**대화형 사용자 인터페이스**

**중간고사**

1. **음성인식의 문제를 정의하는 수식을 서술하고, 이 수식에 대한 의미와 각 기호들에 대한 설명을 서술하시오. [5점]**  
     
   수식 : argwmax P(W|O)  
   의미 : 마이크로 수집된 음성(Observation)데이터가 주어 졌을 때, 문장(W) 가 되는 확률을 구할 때, 가능한 O의 개수가 무한대이기 때문에 P(W|O)를 직접 구할 수 없음.  
   기호 설명 :   
   W : w1…wN (N개의 단어들로 이루어진 문장)  
   O : o1…oT (T개의 윈도우에서 각 윈도우로부터 나온 13차 vector(20ms단위의 소리)의 Sequence.)
2. **1번 문제에서 서술한 수식을 계산이 가능한 수식으로 변환하고, 그 과정을 설명하시오. [5점]**  
     
   Bayesian rule을 적용하여 변환  
   수식 : argwmax P(W|O) = argwmax   
   P(O) : 13 \* T 벡터 공간에서의 한 점의 확률  
   이 확률이 모두 동일하다고 가정한면, argwmax를 찾는 문제이기 때문에 P(O)를 생략 할 수 있음.   
   (13차 vector의 sequence의 확률이 모두 동일하다고 놓고 W를 찾는 문제이므로 생략이 가능.)  
   따라서 argwmax P(W|O)P(W)
3. **2번 문제에서 서술한 수식과 연관 지어 음성인식의 핵심 component 들을 3 가지 이상 기술하시오. [3점]**  
     
   음향 모델 : P(W|O)  
   언어 모델 : P(W)

디코딩 네트워크 : argwmax  
어휘(인식이 가능한 단어 set)의 구현이다.

1. **아래과 같은 2-mic array에서 mic2에 소리가 도착한 뒤 1/680 초 뒤에 mic1에서 동일한 소리가 도착했다면, 2차원 평면상에서 점 0를 기준으로 몇 도 방향에 음원이 위치하는지 풀이 과정과 답을 서술하시오. (단, 소리의 속력은 340m/s 이며 음원을 충분히 먼 거리에 떨어져있어 두 mic에 평행하게 입력된다고 가정한다. [10점]**

|  |  |
| --- | --- |
| θ = arcsin(S \* t/L) or t = L \* /S  t : 시간차  S : 속력 340m/s L : 1m  그리고 L \* = d  = 1 \* /340  = = = 0.5  θ = arcsin(0.5) = -0.5 or θ = arcsin(340\*/1) = arcsin( ) = arcsin(0.5) = -0.5 |  |

1. **HMM의 구성요소 4가지를 기술하시오. [8점]**  
     
   N개의 상태(State)

상태간 천이 확률 : aij 이전의 선택한 바구니를 보고 판단(Index 2개필요)

출력 확률 분포 (Output Probability distribution) : bj(ot) 바구니가 선택 된 후 그 다음 바구니의 확률  
초기 상태

1. **HMM의 3가지 문제를 기술 하시오. [9점]**  
     
   - **인식의 문제** : 관측열이 발생될 확률의 계산   
   모델의 인식 결과를 생성 할 수 있어야 한다  
   인식결과를 어떻게 결정할 것인가?  
   모델 파라미터는 주어진 상황에서 관측열의 생성 확률을 최대로 하는 모델 선정  
   알고리즘 : Total likelihood method , forward algorithm  
   - **Segmentation의 문제** : 관측열이 어떤 상태천이를 거쳐서 발생되었는지를 추정  
   대량의 음성코퍼스로부터 모델의 구분단위(음소)별 학습자료를 자동 생성 할 수 있어야 한다.  
   전체 음성신호 중 특정 단어를 찾아 주는 segmentation시 필요  
   전체 학습자료에서 모델 별 학습에 필요한 자료 자동 추출  
   알고리즘 : viterbi algorithm (auto segmentation)  
   - **학습의 문제** : HMM의 파라미터 추정   
   컴퓨터에 구현 가능한 알고리즘으로 모델이 학습 할 수 있어야 한다. / 학습방법론.  
   모델 학습 시 필요  
   알고리즘 Viterbi training algorithm, Baum-Welch algorithm
2. **가수가 10곡을 수록한 앨범을 발매 했다. 이를 CD에 담았을 때, 전체 CD 중 burning된 부분은 CD 전체 면적의 몇 %인가? [8점]**

한 곡의 길이는 4분으로 가정

CD는 총 700MB를 저장한다고 가정

Sampling rate : 초당 44,100

Sample 당 2byte 사용

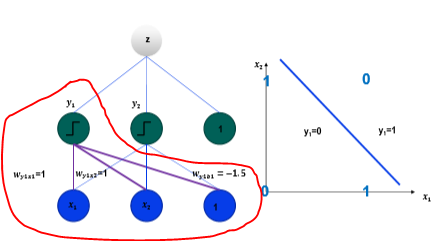
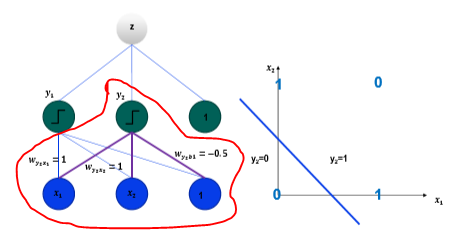
Stereo로 녹음 되었기 때문에 채널은 2개

44,100(samples/sec) \* 2(byte) \* 240(sec) \*10(곡) \* 2(channels) = 423,360,000 = 423.36/700MB = 약 60.48(%)

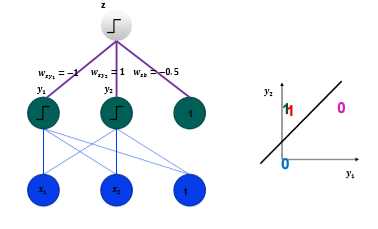
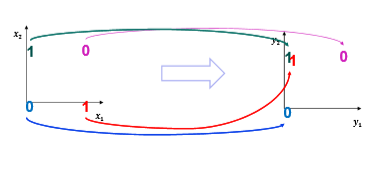
1. **다음 XNOR의 truth Table에 대하여 아래 문제에 답하시오. [14점]**

|  |  |  |
| --- | --- | --- |
| Input | | Output |
| x1 | x2 | y |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

위의 Truth table 의 input vector를 output 값에 따라 분류하는 Neural net을 design 하라. (각 weight의 유도과정을 보이고, input vector가 hidden layer 상의 vector 공간에 어떻게 표현되는지도 보일 것)  
  
XOR : 입력이 다를 경우 1 / 같을 경우 0  
XNOR : 입력이 같을 경우 1 / 다를 경우 0 (XOR연산에 NOT)  
신경망의 XOR 분류 문제를 Multi-layer을 통해 해결 (선 하나 더 그어서 분류..)

y1 = 1(x1, x2) = x1 +x2 – 1.5  
  
  
y2 = 2(x1, x2) = x1 +x2 – 0.5  


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x1 | x2 | y1 | y2 | output |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |

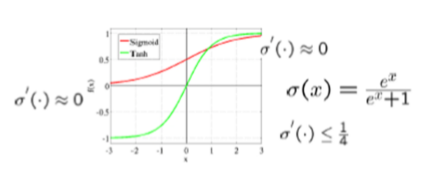


1. **음성 파일과 이에 대한 script가 있을 때, 이로부터 음소별 Labeling을 수행하는 방법을 기술하시오.**

|  |  |
| --- | --- |
|  | **Segmentation : Viterbi Algorithm** Forward 알고리즘의 각 격자점에서 partial path에 대한 local decision을 수행하여, 최적의 상태열의 결정이 가능  알고리즘 :  Фj(t)=o1…ot를 생성하면서 *s(t) = j* 에 도달하는 가장 적합한 path의 likelihood |
| 음성파일의 script는 순서대로 발생되는 시계열 패턴이므로 우상향 할 수 밖에 없다. x축(time정보)이 증가함에 따라 y값이 같거나 큰 둘 중의 하나 값을 가지게 되고, script를 만들 때, 시간상 돌아 갈 수 없다. | - Dynamic programming 기법으로 Ф(t-1)을 이용하여, Ф(t)를 결정 Фj(t) =  최적 상태열에서의 likelihood 값은 아래와 같이 계산됨 Prmax(O|M) =  - 각 격자점에서의 local decision의 결과를 기록했다면, 최적의 상태열은 T로부터 trace back하면서 결정 가능   - 인식의 문제에도 적용 가능함   Viterbi algorithm으로 계산한 likelihood로써 모델로부터의 생성 확률을 근사 |

1. **Expectaion maximization에 대해서 설명하고, 이를 수식으로 표현하시오.[8점]**

HMM의 학습의 문제에서 사용 되는 알고리즘으로 전체 학습자료에서 생성확률이 최대가 되는 모델을 구하기 위한 학습알고리즘. (supervisor learning)  
  
R : 전체 학습자료에서 모델 M에 대응되는 데이터의 수  
*단어당 평균 음소의 수를 10개로 가정할 경우, 코퍼스 전체에서 나타난 음소의 수  
500명 \* 417단어 \* 10개 = 2,085,000개  
인식단위가 음소이고, 전체 음소 수가 40개이면 평균 R = 2,085,000 / 40 = 52,125*  
학습 자료로부터 모델 파라미터의 업데이트는 자동으로 진행이 됨.  
  
  
알고리즘  
Viterbi training algorithm : 근사치로 학습 진행. 속도가 빨라 실무에 사용이 많이 사용됨.  
Baum-Welch algorithm. : 정확한 방법이지만, 수식이 복잡하고 시간이 많이 소요됨.

1. **DNN 에서 Gradient vanshing 문제에 대해서 기술하시오. [6점]**  
     
   학습알고리즘을 back-propagation(역전파)알고리즘으로 수행 할 때, Non-linear activation 함수를 이용하게 되면 가중치를 위한 기울기 값이 소실 되어 학습이 되지 않는 문제가 발생.  
   Gradient가 층을 하나씩 내려가면서 error가 희석되는(작아지는) 문제가 발생   
   – 시그모이드(Sigmoid) 함수의 경우, 값이 급격히 변하는 구간을 제외하면, 미분값(경사도)은 0에 가까움  
   

|  |  |
| --- | --- |
| 해결 방법 :  Activation 함수를 ReLU 함수로 대체 0보다 크면 activation |  |

1. **음성 인식용 DNN-HMM(3문제)**
2. 음성 인식용 DNN-HMM 방식에서 많이 사용 되는 DNN이 hyper parametere를 기술하시오. [5점]

Input layer node 수 : 600  
Hidden layer 수 : 5~7

Hidden layer당 node 수 : 2000  
output layer node수 : 6000~10000

1. Input layer의 node의 의미를 기술하시오.[5점]  
   음성 vector (자기 자진 + 앞뒤context(7\*2) = 15  
   15frames X 40 filterbank outputs = 600차 vector를 의미
2. Output layer의 node의 의미를 기술하시오.[6점]

Seen-tri phone 1만개 \* 3state (HMM 3state가 성능이 제일 좋음)  
Left-to-right모델이므로 3만개 중에서 학습데이터와 Unique 매핑되는 값을  
Decision Tree 기법으로 클러스터링하여 약 6,000~10,000개 state 생성

**기말고사 (문제 없어서 기억 나는 부분으로 추림..)**

3.장. 언어 모델 : **P(W)** = (Wk | Wk-1 Wk-2…W0)

|  |  |
| --- | --- |
| **언어모델** | 3.1. **n-gram 언어모델**  3.1.1. Discounting  **3.1.2 . Back-off 언어모델**  3.1.3. Language Model Scaling  **3.1.4. Text Corpera Normalization**  **3.1.5. 언어 모델 평가 처도(Perplexity)** 3.1.6. Relationship between PP and WER 3.1.7. 언어모델 Interpolation  3.2. 카테고리 기반 언어모델  3.2.1. 품사기반 카테고리 언어모델  3.2.2. 단어 카테고리 자동생성  3.3. DNN 기반 언어모델  3.3.1. DNN 기반 언어모델  3.3.2. Word Vector를 이용한 입력 차원 감소  3.3.3. **Hierarchical Softmax**를 이용한 출력 차원감소 |

1. **n-gram 언어 모델 설명**

문장이 길어짐에 따라 history 또한 길어짐.

확률 계산이 computationally intractable하게 됨

계산이 가능하게 하기 위해서는 history의 길이를 줄여야 함

표준화된 방법은 n-gram임 (hisoty의 제약)  
가정사항: 최근 n-1개의 단어로 구성할 수 있는 모든 history는 같은 history로 다룸  
P(Wk | Wk-1 Wk-2…W0) = P(Wk|Wk-1…Wk-N+1)

n-gramUnigram : 현재 한 단어만 반영  
Bigram : 바로 앞 단어까지 반영  
Trigram : 바로 앞 두 단어까지만 반영

대용량의 학습 코퍼스로부터 통계적 자료 추출하여 생성.

Estimating n-Grams From Counts (tri-gram)  
Expectation Maximization에 따라 P(Wi|Wi-2Wi-1)는 아래와 같이 계산된다.  
P(Wi | Wi-2 ,Wi-1) = =

|  |  |
| --- | --- |
| **장점** | **단점** |
| 통계적 모델로써 계산의 간편함. 대용량 학습 자료를 이용하여 쉽게 모델 생성이 가능함. | N의 제약으로 인하여 loger history에 대한 정보를 표현하지 못함. |

1. **N-gram 언어 모델의 문제점**

**메모리의 문제**어휘의 크기가 10만()인 경유,tri-gram tuple은 센 단어의 index로 표현됨.  
table 형태로 저장하기 위해서는 3차원 integer table이 있어야 함.  
이 경우 tri-gram tuple수는 이며, 필요한 바이트 수는 4\*B = 4\*B= 4,000TB(1TB 외장하드가 4천개가 필요하게 됨.)

**학습 자료의 부족**  
특정 방송국이 100년간 방송한 자료가 있다는 가정에 1년에 300일 하루 20시간 방송했다 치면, 총 수집한 오디오 양은 600,000시간 분량이 된다.  
영미권 뉴스의 경우 100시간당 100만단어가 발화 된다고 알려져 있다. 이 기준을 적용해보면 100년간의 방송자료에 대해 transcription을 만들면 60억(6\*109)단어 = (600,000시간 \*100만 단어/100시간)가 존재  
Table 샐 개수가 1015개 임을 감안하면, tri-gram tuple의 수가 현저히 부족하다.

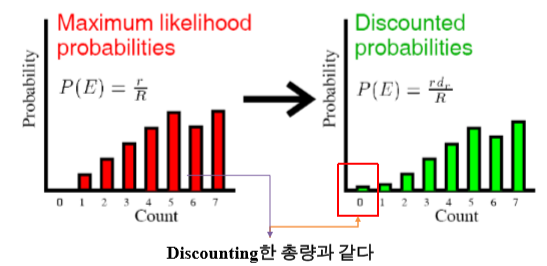
코퍼스를 아무리 많이 모으더라도, 실제 발화에서 나타나는 (wi-2,wi-2,wi), (wi-2,wi-2)를 적절히 추정하지 못하는 경우가 발생하며, 최악의 경우는 0이 되는 경우이다.

(wi-2,wi-2,wi)가 0이 되는 경우(분자가 0이 되는 문제) : 문장생성확률 P(W)=(wi | wi-2, wi-1)중 특정 단어 wi 생성확률P(Wi | Wi-2 ,Wi-1)가 0이 되고, 따라서 P(W)가 0이 되어버림.

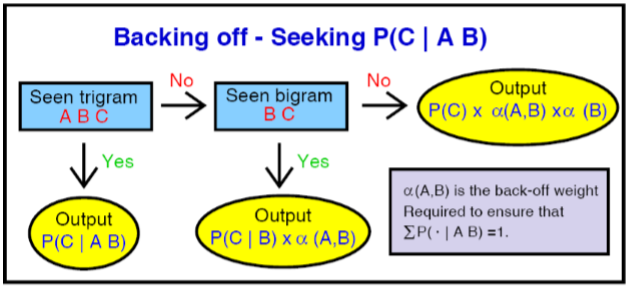
(wi-2,wi-2)가 0이 되는 경우(분모가 0이 되는 문제) : 0으로 나누는 문제가 발생하여 계산이 불가능

분자가 0이 되어 버리면, 음성인식 분류로 해결 되지만, 분모가 0이 되면 치명적

1. **위의 분모/분자가 0이 되는 N-gram 언어 모델의 문제점 해결 방법**

**Discounting**  
0의 값을 가지는 (wi-2,wi-2,wi)에 대해 작은 값으로 flooring 시킴  
flooring하는 과정에서 sum-to-one 제한 ( (wi|wi-1,wi-2) = 1)을 만족하지 않게 됨.  
제한을 만족시키기 위해, 관측된 사건의 수를 discount 하여, 비관측 사건에 할당. 따라서 언어모델 생성확률은 다음과 같이 수정됨.  
((wi|wi-1,wi-2) = d((wi-1,wi-2,wi))  
여기서 d(r)을 discount coefficient(계수)라 한다.  
d(r)의 추정방법에 따라 여러 방법론이 있다.  
  
  
  
X축 : frequency(count)  
Y축 : 해당 frequency를 가지는 tuple들에 대한 언어모델 생성 확률 값들의 총합  
  
**Back-off**  
(wi-2,wi-2,wi)이 작아 적절한 n-gram 언어모델 생성 확률 추정이 어려운 경우, 모델링 파워는 낮지만 적은 양의 코퍼스로부터 적절한 확률 추정이 가능한 (n-1)-gram을 사용하여 추정함.  
Bigram의 예 :  
Pr(wj|wi) =

는 오른쪽 식을 만족해야 함.

C는 n-gram cut-off frequency  
(n-1)-gram언어 모델 추정도 어려운 경우, (n-2)-gram, (n-3)-gram…을 이용하여 추정한다.  
Trigram의 오류를 bigram으로 estimation하고, bigram에서 해결이 안되면, unigram(확실히 막음)으로 계산  
unigram 같은 경운 수동으로 사람이 충분히 계산 할 수 있으므로 unigram에서 해결이 안되면 수동으로 사람이 해결 bigram단계에서는 확률이 많아서 수동 불가  
  


* **Language Model Scaling (logP) :문제는 안 나왔던 듯..**

argwmax P(O|W)P(W) P(O|W)P(W)  
음향모델의 생성확률 P(O|W)는 매 frame마다 aijbj(ot)가 곱해진다.   
반면, P(W)는 매 단어마다 P(Wi | Wi-2 ,Wi-1)가 곱해진다.  
따라서 음향모델의 생성확률의 영향이 상대적으로 훨씬 크다.  
이 문제점을 보완하기 위해, 언어 모델 sacaling factor alpah를 이용하여 P(O|W)P(W)를 아래와 같이 계산한다.

는 실험을 통해 계산된다. 영어 뉴스 인식의 경우 보통 2.0정도가 사용된다.

1. **언어 모델 평가척도(Perplexity)에서 문제 나옴..**텍스트만을 이용한 언어모델 품질 척도가 필요하다.  
   LogProb(LP)는 각 단어 별 로그 n-gram생성확률의 산술평균(n개를 곱해서 n으로 나눔)을 정의한다.  
   LP = 로그를 빼내면 단어 별 생성확률의 기하평균으로 이해 할 수 있음.  
   lim (코퍼스의 사이즈를 무한대)는 수집한 코퍼스의 양이 매우 크다는 것을 의미한다.  
     
   평가척도의 값이 음수가 되는 것을 막기 위해 -1을 곱하여 LP의 값을 양수로 바꾸어 준다.  
   Perplexity(PP)는 LP를 이용하여 다음과 같이 정의함.  
   PP = 2LP( = )를 이용하면 PP는 아래와 같이 정리됨.  
   PP =

PP의 의미는 단어 당 n-gram 생성확률의 기하평균의 역수  
n-gram 생성확률의 기하평균 : 평균확률의 역수를 의미하는데 최악의 언어 모델은 기하평균이 1/10만 이고, P=10만이 됨.  
따라서 PP는 언어모델에 의해 추정되는 다음 언어의 평균수(average branching factor)로 이해할 수 있음.  
동일한 어휘 크기와 도메인에 대해서 PP가 작은 언어 모델이 일반적으로 좋은 언어 모델이다.  
PP의 이론상 최대값은? 어휘의 크기  
PP의 이론상 최소값은? 1 (음성을 듣지 않고 지금까지 나온 단어만을 가지고 다음 단어가 정확히 예측이 된다는 의미이나 사실상 불가능)

PP가 감소할수록 word error rate(WER)도 감소.

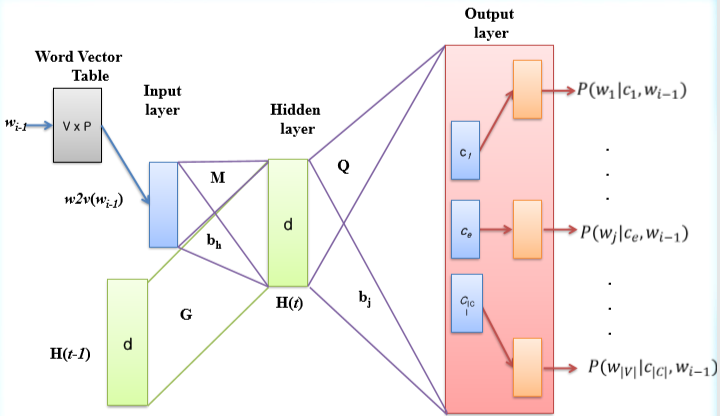
1. **단어 카테고리 자동생성 기본알고리즘 6단계?**기본 알고리즘 (어휘 크기 : , 카테고리 수 : K)  
   ① Unigram 통계 생성(단어 별 빈도수 측정)  
   ② (초기화) 가장 빈도가 높은 단어를 카테고리1에 매핑. 그 다음 높은 빈도의 단어를 카테고리 2에 매핑. 같은 방법으로 K-1번째 카테고리 까지 매핑. (K-1개의 각 카테고리들은 1개의 단어만 매핑되어있음)  
   ③ (초기화) 나머지 단어들(총 -(K-1))을 모두 K번째 카테고리에 매핑  
   ④ 현재의 단어-카테고리 매핑 상에서 모든 가능한 카테고리 이동에 대하여(총 -(K-1)개) 각 이동을 적용했을 때의 perplexity변화를 측정함.  
   ⑤ 4의 과정에서 perplexity를 가장 낮추어 주는 이동을 선택하고 이를 적용한다.  
   ⑥ perplexity를 낮추는 이동이 없을 때 까지 4-5의 과정을 반복한다.
2. **DNN 기반 언어 모델 관련 문제 (차원감소 내용 중요하다고 되어 있네요.. 정확한 문제는 기억이..)**

DNN 기반으로 discrete 입력에 대해서, 대용량 자료에서 기존 n-gram보다 좋은 성능을 보이기 어려움.  
DNN 기반 언어모델의 구조 (Feed-Forward구조)  
가정사항 :  
- 어휘사전 : 약 65K  
- Trigram : (Word history 2개 단어)

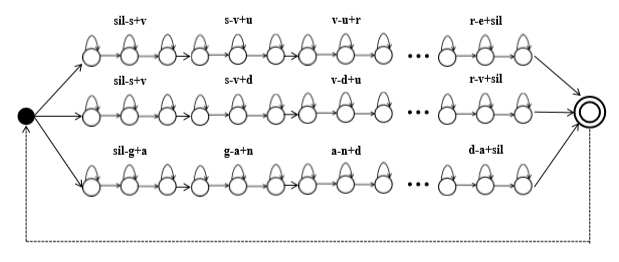
Input layer unit 수 : 130K (2\*65K) // tri-gram

Hidden layer : 2-5 hidden layers

Output layer unit 수 : 65K

1-of-N coding 형태의 discrete입력음향모델과의 비교분석  
- 입력 차이 : 약 200배  
- 출력 차이 : 약 6배  
차원을 줄이는데 연구를 하고 있음.

입력차원 감소 :   
**word vector** space  
미리 학습 시켜서 경험적 단어 관계를 포함 / word간의 관계가 similarity를 통해 표현  
Semantic 또는 syntactic 관점에서 유사한 word들이 continuous vector space상에서 서로 가깝게 표현되면, 다양한 unseen word sequence를 포함하는 문장에 대해서도 generalization을 잘 표현하게 됨.  
ex) 구글 Word2Vec을 이요하여 Distributional Hypothesis에 부합하는 word vector 생성  
Word vector의 차원을 어휘 크기(에서 100~600차원으로 표현  
  
출력차원 감소:  
**Hierarchical Softmax 방법**  
신경망 구조에서 output layer의 특정 output unit이 생성될 조건부 확률을 추정하는 경우 / 모든 output unit의 값 대비 특정 output unit의 값의 비율로 측정하는 normalization방법  
Hierarchical Softmax방법 (Class-based 언어모델 연구에서 제안된 Auto-Clustering 기법 이용)

1. **“서울에 서둘러 간다” Lexical Tree Search network 그리는 문제**

HMM, Context dependency, Lexicon, Grammar가 별도로 구현되어 있음. Runtime에서 T가 증가하며 Network을 차례로 만들어 나가는 방식   
Search space가 너무 커지는 것을 막기 위하여 beam pruning을 적용 

단점 : 속도가 느리다

**4장. Decoder**  
4.1. Lexical Tree 기반 Serch

4.2. Weighted Finie State Transdcer(WFST)

4.2.1. WFST 개요  
4.2.2. WFST 기반 디코딩 구성요소  
 4.2.2.1. Grammar transducer(G) : language모델 그리기 / 알파 구하기

4.2.2.2. Lexicon Transducer(L)

4.2.2.3. Context-dependency transducer(C)

4.2.2.4. HMM topology transducer(H)

4.2.2.5 Utterance transducer(U)  
2.2.3. WFST기반 효율적 디코딩에 필요한 연산

1. **WFST 기반 디코딩**HMM, Context dependency, Lexicon, Grammar를 결합한 효율적인 Network를 미리 만들어 놓는다.  
   Composition, Determinization, Minimization이 필요 

단점

• 메모리를 많이 요구한다   
• 어휘 또는 학습 텍스트의 약간의 변화만 있어도 Network를 다시 구성 해야 함   
• 대용량 언어모델을 사용하는 경우 Network가 커짐

WFST기반 디코딩 구성요소Exploit several knowledge sources (lexicon, grammar, phonetics) to find most likely spoken word sequence [Mohri, 2008]

𝑯𝑪𝑳𝑮 = 𝑯 ∘ 𝑪 ∘ 𝑳 ∘ 𝑮

Create H, C, L, G separately and compose them together

Grammar Transducer : G

Probabilistic grammar or language model acceptor (word)

Lexicon Transducer : L

Lexicon (phones to words)

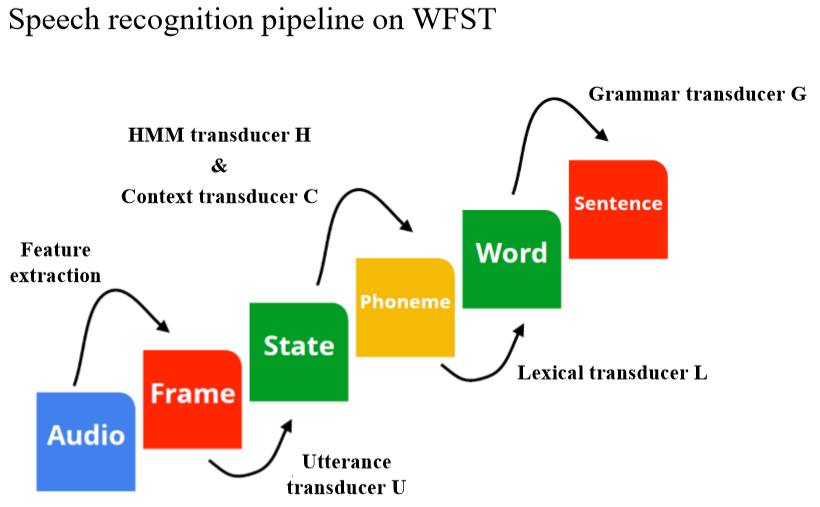
Context dependency Transducer : C

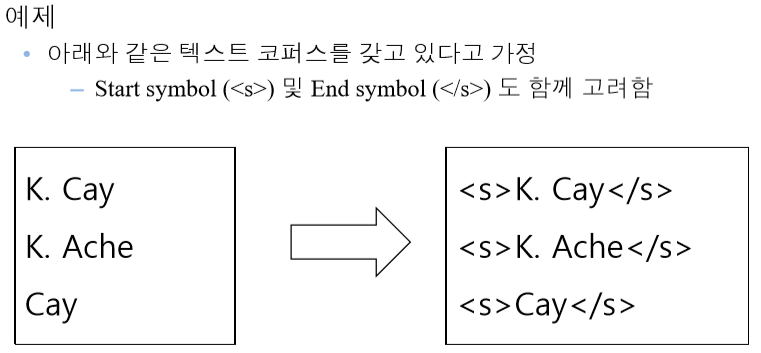
Context-dependent relabeling (context-dependent phone to context-independent phone) 

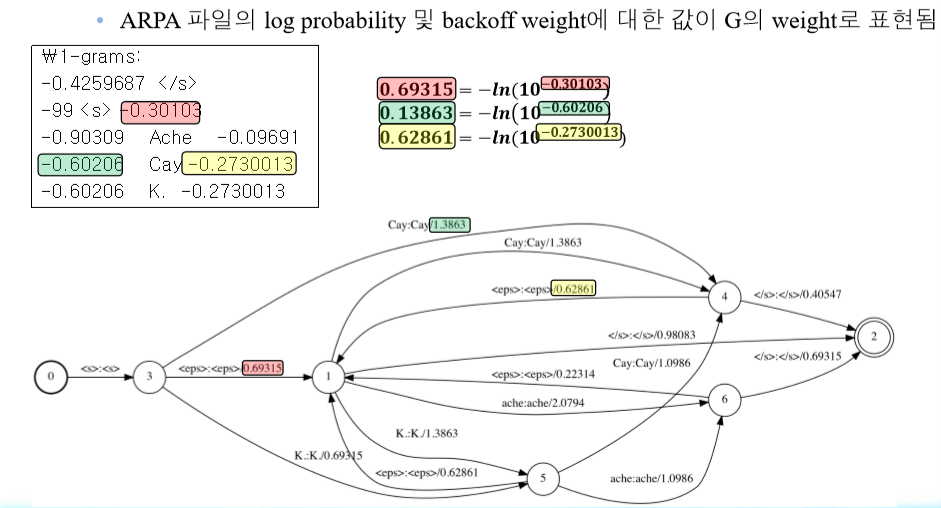
HMM : H

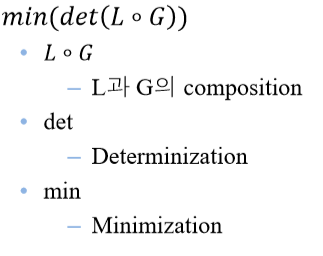
HMM structure (PDF labels to context-dependent phones)

본 강의에서 예제는 Kaldi toolkit에서 구축되는 WFST 기반 음 성인식 디코딩 방법에 따라 진행    
Kaldi toolkit에서 구축되는 WFST 기반 음성인식 디코딩𝐻𝐶𝐿𝐺 = 𝑎𝑠𝑙(min 𝑟𝑑𝑠(det 𝐻𝑎 ∘ min(det 𝐶 ∘ min(det 𝐿 ∘ 𝐺 ))))



1. **Grammar transducer G 생성 그리는 문제  
   **

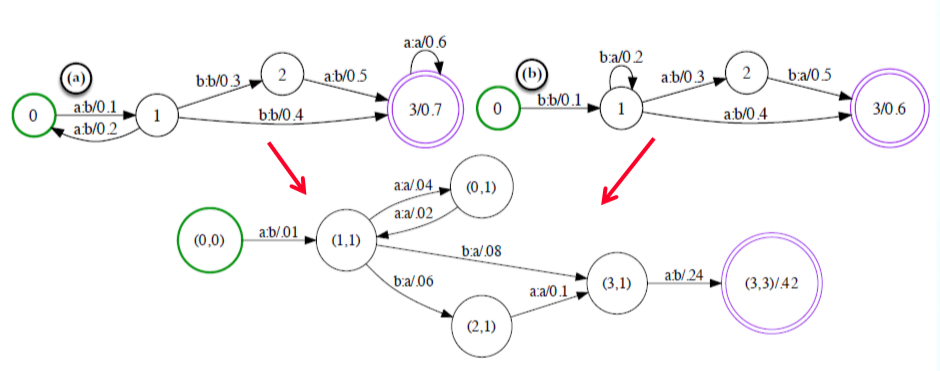
****

1. **HCLG 생성하기 위한 알고리즘**compose L and G, determinization and minimization

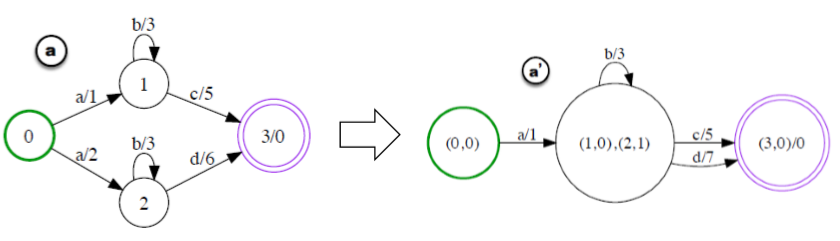
**Compositon :** 발음 모델을 기반으로 가능한 모든 탐색 경로를 구성

• 𝑳 ∘ 𝑮 는 L composed with G를 의미   
– 𝑮 ∘ 𝑳 ≠ 𝑳 ∘ 𝑮 if 𝑳 ≠ 𝑮

• L의 output symbol과 G의 input symbol이 일치하는 edge에 대해서 composition

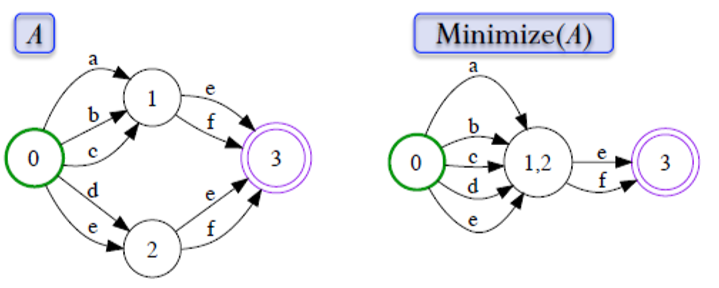
  
**Determinization :** 동일한 입/출력값을 갖는 탐색 경로 제거

입력경로에서 모호성을 제거합니다.   
– Improves efficiency of downstream operations such as shortest-path  
– Each state has at most one outgoing transition containing any particular input label



**Minimization :** 가중치를 고려해 동일한 출력을 갖는 탐색 경로 단일화

Given an input WFST, produces a ‘minimal’ version which is guaranteed to have the smallest possible number of states while preserving the input language and weight/path properties of the original



5/6/7장 (뒷부분은 시험복원은 하지 못했고, 중요하다고 표시된 부분만 요약 했어요..)

1. **Kaldi (문제에 나왔던 듯??)**

Kaldi [Povey, 2011]

DNN-HMM 방식 지원

C++ 기반의 객체 지향 형태로 구현되어 있어, 새로운 입력 및 내부 구조 변경에 대해 확장 가능성 있도록 구현함

디코딩 과정에서 WFST 활용

• Code-level integration with Finite State Transducers (FSTs)

LDC에서 제공하는 음성 코퍼스를 이용한 Recipe를 제공

• 대표적 영어 연속 음성인식 Recipe : WSJ, RM 

새로운 SW 개발시 소스코드 공개 의무 없으며, 상업적 이용에 제한이 없음

1. **Audio fingerprint 음악인식 알고리즘 (여기서도 한 문제 나왔던..)**  
   **Gracenote 구현방법**  
   **Shazam 구현방법**

**Grace note와 Shazam구현방식의 차이**

Gracenote

• 연속된 오디오 신호를 sub-fingerprint의 열로 저장

• 인식 단위 – ingerprint block : 256개의 sub-fingerprint

• Look up table을 해시함수를 이용해 구현

• 일정한 입력길이와 인식 소요시간이 장점

Shazam

• 오디오 신호의 피크 특성을 특징으로 추출

• 피크 정보는 잡음에 강인하지만 fingerprint를 생성하기 위한 입력 길이가 길어져 인식 소요 시간이 오래 걸리는 단점이 있음

1. **End to End 한 문제..(교재 6페이지까지만 보심 되요..)**