Learning Conditional Transducers for Estimating the Distribution of String Edit Costs

Marc Bernard MARC.BERNARD@UNIV-ST-ETIENNE.FR
Jean-Christophe Janodet JANODET@UNIV-ST-ETIENNE.FR
Marc Sebban MARC.SEBBAN@UNIV-ST-ETIENNE.FR
Laboratoire Hubert Curien, 18 r Pr Benoit Lauras, 42000 St-Etienne, France

Abstract

We focus on the Edit Distance and propose an algorithm to learn the costs of the primitive edit operations. The underlying model is a probabilitic transducer computed by using grammatical inference techniques, that is neither deterministic nor stochastic in the standard terminology. Moreover, this transducer is conditional, thus independent from the distributions of the input strings.

Real world applications such as spell checking, speech recognition, DNA analysis or plagiarism detection often use the Edit Distance (ED) (Wagner & Fischer, 1974), to compute similarities of string pairs. The ED is historically defined as the smallest number of insertions, deletions and substitutions required to change one string into another.

The common feature of the majority of ED-based methods is that they are static, in the sense of using *a priori* fixed costs for the primitive edit operations, that leaves little room for adaptation to the string context. Nevertheless, in many real domains, the level of an edit cost should be able to depend not only on the pair of symbols handled but also on the context where the operation occurs. For instance, in computational biology and especially in regular expression analysis, a given edit operation involving the same two symbols can highly depend on its location in the DNA sequence. In handwriting, it is experimentally known that the probability to unintentionally delete a character of a word is higher for symbols after the first one. Thus, an estimate of similarity between two strings can vary a lot depending on the specific domain under consideration.

This work was supported in part by the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

One solution would consist in manually assigning costs to edit operations that reflect the likelihood of the corresponding transformations. But the setting up of this strategy is difficult and seems to be not realistic overall for applications with a low level of expertise. While the main improvements about the ED have above all dealt, so far, with its algorithmic complexity, some recent work tried to overcome the previously mentioned drawbacks by automatically learning the primitive edit costs, rather than hand-tuning them for each domain. Several probabilistic models have been proposed to learn a *stochastic* ED in the form of stochastic transducers (Ristad & Yianilos, 1998; Bilenko & Mooney, 2003; Oncina & Sebban, 2006), conditional random fields (CRF) (McCallum et al., 2005), or pair-Hidden Markov Models (pair-HMM) (Durbin et al., 1998). These models provide a probability distribution over the edit operations and thus over the string pairs. The stochastic ED between two sequences can then be computed from the negative logarithm of the probability of the string pair.

Although these methods have provided some significant improvements on pattern recognition tasks in comparison with the classic non-learned ED, they share at least one of the following two drawbacks (sometimes both). The first one is a *statistical bias* of the inferred model. Actually, the majority of these approaches aim at learning a *generative* model rather than a *discriminative* classifier (Bouchard & Triggs, 2004). In other words, they learn a joint probability distribution $p(x, y)$ over the pairs of strings (x, y), and thus, the resulting conditional density $p(y|x)$, required in classification tasks, is a biased classifier depending on the input distribution $p(x)$. A solution, as proposed in (McCallum et al., 2005; Oncina & Sebban, 2006), consists in directly learning a conditional distribution, called a *discriminative* classifier. The second drawback is a *limitation on the expressive power* of the model. Actually, the structure of the learned model (*i.e.*, the

number of states in the transducer, in the CRF or in the pair-HMM) is always *a priori* fixed in the proposed approaches. The goal is to learn the parameters (the edit costs) assuming that the fixed structure is able to capture the most important configurations which can arise from the alignment of two sequences. Since determining such a structure depends on the domain, this often constitutes a tricky task that can result in a bad adaptation of the model to the string context.

In this work, we propose to take into account both these problems, by learning not only the structure but also the parameters of a so-called *conditional edit transducer*. The motivations that justify the learning of such a transducer are the following. First, we think that an efficient way to model a stochastic ED actually consists in viewing it as a stochastic transduction between the input X and output Y alphabets (Oncina & Sebban, 2006; Ristad & Yianilos, 1998). In other words, it means that the relation constituted by a set of (*input,output*) strings can be compiled in the form of a 2-tape automaton, called a *stochastic finite-state transducer*. The interpretation of the string edit distance as a stochastic transduction naturally leads to two possible string distances (Ristad & Yianilos, 1998): the first one describes the most likely transduction between the two strings, while the second is defined by aggregating all transductions between them. In this paper, we focus on the first stochastic distance, a so-called *Viterbi Edit Distance* (Ristad & Yianilos, 1998). We motivate this choice by the fact that we can use an adaptation of the well known Viterbi algorithm for learning the structure **and** the parameters of the *conditional edit transducer*.

Actually, stochastic transducers suffer from the lack of training algorithms (Eisner, 2002) which generally only learn the parameters of an imposed structure, using the Expectation Maximization algorithm (EM) (Dempster et al., 1977). However, we claim that this drawback can be efficiently overcome using grammatical inference algorithms, that constitutes the second motivation of our work. Basically, a transduction between two strings $x \in X^*$ and $y \in Y^*$, in the specific domain of the ED, can be rewritten using an adapted Viterbi algorithm in the form of an optimal sequence of edit operations $z = z_1...z_n, z_i \in (X \cup \{\lambda\}) \times (Y \cup \{\lambda\}) \setminus \{(\lambda, \lambda)\}$ (where λ is the empty string). Thus, we can exploit grammatical inference algorithms for learning over this new alphabet the structure of the model (and its parameters) in the form of a probabilistic finite state automaton, by using the well-known ALERGIA algorithm (Carrasco & Oncina, 1994). In order to learn a discriminative model, the automaton must be corrected at each step to satisfy constraints of conditional distribution. The conditional edit transducer is then deduced from the automaton by splitting each transition according to the input and output alphabets.

References

Bilenko, M., & Mooney, R. (2003). Adaptive duplicate detection using learnable string similarity measures. *Proc. of the 9th Int. Conf. on Knowledge Discovery and Data Mining (KDD'03)* (pp. 39–48).

Bouchard, G., & Triggs, B. (2004). The trade-off between generative and discriminative classifiers. *Proc. in Computational Statistics (COMPSTAT'04), 16th Symp. of IASC*. Prague: Physica-Verlag.

Carrasco, R. C., & Oncina, J. (1994). Learning stochastic regular grammars by means of a state merging method. *Proc. of 1st Int. Colloquium in Grammatical Inference (ICGI'94)* (pp. 139–150). LNAI 862.

Dempster, A., Laird, M., & Rubin, D. (1977). Maximun likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, B*, 1–38.

Durbin, R., Eddy, S., Krogh, A., & Mitchison, G. (1998). *Biological sequence analysis*. Cambridge University Press.

Eisner, J. (2002). Parameter estimation for probabilistic finite-state transducers. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 1–8). Philadelphia.

McCallum, A., Bellare, K., & Pereira, P. (2005). A conditional random field for discriminatively-trained finite-state string edit distance. *Proc. 21th Annual Conference on Uncertainty in Artificial Intelligence (UAI'05)* (pp. 388–400). Arlington, Virginia: AUAI Press.

Oncina, J., & Sebban, M. (2006). Learning stochastic edit distance: application in handwritten character recognition. *Journal of Pattern Recognition, to appear*.

Ristad, E. S., & Yianilos, P. N. (1998). Learning string-edit distance. *IEEE Trans. on Pattern Analysis and Machine Intelligence, 20*, 522–532.

Wagner, R. A., & Fischer, M. J. (1974). The string-to-string correction problem. *Journal of the ACM, 21*, 168–173.