Comparison Of Local And Global Undirected Graphical Models*

Zhemin Zhu, Djoerd Hiemstra, Peter Apers, Andreas Wombacher

{z.zhu, d.hiemstra, p.m.g.apers, a.wombacher}@utwente.nl
Electrical Engineering, Mathematics and Computer Science (EEMCS)
University of Twente, Enschede, The Netherlands

Abstract. CRFs are discriminative undirected models which are globally normalized. Global normalization preserves CRFs from the label bias problem which most local models suffer from. Recently proposed co-occurrence rate networks (CRNs) are also discriminative undirected models. In contrast to CRFs, CRNs are locally normalized. It was established that CRNs are immune to the label bias problem even they are local models. In this paper, we further compare ECRNs (using fully empirical relative frequencies, not by support vector regression[1]) and CRFs. The connection between Co-occurrence Rate, which is the exponential function of pointwise mutual information, and Copulas is built in continuous case. Also they are further evaluated statistically by experiments.

1 Introduction

Many fundamental applications in natural language processing, computer vision and bioinformatics desire structured outputs rather than a single tag. Conditional random fields [1] are discriminative undirected graphical models which output structured tags conditioned by observations. CRFs avoid weaknesses of existing models, such as hidden Markov models and conditional Markov models [2]. Hidden Markov models assume unnecessary independence realtions between observations, and conditional Markov models suffer from the label bias problem. CRFs address the label bias problem by global normalization in which the global joint probability is normalized. But global normalization brings up the problem of inefficient training. [3] proposed co-occurrence rate networks (CRNs) which are also discriminative undirected models. In contrast to CRFs, CRNs can be locally normalized. It was established in [3] that CRNs also avoid the strong independence assumption of hidden Markov models and the label bias problem of conditional Markov models even they are locally normalized. Hence CRNs are promising models which can be trained much more efficiently while preserving high accuracy. In this paper, we further compare CRFs and ECRNs (using empirical relative frequencies) experimentally. Their differences are clarified. And we futher evaluate performance statistically by experiments. In the remainder of this section we briefly introduce CRFs and ECRNs. In this paper we focus on chain-structured graphs.

*We thank the three reviewers for their very helpful comments. This work has been supported by the Dutch national program COMMIT/.

[1] This is different from our another later paper, in which we use support vector regression.

1.1 CRFs

The factorization of CRFs are based on the Hammersley-Clifford theorem (HCT). Hammersley-Clifford theorem implies a joint distribution can be written as a product of non-negative functions (called potential functions) over cliques. Apply this result to chain-structured graphs, we obtain the following factorization:

$$p(S_1, S_2..., S_n \mid O) = \frac{1}{Z_O} \prod_{i=1}^{n} \phi_i(S_i, O) \prod_{j=1}^{n-1} \psi_j(S_j, S_{j+1}, O), \tag{1}$$

$$Z_O = \sum_{S_1, S_2, ..., S_n} [\prod_{i=1}^{n} \phi_i(S_i, O) \prod_{j=1}^{n-1} \psi_j(S_j, S_{j+1}, O)]. \tag{2}$$

$S = (S_1, S_2, ..., S_n)$ is a tag sequence and O is a observation sequence. We use $p(X)$ to denote probability mass function (pmf) of discrete X. Z_O is a global normalization to ensure p is a probability mass function. Note that there is no pmf constraints on the local factors ϕ and ψ. They are just non-negative functions. In graphical models, the factors ϕ and ψ are traditionally modeled by exponential functions. For example, $\phi_i(S_i, O) = \exp \sum_j \lambda_j [\![f_j, (S_i, O)]\!]$, denoted by $E(S_i, O)$. $[\![f_j, (S_i, O)]\!]$ is a indicator function which equals 1 if feature f_j occurs on (S_j, O); Otherwise, it equals 0. Suppose the training dataset consists of N pairs of tag and observation sequences: $D = \{(s^i, o^i) \mid i = 1, ..., N\}$. Then the likelihood function over D is:

$$\mathcal{L}_{CRF}(D) = \prod_{i=1}^{N} p(s^i \mid o^i) = \prod_{i=1}^{N} [\frac{1}{Z_o{}^i} \prod_{j=1}^{|s^i|} E(s^i_j, o^i) \prod_{k=1}^{|s^i|-1} E(s^i_k, s^i_{k+1}, o^i)]. \tag{3}$$

$|s^i|$ is the length of the tag sequence s^i, which must be equal to $|o^i|$. A maximal likelihood estimate (mle) can be obtained by maximizing the logarithm function of Eq. (3).

1.2 CRNs

The factorization of co-occurrence rate networks (CRNs) are based on the concept of co-occurrence rate (CR) and its two theorems [3]. [4] provides more details about the properties of co-occurrence rate.

Definition 1 (Co-occurrence Rate)**.**

$$\texttt{CR}(X_1; X_2; ...; X_n) = \frac{p(X_1, X_2, ..., X_n)}{p(X_1)p(X_2)...p(X_n)}. \tag{4}$$

Theorem 1 (Partition Operation)**.**

$$\texttt{CR}(X_1; ...; X_j; X_{j+1}; ...; X_n) = \texttt{CR}(X_1; ...; X_j)\texttt{CR}(X_{j+1}; ...; X_n)\texttt{CR}(X_1...X_j; X_{j+1}...X_n).$$

Theorem 2 (Reduce Operation). *If $X \perp Y \mid Z$, then $\texttt{CR}(X;YZ) = \texttt{CR}(X;Z)$.*

Using CR, a chain-structured graph can be factorized as follows [3]:

$$p(S_1, S_2..., S_n \mid O) = \prod_{i=1}^{n} p(S_i \mid O) \prod_{j=1}^{n-1} \texttt{CR}(S_j; S_{j+1} \mid O), \tag{5}$$

where $\texttt{CR}(S_j; S_{j+1}|O) = \frac{p(S_j, S_{j+1}|O)}{p(S_j|O)p(S_{j+1}|O)}$. $p(S_i|O)$ in Eq. (5) is a probability mass function. Hence it can be locally normalized. This is different from $\phi_i(S_i, O)$ in Eq. (1), for which there is not necessarily a probability mass function. Also there is no Z_O in Eq. (5). Hence the global normalization can be avoided.

The more interesting is the quantity $\texttt{CR}(S_j; S_{j+1} \mid O)$. By definition, CR is not a probability mass function. Because its value can be greater than 1. Hence we cannot do local normalization over $\texttt{CR}(S_j, S_{j+1} \mid O)$. Without the normalization, if we maximize the likelihood function, the factors will grow to $+\infty$. Fortunately, $\texttt{CR}(S_j, S_{j+1})$ has close relation to the concept Copula [5] in statistics. We can use a similar idea of estimating Copula to estimate CR. In continuous case, the connection can be built as follows. For convenience, we use (X, Y) instead of (S_j, S_{j+1}). Let F_X and F_Y be the cumulative distribution functions (cdf) of X and Y:

$$\begin{aligned} F_X &: X \to [0,1],\ x \mapsto P(X \leq x) \\ F_Y &: Y \to [0,1],\ y \mapsto P(Y \leq y). \end{aligned}$$

Then the bivariate Copula of (X, Y) is defined as the joint cdf of F_X and F_Y:

$$C_{F_X,F_Y}(u, v) = P(F_X \leq u, F_Y \leq v) \tag{6}$$

If Eq. (6) is differentiable with respect to F_X and F_Y, then we can obtain the Copula density function:

$$c_{F_X,F_Y} = \frac{\partial^2}{\partial F_X \partial F_Y} C_{F_X,F_Y}. \tag{7}$$

The following formula is a simple variable transformation:

$$\begin{aligned} f_{X,Y}(x, y) &= c_{F_X,F_Y}(F_X(x), F_Y(y)) \begin{vmatrix} \frac{\partial F_X}{\partial X} & \frac{\partial F_X}{\partial Y} \\ \frac{\partial F_Y}{\partial X} & \frac{\partial F_Y}{\partial Y} \end{vmatrix} (x, y) \\ &= c_{F_X,F_Y}(F_X(x), F_Y(y)) f_X(x) f_Y(y), \end{aligned}$$

where $\begin{vmatrix} \frac{\partial F_X}{\partial X} & \frac{\partial F_X}{\partial Y} \\ \frac{\partial F_Y}{\partial X} & \frac{\partial F_Y}{\partial Y} \end{vmatrix} = f_X f_Y$ is the Jacobian. Hence we obtain:

$$\texttt{CR}(X = x; Y = y) = \frac{f_{X,Y}(x, y)}{f_X(x) f_Y(y)} = c_{F_X,F_Y}(F_X(x), F_Y(y)). \tag{8}$$

Eq. (8) shows in continuous case $\mathtt{CR}$ is just the Copula density function. In discrete case, $\mathtt{CR}$ cannot be the Copula mass function. Because its value can be greater than 1. But the idea for estimating Copula can still be used for estimating discrete $\mathtt{CR}$. Estimating Copula implies usually that every marginals is estimated and plugged into an estimated joint distribution. Following this idea, [3] plugs the empirical $p(X)$ and $p(Y)$ into $p(X, Y)$ to obtain $\mathtt{CR}(X; Y)$, where $p(X)$, $p(Y)$ and $p(X, Y)$ are estimated using fully empirical frequencies.

Remark I CRNs are different from approximate training methods of CRFs, such as tree-reweighted belief propagation [6] and piecewise training [7, 8]. These methods calculate some approximation of the global partition function. Calculating the approximation can be much more efficient than calculating the exact value. CRNs do not calculate the global partition function approximately or exactly. The factors are trained independently under local constraints. As long as the local constraints are satisfied, the global constraints are automatically met. Hence CRNs do not use message passing process which can be used for calculating the global partition function. This distinguishes our method from these methods. We should emphasize that message-passing is not computationally demanding. By dynamic programming, its complexity is linear to the sequence size. Not using message passing cannot be considered as an advantage of CRNs. We just use it here for distinguishing our method.

Remark II It seems that we can simply estimate $\mathtt{CR}$ in Eq. (5) using maximum likelihood estimation similar to CRFs. As explained, this does not work. Because we can not find a local normalization for $\mathtt{CR}$. A normalization implies constraints. Without these constraints, maximizing the likelihood function leads $\mathtt{CR}$ growing to $+\infty$. Of course we can use the global normalization to constrain $\mathtt{CR}$. Then we go back to CRFs. The purpose of CRNs is to estimate factors independently.

2 Experiments

We adopt CRF++ version 0.57 [9] as the implementation of CRFs. We implement ECRNs in Java. We compare CRFs and ECRNs on two real-world datasets. The first one is a part-of-speech (POS) tagging dataset. In POS tagging, each word in a sentence is assigned a POS tag. The second dataset is for named entity recognition (NER). In NER, each word in a sentence is assigned with a NER tag which indicates if it is a organization, location or person. These two applications can be well modelled as sequence labeling tasks.

The Brown corpus are used for POS tagging. This corpus includes 34,623 sentences and 252 different POS tags. We use the same spelling features as those described by [1]. We use half of the dataset for training and the rest half for testing. We use the the Dutch part of CoNLL-2002 NER Corpus[2] for NER. There are three files in this corpus: ned.train (13,221) for training, ned.testa (2,305) for development and ned.testb (4,211) for testing. There are 9 different NER tags.

[2] http://www.cnts.ua.ac.be/conll2002/ner/

	POS	NER
CRF	1,064,384	794
ECRN	3.3	1.3

Table 1: Training Time In Seconds

Tab. (1) lists the training time token by CRF and ECRN. There is no doubt that ECRNs reduce training time radically.

2.1 Statistical Evaluation On Results

As described, the whole POS tagging dataset (34,623 sentences) is split into two half parts: training part (17,311 sentences) and test part (17,312 sentences). To obtain confidence intervals on the results, we further partition the test dataset into 43 parts, and each part includes 400 sentences. Tab. (2) shows the 95% t-intervals of the total, known and unknown accuracy.

%	Total	Known	Unknown
CRF	93.26 ± 0.30	95.25 ± 0.21	63.21 ± 2.05
ECRN	94.19 ± 0.26	96.36 ± 0.22	61.5 ± 1.94

Table 2: 95% t-interval Of Accuracy On POS Tagging

Total accuracy is the per-word accuracy including both known and unknown words. Known words are those appear in the training dataset. Unknown words are those have not been seen in the training dataset.

The test dataset of NER has 4,211 sentences. Similar to POS tagging experiment, we further partition the NER test dataset into 21 parts and each part has 200 sentences. Tab. (3) shows the 95% t-intervals of F1, precision and recall over these test parts.

%	F1	Precision	Recall
CRF	64.66 ± 3.61	71.31 ± 4.46	59.52 ± 3.41
ECRN	64.97 ± 2.60	80.15 ± 2.32	54.87 ± 2.96

Table 3: 95% t-interval On NER Dataset

F1 is a popular metric for evaluating NER results which is the harmonic mean of precision and recall. In other words, F1 measures the overall performance which considers both precision and recall. Precision is the number of correctly predicted named entities divided by the number of predicted named entities. And recall is the number of correctly predicted named entities divided by number of total named entities. From Tab. (3), we can see that CRF and ECRN have very close F1 scores. But on precision, ECRN performs significantly better than CRF. On recall, in contrast CRF is much better than ECRN.

3 Conclusions And Future Work

We compared the local models (i.e. ECRNs) and the global models (i.e. CRFs) theoretically and experimentally. We conclude that the local models can be trained much faster than global models and also obtain competitive results on overall performance (Total accuracy in POS tagging and F1 in NER). In this paper, we use the fully empirical relative frequencies as the estimation of marginals and CR's. In future, we will employ more sophisticated regression models such as lasso, support vector regression or kernel smoothing methods.

References

[1] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *ICML*, 2001, pp. 282–289.

[2] A. McCallum, D. Freitag, and F. C. N. Pereira, "Maximum entropy markov models for information extraction and segmentation," in *Proceedings of the Seventeenth International Conference on Machine Learning*, ser. ICML '00, 2000, pp. 591–598.

[3] Z. Zhu, D. Hiemstra, P. Apers, and A. Wombacher, "Empirical co-occurrence rate networks for sequence labeling," in *Proceedings of The 12th International Conference on Machine Learning and Applications (ICMLA'13)*. IEEE, 2013, p. to appear. [Online]. Available: https://sites.google.com/site/zhuzhemin/

[4] Z. Zhu, D. Hiemstra, P. M. G. Apers, and A. Wombacher, "Separate training for conditional random fields using co-occurrence rate factorization," http://eprints.eemcs.utwente.nl/22600/, Centre for Telematics and Information Technology, University of Twente, Enschede, Technical Report TR-CTIT-12-29, October 2012.

[5] A. Charpentier, J.-D. Fermanian, and O. Scaillet, *The Estimation of Copulas: Theory and Practice*, 2006.

[6] M. J. Wainwright, T. Jaakkola, and A. S. Willsky, "Tree-reweighted belief propagation and approximate ML estimation by pseudo-moment matching," in *9th Workshop on Artificial Intelligence and Statistics*, January 2003.

[7] C. A. Sutton and A. McCallum, "Piecewise training for undirected models," in *UAI 05*. AUAI Press, 2005, pp. 568–575.

[8] C. Sutton and A. McCallum, "Piecewise pseudolikelihood for efficient training of conditional random fields," ser. ICML '07. ACM, 2007, pp. 863–870.

[9] T. Kudo, "Crf++: Yet another crf toolkit," free software, March 2012. [Online]. Available: http://crfpp.googlecode.com/svn/trunk/doc/index.html