Regional Finite-State Error Repair*

M. Vilares[1], J. Otero[1], and J. Graña[2]

[1] Department of Computer Science, University of Vigo
Campus As Lagoas s/n, 32004 Orense, Spain
{vilares,jop}@uvigo.es

[2] Department of Computer Science, University of A Coruña
Campus de Elviña s/n, 15071 A Coruña, Spain
grana@udc.es

Abstract. We describe an algorithm to deal with error repair over finite-state architectures. Such a technique is of interest in spelling correction as well as approximate string matching in a variety of applications related to natural language processing, such as information extraction/recovery or answer searching, where error-tolerant recognition allows misspelled input words to be integrated in the computational process. Our proposal relies on a regional least-cost repair strategy, dynamically gathering all relevant information in the context of the error location. The system guarantees asymptotic equivalence with global repair strategies.

1 Introduction

An ongoing question in natural language processing (NLP) is how to recover ungrammatical structures for processing text. Focusing on spelling correction tasks, there are few things more frustrating than spending a great deal of time debugging typing or other errors in order to ensure the accuracy of NLP tools over large amount of data. As a consequence, although it is one of the oldest applications to be considered in the field of NLP [4], there is an increased interest in devising new techniques in this area.

In this regard, previous proposals extend the repair region to the entire string, complemented with the consideration of thresholds on an editing distance [7, 8]. This global approach, which seems to be universally accepted, has probably been favored by the consideration of English, a non-concatenative language with a reduced variety of morphological associated processes [11], as running language. However, the application of this kind of techniques to highly inflectional languages such as Latin ones [1], or agglutinative languages such as Turkish [10], could fail to take advantage of the underlying grammatical structure, leading to a significant loss of efficiency.

In this context, we are interested in exploring regional repair techniques, introducing proper tasks for error location and repair region estimation. Our

* Research partially supported by the Spanish Government under projects TIC2000-0370-C02-01 and HP2002-0081, and the Autonomous Government of Galicia under projects PGIDIT03SIN30501PR and PGIDIT02SIN01E.

aim is to avoid examining the entire word, in contrast to global algorithms that expend equal effort on all parts of the word, including those containing no errors.

2 The operational model

Our aim is to parse a word $w_{1..n} = w_1 \ldots w_n$ according to a regular grammar $\mathcal{G} = (N, \Sigma, P, S)$, where N is the set of non-terminals, Σ the set of terminal symbols, P the rules and S the start symbol. We denote by w_0 (resp. w_{n+1}) the position in the string, $w_{1..n}$, previous to w_1 (resp. following w_n). We generate from $\mathcal{G}$ a *numbered minimal acyclic finite state automaton* for the language $\mathcal{L}(\mathcal{G})$. In practice, we choose a device [6] generated using Galena [3]. A *finite automaton* (FA) is a 5-tuple $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ where: $\mathcal{Q}$ is the set of states, Σ the set of input symbols, δ is a function of $\mathcal{Q} \times \Sigma$ into $2^{\mathcal{Q}}$ defining the transitions of the automaton, q_0 the initial state and $\mathcal{Q}_f$ the set of final states. We denote $\delta(q, a)$ by $q.a$, and we say that the FA is *deterministic* when, in any case, $\mid q.a \mid \leq 1$. The notation is transitive, so $q.w$ denotes the state reached by using the transitions labelled by each letter $w_i,\ i \in \{1, ..., n\}$ of w. Therefore, w is *accepted* iff $q_0.w \in \mathcal{Q}_f$, that is, the *language accepted by* $\mathcal{A}$ is defined as $\mathcal{L}(\mathcal{A}) = \{w,$ such that $q_0.w \in \mathcal{Q}_f\}$. An FA is *acyclic* when the underlying graph is. We talk about a *path in the* FA to refer to a sequence of states $\{q_1, \ldots, q_n\}$, such that $\forall i \in \{1, \ldots, n-1\},\ \exists a_i \in \Sigma,\ q_i.a_i = q_{i+1}$.

In order to reduce the memory requirements, we minimize the FA [2]. So, we say that two FAs are *equivalent* iff they recognize the same language. Two states, p and q, are *equivalent* iff the FA with p as initial state and the one that starts in q recognize the same language. An FA is *minimal* iff no pair in $\mathcal{Q}$ is equivalent.

It is important to note that although the standard recognition process is deterministic, the repair process could introduce non-determinism by exploring alternatives associated to possibly more than one recovery strategy. So, in order to get polynomial complexity, we avoid duplicating intermediate computations in the repair of $w_{1..n} \in \Sigma^+$, storing them in a table $\mathcal{I}$ of *items*, $\mathcal{I} = \{[q, i],\ q \in \mathcal{Q},\ i \in [1, n+1]\}$, where $[q, i]$ looks for the suffix $w_{i..n}$ to be analyzed from $q \in \mathcal{Q}$.

We describe our proposal using *parsing schemata* [9], a triple $\langle \mathcal{I}, \mathcal{H}, \mathcal{D} \rangle$, with $\mathcal{H} = \{[a, i],\ a = w_i\}$ an initial set of items called *hypothesis* that encodes the word to be recognized[1], and $\mathcal{D}$ a set of *deduction steps* that allow new items to be derived from already known items. Deduction steps are of the form $\{\eta_1, \ldots, \eta_k \vdash \xi \,/ conds\}$, meaning that if all antecedents η_i are present and the conditions *conds* are satisfied, then the consequent ξ is generated. In our case, $\mathcal{D} = \mathcal{D}^{\text{Init}} \cup \mathcal{D}^{\text{Shift}}$, where:

$$\mathcal{D}^{\text{Init}} = \{ \vdash [q_0, 1] \} \qquad \mathcal{D}^{\text{Shift}} = \{ [p, i] \vdash [q, i+1] \,/ \exists [a, i] \in \mathcal{H},\ q = p.a \}$$

The recognition associates a set of items S_p^w, called *itemset*, to each $p \in \mathcal{Q}$; and applies these deduction steps until no new application is possible. The word is recognized iff a *final item* $[q_f, n+1],\ q_f \in \mathcal{Q}_f$ has been generated. We can

[1] A word $w_{1...n} \in \Sigma^+$, $n \geq 1$ is represented by $\{[w_1, 1],\ [w_2, 2],\ \ldots\ ,\ [w_n, n]\}$.

assume, without lost of generality, that $\mathcal{Q}_f = \{q_f\}$, and that exists an only transition from (resp. to) q_0 (resp. q_f). To get it, we augment the original FA with two states becoming the new initial and final states, and relied to the original ones through empty transitions, a concession to the minimality.

3 The edit distance

The *edit distance* [5] between two strings measures the minimum number of editing operations of insertion, deletion, replacement of a symbol, and transposition of adjacent symbols that are needed to convert one string into another. Let $x_{1..m}$ (resp. $y_{1..n}$) be the misspelled string (resp. a possible partial candidate string), the edit distance, $ed(x, y)$ is computed as follows:

$$
\begin{array}{ll}
\mathrm{ed}(x_{i+1}, y_{j+1}) = & \begin{cases}
\mathrm{ed}(x_i, y_j) & \text{iff } x_{i+1} = y_{j+1} \\
 & \text{(last characters are the same)} \\
1 + \min\{\ \mathrm{ed}(x_{i-1}, y_{j-1}), & \\
\qquad\qquad \mathrm{ed}(x_{i+1}, y_j), & \\
\qquad\qquad \mathrm{ed}(x_i, y_{j+1})\} & \text{iff } x_i = y_{j+1}, x_{i+1} = y_j \\
 & \text{(last two characters are transposed)} \\
1 + \min\{\ \mathrm{ed}(x_i, y_j), & \\
\qquad\qquad \mathrm{ed}(x_{i+1}, y_j), & \\
\qquad\qquad \mathrm{ed}(x_i, y_{j+1})\} & \text{otherwise}
\end{cases} \\
\mathrm{ed}(x_0, y_j) = j & \qquad\qquad\qquad\qquad\qquad 1 \leq j \leq n \\
\mathrm{ed}(x_i, y_0) = i & \qquad\qquad\qquad\qquad\qquad 1 \leq i \leq m
\end{array}
$$

where x_0 (resp. y_0) is ε. We can now extend the concept of language accepted by an FA $\mathcal{A}$, $\mathcal{L}(\mathcal{A})$, to define the *language accepted by an* FA $\mathcal{A}$ *with an error threshold* $\tau > 0$ as $\mathcal{L}_\tau(\mathcal{A}) = \{x, \text{ such that } ed(x, y) \leq \tau,\ y \in \mathcal{L}(\mathcal{A})\}$. We shall consider the edit distance as a common metrical basis in order to allow an objective comparison to be made between our proposal and previous ones.

4 Regional least-cost error repair

We talk about the *error* in a portion of the word to mean the difference between what was intended and what actually appears in the word. So, we can talk about the *point of error* as the point at which the difference occurs.

Definition 1. *Let* $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ *be an* FA, *and let* $w_{1..n}$ *be a word. We say that* w_i *is a* point of error *iff it verifies the following conditions:*

$$(1)\ \ q_0.w_{1..i-1} = q \qquad\qquad (2)\ \ q.w_i \notin \mathcal{Q}$$

The point of error is fixed by the recognizer and it provides the starting point for the repair, in which the following step consists in locating the origin of that error. We aim to limit the impact on the prefix already analyzed, focusing on the context close to the point of error and saving on computational effort. To do so, we first introduce a collection of topological properties that we illustrate in Fig. 1.

Definition 2. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, and let $p, q \in \mathcal{Q}$. We say that p* is lesser than *q iff there exists a path $\{p, \ldots, q\}$. We denote that by $p < q$.*

We have, in Fig. 1, that $q_i < q_{i+1},\ \forall i \in \{1, \ldots, 7\}$. Our order is induced by the transitional formalism, which results in a well defined relation since our FA is acyclic. In this sense, we can also give a direction to the paths.

Definition 3. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, we say that $q_s \in \mathcal{Q}$ (resp. q_d) is a* source *(resp.* drain*)* state *for any path in $\mathcal{A}$, $\{q_1, \ldots, q_m\}$, iff $\exists a \in \Sigma$, such that $q_1 = q_s.a$ (resp. $q_m.a = q_d$).*

Intuitively, we talk about source (resp. drain) states on out-coming (resp. incoming) transitions, which orientates the paths from sources to drains. So, in Fig. 1, q_1 (resp. q_8) is a source (resp. drain) for paths $\{q_9\}$, $\{q_2, q_{10}, q_6, q_7\}$, $\{q_2, q_3, q_{11}, q_5, q_6, q_7\}$ or $\{q_2, q_3, q_4, q_5, q_6, q_7\}$. We can now consider a coverage for FAs by introducing the concept of *region*.

Definition 4. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, we say that a pair $(q_s, q_d),\ q_s, q_d \in \mathcal{Q}$ is a* region in *$\mathcal{A}$, denoted by $\mathcal{R}_{q_s}^{q_d}(\mathcal{A})$, iff it verifies that*

(1) $q_s = q_0$ **and** $q_d = q_f$ *(the global* FA*)*
or
(2) $\{\forall \rho,\ source(\rho) = q_s\} \Rightarrow drain(\rho) = q_d$ **and** $\mid \{\forall \rho,\ source(\rho) = q_s\} \mid > 1$

which we write as $\mathcal{R}_{q_s}^{q_d}$ when the context is clear. We also denote paths$(\mathcal{R}_{q_s}^{q_d}) = \{\rho / source(\rho) = q_s,\ drain(\rho) = q_d\}$ *and, given $q \in \mathcal{Q}$, we say that $q \in \mathcal{R}_{q_s}^{q_d}$ iff $\exists \rho \in paths(\mathcal{R}_{q_s}^{q_d}),\ q \in \rho$.*

This allows us to ensure that any state, with the exception of q_0 and q_f, is included in a region. Applied to Fig. 1, the regions are $\mathcal{A} = \mathcal{R}_{q_0}^{q_f}$, $\mathcal{R}_{q_1}^{q_8}$, $\mathcal{R}_{q_2}^{q_7}$ and $\mathcal{R}_{q_3}^{q_5}$, with $\{q_4, q_{11}, q_{12}\} \subset \mathcal{R}_{q_3}^{q_5} \not\ni q_3$ and $\mathcal{R}_{q_2}^{q_7} \ni q_9 \notin \mathcal{R}_{q_3}^{q_5}$. In a region, all prefixes computed before the source can be combined with any suffix from the drain through the paths between both. This provides a criterion to place around a state a zone for which any change in it has no effect on its context.

Definition 5. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, we say that a region $\mathcal{R}_{q_s}^{q_d}$ is the* minimal region in *$\mathcal{A}$* containing *$p \in \mathcal{Q}$ iff it verifies that $q_s \geq p_s$ (resp. $q_d \leq p_d$), $\forall \mathcal{R}_{p_s}^{p_d} \ni p$. We denote it as $\mathcal{M}(\mathcal{A}, p)$, or simply $\mathcal{M}(p)$ when the context is clear.*

In Fig. 1, $\mathcal{M}(q_4) = \mathcal{M}(q_{11}) = \mathcal{R}_{q_3}^{q_5}$ and $\mathcal{M}(q_3) = \mathcal{M}(q_9) = \mathcal{R}_{q_2}^{q_7}$. At this point, it is trivial to prove the following lemma, which guarantees the consistence of the previous concept based on the uniqueness of a minimal region.

Lemma 1. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, then $p \in \mathcal{Q} \setminus \{q_0, q_f\} \Rightarrow \dot{\exists}\ \mathcal{M}(p)$.*

Proof. Trivial from definition 5.

We can now formally introduce the concept of *point of detection*, the point at which the recognizer detects that there is an error and calls the repair algorithm.

Definition 6. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, and let w_j be a point of error in $w_{1..n} \in \Sigma^+$. We say that w_i is a* point of detection associated to w_j *iff:*

$$\exists q_d > q_0.w_{1..j},\ \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}^{q_d}_{q_0.w_{1..i}}$$

We denote this by $\text{detection}(w_j) = w_i$*, and we say that $\mathcal{M}(q_0.w_{1..j})$ is the* region defining the point of detection w_i.

In our example in Fig. 1, if we assume w_j to be a point of error such that $q_{10} = q_0.w_{1..j}$, we conclude that $w_i = \text{detection}(w_j)$ if $q_2 = q_0.w_{1..i}$ since $\mathcal{M}(q_{10}) = \mathcal{R}^{q_7}_{q_2}$. So, the error is located in the immediate left recognition context, given by the closest source. However, we also need to locate it from an operational viewpoint, as an item in the computational process.

Definition 7. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, let w_j be a point of error in $w_{1..n} \in \Sigma^+$, and let w_i be a point of detection associated to w_j. We say that $[q, j] \in S^w_q$ is an* error item *iff $q_0.w_{j-1} = q$; and we say that $[p, i] \in S^w_p$ is a* detection item associated to w_j *iff $q_0.w_{i-1} = p$.*

Following our running example in Fig. 1, $[q_2, i]$ is a detection item for the error item $[q_{10}, j]$. Intuitively, we talk about error and detection items when they represent states in the FA concerned with the recognition of points of error and detection, respectively. Once we have identified the beginning of the repair region from both the topological and the operational viewpoint, we can now apply the *modifications* intended to recover the recognition process from an error.

Definition 8. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, a* modification to $w_{1..n} \in \Sigma^+$ *is a series of edit operations, $\{E_i\}_{i=1}^n$, in which each E_i is applied to w_i and possibly consists of a sequence of insertions before w_i, replacement or deletion of w_i, or transposition with w_{i+1}. We denote it by $M(w)$.*

We now use the topological structure to restrict the notion of modification, introducing the concept of *error repair*. Intuitively, we look for conditions that guarantee the ability to recover the standard recognition, at the same time as they allow us to isolate repair branches by using the concept of path in a region.

Definition 9. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA*, $x_{1..m}$ a prefix in $\mathcal{L}(\mathcal{A})$, and $w \in \Sigma^+$, such that xw is not a prefix in $\mathcal{L}(\mathcal{A})$. We define a* repair of w following *x as $M(w)$, so that:*

(1) $\mathcal{M}(q_0.x_{1..m}) = \mathcal{R}^{q_d}_{q_s}$ *(the minimal region including the point of error, $x_{1..m}$)*
(2) $\exists \{q_0.x_{1..i} = q_s.x_i, \ldots, q_s.x_{i..m}.M(w)\} \in$ *paths*$(\mathcal{R}^{q_d}_{q_s})$

We denote it by $\text{repair}(x, w)$*, and $\mathcal{R}^{q_d}_{q_s}$ by* $\text{scope}(M)$.

However, the notion of *repair*(x, w) is not sufficient for our purposes, since our aim is to extend the recovery process to consider all possible repairs associated to a given point of error, which implies simultaneously considering different prefixes.

Definition 10. *Let* $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ *be an* FA *and let* $y_i \in y_{1..n}$ *be a point of error, we define the* set of repairs for y_i*, as*

$$\text{repair}(y_i) = \{xM(w) \in \textit{repair}(x, w) / w_1 = \textit{detection}(y_i)\}$$

We now need a mechanism to filter out undesirable repair processes, in order to reduce the computational charges. To do so, we should introduce comparison criteria to select only those repairs with minimal cost.

Definition 11. *For each* $a, b \in \Sigma$ *we assume insert,* $I(a)$*; delete,* $D(a)$*, replace,* $R(a, b)$*, and transpose,* $T(a, b)$*, costs. The* cost of a modification $M(w_{1..n})$ *is given by* $\text{cost}(M(w_{1..n})) = \Sigma_{j \in J_{\dashv}} I(a_j) + \Sigma_{i=1}^{n}(\Sigma_{j \in J_i} I(a_j) + D(w_i) + R(w_i, b) + T(w_i, w_{i+1}))$*, where* $\{a_j,\ j \in J_i\}$ *is the set of insertions applied before* w_i*;* $w_{n+1} = \dashv$ *the end of the input and* $T_{w_n, \dashv} = 0$*.*

In order to take edit distance as the error metric for measuring the quality of a repair, it is sufficient to consider discrete costs $I(a) = D(a) = 1,\ \forall a \in \Sigma$ and $R(a, b) = T(a, b) = 1,\ \forall a, b \in \Sigma,\ a \neq b$. On the other hand, when several repairs are available on different points of detection, we need a condition to ensure that only those with the same minimal cost are taken into account, looking for the best repair quality. However, this is not in contradiction with the consideration of error thresholds or alternative error metrics.

Definition 12. *Let* $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ *be an* FA *and let* $y_i \in y_{1..n}$ *be a point of error, we define the set of* regional repairs for y_i*, as follows:*

$$\textit{regional}(y_i) = \left\{xM(w) \in \textit{repair}(y_i) \middle/ \begin{array}{l} \textit{cost}(M) \leq \textit{cost}(M'),\ \forall M' \in \textit{repair}(x, w) \\ \textit{cost}(M) = \min_{L \in \textit{repair}(y_i)}\{\textit{cost}(L)\} \end{array} \right\}$$

It is also necessary to take into account the possibility of cascaded errors, that is, errors precipitated by a previous erroneous repair diagnosis. Prior to dealing with the problem, we need to establish the existing relationship between the regional repairs for a given point of error and future points of error.

Definition 13. *Let* $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ *be an* FA *and let* w_i, w_j *be points of error in* $w_{1..n} \in \Sigma^+$*,* $j > i$*. We define the set of* viable repairs for w_i in w_j*, as*

$$\textit{viable}(w_i, w_j) = \{xM(y) \in \textit{regional}(w_i) / xM(y) \ldots w_j \textit{ prefix for } \mathcal{L}(\mathcal{A})\}$$

Intuitively, the repairs in $\textit{viable}(w_i, w_j)$ are the only ones capable of ensuring the continuity of the recognition in $w_{i..j}$ and, therefore, the only possible repairs at the origin of the phenomenon of cascaded errors.

Definition 14. *Let* w_i *be a point of error for* $w_{1..n} \in \Sigma^+$*, we say that a point of error* $w_k,\ k > j$ *is a* point of error precipitated by w_j *iff*

$$\forall xM(y) \in \textit{viable}(w_j, w_k),\ \exists \mathcal{R}^{q_d}_{q_0.w_{1..i}} \textit{ defining } w_i = \textit{detection}(w_j)$$

such that $\textit{scope}(M) \subset \mathcal{R}^{q_d}_{q_0.w_{1..i}}$*.*

In practice, a point of error w_k is precipitated by the result of previous repairs on a point of error w_j, when the region defining the point of detection for w_k summarizes all viable repairs for w_j in w_k. This implies that the information compiled from those repair regions has not been sufficient to give continuity to a recognition process locating the new error in a region containing the preceding ones and, therefore, depending on them. That is, the underlying grammatical structure suggests that the origin of the current error could be a mistaken treatment of past errors. Otherwise, the location would be fixed in a zone not depending on these previous repairs.

5 The algorithm

We propose that the repair be obtained by searching the FA itself to find a suitable configuration to allow the recognition to continue, a classic approach in error repair. However, in the state of the art there is no theoretical size limit for the repair region, but only for the edit distance on corrections in it. So, in order to avoid distortions due to unsafe error location, the authors make use of global algorithms limiting the computations by a threshold on the edit distance. This allows them to restrict the section of the FA to be explored by pruning either all repair paths which are more distant from the input than the threshold [7], or those not maintaining a minimal distance no bigger than the threshold [8].

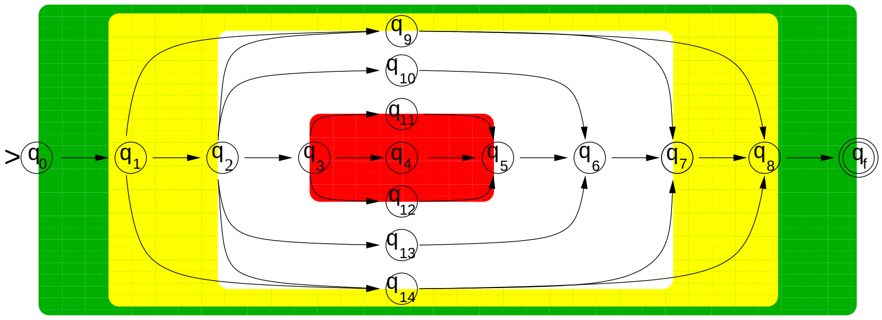

Fig. 1. The concept of region applied to error repair

However, the fact that we are not profiting from the linguistic knowledge present in the FA to locate the error and to delimit its impact may lead to suboptimal computational costs or to precipitating new errors. We eliminate this problem by a construction where all repair phases are dynamically guided by the FA itself and, therefore, inspired by the underlying grammatical structure.

5.1 A simple case

We assume that we are dealing with the first error detected in a word $w_{1..n} \in \Sigma^+$. The major features of the algorithm involve beginning with the error item, whose error counter is zero. So, we extend the item structure, $[p, i, e]$, where e is now the error counter accumulated in the recognition of w at position w_i in state p.

We refer again to Fig. 1. So, given an error item, $[q_{10} = q_0.w_{1..j}, j, e_j]$, the system locates the corresponding detection item, $[q_2 = q_0.w_{1..i}, i, e_i]$, by using a pointer on $\mathcal{M}(q_{10}) = \mathcal{R}^{q_7}_{q_2}$. We then apply all possible transitions in this region beginning at both, the point of error and the its associated point of detection, which corresponds to the following deduction steps in error mode, $\mathcal{D}_{\text{error}} = \mathcal{D}^{\text{Shift}}_{\text{error}} \cup \mathcal{D}^{\text{Insert}}_{\text{error}} \cup \mathcal{D}^{\text{Delete}}_{\text{error}} \cup \mathcal{D}^{\text{Replace}}_{\text{error}} \cup \mathcal{D}^{\text{Transpose}}_{\text{error}}$:

$$\mathcal{D}^{\text{Shift}}_{\text{error}} = \{[p, i, e] \vdash [q, i+1, e], \ \exists [a, i] \in \mathcal{H}, \ q = p.a\}$$

$$\mathcal{D}^{\text{Insert}}_{\text{error}} = \{[p, i, e] \vdash [p, i+1, e + I(a)], \ \nexists \, p.a\}$$

$$\mathcal{D}^{\text{Delete}}_{\text{error}} = \{[p, i, e] \vdash [q, i-1, e + D(w_i)] \Big/ \begin{array}{l} \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}^{q_d}_{q_s} \\ p.w_i = q_d \in \mathcal{R}^{q_d}_{q_s} \textbf{ or } q = q_d \end{array} \}$$

$$\mathcal{D}^{\text{Replace}}_{\text{error}} = \{[p, i, e] \vdash [q, i+1, e + R(w_i, a)], \Big/ \begin{array}{l} \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}^{q_d}_{q_s} \\ p.a = q \in \mathcal{R}^{q_d}_{q_s} \textbf{ or } q = q_d \end{array} \}$$

$$\mathcal{D}^{\text{Transpose}}_{\text{error}} = \{[p, i, e] \vdash [q, i+2, e + T(w_i, w_{i+1})] \Big/ \begin{array}{l} \mathcal{M}(q_0.w_{1..j}) = \mathcal{R}^{q_d}_{q_s} \\ p.w_i.w_{i+1} = q \in \mathcal{R}^{q_d}_{q_s} \textbf{ or } q = q_d \end{array} \}$$

where $w_{1..j}$ looks for the current point of error. Note that, in any case, the error hypotheses apply on transitions behind the repair region. The process continues until a repair covers the repair region, accepting a character in the remaining string. Returning to Fig. 1, the scope of repair for the error detected at $w_i \in \mathit{detection}(w_j)$ is $\mathcal{M}(q_{10}) = \mathcal{R}^{q_7}_{q_2}$, the region defining the detection item $[q_2 = q_0.w_{1..i}, i, e_i]$. Once this has been performed on each recognition branch, we select the regional repairs and the process goes back to standard mode.

5.2 The general case

We now assume that the repair process is not the first one in the word and, therefore, can modify a previous one. This arises when we realize that we come back to a detection item for which some recognition branch includes a previous repair process. To illustrate such a case, we return to Fig. 1 assuming $[q_{10} = q_0.w_{1..k}, k, e_k]$ and $[q_8 = q_0.w_{1..l}, l, e_l]$ to be points of error. As a consequence, $[q_8 = q_0.w_{1..l}, l, e_l]$ would be precipitated by $[q_{10} = q_0.w_{1..k}, k, e_k]$ since $\mathcal{A} = \mathcal{R}^{q_f}_{q_0}$ defining $w_0 = \text{detection}(w_l)$ includes $\mathcal{R}^{q_7}_{q_2 = q_0.w_{1..j}}$, the scope of a previous repair.

To deal with precipitated errors, the algorithm re-takes the previous error counters, adding the cost of the new repair hypotheses to profit from the experience gained from previous recovery phases. At this point, regional repairs have two important properties. First, they are independent of the FA construction and secondly, there is no loss of efficiency in relation to global repair approaches.

Lemma 2. *(The Expansion Lemma) Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA *and let w_k, w_l be points of error in $w_{1..n} \in \Sigma^+$, such that w_l is precipitated by w_k, then:*

$$q_0.w_{1..i} < q_0.w_{1..j}, \ \mathcal{M}(q_0.w_l) = \mathcal{R}^{q_d}_{q_0.w_{1..i}}, \ w_j = y_1, \ xM(y) \in \mathit{viable}(w_k, w_l)$$

Proof. Let $w_j \in \Sigma$, such that $w_j = y_1,\ xM(y) \in \mathit{viable}(w_k, w_l)$ be a point of detection for w_k, for which some recognition branch derived from a repair in $\mathit{regional}(w_k)$ has successfully arrived at w_l. Let also w_l be a point of error precipitated by $xM(y) \in \mathit{viable}(w_k, w_l)$. By definition 14, we can affirm that

$$\text{scope}(M) \subset \mathcal{M}(q_0.w_l) = \mathcal{R}^{q_d}_{q_0.w_{1..i}}$$

Given that $\mathit{scope}(M)$ is the lowest region summarizing $q_0.w_{1..j}$, it follows that $q_0.w_{1..i} < q_0.w_{1..j}$. We conclude the proof by extending it to all repairs in $\mathit{viable}(w_k, w_l)$. □

Intuitively, we prove that the state associated to the point of detection in a cascaded error is lesser than the one associated to the source of the scope in the repairs precipitating it. As a consequence, the minimal possible scope of a repair for the cascaded error includes any scope of those previous repairs.

Corollary 1. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA *and let w_k, w_l be points of error in $w_{1..n} \in \Sigma^+$, such that w_l is precipitated by w_k, then*

$$\mathit{max}\{\text{scope}(M),\ M \in \mathit{viable}(w_k, w_l)\} \subset \mathit{max}\{\text{scope}(\tilde{M}),\ \tilde{M} \in \mathit{regional}(w_l)\}$$

Proof. It immediately follows from lemma 2. □

This allows us to get an asymptotic behavior close to global repair methods. That is, the algorithm ensures a quality comparable to global strategies, but at the cost of a local one. This has profound implications for the efficiency, measured by time, the simplicity and the power of computing regional repairs.

Lemma 3. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA *and let w_i be a point of error in $w_{1..n} \in \Sigma^+$, the time bound for the regional repair is, in the worst case,*

$$\mathcal{O}\left(\frac{n!}{\tau! * (n-\tau)!} * (n+\tau) * 2^\tau * \mathit{fan\text{-}out}_\mu^\tau\right)$$

where τ and fan-out$_\mu$ *are, respectively, the maximal error counter computed and the maximal fan-out of the automaton in the scope of the repairs considered.*

Proof. Here, the proof is a simple extrapolation of the estimation proposed for the Savary's algorithm [8]. In the worst case, there are at most $n!/(\tau! * (n-\tau)!)$ possible distributions of τ modifications over n word positions. For each distribution $(1 + 2 * \text{fan-out}_\mu)^\tau$ paths at most are followed, each path being of length $n + \tau$ at most. So, the worst case complexity is the one proposed. □

However, this lemma does not yet determine the relation with classic global approaches [7, 8], as is our aim, but only an average case estimation of our own time complexity. To reach this, we extend the repair region to the total FA.

Corollary 2. *Let $\mathcal{A} = (\mathcal{Q}, \Sigma, \delta, q_0, \mathcal{Q}_f)$ be an* FA *and let w_i be a point of error in $w_{1..n} \in \Sigma^+$, the time bound for the regional repair is, in the worst case, the same reached for a global approach.*

Proof. It immediately follows from the previous lemma 3 and corollary 1, as well as [8]. In effect, in the worst case, the scope of the repair is the global FA. □

Taking into account the kind of proof applied on lemma 3, this implies that our technique has the same time complexity claimed for Savary's global one [8], in the best of our knowledge the most efficient proposal on spelling correction.

6 Practical aspects

Our aim here is to validate the practical interest of our proposal in relation to classic global ones, trying to corroborate the theoretical results previously advanced. We think that it is an objective criterion for measuring the quality of a repair algorithm, since the point of reference is a technique that guarantees the best quality for a given error metric when all contextual information is available. So, we have compared our algorithm with the Savary's global approach [8]. The restrictions imposed on the length of this paper limit our present discussion to some relevant practical details.

6.1 The running languages

We choose to work with lexicons for languages with a great variety of morphological processes, which make them adequate for our description. In particular, the first preliminary practical tests have been performed on Spanish. The most outstanding features are to be found in verbs, with their highly complex conjugation paradigm, as well as in complex gender and number inflection.

We have taken for Spanish a lexicon with 514,781 different words, to illustrate our work. This lexicon is recognized by an FA containing 58,170 states connected by 153,599 transitions, of sufficient size to allow us to consider this automaton as a representative starting point for our purposes. From this lexicon, we have selected a representative sample of morphological errors for practical evaluation of the algorithm. This sample has the same distribution observed in the original lexicon in terms of lengths of the words dealt with. This is of some importance since, as the authors claim, the efficiency of previous proposals depends on these factors [7, 8], which makes no practical sense. No other dependencies have been detected at morphological level and, therefore, they have not been considered. In each length-category, errors have been randomly generated in a number and position in the input string.

6.2 Preliminary experimental results

We are interested in both computational and quality aspects. In this sense, we consider the concept of item previously defined in order to measure the computational effort. To take into account data related to the performance from both the user's and the system's viewpoint, we have introduced the following two measures, for a given word, w, containing an error:

$$performance(w) = \frac{useful\ items}{total\ items} \qquad recall(w) = \frac{proposed\ corrections}{total\ corrections}$$

that we complement with a global measure on the *precision* of the error repair approach in each case, that is, the rate reflecting when the algorithm provides the correction attended by the user. We use the term *useful items* to refer to the number of generated items that finally contribute to the obtaining of a repair, and *total items* to refer to the number of these structures generated during the process. We denote by *proposed corrections* the number of corrections provided by the algorithm, and by *total corrections* the number of possible ones, in absolute terms.

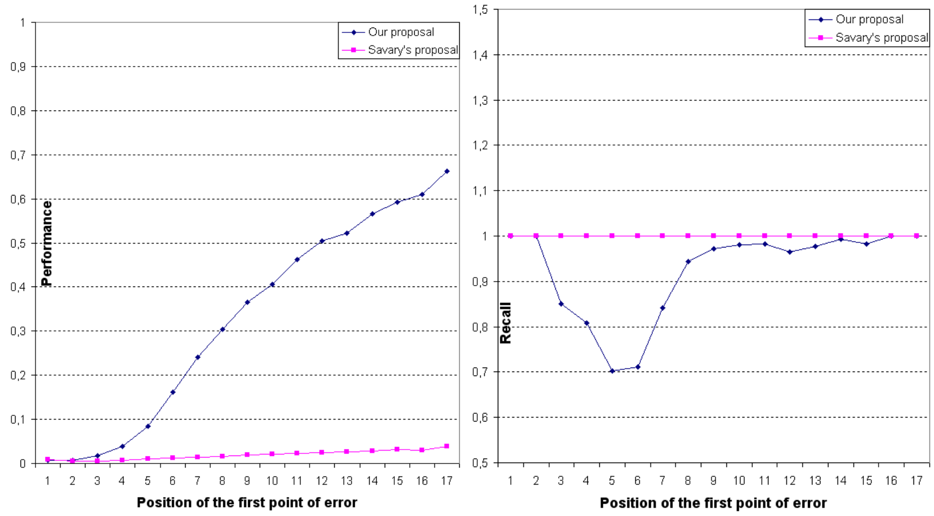

Fig. 2. Performance and recall results.

The practical results shown in Fig. 2 appear to corroborate that not only the performance in our case is better than Savary's, but also that the difference existing between them increases with the location of the first point of error. With respect to the recall relation, Savary's algorithm shows a constant graph since the approach applied is global and, consequently, the set of corrections provided is always the entire one for a fixed error counter. In our proposal, the results prove that the recall is smaller than that for Savary's, which illustrates the gain in computational efficiency in comparison with the global method. Finally, the precision of the regional (resp. the global) method is of 77% (resp. 81%). We must remember that here we are only taking into account morphological information, which has an impact on precision for a regional approach, but not for a global one, which always provides all possible repair alternatives. So, a precision measure represents a disadvantage for our proposal since we base efficiency on limitation of the search space. The future integration of linguistic information from both syntactic and semantic viewpoints should significantly reduce this gap in precision, which is less than 4%, or may even eliminate it.

7 Conclusion

As an extension of a recognition process, error repair is strongly influenced by the underlying grammatical structure, which should be taken into account in order to design efficient handling strategies. In this sense, spelling correction in NLP on FAs applies on states distributed in such a way that the number of path alternatives usually becomes exponential with the length of the input string.

These considerations are of importance in practical systems because they impact both the performance and the implementation techniques. So, most proposals exploit the apparently structural simplicity at word level to apply global techniques that examine the entire word and make a minimum of changes to repair all possible errors, which can be extremely time-consuming on a FA.

Our proposal drastically reduces this impact by dynamically graduating the size of the error repair zone. We describe a least-cost error repair method able to recover and resume the recognition at the point of each error, to avoid the possibility of non-detection of any subsequent errors. This translates into an improved performance without loss of quality in relation to global strategies.

References

1. J.P. Chanod and P. Tapanainen. Creating a tagset, lexicon and guesser for a French tagger. In *ACL SIGDAT Workshop on From Texts to Tags: Issues in Multilingual Language Analysis*, pages 58–64, University College, Dublin, Ireland, 1995.
2. J. Daciuk, S. Mihov, B.W. Watson, and R.E. Watson. Incremental construction of minimal acyclic finite-state automata. *Computational Linguistics*, 26(1):3–16, 2000.
3. J. Graña, F.M. Barcala, and M.A. Alonso. Compilation methods of minimal acyclic automata for large dictionaries. *Lecture Notes in Computer Science*, 2494:135–148, 2002.
4. K. Kukich. Techniques for automatically correcting words in text. *ACM Computing Surveys*, 24(4):377–439, 1988.
5. V.I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–848, 1965.
6. C.L. Lucchesi and T. Kowaltowski. Applications of finite automata representing large vocabularies. *Software-Practice and Experience*, 23(1):15–30, January 1993.
7. K. Oflazer. Error-tolerant finite-state recognition with applications to morphological analysis and spelling correction. *Computational Linguistics*, 22(1):73–89, 1996.
8. A. Savary. Typographical nearest-neighbor search in a finite-state lexicon and its application to spelling correction. *Lecture Notes in Computer Science*, 2494:251–260, 2001.

9. K. Sikkel. *Parsing Schemata.* PhD thesis, Univ. of Twente, The Netherlands, 1993.
10. A. Solak and K. Oflazer. Design and implementation of a spelling checker for Turkish. *Literary and Linguistic Computing*, 8(3), 1993.
11. Richard Sproat. *Morphology and Computation.* The MIT Press, Cambridge, Massachusetts, 1992.