

# 기계 학습 기법으로 스팸 메일 걸러내기

---

소프트웨어 무결점 연구(ROSAEC) 센터  
제 5회 워크샵

튜토리얼  
2011년 1월 8일

카이스트 전산학과  
응용알고리즘 연구실  
곽남주

# 차례

---

- 스팸의 어원
- 왜 스팸 메일을 보내는가?
- 스팸 메일에 당한 사례
- 스팸 방지를 위한 노력
- 기계 학습을 활용한 스팸 걸러내기
- 베이지안 스팸 거름법(Bayesian Spam Filtering)
- 복수 단어 인식 단위로의 확장(Multiple-Word Feature)
- 마르코비안 거름법(Markovian Filtering)
- 은닉 마르코프 모형과 난독화 문제 해법  
(Deobfuscation with Hidden Markov Model)
- 정리
- 질의응답
- 참고자료 및 참고 문헌



# 스팸의 어원

- 1970년대, 영국의 **코미디 프로그램**
  - ‘스팸’이라는 말을 불필요하게 **반복**적으로 함으로써 웃음을 유발
  - 초기 인터넷 사용자들 사이, 온라인 게시판이나 토론장을 **스팸이라는 말로 도배**하는 장난이 유행
- 1980년대
  - **머드**(multi-user-dungeon)게임에서도 성행
  - **Usenet**과 개인 **이메일** 사용자, 지나친 광고글을 게시하는 일을 ‘스패밍’한다고 부르기 시작



영국 코미디  
Monty Python  
Sketches

World of  
Warcraft  
대화창 스팸

```
[20:48:32] [2] [Achieve]: No, it was the Natherzim.  
[20:48:36] [2] [Netzach]: It probably was, Cappa  
[20:48:45] [2] [Healsguy]: WTB Golden rod 5g  
[20:48:47] [2] [Achieve]: However you spell it. The vampire dudes like  
Tichondrius.  
[20:48:47] [2] [Jeicen]: Sargerass went insane due to the chaos of demons in  
the Twisting Nether  
[20:48:54] [2] [Ontargre]: it was never clearly stated whhat corrupted  
Sargerass  
[20:48:55] [2] [Kish]: I believe Red Shirt Guy corrupted Sargerass  
[20:49:10] [2] [Killiamós]: LFG for razorscale. Tank or dps  
[20:49:16] [2] [Netzach]: Either wya, Sargerass is the final expansion... will  
probably be in the Twisting Nether xpack  
[20:49:35] [2] [Achieve]: It was hinted at. They said Sargerass began to  
realize no matter how many worlds he purified there was still an infinite  
number of demon worlds out there which made him doubt his mission.  
[20:49:41] [2] [Jeicen]: Twisting Nether is a big place, will probably focus on  
Argus, the draenei homeworld  
[20:49:47] [2] [Lilnaynay]: the legion basidly tricked azshara to help them,  
and she betrayed the night elves on suramar but she soon found herself  
when the well of eternity was destroyed in the seas, and she then  
transformed.  
[20:49:53] [2] [Shiftry]: needs big guild. please invite
```

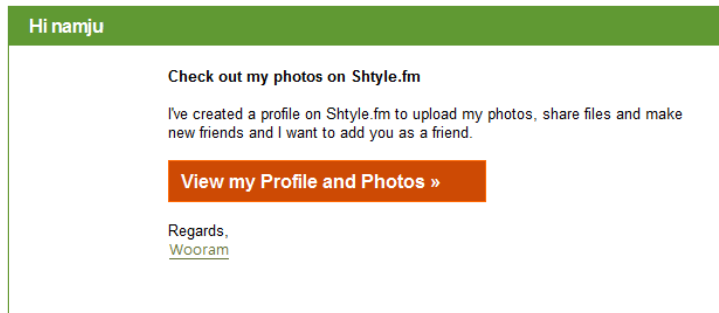
# 왜 스팸 메일을 보내는가?

- 통계 수치에 따르면 **12,500,000통**의 상품 판매 목적 스팸 메일을 보내면, **한 건 정도는 매출**로 이어진다고 함(2010년 3월 자료).
- McAfee에 의하면, **미국인의 절반이 매일 이메일**을 사용하고, 그 중 **절반이 귀가 얇아** 잘 속는 경향이 있고, 그 중 **1%가 구매를 시도**하다 **신용사기**의 희생양이 되어, **\$20씩을 지불**해야 한다면, 잠재적 시장 규모가 미국 내에서만 **일간 1500만 달러, 주간 1억 500만 달러, 연간 55조 달러**에 이릅니다.



# 스팸 메일에 당한 사례

- Shtyle.fm

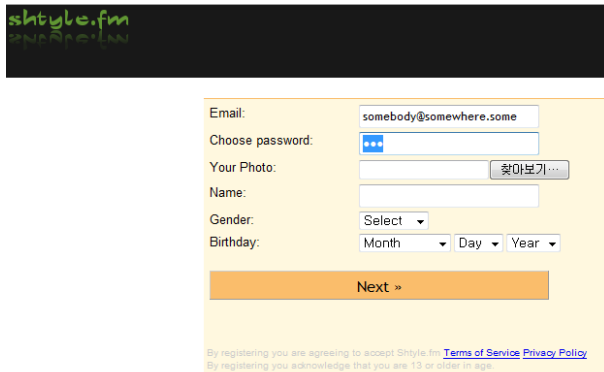


뭔가 페이스북  
같은 곳일까?



# 스팸 메일에 당한 사례

- Shtyle.fm



The screenshot shows the Shtyle.fm registration page. It includes a header with the Shtyle.fm logo. The registration form has the following fields: Email (with the example 'somebody@somewhere.some'), Choose password (with a strength indicator), Your Photo (with a '찾아보기...' button), Name, Gender (a 'Select' dropdown), and Birthday (with 'Month', 'Day', and 'Year' dropdowns). A 'Next >' button is at the bottom. Below the form, there is a small disclaimer: 'By registering you are agreeing to accept Shtyle.fm Terms of Service Privacy Policy. By registering you acknowledge that you are 13 or older in age.'

아주 간단한 비밀번호도 통과되고 세부 정보는 제공하지 않아도 가입이 가능  
단, 아이디는 이메일 주소

가입기념으로 친구에게 선물을  
줄 수 있다고 하면서,  
가입시 사용했던 이메일 주소의  
비밀 번호를 요구함

## Send a Gift to Your Friends

Your Email:

Enter the password for your email account.

Email Password:

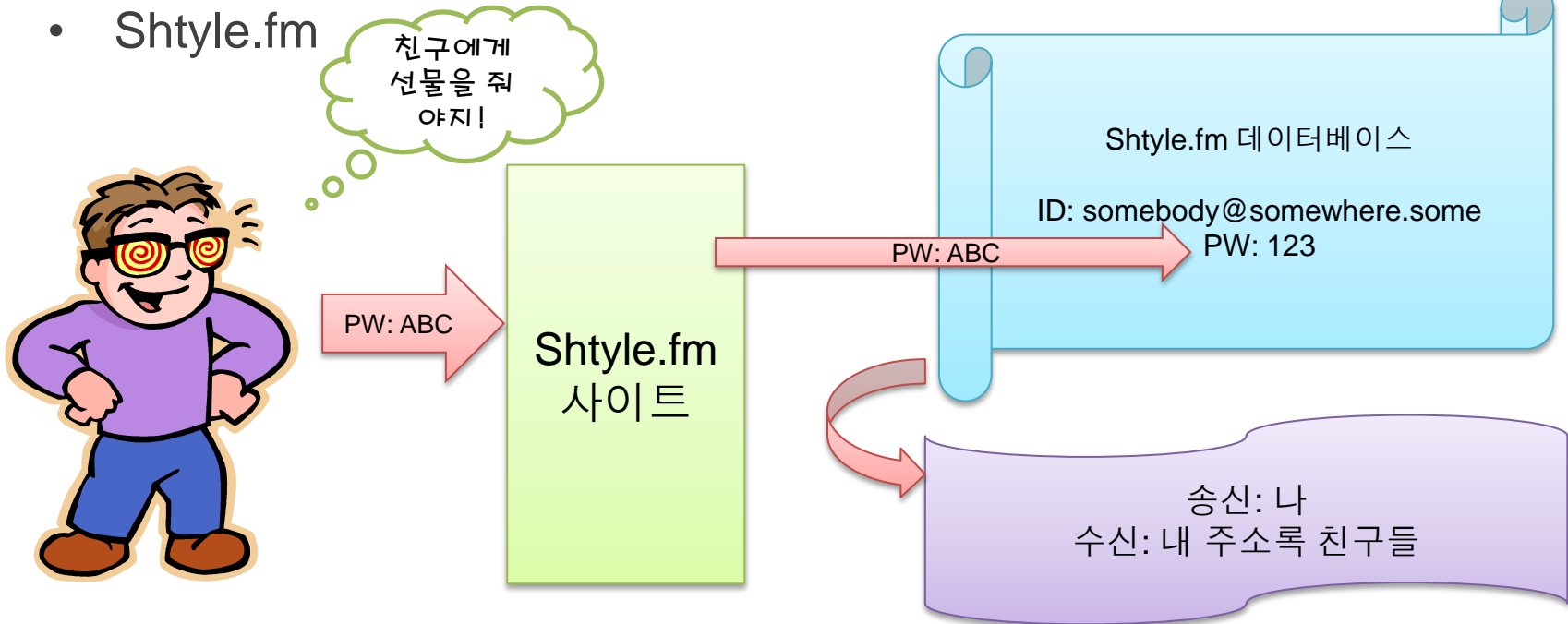
Choose a gift to send to your friends:



This is a fun way to exchange virtual gifts with your friends. Your email contacts will be notified of your Shtyle.fm profile on your behalf through email. When they register through the email they will become your friend on Shtyle and receive the gift you have selected on their profile.

# 스팸 메일에 당한 사례

- Shtyle.fm



- 가입자의 이메일 계정 **주소록을 조회**해서, 발신자는 가입자의 이메일 주소, 수신자는 주소록에서 얻어진 이메일 주소들로 하여 **자사 홍보 메일**을 보낸다.

# 스팸 방지를 위한 노력

- 사용자 입장
  - 이메일 주소는 지인들에게만 공개
  - 주소 일그러뜨리기(personNOS@PAMdomain.com)
  - 스팸에 반응 보이지 않기
    - “더 이상 스팸 보내지 마세요!”라고 반응하는 것은 “당신이 스팸 보낸 주소는 실제로 존재하는 주소입니다. 감사합니다.”라고 하는 것과 같다.
  - 외부용 주소를 사용하고, 실제 사용은 전달(forward)받아서 하기
  - 응징 및 복수(발신자 추적해 스팸 더 보내기, 발신자 컴퓨터 찾아서 괴롭히기, 스팸 광고하는 사이트 가서 악성 게시물 올리기)





# 스팸 방지를 위한 노력

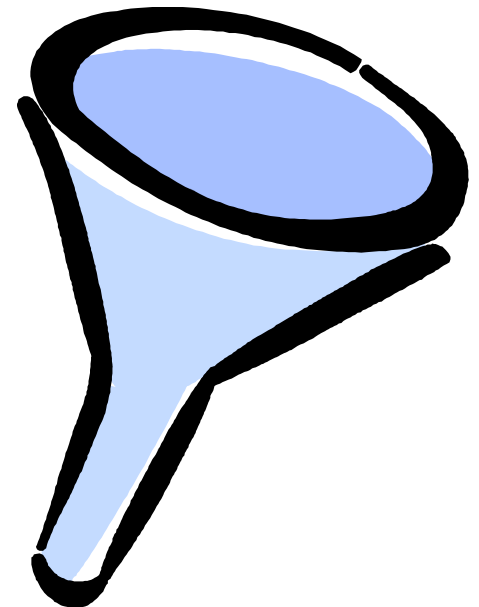
---

- 이메일 관리자 입장
  - 스팸 발신자가 없다고 검증된 이메일 서버만 취급
  - 발신 때마다 스팸 발신이 아닌지 검사(Captcha 등)
  - 스팸 메일의 검사 합계(checksum)를 수집하여, 걸러내기
  - RFC 표준 준수 여부 확인(스팸 메일은 보통 표준을 엄두에 두지 않음)
  - 스팸 메일 필터 설치 후 걸린 발신자 차단
  - 기계 학습 및 통계적 방법으로 걸러내기



# 기계 학습을 활용한 스팸 걸러내기

- 베이지안 스팸 거름법
- 복수 단어 인식 단위로의 확장
- 마르코비안 거름법



# 베이지안 스팸 거름법

(Bayesian Spam Filtering)

- 베이즈(Bayes)의 법칙

$$\Pr(A|B) = \frac{\Pr(B|A) \Pr(A)}{\Pr(B|A) \Pr(A) + \Pr(B|\neg A) \Pr(\neg A)}$$

- 베이지안 스팸 거름법의 기본 원리
  - 메일에 포함된 개별 단어들의 스팸성(spamcity, spamness)을 측정한다.
  - 메일에 포함된 단어들의 스팸성을 결합하여 메일 자체가 스팸일 가능성을 측정한다.

# 베이지안 스팸 걸름법

(Bayesian Spam Filtering)

- 메일에 포함된 개별 단어들의 스팸성(spamicity, spamness)을 측정
  - $S$ : 임의의 메일이 **스팸**일 사건
  - $H$ : 임의의 메일이 **햄**일 사건( $\text{스팸} \leftrightarrow \text{햄}$ ,  $H = \neg S$ )
  - $W$ : 임의의 **단어**가 주어질 사건
  - $\Pr(S|W)$ : 그 **단어**를 포함할 때, 그 메일이 **스팸**일 확률

$$\Pr(S|W) = \frac{\Pr(W|S) \Pr(S)}{\Pr(W|S) \Pr(S) + \Pr(W|H) \Pr(H)}$$

# 베이지안 스팸 거름법

(Bayesian Spam Filtering)

- 메일에 포함된 개별 단어들의 스팸성(spamcity, spamness)을 측정

$$\Pr(S|W) = \frac{\Pr(W|S) \Pr(S)}{\Pr(W|S) \Pr(S) + \Pr(W|H) \Pr(H)}$$

- 통계적으로 임의의 메일이 스팸일 확률은 80%이므로,  $\Pr(S) = 0.8$ 이고,  $\Pr(H) = 0.2$ 이다.
- 대부분 베이지안 스팸 감지 소프트웨어는 임의의 유입 메일이 햄이 아니고 스팸일 것이라고 예측할 근거를 갖지 못한다 가정하고,  $\Pr(S) = \Pr(H) = 0.5$ 라 설정하기도 한다.

$$\Pr(S|W) = \frac{\Pr(W|S)}{\Pr(W|S) + \Pr(W|H)}$$

# 베이지안 스팸 거름법

(Bayesian Spam Filtering)

- 메일에 포함된 단어들의 스팸성을 결합하여 메일 자체가 스팸일 가능성을 측정
  - 임의의 메일이  $W_1, W_2, \dots, W_N$ 의  $N$ 개의 단어들을 포함한다고 가정한다.
  - 임의의 메일에 단어가 등장하는 사건들은 독립 사건이라고 가정한다.

$$\begin{aligned}\Pr(S|W_1, W_2, \dots, W_N) &= \frac{\Pr(W_1, W_2, \dots, W_N|S) \Pr(S)}{\Pr(W_1, W_2, \dots, W_N|S) \Pr(S) + \Pr(W_1, W_2, \dots, W_N|H) \Pr(H)} \\ &= \frac{\Pr(W_1|S) \cdots \Pr(W_N|S) \Pr(S)}{\Pr(W_1|S) \cdots \Pr(W_N|S) \Pr(S) + \Pr(W_1|H) \cdots \Pr(W_N|H) \Pr(H)}\end{aligned}$$

- 베이즈의 법칙에 의해

$$\Pr(W_i|S) = \frac{\Pr(S|W_i) \Pr(W_i)}{\Pr(S)}$$

# 베이지안 스팸 거름법

(Bayesian Spam Filtering)

- 메일에 포함된 단어들의 스팸성을 결합하여 메일 자체가 스팸일 가능성을 측정
  - $\Pr(W_i|S)$ 를 본 식에 대입하여 정리한다.

$$\begin{aligned}\Pr(S|W_1, W_2, \dots, W_N) &= \frac{\Pr(W_1|S) \cdots \Pr(W_N|S) \Pr(S)}{\Pr(W_1|S) \cdots \Pr(W_N|S) \Pr(S) + \Pr(W_1|H) \cdots \Pr(W_N|H) \Pr(H)} \\ &= \frac{[\prod_{i=1}^N \Pr(S|W_i)] \Pr(S)^{1-N}}{[\prod_{i=1}^N \Pr(S|W_i)] \Pr(S)^{1-N} + [\prod_{i=1}^N \Pr(H|W_i)] \Pr(H)^{1-N}}\end{aligned}$$

# 베이지안 스팸 거름법

(Bayesian Spam Filtering)

- 메일에 포함된 단어들의 스팸성을 결합하여 메일 자체가 스팸일 가능성을 측정

–  $\Pr(S) = \Pr(H) = 0.5$ 라 설정하여 정리하면 다음의 결과를 얻는다.

$$\Pr(S|W_1, W_2, \dots, W_N) = \frac{[\prod_{i=1}^N \Pr(S|W_i)]}{[\prod_{i=1}^N \Pr(S|W_i)] + [\prod_{i=1}^N (1 - \Pr(S|W_i))]}$$

– 이 확률이 일정 한계치(threshold)를 넘으면, 스팸으로 간주한다.

$$\frac{\Pr(S|W_1, W_2, \dots, W_N)}{\Pr(H|W_1, W_2, \dots, W_N)} > \lambda \text{ 또는 } \Pr(S|W_1, W_2, \dots, W_N) > \frac{\lambda}{1+\lambda}$$



# 복수 단어 인식 단위로의 확장

## (Multiple-Word Feature)

- 베이지안 스팸 거름법은 **단어들의 이웃함**을 고려하지 않는다.
  - “대출 한도”와 “대출” “한도”의 차이
- 최대  **$k$ 개의 단어 순서열**을 인식의 단위로 간주하면 어떨까?
- 크기  $k$ 인 창을 움직이면서, 창의 첫 단어를 반드시 포함하되, 다른 단어들은 생략 가능하며, 단, 그 단어들의 순서가 유지되어야 한다.
- $k=3$ 이라고 할 때의 생성되는 일부 인식 단위들의 예
  - 빠르게 **알아보는 나의** 대출한도 조회 기록 ...

빠르게	알아보는	나의
사용	사용	사용
사용	사용	
사용		사용
사용		

빠르게 알아보는 나의  
빠르게 알아보는  
빠르게 나의  
빠르게

알아보는	나의	대출한도
사용	사용	사용
사용	사용	
사용		사용
사용		

알아보는 나의 대출한도  
알아보는 나의  
알아보는 대출한도  
알아보는

# 복수 단어 인식 단위로의 확장

## (Multiple-Word Feature)

- $k=3$ 이라고 할 때의 생성되는 일부 인식 단위들의 예
  - 빠르게 알아보는 나의 대출한도 조회 기록 ...

빠르게 알아보는 나의	알아보는 나의 대출한도	나의 대출한도 조회	대출한도 조회 기록
빠르게 알아보는	알아보는 나의	나의 대출한도	대출한도 조회
빠르게 나의	알아보는 대출한도	나의 조회	대출한도 기록
빠르게	알아보는	나의	대출한도

$$\Pr(S|F_1, F_2, \dots, F_N) = \frac{[\prod_{i=1}^N \Pr(S|F_i)]}{[\prod_{i=1}^N \Pr(S|F_i)] + [\prod_{i=1}^N (1 - \Pr(S|F_i))]}$$

- $F$ 는 인식 단위(feature)를 의미하고, 총  $N$ 개 있다고 가정한다.

# 마르코비안 거름법

(Markovian Filtering)

- 스팸 메일에 포함된  $k$ 개의 단어들을 **순서**와 **이웃함**을 유지한 채 **포함**하고 있는 경우가 많을 수록 스팸일 가능성이 높지 않을까?
  - “당일 바로 대출 5000”과 “당일 바로”의 차이
- 인식 단위의 길이**에 **지수적으로 증가**하는 **가중치**를 부여
- $k=5$ 이라고 할 때의, 각 인식 단위에 주어지는 가중치의 예
  - 당일 바로 대출 최대 5000 ...

당일 바로 대출 최대 5000	256	당일 바로 5000	16
당일 대출 최대 5000	64	당일 바로 최대	16
당일 바로 최대 5000	64	당일 바로 대출	16
당일 바로 대출 5000	64	당일 5000	4
당일 바로 대출 최대	64	당일 최대	4
당일 최대 5000	16	당일 대출	4
당일 대출 5000	16	당일 바로	4
당일 대출 최대	16	당일	1

# 마르코비안 거름법

(Markovian Filtering)

- 스팸성 측정 방식(CRM114의 구현)

$$\Pr(S|F) = 0.5 + \frac{(NS(F) - NH(F)) \cdot Weight(F)}{C_1(NS(F) + NH(F) + C_2) \cdot Weight_{max}}$$

- $NS(F)$ : F가 스팸인 경우의 수
- $NH(F)$ : F가 햄인(스팸이 아닌) 경우의 수
- $Weight(F)$ : F의 가중치
- $Weight_{max}$ : 가능한 가중치 중 최대( $= 2^{2k}$ )
- CRM114에서  $C_1 = 16$ ,  $C_2 = 1$ 이다.

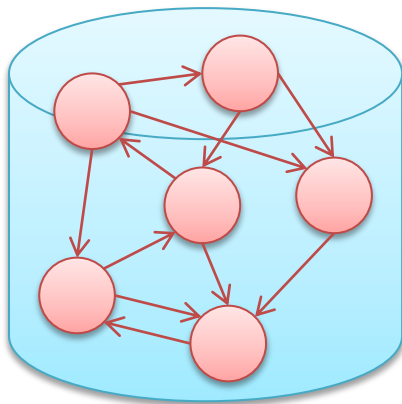
# 은닉 마르코프 모형과 난독화 문제 해법

## (Deobfuscation with Hidden Markov Model)

- 난독화 문제 (obfuscation)

기존 단어	난독화 단어
refinance	r.efina.nce, r-efin-ance, re xe finance
mortgage	mort gage, mo>rtglage, mor;tg2age
viagra	v*1agra, v-i-a-g-r-a, v1@gra, vjaggra
unsubscribe	u.n sabcjbe, un susc ribe

- 은닉 마르코프 모형 (Hidden Markov Model, HMM)



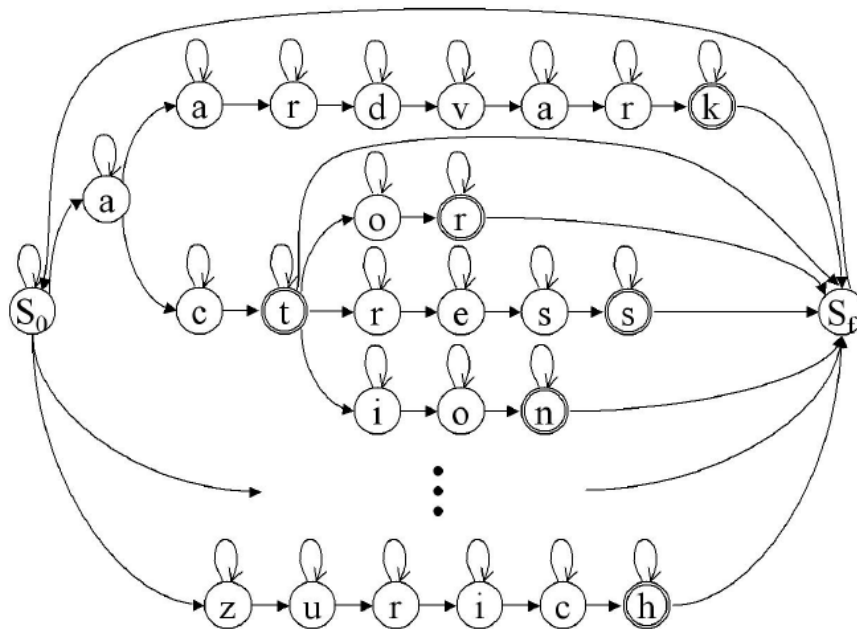
초기 상태 확률 벡터 ( $\Pr(X_0 = i)$ )  
 상태 전이 확률 행렬 ( $\Pr(X_{t+1} = j | X_t = i)$ )  
 관찰 발생 확률 행렬 ( $\Pr(O_t = j | X_t = i)$ )

비터비 (Viterbi) 알고리즘으로 가장 가능성이 높은 상태의  
 순서열을 구할 수 있음.

# 은닉 마르코프 모형과 난독화 문제 해법

## (Deobfuscation with Hidden Markov Model)

- 사전식 수형 구조(lexicon tree) 은닉 마르코프 모형을 생성
  - 표준 영어 사전(45475 단어)로 구성
  - 시작 상태( $S_0$ )와 종료 상태( $S_f$ )를 포함
  - 다른 상태들은 사전에 등장하는 단어들의 철자에 해당



# 은닉 마르코프 모형과 난독화 문제 해법

## (Deobfuscation with Hidden Markov Model)

- 초기 상태 확률 벡터, 상태 이전 확률 행렬, 관찰 발생 확률 벡터의 정의

$$P_0(X_{t+1}|X_t) = \begin{cases} 1 - \eta & \text{if } X_{t+1} = X_t \\ \eta f_{X_{t+1}}/f_{X_t} & \text{if } X_{t+1} \text{ is } X_t\text{'s child} \\ \eta h_{X_t}/f_{X_t} & \text{if } X_{t+1} = S_f, X_t \text{ is a terminal node} \\ 0 & \text{otherwise.} \end{cases}$$

$\xrightarrow{\text{자가 이전 제어 인자}}$   $\xrightarrow{(S_0 \dots X) \text{를 접두어로 갖는 단어의 총 빈도수}}$   
 $\xrightarrow{\text{단어 } (S_0 \dots X) \text{의 총 빈도수}}$

$Q$ 는 이전하면서 공백문자를 생성할 확률,  $P$ 는 이전하면서 비공백문자를 생성할 확률

$$Q(X_{t+1}|X_t) = \epsilon P_0(X_{t+1}|X_t)$$

$$P(X_{t+1}|X_t) = (1 - \epsilon)P_0(X_{t+1}|X_t).$$

모형 자체의 또 다른인자

$$P(O_{t+1}|X_t \rightarrow X_{t+1}) = \begin{cases} (1 - \tau)P_{\text{emit}}(O_{t+1}|X_{t+1}) + \tau P_{\text{random}}(O_{t+1}) & \text{if } X_{t+1} = X_t \neq S_0 \\ P_{\text{random}}(O_{t+1}) & \text{if } X_{t+1} = X_t = S_0 \\ P_{\text{emit}}(O_{t+1}|X_{t+1}) & \text{otherwise,} \end{cases}$$

상태가 나타내는 문자를 나타내기 위한 관찰된 문자의 확률 분포

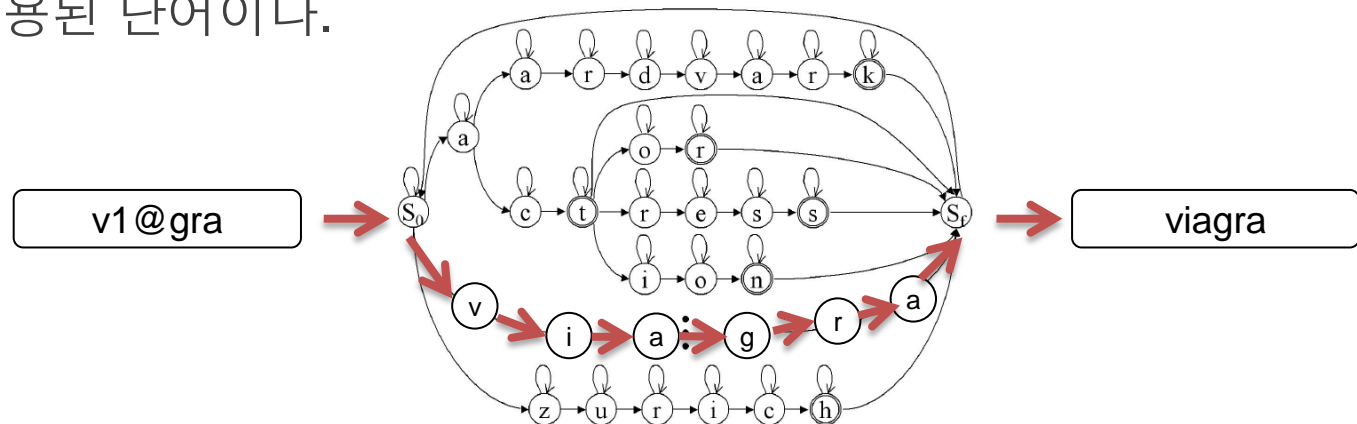
난독화를 위해 삽입되는 무의미한 문자의 확률분포

훈련 데이터(training data)의 로그 가능성(likelihood)를 최대화하도록  $\eta, \epsilon, \tau$ 를 학습한다.

# 은닉 마르코프 모형과 난독화 문제 해법

(Deobfuscation with Hidden Markov Model)

- 학습된 모형에 **난독화된 단어를 입력**으로 넣고 **비터비 알고리즘**을 수행하면, 가장 가능성이 높은 상태의 **순서열이 얻어지고**, 이것이 바로 비난독화가 적용된 단어이다.



- 상태 전이 확률 행렬의 희소(sparse) 표현을 적용하고, Jelinek의 1999년 저서에서 소개된 방법(**beam search**)을 사용하면, 알고리즘을 더욱 빠르게 할 수 있다.
  - F. Jelinek, *Statistical Methods for Speech Recognition*. MIT Press, 1999.



# 정리

---

- **베이지안 거름법**: 단어의 이웃 관계를 무시하고 단어의 출현이 독립적인 사건이라는 가정 하에, 메일에 등장하는 각 단어의 스팸성을 구해 종합함으로써 스팸 메일일 확률을 추정한다.
- **복수 단어 인식 단위로의 확장**: 단어의 이웃 관계를 제한적으로 감안하고 단어의 출현의 종속성을 다소 반영한 인식 단위를 이용하여, 스팸 메일일 확률을 추정한다.
- **마르코비안 거름법**: 단어의 이웃 관계 및 출현의 종속성이 잘 반영된 인식 단위일 수록 높은 가중치를 제공하여, 스팸 메일일 확률을 추정한다.
- **은닉 마르코프 모형과 난독화 문제**: 은닉 마르코프 모형을 이용하여 난독화된 단어를 원래 단어로 해독하고, 이를 바탕으로 거름법을 수행하면 더 좋은 결과를 기대할 수 있을 것이다.

# 질의 응답

---

# 참고자료 및 참고 문헌

---

- 참고자료 및 참고문헌
  - Wikipedia - Spam (Monty Python) ([http://en.wikipedia.org/wiki/Spam \(Monty Python\)\)](http://en.wikipedia.org/wiki/Spam_(Monty_Python))))
  - Spam Experts (<http://www.spamexperts.com/spam-experts/news-archive/article/motivation-for-spammers.html>)
  - Consumer Fraud Reporting ([http://www.consumerfraudreporting.org/spam\\_costs.php](http://www.consumerfraudreporting.org/spam_costs.php))
  - Wikipedia - Bayesian spam filtering ([http://en.wikipedia.org/wiki/Bayesian spam filtering](http://en.wikipedia.org/wiki/Bayesian_spam_filtering))
  - Ben O'connor, *Markovian Spam Filtering*, 2007.
  - Gary Robinson's Rants (<http://radio-weblogs.com/0101454/stories/2002/09/16/spamDetection.html>)
  - Jonathan A. Zdziarski, *Ending Spam: Bayesian Content Filtering and the Art of Statistical Language Classification*, No Starch Press, 2005.
  - Raju Shrestha and Yaping Lin, *Improved Bayesian Spam Filtering Based on Co-weighted Multi-area Information*, Advances in Knowledge Discovery and Data Mining, 2005.
  - William S. Yeraunus, *Sparse Binary Polynomial Hashing and the CRM114 Discriminator* (slides)
  - Shalendra Chhabra, William S. Yeraunus, and Christian Siefkes, *Spam Filtering using a Markov Random Field Model with Variable Weighting Schemas*, ICDM'04, 2004.
  - William S. Yeraunus, *The Spam-Filtering Accuracy Plateau at 99.9 percent Accuracy and How to Get Past It*, MIT Spam Conference, 2004
  - William S. Yeraunus, et al., *A Unified Model of Spam Filtration*, MIT Spam Conference, 2005.
  - Honglak Lee and Adrew Y. Ng, *Spam deobfuscation using a hidden Markov model*, Conference on Email and Anti-Spam, 2005.
  - Seunghak Lee, Iryoung Jeong, and Seungjin Choi, *Dynamically Weighted Hidden Markov Model for Spam Deobfuscation*, Proceedings of the 20th International Joint Conference on Artificial Intelligence, 2007.