N-gram Language Model

국민대학교 인공지능학부
강 승 식

개요

- N-gram 언어 모델
  - “A probability distribution over sequences of words”
  - 활용 분야: 음성인식, 기계번역, 딥러닝 자연어처리(BERT, GPT) 등

- Smoothing 기법
  - Back-off smoothing
  - Linear interpolation
  - Maximum entropy

N-gram이란?

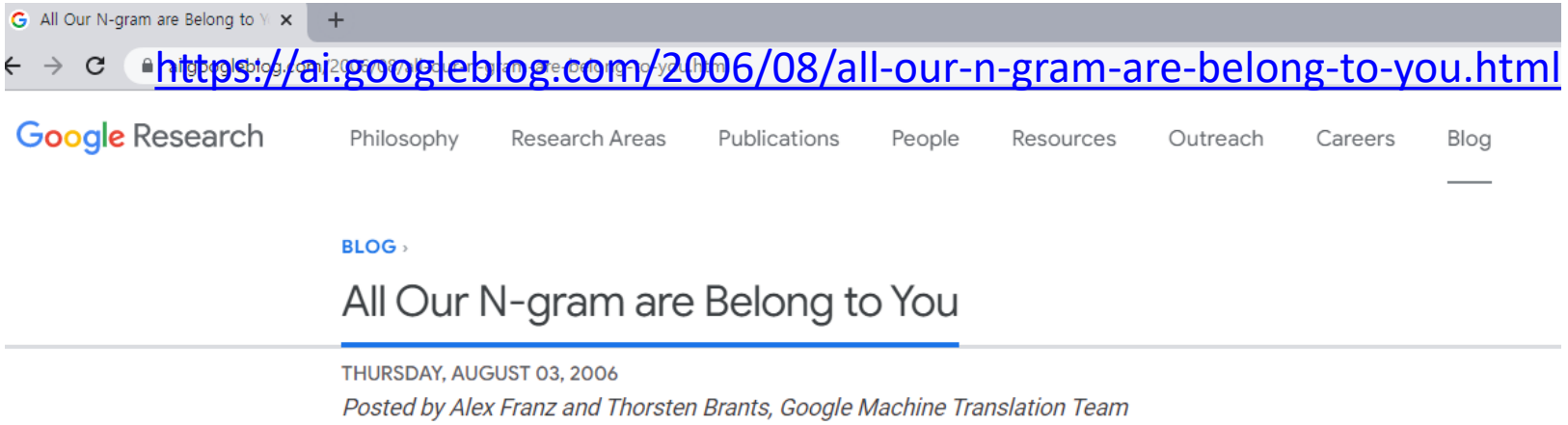
https://en.wikipedia.org/wiki/N-gram

- Contiguous sequence of *n* items from a given text or speech.
  - The items can be phonemes, syllables, letters, words or base pairs according to the application.
  - N-grams are collected from a text or speech corpus.

Figure 1 *n*-gram examples from various disciplines

Field	Unit	Sample sequence	1-gram sequence	2-gram sequence	3-gram sequence
Vernacular name			unigram	bigram	trigram
Order of resulting Markov model			0	1	2
Protein sequencing	amino acid	... Cys-Gly-Leu-Ser-Trp ...	..., Cys, Gly, Leu, Ser, Trp, ...	..., Cys-Gly, Gly-Leu, Leu-Ser, Ser-Trp, ...	..., Cys-Gly-Leu, Gly-Leu-Ser, Leu-Ser-Trp, ...
DNA sequencing	base pair	...AGCTTCGA...	..., A, G, C, T, T, C, G, A, ...	..., AG, GC, CT, TT, TC, CG, GA, ...	..., AGC, GCT, CTT, TTC, TCG, CGA, ...
Computational linguistics	character	...to_be_or_not_to_be...	..., t, o, _, b, e, _, o, r, _, n, o, t, _, t, o, _, b, e, ...	..., to, o_, _b, be, e_, _o, or, r_, _n, no, ot, t_, _t, to, o_, _b, be, ...	..., to_, o_b, _be, be_, e_o, _or, or_, r_n, _no, not, ot_, t_t, _to, to_, o_b, _be, ...
Computational linguistics	word	... to be or not to be ...	..., to, be, or, not, to, be, ...	..., to be, be or, or not, not to, to be, ...	..., to be or, be or not, or not to, not to be, ...

예) Google n-gram corpus



File sizes: approx. 24 GB gzip'ed text files

Number of tokens: 1,024,908,267,229
Number of sentences: 95,119,665,584
Number of unigrams: 13,588,391
Number of bigrams: 314,843,401
Number of trigrams: 977,069,902
Number of fourgrams: 1,313,818,354
Number of fivegrams: 1,176,470,663

- 3-grams
  - ceramics collectables collectibles (55)
  - ceramics collectables fine (130)
  - ceramics collected by (52)
  - ceramics collectible pottery (50)
  - ceramics collectibles cooking (45)
- 4-grams
  - serve as the incoming (92)
  - serve as the incubator (99)
  - serve as the independent (794)
  - serve as the index (223)
  - serve as the indication (72)
  - serve as the indicator (120)

N-gram 모델

- An n-gram model models sequences, using the statistical properties of n-grams.

- Shannon's information theory
  - Question: given a sequence of letters (for example, the sequence “for ex”)
    - What is the likelihood of the next letter?
  - Answer: argmax (P(‘a’ | ‘for ex’), P(‘b’ | ‘for ex’), …, P(‘z’ | ‘for ex’))
    - a = 0.4, b = 0.00001, c = 0, ... → P(a) + P(b) + … + P(z) = 1.0
    - From training data, derive a probability distribution for the next letter given a history of size *n*.
- Issue: OOV(out-of-vocabulary) words

배경 1. 음성인식 후처리 문제

- Speech recognition (STT: Speech-To-Text)
  - A sentence is uttered in a sequence of words. (left-to-right)
- "I ate a cherry"
  - I ate a cherry.
  - I eight a cherry.
  - I aid a cherry.
  - I hate a cherry.
  - I wait a cherry.
  - Eye ate a cherry.
  - Eye eight a cherry.
  - Eye aid a cherry.
  - Eye hate a cherry.
  - Eye wait a cherry.

배경 2. Next Word Prediction

- 첫 1~2어절이 주어졌을 때 다음 어절로 올 가능성이 가장 높은 단어들을 순서대로 생성

- Shannon Game
  - Claude E. Shannon. “Prediction and Entropy of Printed English”, *Bell System Technical Journal*, 1951.
  - Predict the next word, given (*n-1)* previous words

“large green _____”
tree? mountain? frog? car?
“swallowed the large green _____”
pill? broccoli?

Please turn off your cell _____.
Your program does not _____.

Next Word Prediction

- Given the observed training data
- Develop a model (probability distribution) to predict future events

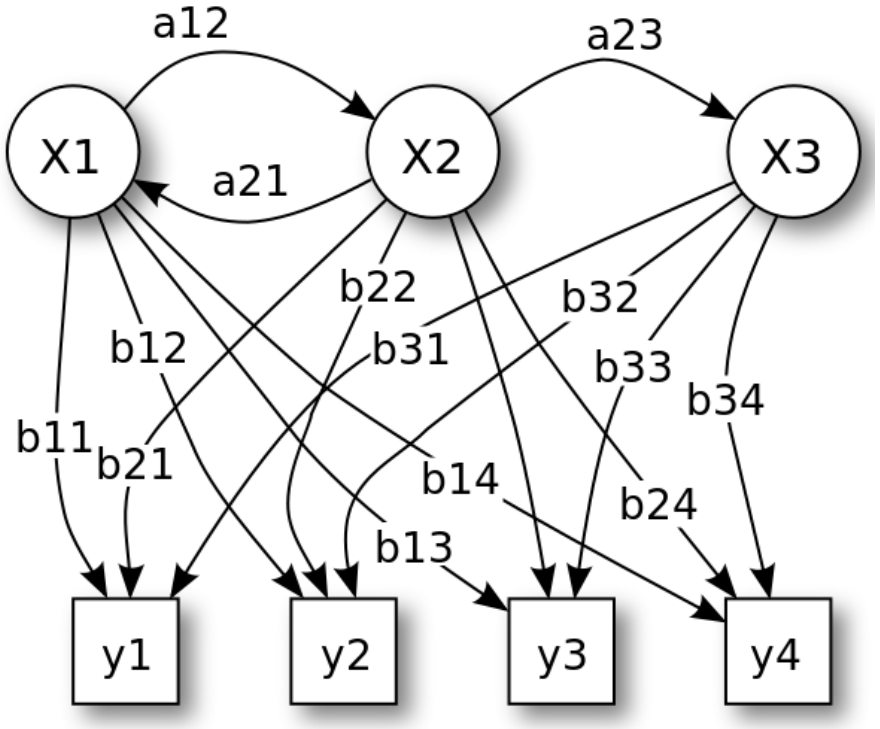
- Example:
  - Corpus: five Jane Austen novels
    - N = 617,091 words
    - V = 14,585 unique words
  - Task: predict the next word of the trigram

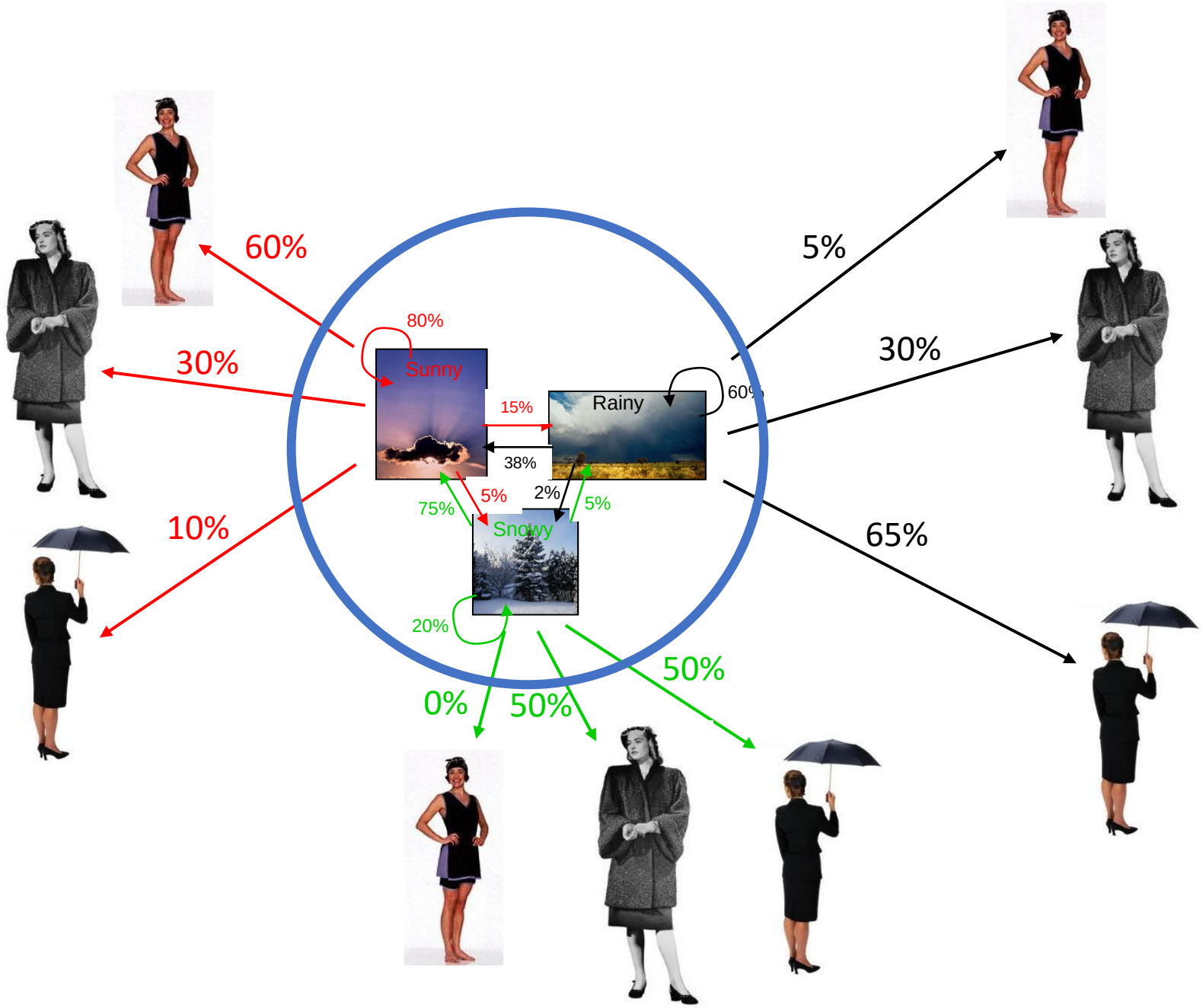
    “inferior to _____” (the, her, cherries, Maria)

- From a NY Times story...
  - Stocks ...
  - Stocks plunged this ….
  - Stocks plunged this morning, despite a cut in interest rates
  - Stocks plunged this morning, despite a cut in interest rates by the Federal Reserve, as Wall ...
  - Stocks plunged this morning, despite a cut in interest rates by the Federal Reserve, as Wall Street began
  - Stocks plunged this morning, despite a cut in interest rates by the Federal Reserve, as Wall Street began trading for the first time since last …
  - Stocks plunged this morning, despite a cut in interest rates by the Federal Reserve, as Wall Street began trading for the first time since last Tuesday's terrorist attacks.

HMM 모델의 활용: 음성인식, POS-tagging 등

- Markov Assumption
  - A word is affected only by its "prior local context“.
  - Examine short sequences of words
    - How likely is each sequence? (관측 확률)
- Observation sequence에 대한 state sequence 예측
  - 관측값 분포를 학습데이터를 이용하여 추정
  - 예) http://ko.wikipedia.org
    - 상태: x, 출력/관측: y
    - 상태전이 확률: aij, 출력 확률: bij
- Training
  - EM algorithm(Baum-Welch 또는 forward-backward algorithm)
- Viterbi algorithm으로 구현





1. Compute P(O| HMM)

$$P(O) = P(O_{coat}, O_{coat}, O_{umbrella}, \ldots, O_{umbrella})$$

2. Given output sequence O,
   compute most likely state sequence
   q* = argmax_q P(q|O, HMM)

3. Training (aka learning):
   HMM* = argmax_HMM P(O|HMM)

Corpora: 대규모 텍스트 말뭉치

- Corpora are online collections of text and speech
  - Brown corpus, LOB corpus, BNC corpus
  - Wall Street Journal, AP newswire
  - DARPA/NIST text/speech corpora
  - Boston radio speech corpus

  - Wikipedia text dump
    - 영어 등 -- https://dumps.wikimedia.org/backup-index.html
    - Kowiki -- https://dumps.wikimedia.org/kowiki/20230501/
  - Webpages, blog, SNS corpus

- 한국어 텍스트 말뭉치
  - http://nlp.kookmin.ac.kr/kcc/
  - https://cafe.naver.com/nlpk/60
  - https://cafe.naver.com/nlpkang/22

대용량 한국어 텍스트 데이터(원시 말뭉치)

1) 한글 위키피디아 말뭉치
   https://dumps.wikimedia.org/kowiki/
   http://nlp.kookmin.ac.kr/down/data/ko_wiki_text_EUCKR.zip (KS완성형)
   http://nlp.kookmin.ac.kr/kcc/word2vec/ko_wiki_text.zip
2) 세종말뭉치
   http://nlp.kookmin.ac.kr/down/data/RAW2169-CORE.zip (KS완성형)
3) KMU KCC 한국어 뉴스 말뭉치
   http://nlp.kookmin.ac.kr/kcc
   http://nlp.kookmin.ac.kr/kcc/word2vec
4) KT set -- 주로 신문기사
   http://nlp.kookmin.ac.kr/down/data/KT_93_95.zip (KS완성형)
5) 한국어 대용량 학습 데이터셋
   https://www.aihub.or.kr/ (AI Hub)

한국어 문서분류 말뭉치

6) 문서분류 학습말뭉치(고려대)
   http://nlp.kookmin.ac.kr/down/data/koreaWebdoc-TC.zip
   http://nlp.kookmin.ac.kr/down/data/koreaWebdoc-TC-ver2.zip
7) 문서분류 학습말뭉치(서강대)
   http://nlp.kookmin.ac.kr/down/data/sogang-TC.zip

기타 뉴스기사, 특허 문서 등 (KS완성형)

8) IT 분야 관련 뉴스기사
   http://nlp.kookmin.ac.kr/down/data/ITnews.zip (600여개)
   http://nlp.kookmin.ac.kr/down/data/ITnews623_sim383.zip (ITnews 623개 + 유사문서 383개)
9) 유사문서 383개 뉴스기사 -- http://nlp.kookmin.ac.kr/down/data/SimNews383.zip
10) 이규태 칼럼 -- http://nlp.kookmin.ac.kr/down/data/gtlee.zip
11) 특허문서 1229개 -- http://nlp.kookmin.ac.kr/down/data/patent-1229.zip
12) 신문기사 -- 중복제거 필요!
   http://nlp.kookmin.ac.kr/down/data/NewsML-sskang.zip

13) 네이버 영화평 데이터셋 -- https://github.com/e9t/nsmc
   자동띄어쓰기 -- http://nlp.kookmin.ac.kr/corpus/NaverSentiMovieCorpus_ASP.zip
   NSMC extended -- http://nlp.kookmin.ac.kr/corpus/NSMC_extended_282K.zip

14) Dacon 자연어 기반 기후기술분류 AI 경진대회 데이터셋
   http://nlp.kookmin.ac.kr/down/2021_Dacon_NLP_weather_AI_challenge.zip
   <참고> 파이썬 소스 -- 파일 로딩 테스트 dacon.py (첨부파일)

15) 일베 댓글 데이터셋 -- http://nlp.kookmin.ac.kr/down/data/ilbe_comments.zip
16) 한국어 혐오성 표현 데이터셋 및 kaggle competition (아래 github 또는 첨부파일)
   https://github.com/kocohub/korean-hate-speech/

Kaggle competition

Gender-bias detection -- http://www.kaggle.com/c/korean-gender-bias-detection
Bias detection -- http://www.kaggle.com/c/korean-bias-detection
Hate speech detection -- http://www.kaggle.com/c/korean-hate-speech-detection

17) 감정분석 데이터셋 (영한 번역) -- http://nlp.kookmin.ac.kr/corpus/emotion_korTran.zip
18) SentimentTreebank (영한 번역) -- http://nlp.kookmin.ac.kr/corpus/Sentiment_Treebank_en_ko.zip

19) 한영 법률문장 70만개
   http://nlp.kookmin.ac.kr/down/data/KE-kor-cp949(788136).zip
   http://nlp.kookmin.ac.kr/down/data/KE-eng-sentences(716566).zip

문장 생성 확률

- Chain rule of sentence probability

$P(s) = P(w_1 w_2 w_3 \ldots w_n)$

$$P(w_1 \ldots w_n) = P(w_1)P(w_2|w_1) \ldots P(w_i|w_1 \ldots w_{i-1})$$
$$= \prod_{i=1}^{n} P(w_i|w_1 \ldots w_{i-1})$$

P(your cell phone is good) =
P(your) × P(cell | your) × P(phone | your cell)
× P(is | your cell phone)
× P(good | your cell phone is)

N-gram 언어 모델

- Estimate probability of each word given prior context.
  - P(phone | Please turn off your cell)

$$= \frac{freq(Please\ turn\ off\ you\ cell\ phone)}{freq(Please\ turn\ off\ you\ cell)}$$

- ngram 확률
  - Unigram: P(phone)
  - Bigram: P(phone | cell)
  - Trigram: P(phone | your cell)

N-gram 언어 모델

- Bigram 모델

$$P(w_n|w_{1:n-1}) \approx P(w_n|w_{n-1}) \quad \Longrightarrow \quad P(w_{1:n}) \approx \prod_{k=1}^{n} P(w_k|w_{k-1})$$

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$$

- 문장의 시작, 끝 기호 추가

  <s> Please turn off your cell phone </s>

https://web.stanford.edu/~jurafsky/slp3/3.pdf

- Unigram approximation

$$P_{uni}(w_1 w_2 \ldots w_n) = \prod_i P(w_i)$$

- Bigram approximation → $P(w_n|w_{1:n-1}) \approx P(w_n|w_{n-1})$

$$P_{bi}(w_1 w_2 \ldots w_n) = \prod_i P(w_i|w_{i-1})$$

- Trigram approximation

$$P_{tri}(w_1 w_2 \ldots w_n) = \prod_i P(w_i|w_{i-2} w_{i-1})$$

- N-gram approximation → $P(w_n|w_{1:n-1}) \approx P(w_n|w_{n-N+1:n-1})$

$$P(w_1^n) = \prod_{k=1}^{n} P(w_k \mid w_{k-N+1}^{k-1})$$

$$P(w_n \mid w_{n-N+1}^{n-1}) = \frac{C(w_{n-N+1}^{n-1} w_n)}{C(w_{n-N+1}^{n-1})}$$

[참고] N-gram 언어 모델

Unigram model: $P(w_1)P(w_2)P(w_3) \dots P(w_n)$

Bigram model:

$$P(w_1)P(w_2|w_1)P(w_3|w_2) \dots P(w_n|w_{n-1})$$

Trigram model:

$$P(w_1)P(w_2|w_1)P(w_3|w_2,w_1) \dots P(w_n|w_{n-1}w_{n-2})$$

N-gram model:

$$P(w_1)P(w_2|w_1) \dots P(w_n|w_{n-1}w_{n-2} \dots w_{n-N})$$

언어모델의 활용 분야

- 음성 인식, 문자 인식, 기계 번역, 철자 교정 등
  - 후보 문장들 중에서 가장 그럴듯한(생성 확률이 높은) 문장 선택

    (Most likely or probable sentence is selected.)

- 딥러닝 자연어 처리 분야
  - seq2seq 모델: 기계번역, chatbot 등
  - 임베딩(문서표현) 모델: ELMo, BERT 등
  - 자연어 생성 모델: GPT

Example. Sentences Generated from WSJ

unigram: Months the my and issue of year foreign new exchange's september were recession exchange new endorsed a acquire to six executives

bigram: Last December through the way to preserve the Hudson corporation N. B. E. C. Taylor would seem to complete the major central planners one point five percent of U. S. E. has already old M. X. corporation of living on information such as more frequently fishing to keep her

trigram: They also point to ninety nine point six billion dollars from two hundred four oh six three percent of the rates of interest stores as Mexico and Brazil on market conditions

Smoothing 기법

- Unseen n-grams: OOV 문제 (학습말뭉치에 미출현 어휘)

- Smoothing
  - 확률값이 0 이 되지 않도록 하는 방법

  - Laplace(add-one) smoothing
  - Back-off smoothing
  - Class-based smoothing
  - Linear interpolation

Laplace(add-one) Smoothing

- For unigrams:
  - Add 1 to every word (type) count to get an adjusted count c^*
  - Normalize by N (#tokens) + V (#types)

  - Original unigram probability

$$P(w_i)=\frac{c_i}{N}$$

  - New unigram probability

$$P_{LP}(w_i)=\frac{c_i+1}{N+V}$$

Unigram Smoothing Example

Tiny Corpus, V=4; N=20

$$P_{LP}(w_i)=\frac{c_i+1}{N+V}$$

Word	True Ct	Unigram Prob	New Ct	Adjusted Prob
eat	10	.5	11	.46
British	4	.2	5	.21
food	6	.3	7	.29
happily	0	.0	1	.04
	20	1.0	~20	1.0

- For bigrams:
  - Original

$$P(w_n|w_{n-1}) = \frac{c(w_n|w_{n-1})}{c(w_{n-1})}$$

  - New

$$P(w_n|w_{n-1}) = \frac{c(w_n|w_{n-1})+1}{c(w_{n-1})+V}$$

Back-off smoothing

- 학습 데이터에 출현하지 않는 N-gram 확률을 (N-1)-gram 확률로부터 추정하는 방법

- Trigram 확률의 예

$$\hat{P}(w_i \mid w_{i-2}w_{i-1}) = \begin{cases} P(w_i \mid w_{i-2}w_{i-1}), & \text{if } C(w_{i-2}w_{i-1}w_i) > 0 \\ \alpha_1 P(w_i \mid w_{i-1}), & \text{if } C(w_{i-2}w_{i-1}w_i) = 0 \\ & \text{and } C(w_{i-1}w_i) > 0 \\ \alpha_2 P(w_i), & \text{otherwise.} \end{cases}$$

Katz's back-off model

$$P_{bo}(w_i \mid w_{i-n+1} \cdots w_{i-1})$$

$$= \begin{cases} d_{w_{i-n+1} \cdots w_i} \dfrac{C(w_{i-n+1} \cdots w_{i-1} w_i)}{C(w_{i-n+1} \cdots w_{i-1})} & \text{if } C(w_{i-n+1} \cdots w_i) > k \\ \alpha_{w_{i-n+1} \cdots w_{i-1}} P_{bo}(w_i \mid w_{i-n+2} \cdots w_{i-1}) & \text{otherwise} \end{cases}$$

- *d* : discounting 계수
- *k* : 최저빈도, usually 0
- α : 확률의 총합이 1이 되도록 하기 위한 정규화 계수

Class-based smoothing

- For certain types of n-grams, back off to the count of its syntactic class or semantic category

- E.g., Count ProperNouns in place of names (e.g., Obama)

Linear Interpolation

- 여러 개의 확률에 가중계수를 곱하여 평균한 전체의 확률을 추정하는 것

- Ex) trigram 모델의 확률을 linear interpolation 법에 의해 구할 경우

$$P(w_i|w_{i-2}^{i-1}) = \lambda_3 f(w_i|w_{i-2}^{i-1}) + \lambda_2 f(w_i|w_{i-1}) + \lambda_1 f(w_i) + \lambda_0$$

$$\lambda_0 + \lambda_1 + \lambda_2 + \lambda_3 = 1, \quad \lambda_i > 0$$

Maximum Entropy

- 여러 개의 정보원으로부터의 정보를 작성하는 일반적인 방법
  - Ex) bigram 모델과 trigram 모델을 계산할 경우

$$P(w_i|h) = \prod_{v} \mu_{v,w_i}^{f_{v,w_i}(h)} \prod_{u,v} \mu_{u,v,w_i}^{f_{u,v,w_i}(h)}$$

  - h : w_i의 이전 단어열 $w_1 \dots w_{i-1}$
  - μ_{u,w_i} , μ_{u,v,w_i} : 추정할 수 있는 파라미터
  - $f_{vw_i(h)}$, $f_{uvw_i(h)}$: 제약함수

$$f_{vw}(h,w_i) = \begin{cases} 1 & w = w_i\, h \text{ 최후의 단어가 } v \text{ 인 경우} \\ 0 & otherwise \end{cases}$$

$$f_{uvw}(h,w_i) = \begin{cases} 1 & w = w_i\, h \text{ 최후의 2단어가 } u,v \text{ 인 경우} \\ 0 & otherwise \end{cases}$$