Ergodic-HMMを用いた確率付きネットワーク文法の自動獲得

・確率付きネットワーク文法

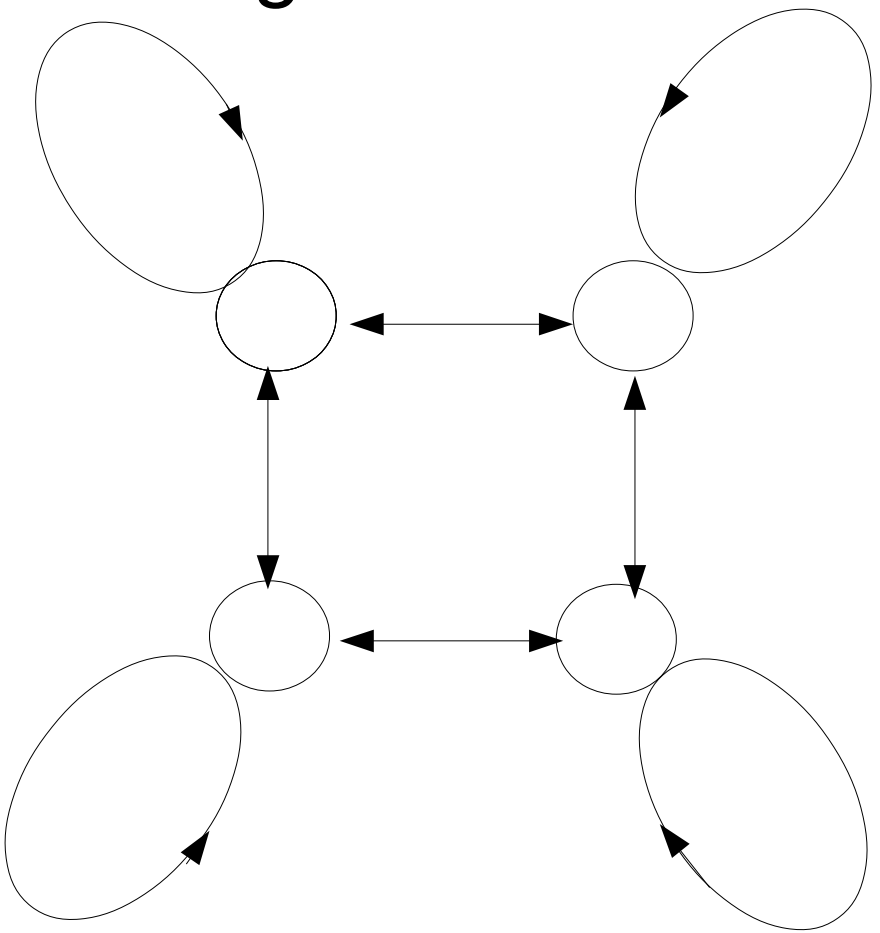
等値
（シンボル出力確率
＝単語の出力確率）

・Ergodic HMM

Ergodic HMM

言語データ

もしもし
はい そう です
えーっと ちょっと その
会議 の こと でね
はい どうぞ
えーっと それ を ちょっ
と クレジットカード を
持っ て い ない 者 が
いる ん です けれども



Baum-Welch　Algorithm

確率つきネットワーク文法

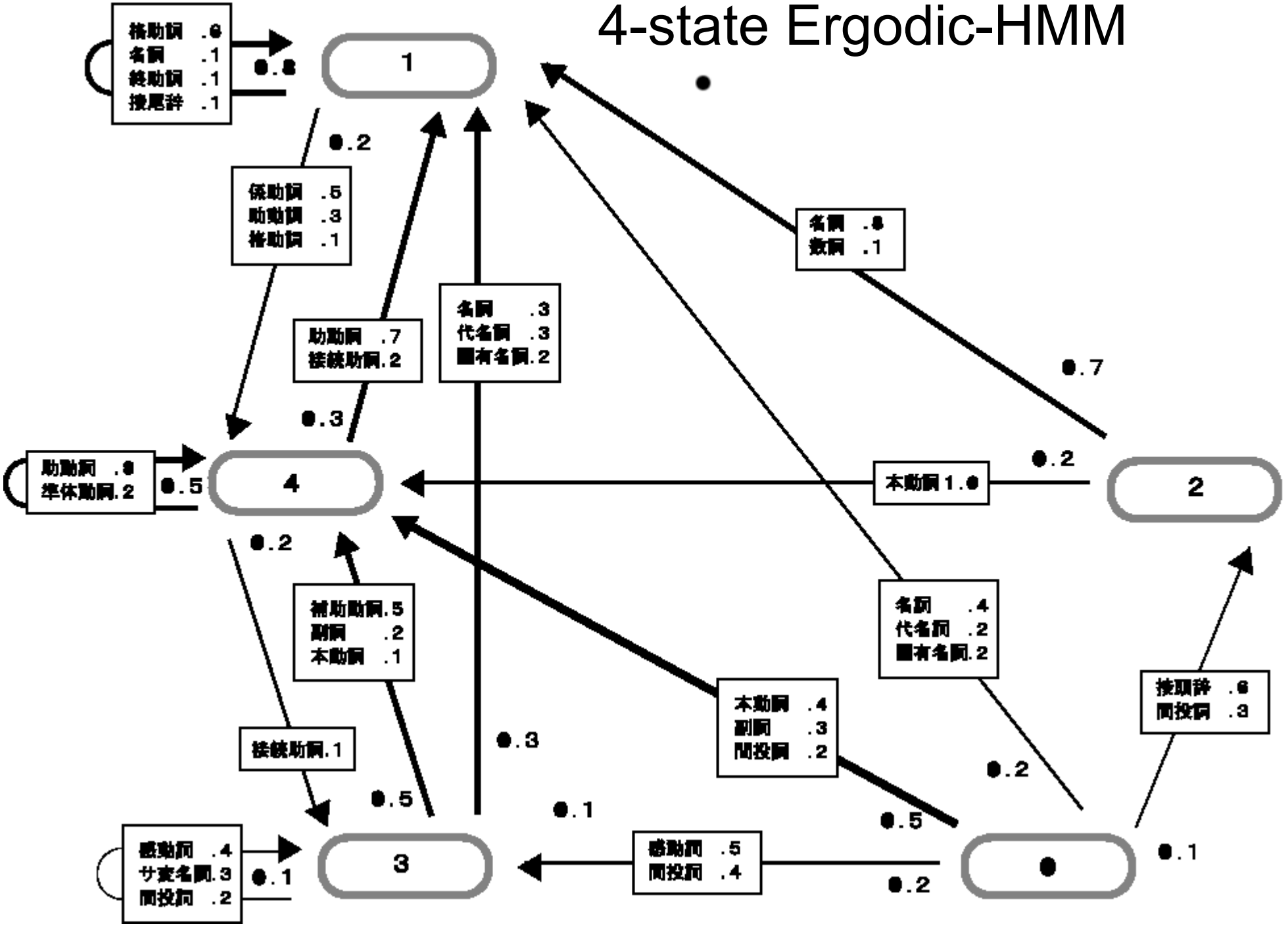
文の例

はい もしもし
えーっと そちら 第 1 回 の 通訳 電話 国際 会議 の 事務 局 で しょう か
はい そう です
えーっと ちょっと その 会議 の こと でね
はい どうぞ
えーっと 今 手元 に あの 登録 用紙 が ある ん です けれども
えーっと その 中 で ちょっと あの クレジットカード を ね
あのー クレジットカード の 名前 と なん か ナンバー を 書く ところ が ある ん です が
はい そう です
えーっと それ を ちょっと クレジットカード を 持っ て い ない 者 が いる ん です けれども
その 場合 は どう な ん でしょ う か

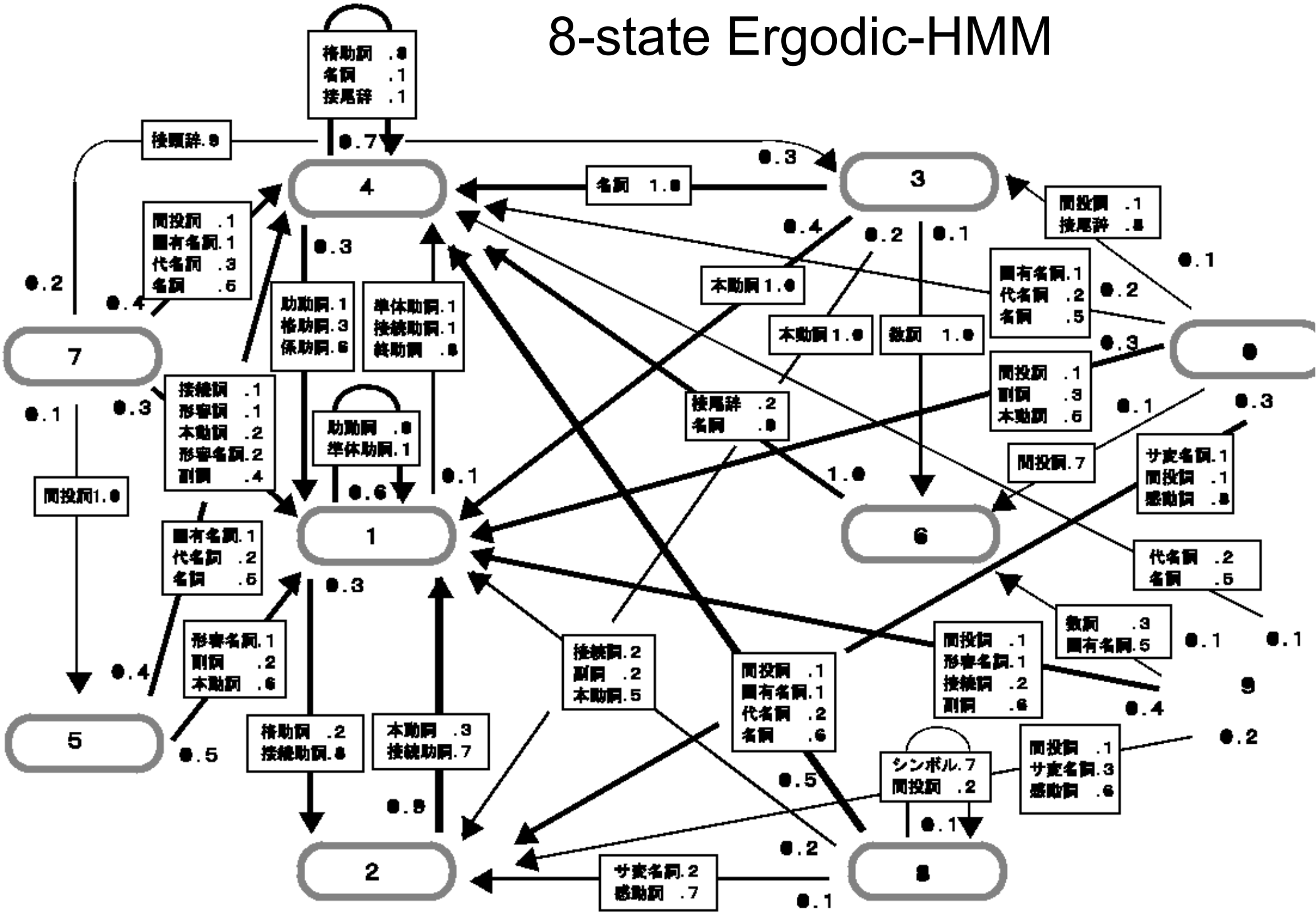
言語モデル生成実験の条件

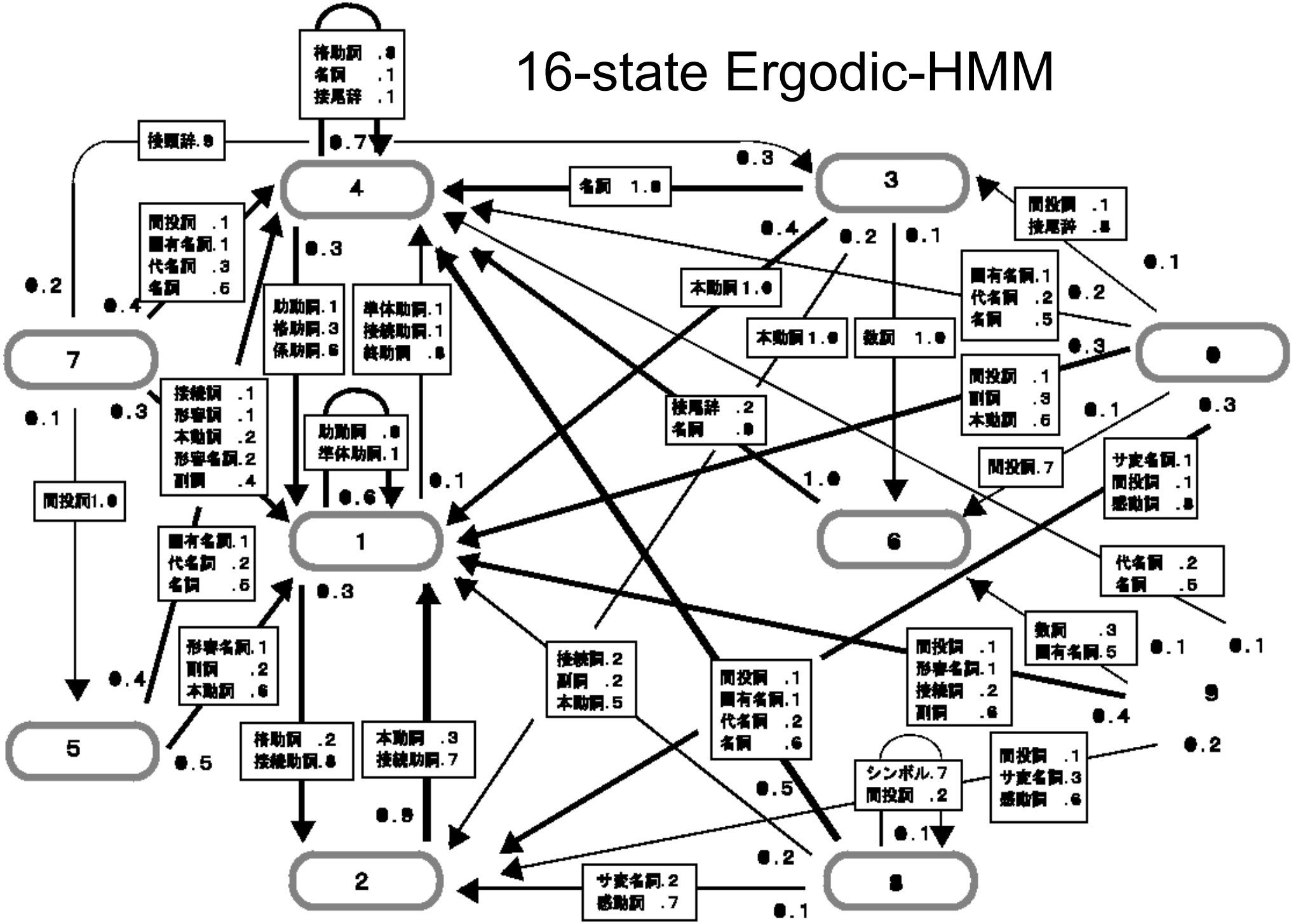
HMMの構造	状態遷移出力型Ergodic HMM
HMMの状態数	4状態,8状態,16状態
HMMの出力シンボル	単語
開始 終了状態	任意
初期状態遷移確率	ランダム
初期シンボル出力確率	ランダム
初期状態確率	均等
語彙数	6418
学習データ set	odd4000,odd2000,odd1000
学習データ数	4000文,2000文,1000文
単語総数	57354単語,20730単語,13299単語
学習終了条件	尤度上昇率1%未満

4-state Ergodic-HMM



8-state Ergodic-HMM




16-state Ergodic-HMM
格助詞 .8
名詞 .1
接尾辞 .1
接頭辞.9
.7
.3
4
3
名詞 1.0
間投詞 .1
固有名詞.1
代名詞 .3
名詞 .5
.2
.4
.3
助動詞.1
格助詞.3
係助詞.6
準体助動詞.1
接続助動詞.1
終助詞 .8
本動詞1.0
.4
.2
.1
.1
固有名詞.1
代名詞 .2
名詞 .5
.2
本動詞1.0
数詞 1.0
7
8
.3
.1
.3
接続詞 .1
形容詞 .1
本動詞 .2
形容名詞.2
副詞 .4
助動詞 .8
準体助動詞.1
接尾辞 .2
名詞 .8
間投詞 .1
副詞 .3
本動詞 .5
.1
.3
間投詞1.0
.6
.1
1.0
間投詞.7
サ変名詞.1
間投詞 .1
感動詞 .8
1
6
固有名詞.1
代名詞 .2
名詞 .5
.3
代名詞 .2
名詞 .5
形容名詞.1
副詞 .2
本動詞 .6
.4
接続詞.2
副詞 .2
本動詞.5
間投詞 .1
固有名詞.1
代名詞 .2
名詞 .6
数詞 .3
固有名詞.5
.1
.1
間投詞 .1
形容名詞.1
接続詞 .2
副詞 .6
9
.4
.2
5
.5
格助詞 .2
接続助詞.8
本動詞 .3
接続助詞.7
.5
シンボル.7
間投詞 .2
間投詞 .1
サ変名詞.3
感動詞 .6
.8
.1
.2
2
サ変名詞.2
感動詞 .7
8
.1

まとめ

- ４状態　　体言と用言のグループ化
- ８状態　　活用系でのグループ化
- １６状態　　品詞でのグループ化

・確率付きネットワーク文法の自動獲得が可能

メモリ量および計算量を削減した Baum-Welch アルゴリズムの提案と言語モデルへの適用

1: 目的

状態数が大きいErgodic HMMの
Baum-Welch学習

1.2: 問題点

状態数が小さい
　　→　Perplexityが高い
状態数が多い
　　→　Baum-Welch 学習が不可能

1.3: 解決方法

メモリ量および計算量を削減した
Baum-Welchアルゴリズム

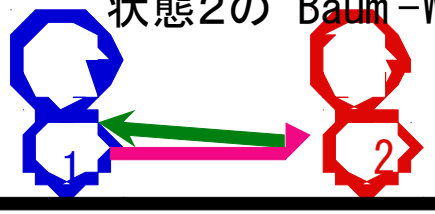
小さいシンボル出力確率の削除
（メモリ量，計算量の削減）

シンボル出力確率が閾値　　（10^{-300}）より小さいとき　0.0

再推定およびメモリから削除。

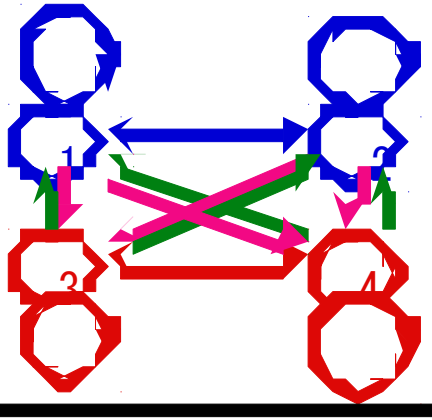
状態数の逐次増加

状態2の Baum-Welch　学習の終了後のパラメータ



$\pi_1^1=0.3$
$\pi_2^2=0.7$

$a_{11}^1=0.4$
$a_{12}^2=0.6$
$a_{21}^2=0.1$
$a_{22}^2=0.9$



状態4の初期モデルのパラメータ

$\pi_4^1=\pi_2^1=0.3$
$\pi_4^2=\pi_2^1=0.3$
$\pi_4^3=\pi_2^2=0.7$
$\pi_4^4=\pi_2^2=0.7$

$a_{11}^4=a_{11}^2=0.4$
$a_{12}^4=a_{11}^2=0.4$
$a_{13}^4=a_{12}^2=0.6$
$a_{14}^4=a_{12}^2=0.6$

：

ただしシンボル出力確率は乱数を使用

N状態の Ergodic HMM　のパラメータ

$\pi_N(i)$　; i=1,..,N　; 初期状態確率

$a_N(i,j)$　; i=1,..,N, j=1,..,N　; 状態遷移確率

$b_N(i,j,w)$　; i=1,..,N, j=1,..,N, w=1,..,V　; シンボル出力確率

V ; 語彙数

2N状態の Ergodic HMM の

初期状態確率および状態遷移確率の初期パラメータ

$\pi_{2N}(i) = 0.5 \times \pi_N(i/2)$　; i=1,..,2N

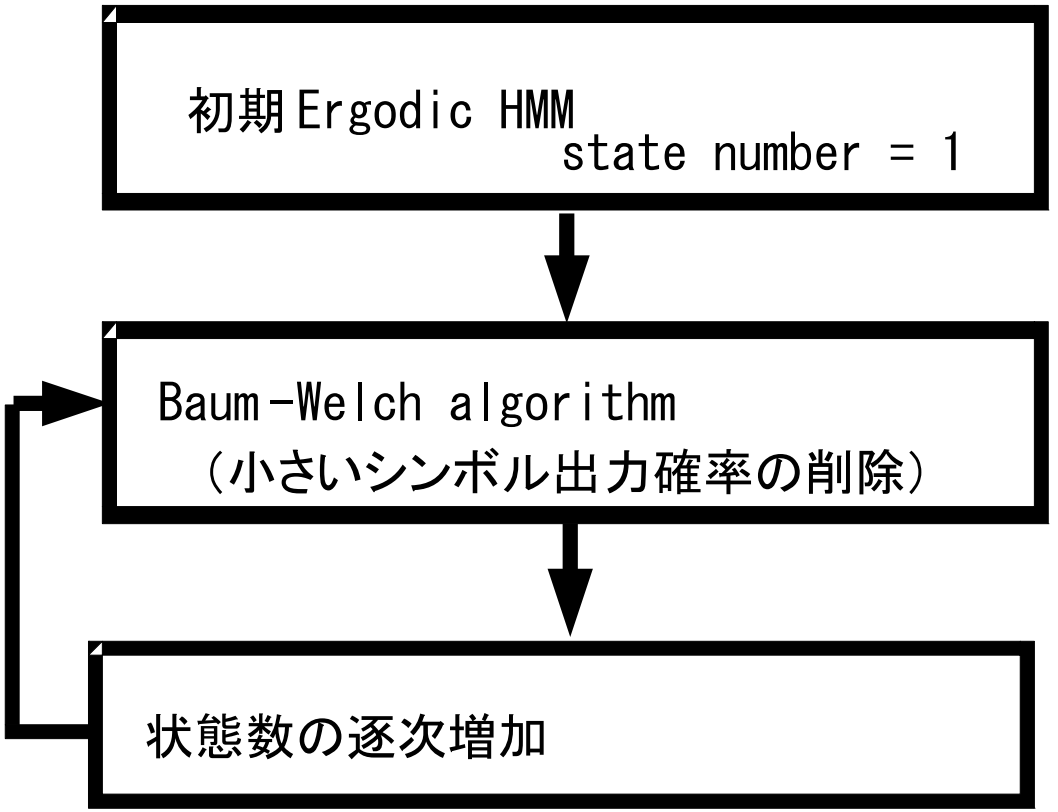
$a_{2N}(i,j) = 0.5 \times a_N(i/2, j/2)$　; i=1,..,2N, j=1,..,2N

$b_{2N}(i,j,w) = b_N(i/2, j/2, w) \times random(i,j,w)$;

i =1,..,2N, j=1,..,2N, w=1,..,V.　ただし $\sum_w b_{2N}(i,j,w)=1.0$

″ / ″　切り上げを意味

具体的なアルゴリズム



π, A , B ; random
state number = 1

state number = N

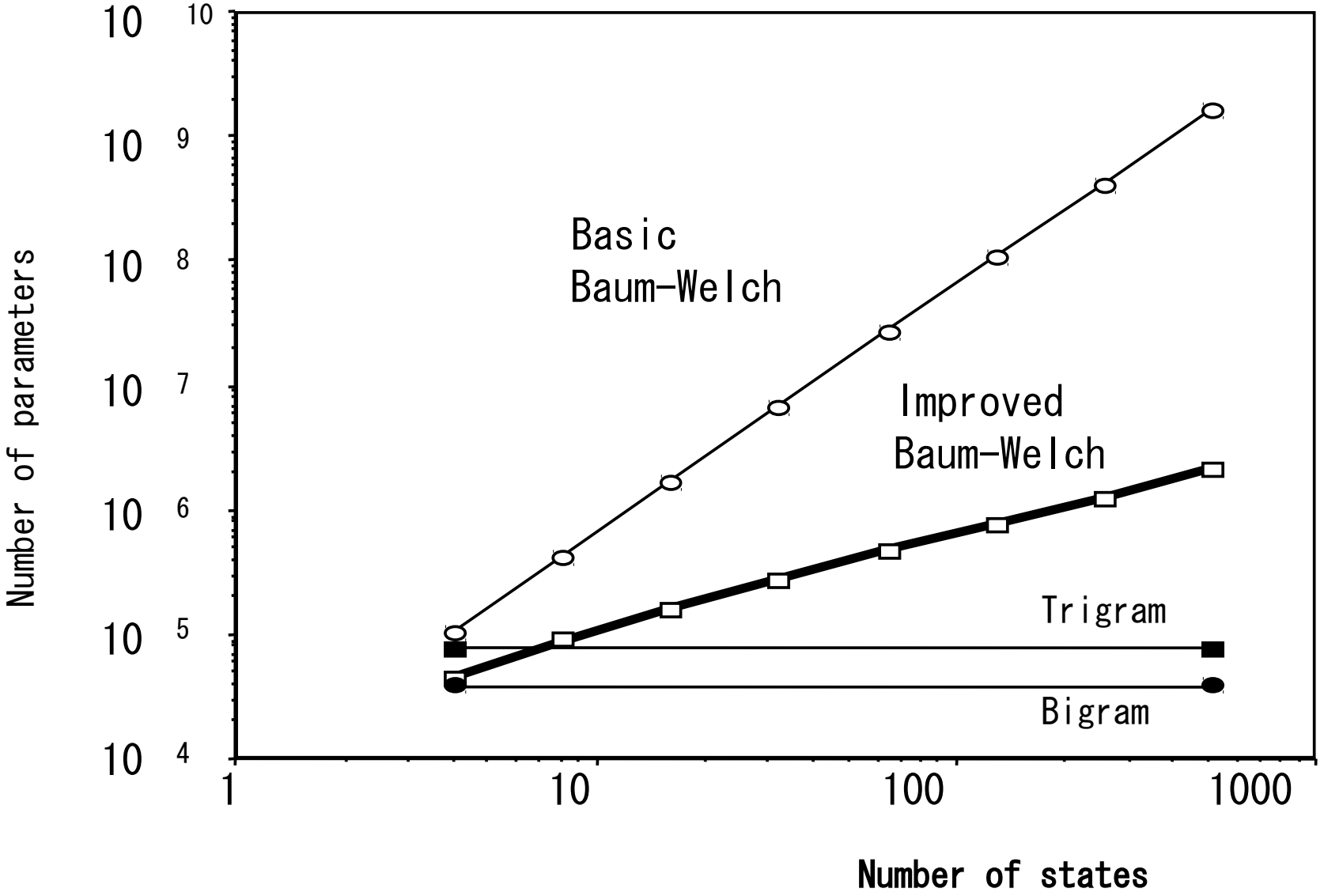
state number = 2N

確率付きネットワーク文法の獲得の実験

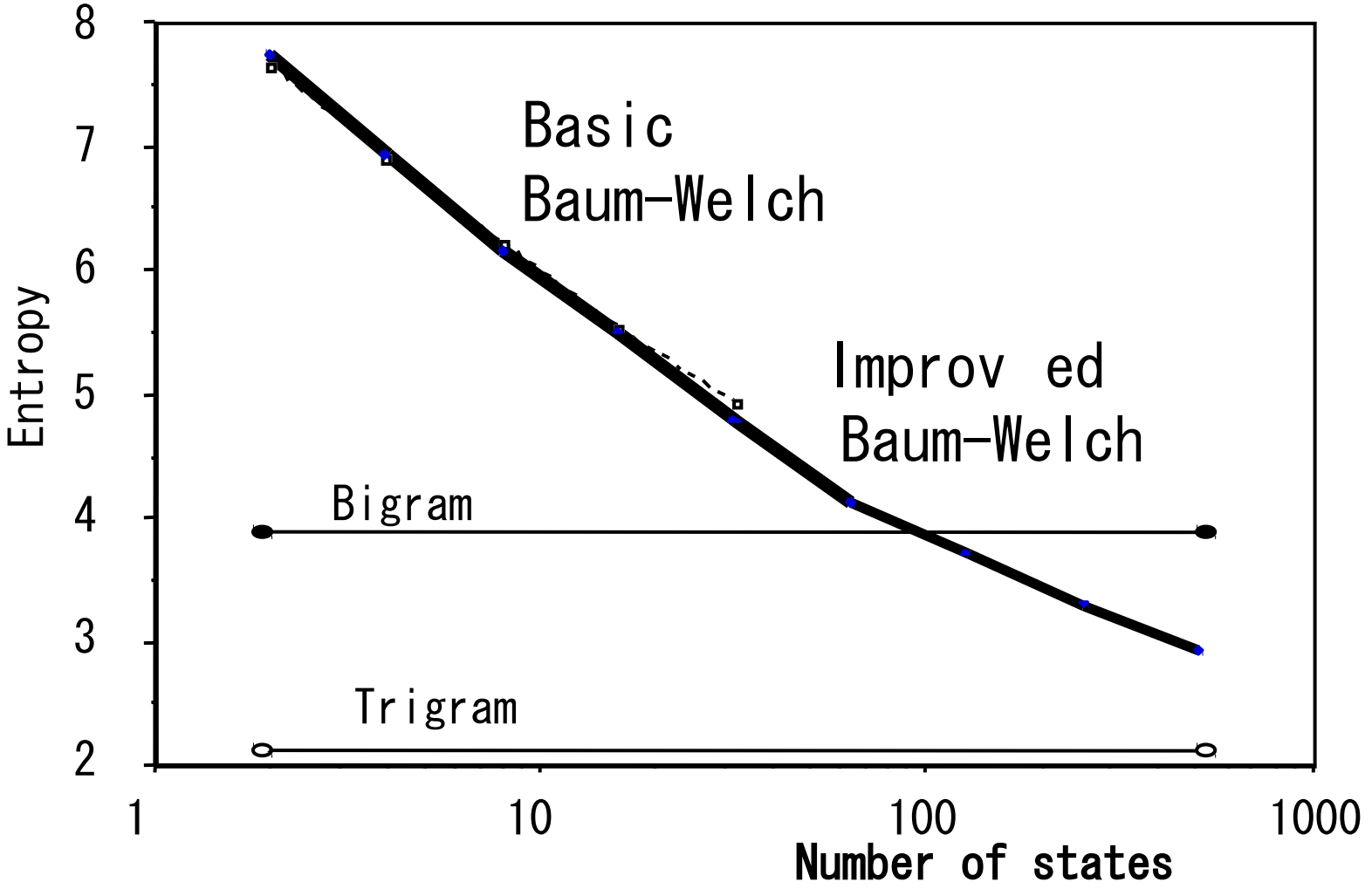
HMMの構造	状態遷移出力型
学習語彙数	6420 単語
学習データ数	8475
総単語数	57354
Baum-Welchアルゴリズムの終了条件	40回の繰り返し

言語モデル生成実験の条件

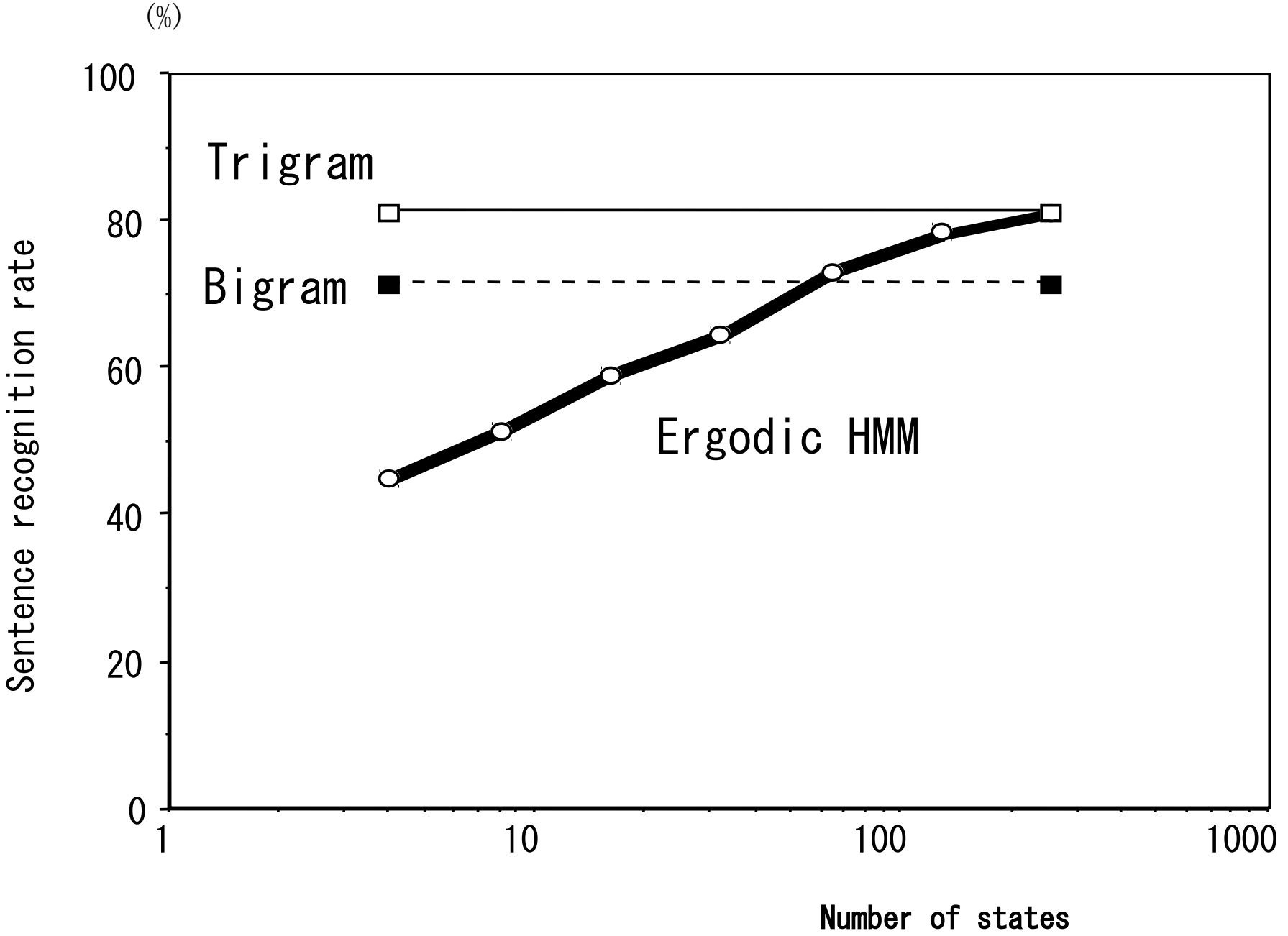
パラメータの数　vs 状態数



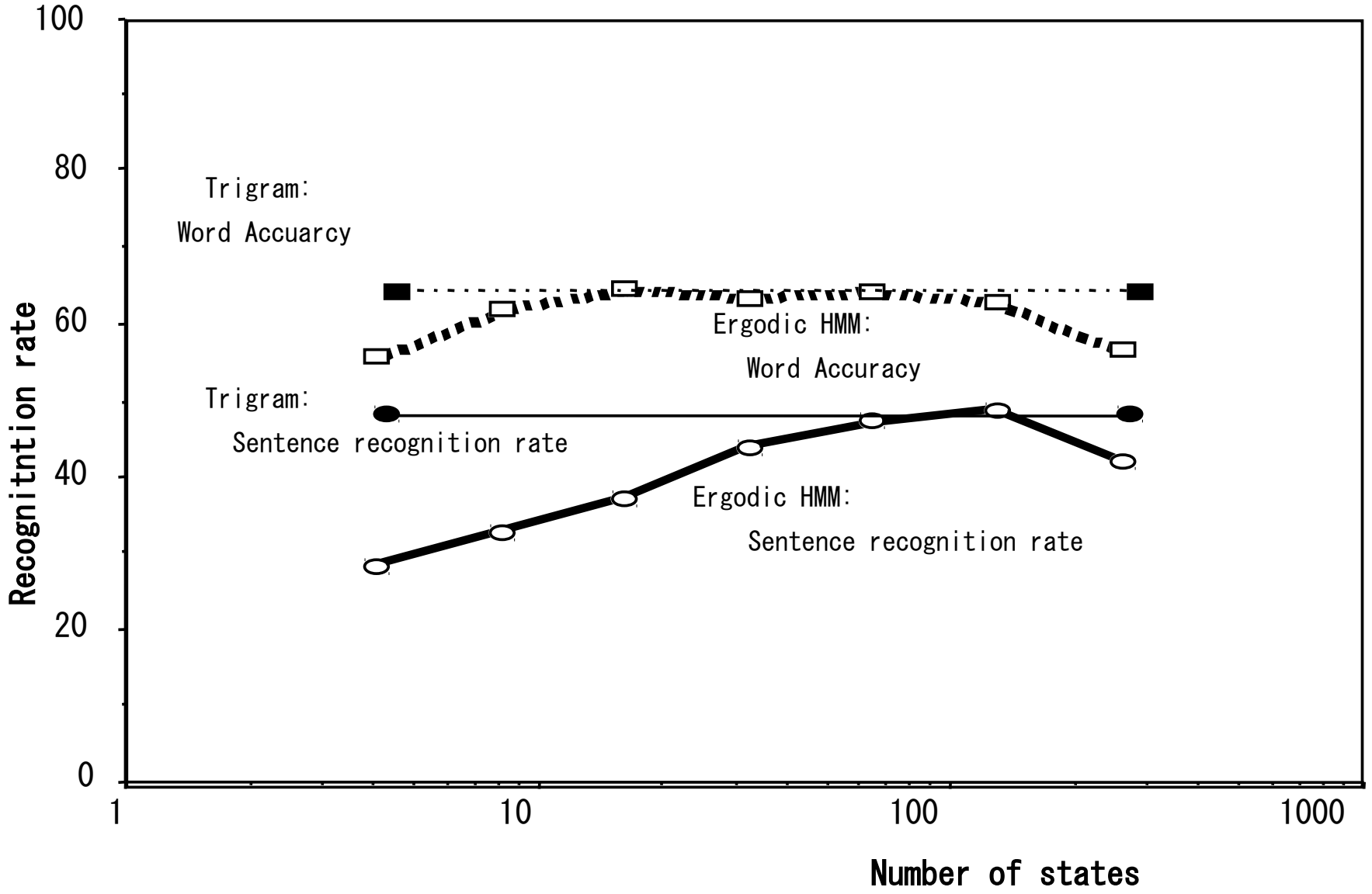
エントロピーの変化


8
7
6
5
4
3
2
Entropy
Basic
Baum-Welch
Improv ed
Baum-Welch
Bigram
Trigram
1
10
100
1000
Number of states

認識実験の結果　closed-test



認識実験の結果　open-test



まとめ

メモリ量および計算量を削減した
　Baum-Welch　アルゴリズムを提案

確率つきネットワーク文法の獲得
perplexity:
　bigram ＞ Ergodic HMM ＞ trigram

言語モデルとして
　連続音声認識に利用
認識率
　text-closed:　bigram ＜ Ergodic HMM ＜ trigram
　text-open :　trigram ＜　Ergodic HMM