Multi-Grain Relations

François Barthélemy

Conservatoire National des Arts-et-Métiers (Cedric) Paris
INRIA (Atoll) Rocquencourt
France

CIAA 2007

Motivation

Goal

Expressing relations between different representations with two (or more) levels of analysis.

Example

Natural Language Processing, morphological analysis.

[pos=V]	[pos=N][from=V]
move	ment

The two levels: symbols (letters, features), morphemes.

Important operations

The formalism and its compilation

- Linguistic description using *contextual rules*.
- For example *Generalized restriction* (Yli-Jyrä and Koskenniemi 2005).

Important operations

The formalism and its compilation

- Linguistic description using *contextual rules*.
- For example *Generalized restriction* (Yli-Jyrä and Koskenniemi 2005).
- Rule example:
  $\Sigma^* \diamond ([\text{num=pl}] : s) \diamond \Rightarrow \Sigma^* ([\text{pos=N}]\mathit{Feature}^* : \mathit{Letter}^*) \diamond \Sigma^* \diamond$

Important operations

The formalism and its compilation

- Linguistic description using *contextual rules*.
- For example *Generalized restriction* (Yli-Jyrä and Koskenniemi 2005).
- Rule example:
  $\Sigma^* \diamond ([\text{num=pl}] : s)\diamond \Rightarrow \Sigma^*([\text{pos=N}]\mathit{Feature}^* : \mathit{Letter}^*) \diamond \Sigma^* \diamond$
- Reading: a plural suffix `s` may be added to nouns only.

Important operations

The formalism and its compilation

- Linguistic description using *contextual rules*.
- For example *Generalized restriction* (Yli-Jyrä and Koskenniemi 2005).
- Rule example:
  $\Sigma^* \diamond ([\text{num=pl}] : s) \diamond \Rightarrow \Sigma^* ([\text{pos=N}] \mathit{Feature}^* : \mathit{Letter}^*) \diamond \Sigma^* \diamond$
- Reading: a plural suffix s may be added to nouns only.
- Rules are simultaneous constraints.

Important operations

The formalism and its compilation

- Linguistic description using *contextual rules*.
- For example *Generalized restriction* (Yli-Jyrä and Koskenniemi 2005).
- Rule example:
  $\Sigma^* \diamond ([\text{num=pl}] : s) \diamond \Rightarrow \Sigma^* ([\text{pos=N}]\mathit{Feature}^* : \mathit{Letter}^*) \diamond \Sigma^* \diamond$
- Reading: a plural suffix s may be added to nouns only.
- Rules are simultaneous constraints.
- Compilation formula: $\cap_i(\Sigma^* - d_\diamond(W_i - W_i'))$
  where *i* is an index over rules

Important operations

The formalism and its compilation

- Linguistic description using *contextual rules*.
- For example *Generalized restriction* (Yli-Jyrä and Koskenniemi 2005).
- Rule example:
  $\Sigma^* \diamond ([\text{num=pl}] : s)\diamond \Rightarrow \Sigma^*([\text{pos=N}]\mathit{Feature}^* : \mathit{Letter}^*) \diamond \Sigma^* \diamond$
- Reading: a plural suffix s may be added to nouns only.
- Rules are simultaneous constraints.
- Compilation formula: $\cap_i(\Sigma^* - d_\diamond(W_i - W_i'))$
  where *i* is an index over rules

Important operations

rational operations + Cartesian product + intersection + difference

More about the two grains

The two grains

Symbols and morphemes.

Operations on symbols

- e.g.: recognizing a word from a text.
- for instance `movement`

More about the two grains

The two grains

Symbols and morphemes.

Operations on symbols

- e.g.: recognizing a word from a text.
- for instance `movement`

Operations on morphemes

- e.g.: description of word structure (morphotactics).
- ([pos=V] : $Letter^*$)([tense=preterit] : ed)

More about the two grains

The two grains

Symbols and morphemes.

Operations on symbols

- e.g.: recognizing a word from a text.
- for instance `movement`

Operations on morphemes

- e.g.: description of word structure (morphotactics).
- ([pos=V] : $Letter^*$)([tense=preterit] : ed)

Synchronization between the two representations: using the morpheme notion.

Two-grain Sets

Definition

- $L \in \mathrm{Rat}_1(\Sigma^*)$
- $\mu : \Sigma \longrightarrow \mathrm{Rat}_2(A^* \times B^*)$
- $TGS(L, \mu) = \{(v_1, w_1) \ldots (v_n, w_n) \in (A^* \times B^*)^* |$
  $\exists a_1 \ldots a_n \in L, (v_1, w_1) \in \mu(a_1), \ldots, (v_n, w_n) \in \mu(a_n)\}.$

Two-grain Sets

Definition

- $L \in \mathrm{Rat}_1(\Sigma^*)$
- $\mu : \Sigma \rightarrow \mathrm{Rat}_2(A^* \times B^*)$
- $TGS(L, \mu) = \{(v_1, w_1) \ldots (v_n, w_n) \in (A^* \times B^*)^* |$
  $\exists a_1 \ldots a_n \in L, (v_1, w_1) \in \mu(a_1), \ldots, (v_n, w_n) \in \mu(a_n)\}.$

Example

- $L = a(a|b)b$
- $\mu(a) = (x^*, x) \;\; \mu(b) = (y, z^+)$
- $$\begin{array}{cccc} a & a & b & \in L \\ (xxx, x) & (xx, x) & (y, z) & \in TGS(L, \mu) \end{array}$$

Two-grain Sets

Definition

- $L \in \mathrm{Rat}_1(\Sigma^*)$
- $\mu : \Sigma \to \mathrm{Rat}_2(A^* \times B^*)$
- $TGS(L, \mu) = \{(v_1, w_1) \ldots (v_n, w_n) \in (A^* \times B^*)^* |$
  $\exists a_1 \ldots a_n \in L, (v_1, w_1) \in \mu(a_1), \ldots, (v_n, w_n) \in \mu(a_n)\}.$

Example

- $L = a(a|b)b$
- $\mu(a) = (x^*, x)\ \ \mu(b) = (y, z^+)$
- $\begin{matrix} a & a & b & \in L \\ (xxx, x) & (xx, x) & (y, z) & \in TGS(L, \mu) \end{matrix}$

Terminology

- coarse grains: elements from Σ
- fine grains: elements from A and B

Remarks about two-grain Sets

Remark 1

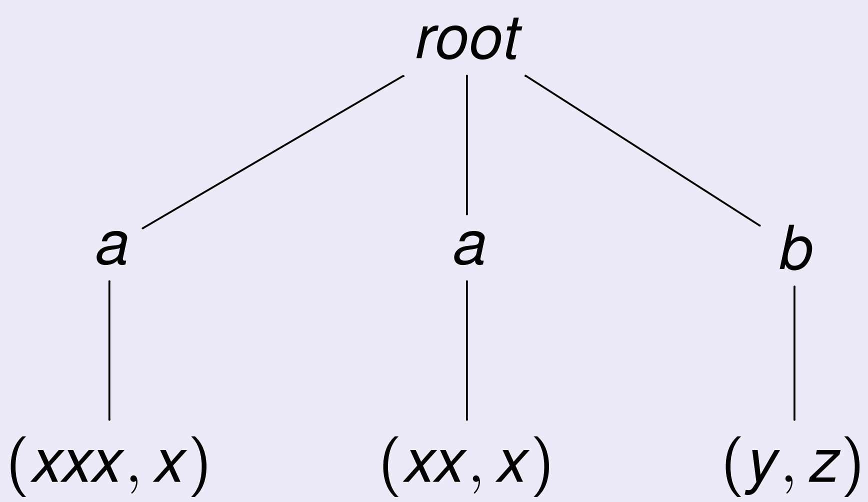
- The symbols of the rational language do not appear in members of two-level sets.
- They are non-terminals which structure forms.

Remarks about two-grain Sets

Remark 1

- The symbols of the rational language do not appear in members of two-level sets.
- They are non-terminals which structure forms.

Example



Remarks about two-grain Sets

Remark 1

- The symbols of the rational language do not appear in members of two-level sets.
- They are non-terminals which structure forms.

Remark 2

- Some disjunctions may be expressed at both levels.
- Example:
  $L_1 = (a|b) \;\; \mu_1(a) = (x, x) \;\; \mu_1(b) = (y, y)$
  $L_2 = c \;\; \mu_2(c) = (x, x)|(y, y)$
  $TGS((a|b), \mu_1) = TGS(c, \mu_2) = \{(x, x), (y, y)\}$
- It is not the case of product (and star):
  $(x, x)(y, y) \neq (xy, xy)$

Two-grain Relations

Idea

- flatening the two levels of the description in one rational description
- marking the structure using a special symbol ω denoting the end of pairs

Two-grain Relations

Idea

- flatening the two levels of the description in one rational description
- marking the structure using a special symbol ω denoting the end of pairs

Definition

A Two-Grain Relation is a rational subset of $(A^*\omega \times B^*\omega)^*$.

Two-grain Relations

Idea

- flatening the two levels of the description in one rational description
- marking the structure using a special symbol ω denoting the end of pairs

Definition

A Two-Grain Relation is a rational subset of $(A^*\omega \times B^*\omega)^*$.

Proposition

Two-grain sets and two-grain relations are equivalent.

Closure properties

Closed

- union
- product
- star

Not closed

- intersection
  basically because rational relations are not

Sub-classes closed under intersection

Sufficient condition

If the relations in the images of μ_1 and μ_2 belong to a subclass of rational relation closed under intersection, then $TGS(L_1, \mu_1) \cap TGS(L_2, \mu_2)$ is a two-grain set.

Three instances

- recognizable relations
  Cartesian product of independent languages
- length-preserving relations
  Rational language over symbol pairs
- synchronized relations
  product of a length preserving relation and a rational language

Generalization

Three ways

- more than two sizes of grains
- more than two components in the relation (n-tuples instead of pairs)
- possibly different number of grains in the different components

Example

Example

- three components: morpheme description, grapheme, phoneme
- three grains: morpheme, grapheme to phoneme matching unit, symbol
- the morpheme description has only two grains.

Example

Example

- three components: morpheme description, grapheme, phoneme
- three grains: morpheme, grapheme to phoneme matching unit, symbol
- the morpheme description has only two grains.

3-tuple

morph. desc.	[type=root][cat=nom]		[type=suffix][cat=adv]	
graphemes	s h	y	l	y
phonemes	$\int$	a ɪ	l	ɪ

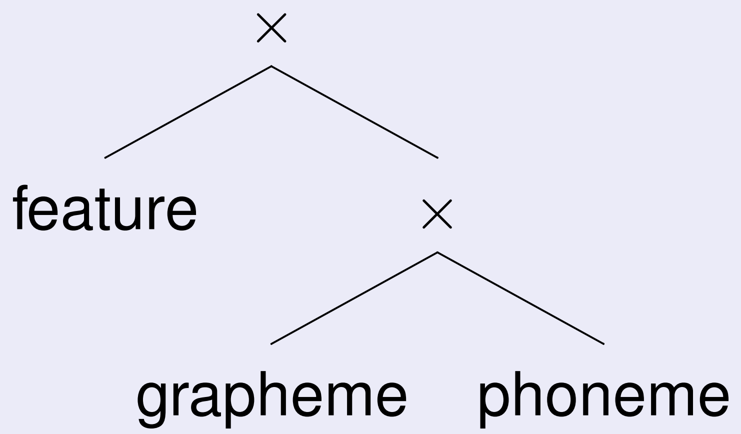
Tree structured description

Grains definition

A tree structure defines:

- the number of components (= number of leaves)
- the number of grains (1+ # of inner nodes)
- the grains used by each component (path from the root to the leave)
- the embedding of grains (daughter embedded in the parent)

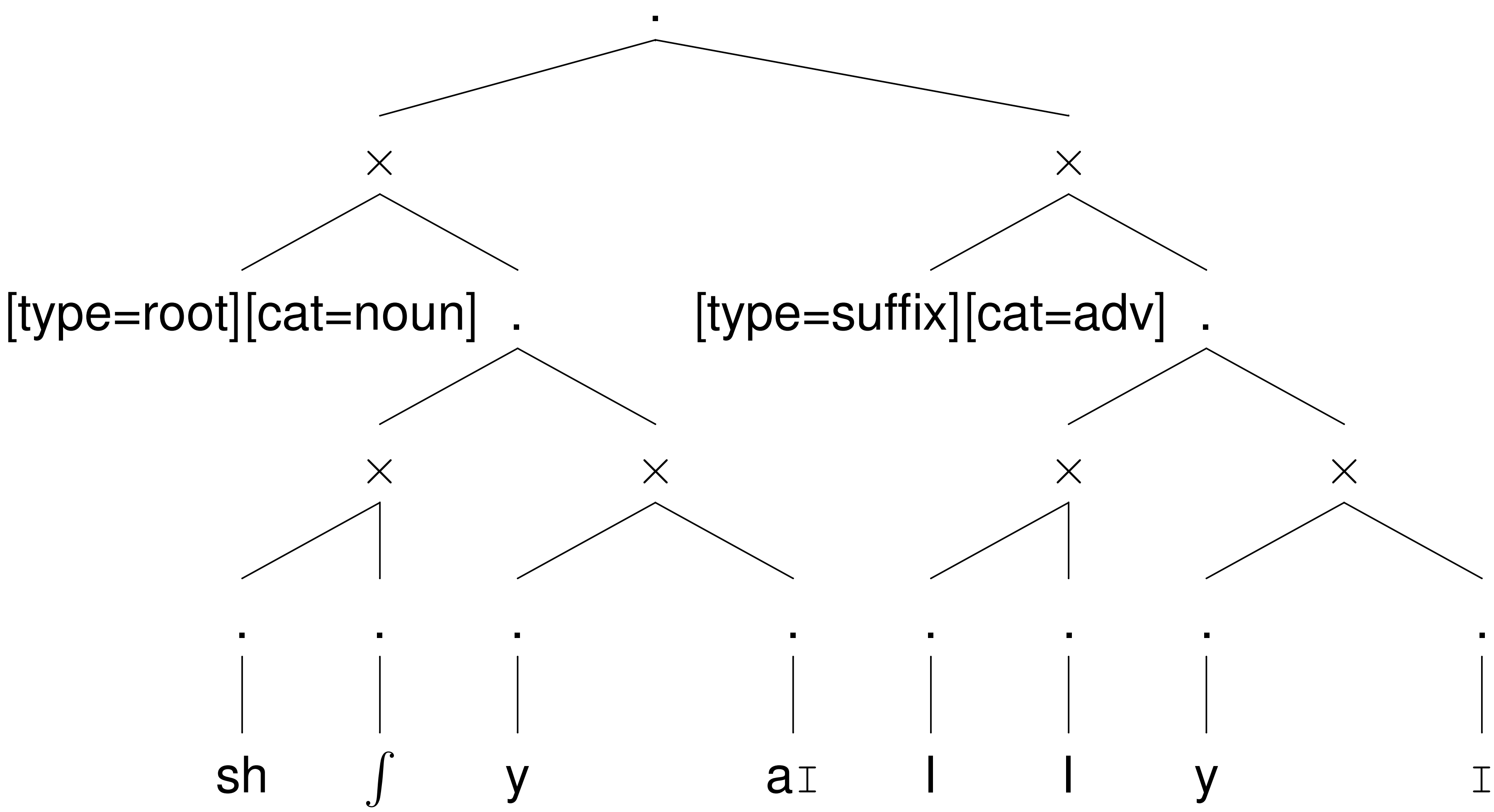
Example



Implicit

Sequences of symbols and grains remain implicit.

The product-Cartesian product structure



Tree linearization

- a special symbol used as terminator for each different grain type.
- terminator $\Leftrightarrow$ postfix notation of the tree structure.

Tree linearization

- a special symbol used as terminator for each different grain type.
- terminator $\Leftrightarrow$ postfix notation of the tree structure.

Example

[type=root][cat=noun]				ω_2	[type=suffix][cat=noun]				ω_2
s h	ω_1	y	ω_1	ω_2	l	ω_1	y	ω_1	ω_2
$\int$	ω_1	a ɪ	ω_1	ω_2	l	ω_1	ɪ	ω_1	ω_2

Tree linearization

- a special symbol used as terminator for each different grain type.
- terminator ⇔ postfix notation of the tree structure.

Example

[type=root][cat=noun]				ω_2	[type=suffix][cat=noun]				ω_2
s h	ω_1	y	ω_1	ω_2	l	ω_1	y	ω_1	ω_2
ʃ	ω_1	a ɪ	ω_1	ω_2	l	ω_1	ɪ	ω_1	ω_2

Rational descriptions

Goal

- description of rational sets of tree linearization
- compilable in a finite-state machine

Rational descriptions

Goal

- description of rational sets of tree linearization
- compilable in a finite-state machine

Ideas

- rational expressions including the terminators

Rational descriptions

Goal

- description of rational sets of tree linearization
- compilable in a finite-state machine

Ideas

- rational expressions including the terminators
- extended with the Cartesian product

Rational descriptions

Goal

- description of rational sets of tree linearization
- compilable in a finite-state machine

Ideas

- rational expressions including the terminators
- extended with the Cartesian product
- using grain (terminator) as a *type* used to define well-formed expressions:

Rational descriptions

Goal

- description of rational sets of tree linearization
- compilable in a finite-state machine

Ideas

- rational expressions including the terminators
- extended with the Cartesian product
- using grain (terminator) as a *type* used to define well-formed expressions:
    - all grains in a sequence have the same type

Rational descriptions

Goal

- description of rational sets of tree linearization
- compilable in a finite-state machine

Ideas

- rational expressions including the terminators
- extended with the Cartesian product
- using grain (terminator) as a *type* used to define well-formed expressions:
  - all grains in a sequence have the same type
  - both operands of binary operators have the same type

Rational descriptions

Goal

- description of rational sets of tree linearization
- compilable in a finite-state machine

Ideas

- rational expressions including the terminators
- extended with the Cartesian product
- using grain (terminator) as a *type* used to define well-formed expressions:
  - all grains in a sequence have the same type
  - both operands of binary operators have the same type
  - Cartesian product restricted with respect to grain composition and closure under intersection constraint (e.g. length-preserving constraint).

Rational descriptions

Goal

- description of rational sets of tree linearization
- compilable in a finite-state machine

Ideas

- rational expressions including the terminators
- extended with the Cartesian product
- using grain (terminator) as a *type* used to define well-formed expressions:
  - all grains in a sequence have the same type
  - both operands of binary operators have the same type
  - Cartesian product restricted with respect to grain composition and closure under intersection constraint (e.g. length-preserving constraint).

In the paper, definitions of of tree sets, linearization sets, etc.

Conclusion

- a richer structure for morphological description

Conclusion

- a richer structure for morphological description
- a finite structure of grains + linear sequences

Conclusion

- a richer structure for morphological description
- a finite structure of grains + linear sequences
- remaining finite-state

Conclusion

- a richer structure for morphological description
- a finite structure of grains + linear sequences
- remaining finite-state
- closure under intersection possible

Conclusion

- a richer structure for morphological description
- a finite structure of grains + linear sequences
- remaining finite-state
- closure under intersection possible
- each grain is a level of synchronization between components of the relation

Conclusion

- a richer structure for morphological description
- a finite structure of grains + linear sequences
- remaining finite-state
- closure under intersection possible
- each grain is a level of synchronization between components of the relation
- in practice, more convenient than other subclasses of relation closed under intersection

Future work

Achieved work

- Definitions and closure properties.

To do

Future work

Achieved work

- Definitions and closure properties.

To do

- improving the formalization of the generalization.

Future work

Achieved work

- Definitions and closure properties.

To do

- improving the formalization of the generalization.
- considering parenthesis instead of terminators to denote the structure in strings.

Future work

Achieved work

- Definitions and closure properties.

To do

- improving the formalization of the generalization.
- considering parenthesis instead of terminators to denote the structure in strings.
- tree operations implementation within the finite-state framework

Thank you