Towards the Cross-Roads of MWE Identification and Tree Correction [WG 4-2-3]

Agata Savary

Université François-Rabelais Tours, Blois, France

Challenge: variability of MWEs

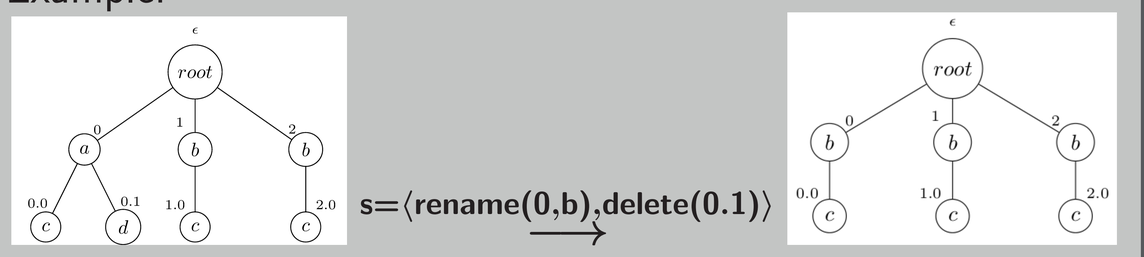
- orthographic
  - ▷ *to see the col**o**r of sb's money* → *to see the col**ou**r* . . . ,
- morphological
  - ▷ *image converter**s***, *image conver**sion*** → *image converter*,
- syntactic
  - ▷ *the beens have been spilled* → *to spill the beans*,
- lexical semantic
  - ▷ (FR) *se* **fourrer** *le doigt dans l'oeil* → *se* **mettre** *le doigt dans l'oeil*
    (lit.) 'to put one's finger in one's eye' = 'to cherish illusions'

MWE identification in a syntax tree

- For a **contiguous** MWE:
  - ▷ generate all possible variants [3],
  - ▷ match them against the leaves of the syntax tree.
- For a **non-contiguous** MWE:
  - ▷ all instantiations correspond to a possibly infinite set of syntactic subtrees (formally: a **tree language**).

Tree-to-tree distance

- Elementary **edit operations** with **costs**, e.g.
  - ▷ relabeling a node,
  - ▷ inserting a leaf,
  - ▷ deleting a leaf (all of cost 1).
- **Edit sequences** transforming one tree into another:
  $\mathbf{t} \xrightarrow{\mathbf{seq}} \mathbf{t'}$
- **Edit distance** between trees $\mathbf{t}$ and $\mathbf{t'}$ – minimal cost of all edit sequences which transform $\mathbf{t}$ into $\mathbf{t'}$
  $$\mathbf{dist(t, t') = min_{t \xrightarrow{seq} t'} cost(seq)}$$
- Example:



$\mathbf{s} = \langle \mathbf{rename(0,b), delete(0.1)} \rangle$ →

t t'

$\mathbf{dist(t, t') = cost(s) = 2}$

Tree-to-language distance

- **Distance** between a tree $\mathbf{t}$ and a **tree language** $\mathbf{L}$ – minimal distance between $\mathbf{t}$ and any tree in $\mathbf{L}$:
  $$\mathbf{DIST(t, L) = min_{t' \in L}\{dist(t, t')\}}$$

Tree-to-language correction [2]

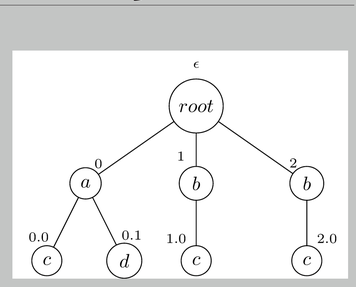
- Input:
  - ▷ tree $\mathbf{t}$,
  - ▷ tree language $\mathbf{L}$,
  - ▷ non-negative threshold $\mathbf{th}$.
- Output – all trees in $\mathbf{L}$ whose distance from $\mathbf{t}$ is no higher than $\mathbf{th}$
  $$\mathbf{corr(t, L, th) = \{t' \in L : dist(t, t') \leq th\}}$$

Tree-to-language correction – example

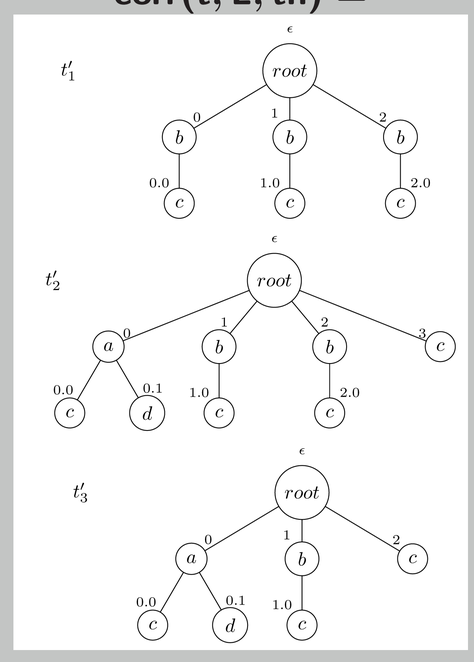
$\mathbf{th = 2}$

$\mathbf{DTD(L) = \{root \to b^*|ab^*c,\ a \to cd,\ b \to c,\ c \to \epsilon,\ d \to \epsilon\}}$

$\mathbf{t \notin L}$=



$\mathbf{corr(t, L, th)}$ =



MWE identification as a tree-to-tree correction problem

- elementary operations on trees – e.g. in LTAG [1]:
  - ▷ substitution and adjunction – cost 0,
  - ▷ inserting or deleting a subtree $\mathbf{t'}$ at a syntactically non-allowed position – cost equal to the size of $\mathbf{t'}$.
- MWE = a tree $\mathbf{t}$ (or a family of trees)
- occurrence of a MWE in a syntax tree = subtree $\mathbf{t'}$,
- MWE identification = finding the distance between $\mathbf{t}$ and $\mathbf{t'}$.

MWE identification as a tree-to-language correction problem

- elementary operations on trees:
  - ▷ syntactically incorrect transformations – non-negative costs.
- MWE = tree language $\mathbf{L}$ (possibly infinite set of trees) – e.g.
  - ▷ *at once*: $\mathbf{[[[at]_{Prep}[once]_{Adv}]_{AdvP}[]_S]_S}$
  - ▷ $\mathbf{L_{at\_once}}$ – set of all trees that result from its auxiliary tree by its adjunction to any other tree.
- occurrence of a MWE in a syntax tree = subtree $\mathbf{t}$,
- MWE identification = correcting $\mathbf{t}$ with respect to $\mathbf{L}$ under a given threshold $\mathbf{th}$.

Applications

- post-annotating MWEs in treebanks,
- detecting MWEs in a post-parsing stage,
- when $\mathbf{th > 0}$:
  - ▷ processing noisy data (spontaneous speech, social networks),
  - ▷ detecting errors in corpus annotation, grammar, or MWE lexicon.

Bibliography

[1] Anne Abeillé and Yves Schabes.
Parsing idioms in lexicalized tags.
In *EACL'89, Manchester*, pages 1–9, 1989.

[2] Joshua Amavi, Béatrice Bouchou, and Agata Savary.
On Correcting XML Documents with Respect to a Schema.
The Computer Journal, 2013.

[3] Agata Savary.
Multiflex: A Multilingual Finite-State Tool for Multi-Word Units.
volume 5642 of *Lecture Notes in Computer Science*, pages 237–240.
Springer, 2009.

agata.savary@univ-tours.fr