ME-CSSR: an Extension of CSSR using Maximum Entropy Models

Muntsa Padró and Lluís Padró

TALP Research Center
Universitat Politècnica de Catalunya
{mpadro,padro}@lsi.upc.edu

July 2007

Abstract

In this work an extension of CSSR algorithm using Maximum Entropy Models is introduced. Preliminary experiments to perform Named Entity Recognition with this new system are presented.

1 Introduction

The Causal State Splitting Reconstruction (CSSR) algorithm (Shalizi and Shalizi 2004) infers the causal states of a process from data, building a deterministic automaton that is expected to capture the patterns of data. These data are sequences of symbols drawn from a discrete alphabet Σ. Consider, for example $\Sigma = \{M, m\}$ to represent capitalized words (M) and not capitalized words (m). A history x is defined as a suffix formed by alphabet symbols (i.e. Mmm, $MMmM$, etc). CSSR studies each possible history (up to a preestablished maximum length l_{max}), comparing them in terms of their future probability distributions $P(Z|x)$, where Z is a random variable taking any value in Σ. Two histories, x and y, are equivalent when $P(Z|x) = P(Z|y)$, i.e. when they have the same probability distribution for the future. The different future distributions build the equivalence classes, named *causal states*. CSSR iteratively builds these causal states. The algorithm performs the comparison between probability distributions performing a hypothesis test.

CSSR has been applied to different research areas. For example, it has been used to learn the patterns of physical systems in crystallography (Varn and Crutchfield 2004) and to anomaly detection in dynamical systems (Ray 2004). These systems use CSSR to capture patterns representing data that can be then used for different purposes.

This algorithm has been also used in the field of Natural Language Processing (NLP) to learn automata that can be afterwards used to tag new data in tasks such as Named Entity Recognition (NER) and Chunking (Padró and Padró 2005). The results obtained in those experiments show that this technique can provide state-of-the-art results in some NLP tasks. Given these results, the challenge is to improve them, developing systems rivalling best state-of-the-art systems. In this work, we propose an approach to combine CSSR with Maximum Entropy (ME) models in order to introduce more information into the system and study if the performance improves. For these preliminary experiments we focus on NER task.

To apply CSSR to NER and to other NLP tasks, it is necessary to encode each word as a symbol of the alphabet Σ. This symbol has to take into account the relevant features for the task as well as the hidden information about whether the word belongs to a named entity (NE)[1]. For example, if the only features taken into account are if the word is capitalized or not (M or m), the alphabet will be the combination of each feature with the corresponding "B-I-O" tag: $\Sigma = \{M_B, M_I, M_O, m_B, m_I, m_O\}$

This approach is rather limited, since all information we want to take into account has to be encoded in the alphabet. Furthermore, the amount of necessary data to build a correct automaton grows exponentially with the alphabet size. For that reason, a method to introduce more information about the words independently of the alphabet has been devised.

2 Introducing ME models into CSSR

The main idea of the proposed approach is based on generalizing the concept of history. Instead of considering histories as sequences of alphabet symbols corresponding to the last l_{max} words, we define histories as sets of relevant information about the last l_{max} words. Thus, histories can be encoded as collections of features of the words in a window of size l_{max}. In this way, causal states can still be defined as sets of histories with the same distribution for the future and can be calculated following the structure of CSSR.

This work uses ME models (Berger, Pietra, and Pietra 1996) to compute the probability distribution of the future. The classes of ME models are the alphabet symbols used with CSSR, and they define the possible transitions of each state in the automaton. The relevant information associated to each word is encoded as different features, and ME models are used to compute the probability distribution of the next symbol given the active features. If with CSSR the probability to be computed had the form of $P(m_B|M_B M_I m_O)$ now the probability will be computed as $P(m_B|h)$ where h is a history including relevant features of last words.

We present three different approaches to use this extended concept of history and ME models in combination with CSSR:

1. **Plain ME:** Using the learned ME models, compute the probability of each word in test corpus of having the tag B, I or O (taking into account that there is a known part of the symbol, i.e. we know if it is M or m), and compute the best sequence of tags using the Viterbi algorithm. Note that this first approach doesn't use CSSR in any way, but it can be used as a baseline of ME models performance.
2. **ME-over-CSSR:** Use CSSR to learn an automaton as in (Padró and Padró 2005), using a simple alphabet. The ME model is used only during the tagging task, and its predicted probabilities are combined with the transition probabilities learned by the automaton. This is a simple way to introduce more complicated features without changing CSSR algorithm.
3. **ME-CSSR:** An extended version of CSSR algorithm that defines histories as sets of features instead of simple symbol suffixes. In this way all the

[1] This information is encoded using "B-I-O" approach (Ramshaw and Marcus 1995): B for words at NE beginning, I for words internal to a NE, and O for words outside a NE

information encoded in the features is taken into account when building the automaton and the automaton is expected to better capture the patterns of sequences since it has more information.

2.1 Experiments and Results

Different experiments with these three different methods were performed. These are preliminary experiments as the system is still under development.

The used alphabet and data are the same used in (Padró and Padró 2005). The alphabet has 5 symbols combining different orthographical and syntactic information, which combined with the B-I-O tags lead to a 15 symbol alphabet. The data are those of CoNLL-2000 shared task (Tjong Kim Sang and Buchholz 2000).

The experiments presented in this work were performed with two different feature sets. These sets include few and simple features, and will be extended in further work. First feature set (FS_1) takes into account just the alphabet symbol and the PoS tag. The second one (FS_2) includes the same features than FS_1 plus 4 more boolean features: capitalized word, word containing numbers, all letters capitalized, and auxiliary word (words that often appear inside NEs). Note that the feature corresponding to the alphabet symbol includes the hidden B-I-O information which is not available in the test corpus. When performing tagging step this feature is set to the symbol assumed by the Viterbi algorithm in the currently analyzed path. All these features are taken into account for each word in a window of size l_{max} to the left of the current word. To maintain the idea of histories it is necessary to consider the same maximum length for all features which will be the length used by CSSR to learn the automaton. Both feature sets also include the known part of the symbol (i.e. m or M) and the PoS tag of the current word.

The implication of taking into account different lengths for different features, of introducing features of future words, and how to combine it with CSSR algorithm, will be studied in the future.

The experiments were conducted with both feature sets and with l_{max} from 2 to 4. Table 1 shows the best F_1 scores obtained. The results with $l_{max} = 4$ are not presented as they are far behind the other results, since the available training data is insufficient to learn reliable automata with this history length.

System	FS_1		FS_2	
	l=02	**l=03**	**l=02**	**l=03**
Plain ME	87.00	86.37	86.56	86.28
ME-over-CSSR	**88.51**	86.63	88.26	86.61
ME-CSSR	85.89	85.61	85.97	85.18

Table 1: Obtained F_1 results with different feature sets and different approaches

From these results it can be seen that the simple combination ME-over-CSSR leads to better results than using plain ME models with the Viterbi algorithm, and that the proposed ME-CSSR method leads to worse results. The best result of using only CSSR reported in (Padró and Padró 2005) is $F_1 = 88.96\%$ which is not significantly different (at 95% confidence degree) from the best result presented here. Also the figures show that increasing l_{max} or the number of features leads to a lose in performance, which is surprising specially in the case of using plain

ME models. This can be due to the sparseness of data, or to using over-simplistic feature sets, and further research is required on this issue.

Another point requesting further study is the trade-off between the data-sparseness caused by the fact of viewing histories as feature sets. Since the richer feature set we use, the less occurrences we'll have of each particular history, the CSSR algorithm will have less evidence to accurately build the causal states. On the other hand, richer feature sets should produce better ME models, which can compensate this lack of evidence.

3 Conclusions and Further Work

An extension of CSSR using ME models has been presented. The best results obtained are similar to the ones obtained with CSSR without ME models, but the experiments are very preliminary and the used features very simple, so there is still room for improvement. We expect to attain better performance when introducing more complicated features into the system, as ME models estimate better the probability distributions when rich feature sets are taken into account.

While the ME-over-CSSR approach yields better results than using only plain ME models, the ME-CSSR proposal leads to worse results in the performed experiments. One reason for this can be that the hypothesis test to determine if two probability distributions are different is performed using χ^2 statistics, and this may not be adequate when dealing with histories containing many features, as the number of occurrences for each history will be low, and χ^2 test depends on the counts of seen events being a poor test if the counts are low. Additionally, since ME models provide conditional probability distributions, a test comparing distributions regardless of the counts behind would be much more appropriate.

In the future, experiments introducing more features into the combined systems will be performed, searching for better results of the approaches combining CSSR and ME models. Also, other hypothesis tests have to be checked to learn automata with ME-CSSR, as χ^2 seems not to be the most adequate.

References

Berger, A., S. D. Pietra, and V. D. Pietra (1996). A maximum entropy approach to natural language processing. *Computational Linguistics 22*, 39–71.

Padró, M. and L. Padró (2005). Applying a finite automata acquisition algorithm to named entity recognition. In *Proceedings of FSMNLP'05*, Helsinki.

Ramshaw, L. and M. P. Marcus (1995). Text chunking using transformation-based learning. In *Proceedings of the Third ACL Workshop on Very Large Corpora*.

Ray, A. (2004). Symbolic dynamic analysis of complex systems for anomaly detection. *Signal Process. 84*(7), 1115–1130.

Shalizi, C. R. and K. L. Shalizi (2004). Blind construction of optimal nonlinear recursive predictors for discrete sequences. In *Uncertainty in Artificial Intelligence: Proceedings of the Twentieth Conference*.

Tjong Kim Sang, E. F. and S. Buchholz (2000). Introduction to the conll-2000 shared task: Chunking. In *Proceedings of CoNLL-2000*. Lisbon, Portugal.

Varn, D. P. and J. P. Crutchfield (2004). From finite to infinite range order via annealing: The causal architecture of deformation faulting in annealed close-packed crystals. *Physics Letters A 324*, 299–307.