A Note on an Approximate Learning Algorithm with Limited Parameters for Language Models

Yasunari Maeda, Fumito Masui, Hideki Yoshida and Masakiyo Suzuki

Kitami Institute of Technology, 165 Koen-cho, Kitami-shi, Hokkaido 090-8507 Japan

Summary
A lot of research in the field of NLP(natural language processing) for AI(artificial intelligence) has the goal of learning language models. In general, the aim is to minimize the divergence between the approximate model and the true model, but most learning algorithms are based on the maximum likelihood method. The existence of finite samples with high likelihood doesn't mean that the divergence between the approximate model and the true model is small. This paper proposes a new learning algorithm, the measure of which is divergence. The proposed algorithm is compared to previous algorithms using simulations.

Key words:
language model, approximate learning algorithm, Markov chain, divergence

1. Introduction

Many researchers are now tackling language models in the field of NLP(natural language processing) for AI(artificial intelligence). Most papers used n-gram models as the language model. n-gram models can be represented as $(n-1)$ -th order Markov chains, so training an n-gram model can be regarded as training a Markov chain with unknown parameters.

Some researchers have tried training Markov chains with an extremely large number of parameters; since the number of parameters of the true model is unknown, overcoverage is common. This requires basically unlimited memory for the parameters and so is rather impractical. Another research approach, which assumes that the number of parameters of the true model is known, is to train the Markov chain with enough parameters. Another research approach, which also assumes that the number of parameters of the true model is known, is to first train the Markov chain with enough parameters, and then develop approximate models with limited parameters to allow their implementation on computers that have limited memory. This paper adopts this last approach but introduces a better learning method.

Though the purpose of the approximate learning algorithm with limited parameters is to minimize the divergence between the approximate model and the true model, Brown's algorithm[1] uses the maximum likelihood method. Its performance given finite samples is weak, since a high likelihood doesn't mean that the divergence is small.

This paper introduces an approximate learning algorithm in which divergence is used as a measure. I show that the algorithm can develop an approximate model that is closer to the true model than those yielded by Brown's algorithm.

I describe Markov chains in section 2 and previous research in section 3. The proposed algorithm is introduced in section 4. It is compared to Brown's algorithm in section 5. Section 6 concludes the paper.

2. Markov Chains

Almost all previous research studied n-gram models. These are language models that can be represented by $(n-1)$ -th order Markov chains as follows:

$$p\left(w_{i_t} \middle| w_{i_0} w_{i_1} \cdots w_{i_{t-1}}, \theta\right) = p\left(w_{i_t} \middle| w_{i_{t-n+1}}, w_{i_{t-n+2}}, \cdots w_{i_{t-1}}, \theta\right), \qquad (1)$$

where w_i is a word, $w_i \in W$, W is a finite word set, θ is a $|W|^{n-1}(|W|-1)$ -dimensional vector of real-valued parameters, $\theta \in \Theta$, Θ is the set of parameters. This paper also adopts the Markov chain approach.

3. Previous Research

Many previous papers addressed the training of $(n-1)$ -th order Markov chains. Prior work is described below.

3.1 Bayes coding

When the order of the true Markov model $(n-1)$ is known, the true parameter vector θ^*, $\theta^* \in \Theta$ is unknown and using $|W|^{n-1}(|W|-1)$ -dimensional vector of real-valued parameters, divergence is minimized, in terms of the Bayes criterion, by Bayes coding[2]. The divergence is as follows:

Manuscript received November 5, 2010
Manuscript revised November 20, 2010

$$D\left(p\left(\cdot\middle|\cdot,\theta^*\right)\middle\|p\left(\cdot\middle|\cdot,\theta\right)\right)= \sum_{w^{n-1}}\pi\left(w^{n-1},\theta^*\right)\sum_{w}p\left(w\middle|w^{n-1},\theta^*\right)\log\frac{p\left(w\middle|w^{n-1},\theta^*\right)}{p\left(w\middle|w^{n-1},\theta\right)}, \quad (2)$$

where $\pi\left(w^{n-1},\theta^*\right)$ is the stationary state probability of the true model. The approximate model in Bayes coding is shown as follows:

$$p_{Bayes}\left(w\middle|w^{n-1},\hat{\theta}\right)=\frac{h\left(w\middle|w^{n-1},x^M\right)+\beta\left(w\middle|w^{n-1}\right)}{\sum_{w_i}h\left(w_i\middle|w^{n-1},x^M\right)+\beta\left(w_i\middle|w^{n-1}\right)}, \quad (3)$$

where x^M is a string of length M for learning, $x\in W$, $h\left(w\middle|w^{n-1},x^M\right)$ is the number of times w appears next to w^{n-1} in x^M, and $\beta\left(w\middle|w^{n-1}\right)$ is a parameter of Dirichlet distribution for the prior distribution of θ.

3.2 Brown's algorithm

There is a strong need to reduce the dimension of the parameter vector if we are to implement an approximate model on a computer with limited memory. I now describe Brown's algorithm for the case of 2-gram models. At first, an approximate model with a $|W|(|W|-1)$ -dimensional parameter vector is learned based on the maximum likelihood method as follows:

$$p_{ml}\left(w_j\middle|w_i,\hat{\theta}\right)=\frac{h\left(w_j\middle|w_i,x^M\right)}{\sum_{w_j}h\left(w_j\middle|w_i,x^M\right)}. \quad (4)$$

Next, the dimension of the approximate model is reduced as follows:

$$\hat{p}_{Brown}\left(w_{i_t}\middle|w_{i_{t-n+1}}\cdots w_{i_{t-1}}\right)=p\left(w_{i_t}\middle|c_{i_t}\right)p\left(c_{i_t}\middle|c_{i_{t-n+1}}\cdots c_{i_{t-1}}\right), \quad (5)$$

where c_i is a subset of W, $c_i\cap c_j=\phi$, $i\neq j$, $\bigcup_i c_i=W$, $C=\left\{c_0,c_1,\cdots,c_{|C|-1}\right\}$, C is a partition of W, $|\cdot|$ is the cardinality of a set. The cardinality of C is reduced to $|C|-1$ by merging c_i and c_j. The initial C is W, $C=\left\{c_0,c_1,\cdots,c_{|W|-1}\right\}=\left\{w_0,w_1,\cdots,w_{|W|-1}\right\}$. The pair of $\left(c_i,c_j\right)$ to be merged is determined as follows:

$$\left(c_i,c_j\right)=\arg\min_{c_i,c_j}\left(I\left(C;C\right)-I\left(C'\left(c_i,c_j\right);C'\left(c_i,c_j\right)\right)\right), \quad (6)$$

where $C'\left(c_i,c_j\right)$ is the new partition of W created by merging c_i and c_j, $I\left(C;C\right)$ is the mutual information of C and C as follows:

$$I\left(C;C\right)=\sum_{c_k,c_l}p\left(c_kc_l\right)\log\frac{p\left(c_l\middle|c_k\right)}{p\left(c_l\right)}, \quad (7)$$

where each probability distribution in the right hand side of formula (7) is calculated by the maximum likelihood method by formula (4). When the merging step is repeated T times, the dimension of the parameter vector is reduced to $\left(|W|-T\right)\left(2|W|-T-2\right)$. Formula (6) is also based on the maximum likelihood method.

Unfortunately, the maximum likelihood method doesn't offer good performance if the sample number is finite. A high likelihood doesn't mean that the divergence between the approximate model and the true model is small.

4. Proposed Algorithm

This section proposes a new approximate learning algorithm for $(n-1)$ -th order Markov chains. This algorithm uses divergence as a measure of fitness. At first, an approximate model with $|W|^{n-1}\left(|W|-1\right)$ -dimensional parameter vector is calculated by formula (3), which minimizes the divergence between the approximate model and the true model using the Bayes criterion. Next, the dimension of the approximate model is reduced as follows:

$$\hat{p}_{Proposed}\left(w_{i_t}\middle|w_{i_{t-n+1}}\cdots w_{i_{t-1}}\right)=p\left(w_{i_t}\middle|s_{i_{t-1}}\right),\ w_{i_{t-n+1}}\cdots w_{i_{t-1}}\in s_{i_{t-1}}, \quad (8)$$

where s_i is a subset of W^{n-1}, $s_i\cap s_j=\phi$, $i\neq j$, $\bigcup_i s_i=W^{n-1}$, $S=\left\{s_0,s_1,\cdots,s_{|S|-1}\right\}$, S is a partition of W^{n-1}. The cardinality of S is reduced to $|S|-1$ by merging s_i and s_j. The initial S is W^{n-1},

$$S=\left\{s_0,s_1,\cdots,s_{|W|^{n-1}-1}\right\}=\left\{w_0\cdots w_0w_0,w_0\cdots w_0w_1,\cdots,w_{|W|-1}\cdots w_{|W|-1}w_{|W|-1}\right\}.$$

The pair of $\left(s_i,s_j\right)$ to be merged is determined as follows:

$$\begin{aligned}\left(s_i,s_j\right)&=\arg\min_{s_i,s_j}D\left(p\left(\cdot\middle|\cdot,S\right)\middle\|p\left(\cdot\middle|\cdot,S'\left(s_i,s_j\right)\right)\right)\\&=\pi\left(s_i,S\right)\sum_{w_k}p\left(w_k\middle|s_i,S\right)\log\frac{p\left(w_k\middle|s_i,S\right)}{p\left(w_k\middle|s_i\cup s_j,S'\left(s_i,s_j\right)\right)}\\&\quad+\pi\left(s_j,S\right)\sum_{w_k}p\left(w_k\middle|s_j,S\right)\log\frac{p\left(w_k\middle|s_j,S\right)}{p\left(w_k\middle|s_i\cup s_j,S'\left(s_i,s_j\right)\right)},\end{aligned} \quad (9)$$

where $S'\left(s_i,s_j\right)$ is the new partition of W^{n-1} created by merging s_i and s_j, $\pi\left(s_i,S\right)$ is the stationary state probability of the approximate model based on partition S, the initial $\pi\left(s_i,S\right)$ is calculated based on the model of formula (3), the initial $p\left(w_k\middle|s_i,S\right)$ is equal to $p_{Bayes}\left(w_k\middle|w^{n-1},\hat{\theta}\right)$, where $w^{n-1}=s_i$, calculated by formula

(3), $p\left(w_k \middle| s_i \cup s_j, S'\left(s_i, s_j\right)\right)$ is calculated as follows:

$$p\left(w_k \middle| s_i \cup s_j, S'\left(s_i, s_j\right)\right) = \frac{\pi\left(s_i, S\right)}{\pi\left(s_i, S\right)+\pi\left(s_j, S\right)} p\left(w_k \middle| s_i, S\right) + \frac{\pi\left(s_j, S\right)}{\pi\left(s_i, S\right)+\pi\left(s_j, S\right)} p\left(w_k \middle| s_j, S\right), \tag{10}$$

$p\left(w_k \middle| s_l, S'\left(s_i, s_j\right)\right) = p\left(w_k \middle| s_l, S\right)$, $l \neq i$, $l \neq j$,

$\pi\left(s_i \cup s_j, S'\left(s_i, s_j\right)\right)$ is calculated as follows:

$$\pi\left(s_i \cup s_j, S'\left(s_i, s_j\right)\right) = \pi\left(s_i, S\right) + \pi\left(s_j, S\right), \tag{11}$$

$\pi\left(s_l, S'\left(s_i, s_j\right)\right) = \pi\left(s_l, S\right)$, $l \neq i$, $l \neq j$.

The increase in divergence caused by merging is minimized by formula (9). When merging is repeated T times, the dimension of the parameter vector is reduced to $\left(|W|^{n-1} - T\right)\left(|W| - 1\right)$.

Divergence is used as the convergence measure in the proposed algorithm. The idea of this algorithm is based on a part of Nomura's research[3]. The next section uses simulations to show that the approximate model calculated by the proposed algorithm offers better convergence to the true model than the approximate model calculated by Brown's algorithm.

5. Simulations

In this section I show a result of simulating the 2-gram model shown in Fig.1(at the end of this paper). Let $|W|$ be equal to 10. The true model is as follows:

$$\left|p\left(w_j \middle| w_i, \theta^*\right)\right| = \begin{vmatrix} .24556 & .00002 & .00102 & .46333 & .25257 & .01985 & .01691 & .00065 & .00007 & .00002 \\ .78176 & .01112 & .00885 & .10790 & .00157 & .08864 & .00007 & .00001 & .00007 & .00001 \\ .28659 & .05758 & .00001 & .00040 & .64797 & .00003 & .00542 & .00000 & .00197 & .00004 \\ .02189 & .22285 & .59295 & .00114 & .00032 & .00000 & .00023 & .16062 & .00000 & .00000 \\ .02851 & .03988 & .04005 & .04520 & .11245 & .08491 & .02554 & .36051 & .22218 & .04077 \\ .00051 & .05621 & .00621 & .35798 & .27244 & .02884 & .16469 & .10585 & .00050 & .00677 \\ .30394 & .03310 & .03919 & .00011 & .00010 & .00012 & .04755 & .57360 & .00226 & .00003 \\ .02786 & .30086 & .00016 & .00110 & .39047 & .25020 & .00246 & .00068 & .02603 & .00018 \\ .00498 & .00181 & .69491 & .11962 & .00284 & .00400 & .01403 & .09930 & .05549 & .00302 \\ .01825 & .45628 & .51917 & .00038 & .00056 & .00004 & .00508 & .00007 & .00006 & .00011 \end{vmatrix}, \tag{12}$$

where the value of an element located on the i-th row, j-th column is $p\left(w_j \middle| w_i, \theta^*\right)$.

Divergence is calculated between the true model and each approximate model. Brown65 means an approximate model with a 65-dimensional parameter vector calculated by Brown's algorithm, Brown48 means an approximate model with a 48-dimensional parameter vector calculated by Brown's algorithm, and pro45 means an approximate model with a 45-dimensional parameter vector calculated by the proposed algorithm. Fig.1 shows the average results of 100 runs. A uniform distribution was used as the prior distribution of Bayes coding.

The approximate model yielded by the proposed algorithm diverges less from the true model than those of Brown's algorithm, though the dimension of pro45 is less than those of Brown65 and Brown48. Similar results were gained in other simulations.

6. Conclusion

There are many learning algorithms for language models with limited parameters. Brown's algorithm uses likelihood as a measure, but likelihood is not appropriate for obtaining approximate models that are close to the true model. Because the goal is to minimize the divergence between the approximate model and the true model, divergence should be used instead of likelihood.

This paper proposed a new algorithm that uses divergence as a measure of fitness. Simulations showed that the approximate models calculated by the algorithm are closer to the true models than those calculated by Brown's algorithm.

Since the algorithm is a kind of greedy algorithm, the approximate models yielded by the algorithm are not the best in terms of any one criterion. Further work will be to create some theoretical guarantee of the algorithm.

Learning language models can be regarded as the clustering of words. Therefore, the proposed algorithm may be useful as a clustering algorithm.

References

[1] Peter F. Brown, Vincent J. Della Pietra, "Class-Based n-gram Models of Natural Language", Computaitional Linguistics,Vol.18,No.4,pp.467-479, 1992.

[2] Matsushima, Hirasawa, "On universal codes for FSMX sources", The 18th Symposium on Information Theory and Its Applications (SITA95), pp.377-380, 1995.

[3] Nomura, Matsushima, Hirasawa, "On Bayes Coding in the case of limited memories", TECHNICAL REPORT OF IEICE,IT96-15, pp.31-36, 1996.

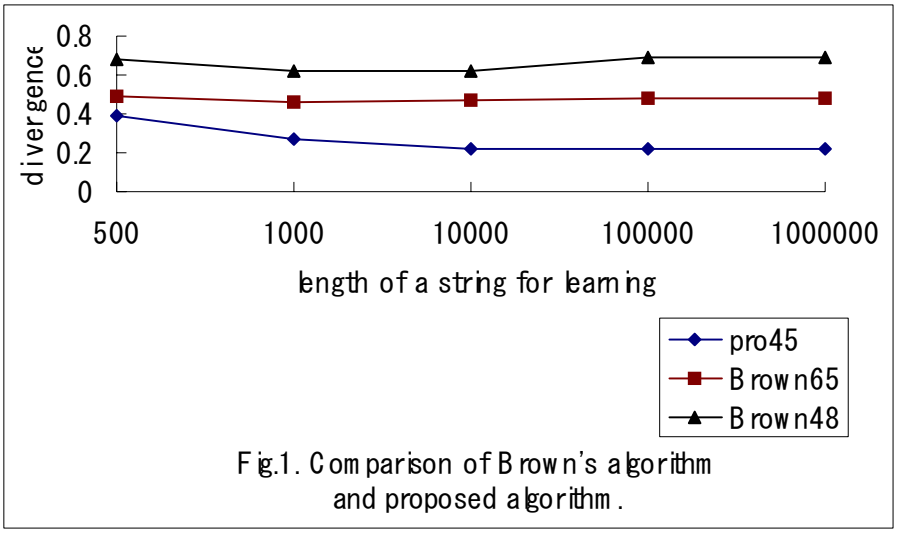

Fig.1. Comparison of Brown's algorithm and proposed algorithm.