Nested Words: representation of data with both a linear ordering and a hierarchically nested matching of items.

refs

> Alur and Madhusudan Adding Nesting Structure to Words JACM 56(3), 2009. https://dl.acm.org/doi/abs/10.1145/1516512.1516518

and

> Marcelo Arenas, Pablo Barceló & Leonid Libkin Regular Languages of Nested Words: Fixed Points, Automata, and Synchronization Theory of Computing Systems volume 49, pages639–670, 2011. https://homepages.inf.ed.ac.uk/libkin/papers/tocs11.pdf

see also

- Nested Words and Visibly Pushdown Languages summary and biblio on newsted words and VPL by R. Alur
- Motley-word automata A note on nested words by A. Blass, Y. Gurevich

---

def. nested word

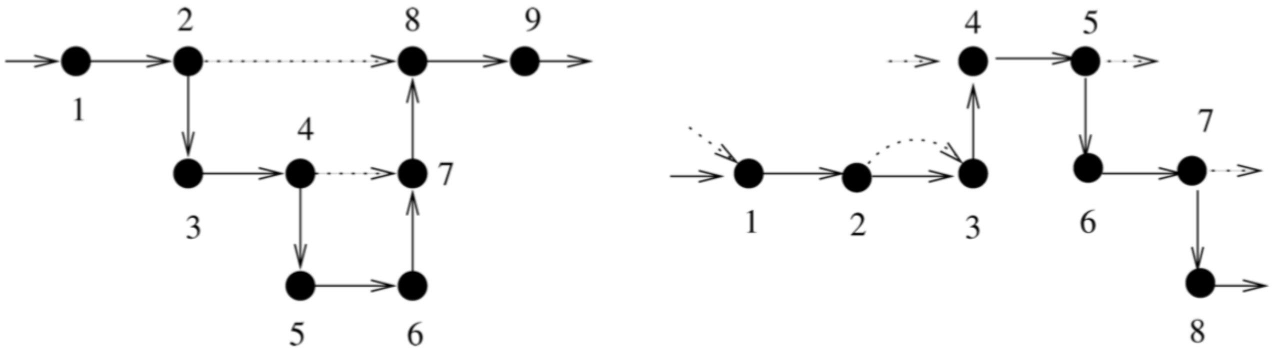
- sequence in $[1..n]$ and
- matching relation $\dashrightarrow$ in $\{-\infty, 1, \ldots, n\} \times \{1, \ldots, n, +\infty\}$

such that:

- match always forward: if $i \dashrightarrow j$ then $i < j$
- match do not share position: $|\{i \mid i \dashrightarrow j\}| \leq 1$ and $|\{j \mid i \dashrightarrow j\}| \leq 1$
- match do not cross: no $i \dashrightarrow j$ and $i' \dashrightarrow j'$ and $i < i' \leq j < j'$

ALT Gurevitch, Blass definition (see Motley-word automata) drops second condition:

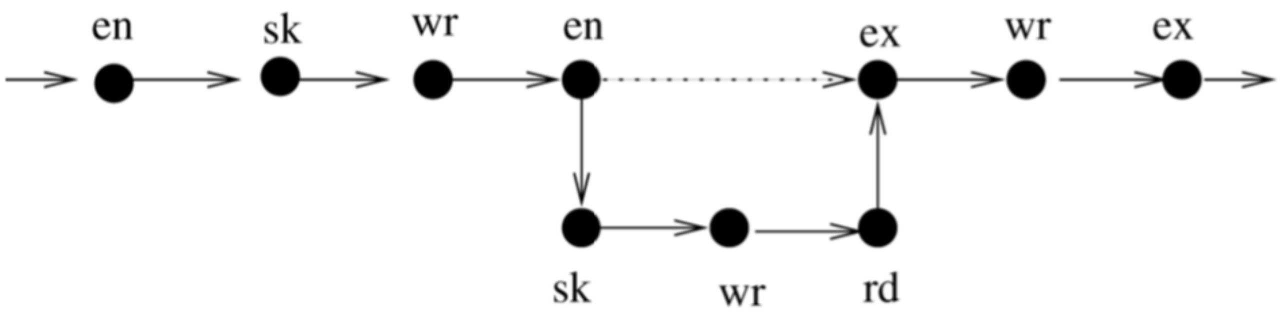
- if $i \dashrightarrow j$ and $i' \dashrightarrow j'$ and $i \leq i'$ then either $i < j < i' < j'$ or $i < i' < j' < j$



---

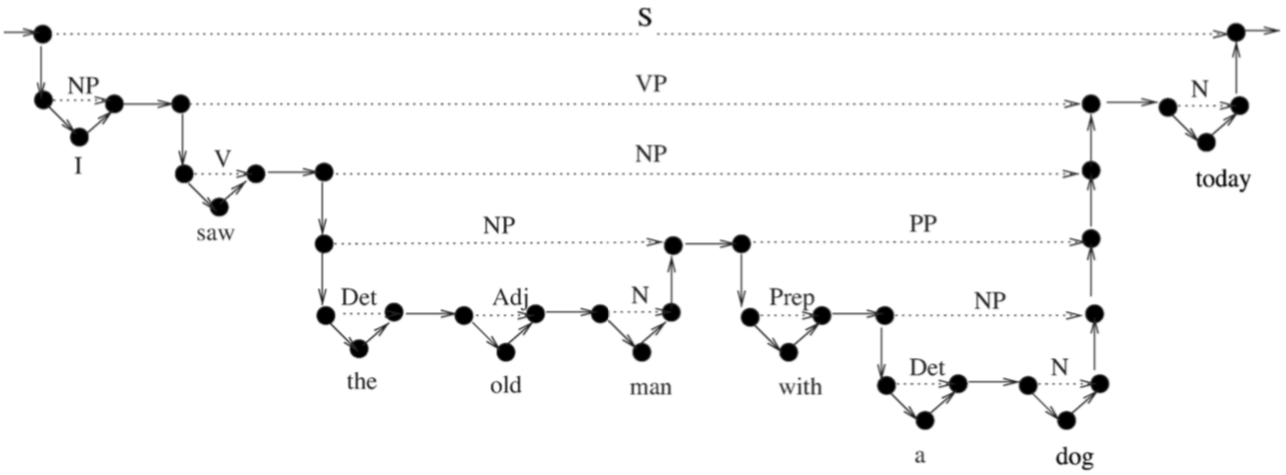
applications:

- Executions of sequential structured programs: matchs = *calls* and *returns*



program execution. en = new scope = call, ex = exit scope = return, rd = read, wr = write, sk = other.

- XML docs: matchs = *open-* and *close- tags*
- Annotated linguistic data: tree bank = repository (corpora) with sentences (linear order) + anotation (hierarchical structure)



parsed sentence as nested word: I saw the old man with a dog today.

---

usefulness:

- queries that refer to both hierarchical and linear structure (not solely on word or tree)
- automata model (*Nested Word Automata*) for reading linear & hierarchical structure in the same time.

Nested-Word Automata

simple definition Blass and Alur (Q, Q_{in}, Q_f, δ) over Σ where $\delta = (\delta_c, \delta_i, \delta_r)$,

- $\delta_c \subseteq Q \times \Sigma \times Q$,
- $\delta_i \subseteq Q \times \Sigma \times Q$,
- $\delta_r \subseteq Q \times Q \times \Sigma \times Q$.

run over nested word $(a_1 \ldots a_k, \dashrightarrow)$ = sequence $q_0, \ldots, q_k$ such that

- $q_0 \in Q_{in}$
- for all i call position of $\dashrightarrow$, $(q_{i-1}, a_i, q_i) \in \delta_c$,
- for all i internal position of $\dashrightarrow$, $(q_{i-1}, a_i, q_i) \in \delta_i$,
- for all i return position of $\dashrightarrow$, with $j \dashrightarrow i$, $(q_{i-1}, q_{j-1}, a_i, q_i) \in \delta_r$.

it means that at call position, the current state is pushed to the stack, and it is popped at return positions.