A New Method for Compiling Parallel Replacement Rules

Anssi Yli-Jyrä[1,2] and Kimmo Koskenniemi[2]
[1]CSC – Scientific Computing Ltd.
[2]Department of General Linguistics, University of Helsinki
email: {firstname.lastname}@helsinki.fi

Replacement Rules

A *replacement rule* is a particular kind of rewriting rule whose interpretation gives a rational relation between two string sets.

{d i e B u c h e r, ...}
↕
`u -> ü // _LETTER* e r`
↕
{d i e B ü c h e r, ...}

Kempe and Karttunen (1996) presented previosly a method that compiles a rational relation from a set of *parallel replacement rules*.

{a a b, ...}
↕
`a->b // a_ ,, b->c // b_`
↕
{a b c, ...}

The Problem

Parallel replament rules can be very complex, having the form:

$$U_{1,1}\text{->}L_{1,1}, \ldots, U_{1,n_1}\text{->}L_{1,n_1}\text{//}\lambda_{1,1}\_\rho_{1,1}, \ldots, \lambda_{1,m_1}\_\rho_{1,m_1},,$$
$$\ldots$$
$$U_{k,1}\text{->}L_{k,1}, \ldots, U_{k,n_k}\text{->}L_{k,n_k}\text{//}\lambda_{k,1}\_\rho_{k,1}, \ldots, \lambda_{k,m_k}\_\rho_{k,m_k}$$

We need a simple and efficient method for compiling a large set of rules. The method of Kempe and Karttunen is not satisfactory because (i) it is very complicated to reimplement and verify and (ii) because it employs internally $O(\sum_{i=1}^{k} m_i)$ auxiliary markers.

The Main Result

We present a new compilation method for parallel replacement rules, consisting of the following sub-methods:

- Generalized Two-Level Grammars, GTWOL, (Yli-Jyrä & Koskenniemi 2006)
- Optimality through Matching (Gerdemann & van Noord 2000).

Further Information:

www.ling.helsinki.fi/~aylijyra/replace

A Radically Different Approach

The prior methods of Kaplan & Kay (1994), Karttunen (1995,1996), Kempe & Karttunen (1996) and Mohri & Sproat (1996) are best described using the closure property of rational relations under *composition* e.g.:

$G_{\text{parses}} =$ *Constraints*
∘ *RightContexts*
∘ *Replace*
∘ *LeftContexts*
$G_{\text{result}} =$ *InsertBrackets*
∘ G_{parses}
∘ *RemoveBrackets*

In contrast to the prior art, our new method is best described using other closure properties of rational relations::

$G_{\text{TWOL}} =$ $\mu(D) - \mu(G)$ *EliminateAnchors*$_{\{@_L,@_R\}}$

$G_{\text{parses}} =$ G_{TWOL} ω_{T_1} ω_{T_2} $\ldots\omega_{T_n}$

$G_{\text{result}} =$ G_{parses} *EliminateBrackets*$_{\{<,>\}}$

A Subroutine: Generalized TWOL

Parallel finite-state rules were already used in the *classical two-level grammars* (Koskenniemi 1983) and *partition-based two-level grammars* (Grimley-Evans *et al.* 1996). However, the *generalized two-level grammars* (GTWOLs) of Yli-Jyrä and Koskenniemi (2006) come closest to the meaning of the replacement by supporting default identity mapping, multi-segment rewriting and conflict resolution. The *context restriction* rules (with `=>`) of GTWOL provide immediately a way to express *optional* replace rules where centres can overlap. However, some *obligatory* replaces, such as `AB->ap, BC->bc`, are not directly expressible in GTWOL:

`A:a B:p <=> _; B:b C:c <=> _` (1)

The sample rules of (1) try to rewrite the prefix and suffix of string `ABC` respectively as `ap` and `bc`, but these substrings cannot coincide in any output string. Thus, string `ABC` gets no output. In comparison, `AB->ap, BC->bc` maps string `ABC` in two ways: `apC` and `Abc`.

From Replace Rules to GTWOL Rules

All parallel replacements consist of elementary rules. An elementary replament rule is of the form:

$$U_i \text{ => } L_i \text{ // .\#. } \lambda_i\_\rho_i \text{ .\#.} \quad (2)$$
where $U_i, L_i, \lambda_i, \rho_i \subseteq \Sigma^*$.

A set of such rules is converted to a GTWOL as follows:

STEP 1A: MAKE CONTEXTS ORIENTED:
$$\lambda_i' = \Sigma^* \times \lambda_i, \rho_i' = \rho_i \times \Sigma^*.$$

STEP 1B: INSERT BRACKETS:
$$B = \{<,>\}, \Sigma' = \Sigma \cup B, \lambda'' = \lambda' \sqcup B^*, \rho'' = \rho' \sqcup B^*.$$

STEP 1C: EMBRACE CENTRES WITH BRACKETS:
$$X_i = <(U_i \times L_i)>.$$

STEP 1D: BAN BRACKETS BY DEFAULT:
$$B \Rightarrow \emptyset\_\emptyset.$$

STEP 2A: REDUCE (2) TO A CR RULE:
$$X_i \Rightarrow \lambda_i''\_\rho_i''.$$

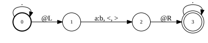
STEP 2B: COMPILE THE GTWOL GRAMMAR:
$$\Pi = \Sigma' \times \Sigma', \quad N = \Pi - \{a{:}a | a \in \Sigma'\}$$
$$D = \Pi^* @_L (N \cup B) @_R \Pi^*, \quad G = \cup_i \ \lambda_i'' @_L X_i @_R \rho_i''$$
$$G_{\text{TWOL}} = \mu(D) \overset{@_L,@_R}{\Longrightarrow} \mu(G),$$

where $\mu(L) = \{vx_1@_L x_2 @_R x_3 y | v @_L x_1 x_2 x_3 @_R y \in L\}$. Operation $W \overset{@_L,@_R}{\Longrightarrow} W'$ is a *generalized restriction* (Yli-Jyrä and Koskenniemi 2004):

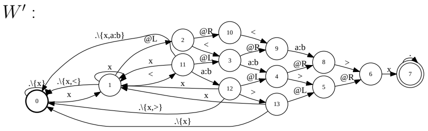
$$W \overset{@_L,@_R}{\Longrightarrow} W' = \Pi^* - \textit{EliminateAnchors}_{\{@_L,@_R\}}(W - W')$$

Example: `a -> b // x _ x`

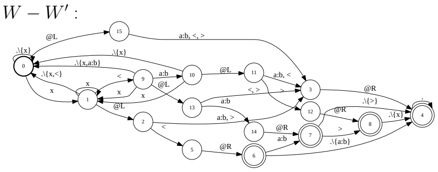
W :



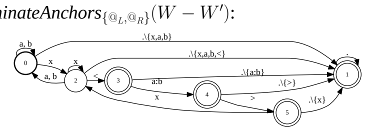
W' :



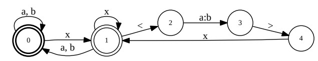
$W - W'$:



EliminateAnchors$_{\{@_L,@_R\}}(W - W')$:



$G_{\text{TWOL}} = \Pi^* - \textit{EliminateAnchors}_{\{@_L,@_R\}}(W - W')$:

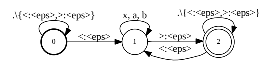


Optimality through Matching

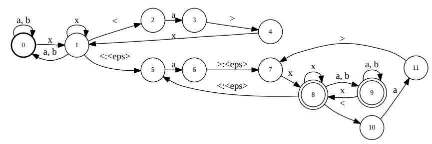
Gerdemann and van Noord (2000) approximated gradient OT constraints with a matching method. We use a similar method to compile obligatory replacement:

STEP 3: MATCH BETTER AND WORSE PARSES:
$$Bw(G) = G \circ T_{\text{mostobl}} \circ G.$$

T_{mostobl} :

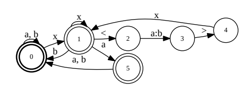


$Bw(\textit{Domain}(G_{\text{TWOL}}))$:



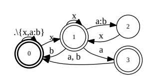
STEP 4A: DELETE WORSE PARSES FROM G_{TWOL}:
$$G_{\text{parse}} = \omega_{T_{\text{mostobl}}}(G_{\text{TWOL}}), \text{ where}$$
$$\omega_{T_{\text{mostobl}}}(C) = \overline{\textit{Range}(Bw(\textit{Domain}(C)))} \circ C.$$

G_{parse} :



STEP 4B: ELIMINATE BRACKETS:
$$G_{\text{result}} = \textit{EliminateBrackets}_{\{<,>\}}(G_{\text{parse}}).$$

G_{result} :



Concluding Remarks

Kempe and Karttunen (1996) used up to $4m$ brackets $\ldots, <_{mE}, <_m, >_{mE}, <_m$ to indicate the presence of m contexts. Their occurrences were constrained with Kaplan and Kay's (1994) *P-iff-S* idiom:

$$\textit{P-iff-S}(\lambda_i, <_i) = \overline{\Pi^*\lambda_i, \overline{<_i \ \Pi^*}} \cap \overline{\overline{\Pi^*\lambda_i}, <_i \ \Pi^*}$$

Both the *P-iff-S* idiom and the generalized restriction operation involve a double negation, but our method needs all together only four auxiliary symbols: $<, >, @_L, @_R$.
The construction of G_{TWOL} can be described using only star-free operations (including complementation and generalized restriction).

References

[1] Dale Gerdemann and Gertjan van Noord. Approximation and exactness in Finite-State Optimality Theory. In Jason Eisner, Lauri Karttunen, and Alain Thériault, editors, *SIGPHON 2000, Finite State Phonology*, 2000.

[2] Edmund Grimley-Evans, George Anton Kiraz, and Stephen G. Pulman. Compiling a partition-based two-level formalism. In *16th COLING 1996, Proc. Conference*, volume 1, pages 454–459, Copenhagen, Denmark, 1996.

[3] Ronald M. Kaplan and Martin Kay. Regular models of phonological rule systems. *Computational Linguistics*, 20(3):331–378, September 1994.

[4] Lauri Karttunen. The replace operator. In *33th ACL 1995, Proceedings of the Conference*, pages 16–23, Cambridge, MA, USA, 1995.

[5] Lauri Karttunen. Directed replace operator. In Emmanuel Roche and Yves Schabes, editors, *Finite-state language processing*, pages 117–147, Cambridge, Massachusetts, 1996. A Bradford Book. The MIT Press.

[6] André Kempe and Lauri Karttunen. Parallel replacement in finite state calculus. In *16th COLING 1996, Proc. Conference*, volume 2, pages 622–627, Copenhagen, Denmark, 1996.

[7] Mehryar Mohri and Richard Sproat. An efficient compiler for weighted rewrite rules. In *34th ACL 1996, Proc. Conference*, pages 231–238, Santa Cruz, CA, USA, June 24-27 1996.

[8] Anssi Yli-Jyrä and Kimmo Koskenniemi. Compiling generalized two-level rules and grammars. In *Proceedings of FinTAL 2006*, LNAI, 2006.

[9] Anssi Mikael Yli-Jyrä and Kimmo Koskenniemi. Compiling contextual restrictions on strings into finite-state automata. In Loek Cleophas and Bruce W. Watson, editors, *The Eindhoven FASTAR Days, Proceedings*, number 04/40 in Computer Science Reports, Eindhoven, The Netherlands, December 2004. Technische Universiteit Eindhoven.

†Address for correspondence: Department of General Linguistics, University of Helsinki, PL 9, 00014 University of Helsinki, Finland