönöm
akun
dankon
ačiū
djiere dieuf
tau
дякую
mamnun
chokrane
murakoze
tenki
хвала
asante
manana
obrigada
chnorakaloutioun
gratias ago
gràcies
sulpáy
taiku
takk
trugarez
sobodi
děkuji
mèsi
sagolun
didi madloba
najis tuke
ありがとう
tanemirt
rahmet
rahmat
kam sah hamnida
তোমাকে ধন্যবাদ
diolch
dhanyavadagalu
shukriya
merçé
мерси
xièxie