Automatic Spelling Correction in Galician

M. Vilares\textsuperscript{1}, J. Otero\textsuperscript{1}, F.M. Barcala\textsuperscript{2}, and E. Domínguez\textsuperscript{2}

\textsuperscript{1} Department of Computer Science, University of Vigo
Campus As Lagoas s/n, 32004 Ourense, Spain
\{vilares,jop\}@uvigo.es
\textsuperscript{2} Ramón Piñeiro Research Center for Humanities
Estrada Santiago-Noia, Km. 3, A Barcia, 15896 Santiago de Compostela, Spain
\{fbarcala,edomin\}@cirp.es

Abstract. We describe a proposal on spelling correction intended to be applied on Galician, a Romance language. Our aim is to put into evidence the flexibility of a novelty technique that provides a quality equivalent to global strategies, but with a significantly minor computational cost. To do it, we take advantage of the grammatical background present in the recognizer, which allows us to dynamically gather information to the right and to the left of the point at which the recognition halts in a word, as long as this information could be considered as relevant for the repair process. The experimental tests prove the validity of our approach in relation to previous ones, focusing on both performance and costs.

1 Introduction

Galician belongs to the group of Latin languages, with influence of peoples living here before the Roman colonization, as well as contributions from other languages subsequent to the breaking-up of this empire. Long time relegated to informal usage, it has managed to survive well into the 20\textsuperscript{th} century until it was once again granted the status of official language for Galicia, together with Spanish. Although there several dialects exist, it has been recently standardized and, as a consequence, there is a pressing need for tools in order to permit a correct linguistic treatment. A main point of interest is the development of efficient error repair tools, in particular for spelling correction purposes.

In this context, the state of the art focuses on global techniques based on the consideration of error thresholds to reduce the number of repair alternatives, a technique often dependent on the recognizer. So, Oflazer \cite{oflazer} introduces a \textit{cut-off distance} that can be performed efficiently by maintaining a matrix \cite{vicedo} which help the system to determine when a partial repair will not yield any result by providing non-decreasing repair paths. In order to save this maintaining, Savary \cite{savary} embeds the distance in the repair algorithm, although this allows to partial corrections may be reached several times with different intermediate

\textsuperscript{*} 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.
distances; which is not time-efficient for error threshold values bigger than one. Anyway, these pruning techniques are strongly conditioned by the estimation of the repair region and their effectiveness is relative in global approaches.

In contrast to global algorithms, that expend equal effort on all parts of the word, also on those containing no errors; we introduce regional repairs avoiding to examine the entire word. This is of importance since Galician is an inflectional language with a great variety of morphological processes, and a non-global strategy could drastically reduce the costs. In effect, work underway focusing on word processing, the descriptive model is a regular grammar (rg) and the operational one is a finite automaton (fa). At this point, repairs on rg’s are explored breadth-wise; whilst the number of states in the associated finite automaton (fa) is massive. So, a complex morphology impacts both time and space bounds, that can even become exponential; which justifies our approach.

2 The Error Repair Model

Our aim is to parse a word \( w_{1..n} = w_1 \ldots w_n \) according to a rg \( G = (N, \Sigma, P, S) \). 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 \( G \) a **numbered minimal acyclic finite automaton** for the language \( L(G) \). In practice, we choose a device \([4]\) generated by **Galena** \([3]\). A fa is a 5-tuple \( A = (Q, \Sigma, \delta, q_0, Q_f) \) where: \( Q \) is the set of states, \( \Sigma \) the set of input symbols, \( \delta \) is a function of \( Q \times \Sigma \) into \( 2^Q \) defining the transitions of the automaton, \( q_0 \) the initial state and \( Q_f \) the set of final states. We denote \( \delta(q,a) \) by \( q.a \), and we say that \( A \) is **deterministic** iff \( |q.a| \leq 1, \forall q \in Q, a \in \Sigma \). The notation is transitive, \( q.w_{1..n} \) denotes the state \( (n \ldots (q.w_1)^{n-1}).w_n \). As a consequence, \( w \) is **accepted** iff \( q_0.w \in Q_f \), that is, the **language accepted by** \( A \) is defined as \( L(A) = \{ w, \text{ such that } q_0.w \in Q_f \} \). A fa is **acyclic** when the underlying graph it is. We define a **path in the fa** as 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 apply a minimization process \([1]\). Two fa’s 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 \( Q \) is equivalent.

2.1 The Dynamic Programming Frame

Although the standard recognition process is deterministic, the repair one 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 \( I \) of **items** \( I = \{ [q,i], q \in Q, i \in [1, n+1] \} \), where \( [q,i] \) looks for the suffix \( w_{i..n} \) to be analyzed from \( q \in Q \).

We describe our proposal using **parsing schemata** \([7]\), a triple \( \langle I, \mathcal{H}, \mathcal{D} \rangle \), with \( \mathcal{H} = \{ [a,i], a = w_i \} \) an initial set of items called **hypothesis** that encodes the