

# { WEEK2. Word Representation }

# Word Representation

1

## 1-hot representation

- 각 단어를 하나의 object로 바라본다.
- 단어 간의 관계 추론 불가능

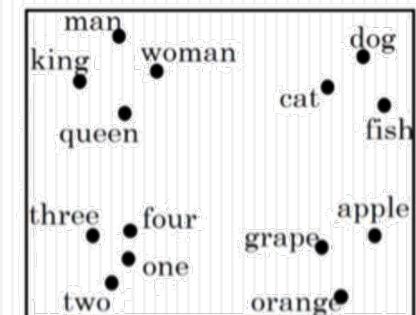
2

## Featurized representation

- Row에는 feature
- Column에는 vocab

	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	-1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03	0.02	0.70	0.69	0.03	-0.02
Food	0.09	0.01	0.02	0.01	0.95	0.97

$e_{4914}$



t-SNE

## Analogies

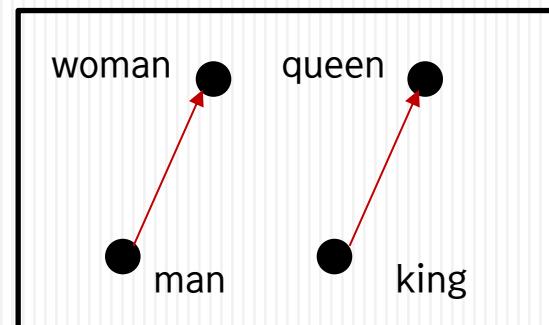
**Man → Woman vs King → ?**

	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	-1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03	0.02	0.70	0.69	0.03	-0.02
Food	0.09	0.01	0.02	0.01	0.95	0.97

$$e_{man} - e_{woman} \approx \begin{bmatrix} -2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad e_{man} - e_{woman} \approx e_{king} - e_w$$

find word w:  
 $\text{argmax sim}(e_w, e_{king} - e_{man} + e_{woman})$

$$e_{king} - e_{queen} \approx \begin{bmatrix} -2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



## Embedding Matrix

$E$	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	-1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03	0.02	0.70	0.69	0.03	-0.02
Food	0.09	0.01	0.02	0.01	0.95	0.97

10,000 vocab  
300 features

$$e_j = E \cdot O_j$$

Embedding for word j      One-hot vector  
↓  
Embedding matrix

## Word2Vec : Skip grams

---

I want a glass of orange juice to go along with my cereal.

$$O_c \rightarrow E \rightarrow e_c \rightarrow softmax \rightarrow \hat{y}$$

$$p(t|c) = \frac{e^{\theta_t^T e_c}}{\sum_{j=1}^{10,000} e^{\theta_j^T e_c}}$$

$$\mathcal{L}(\hat{y}, y) = - \sum_{i=1}^{10,000} y_i \log \hat{y}_i$$

## Negative Sampling

---

I want a glass of orange juice to go along with my cereal.

<u>context</u>	<u>word</u>	<u>target?</u>
orange	juice	1
orange	king	0
orange	book	0
orange	the	0
orange	of	0

$$p(t|c) = \frac{e^{\theta_t^T e_c}}{\sum_{j=1}^{10,000} e^{\theta_j^T e_c}}$$

연산량 감소!

어떻게 negative sample 선택?

$$p(w_i) = \frac{f(w_i)^{3/4}}{\sum_{j=1}^{10,000} f(w_i)^{3/4}}$$

I want a glass of orange juice to go along with my cereal.

$$X_{ij} = \# \text{ times } j \text{ appears in context of } i$$

$$\text{minimize} \sum_{i=1}^{10,000} \sum_{j=1}^{10,000} f(X_{ij})(\Theta_i^T e_j + b_i + b'_j - \log X_{ij})^2$$

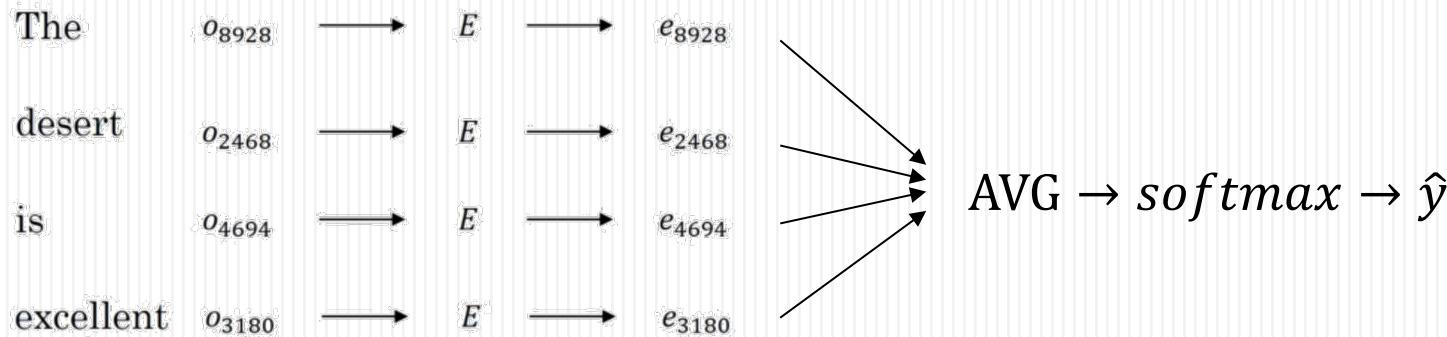


Weighting term

- 1) 한번도 등장하지 않은 경우에 0으로 설정
- 2) 지나치게 빈도가 높거나 낮은 단어로 인해서  $X_{ij}$  값이 특정 값 이상으로 되는 것을 방지하는 역할

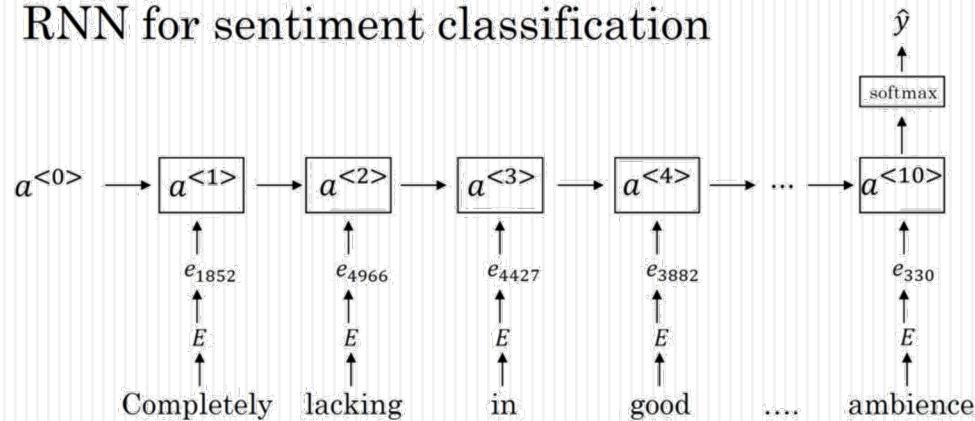
# Sentiment Classification

The dessert is excellent  
8928 2468 4694 3180



**Completely lacking in good taste, good service, and good ambience**

RNN for sentiment classification



# Debiasing word embeddings

---

많은 결정에 AI가 사용되므로, 단어 임베딩에서 성별이나 나이, 인종 등의 편견을 없애는 것은 매우 중요하다.

## 1. bias의 direction을 구한다.

만약 성별의 direction을 구한다면, 성별을 나타낼 수 있는 단어들의 차이를 구해서 구할 수 있는데, 남성성의 단어와 여성성의 단어의 차이를 구해서 평균으로 그 방향을 결정할 수 있다.

## 2. Neutralize(중성화) 작업

bias가 없어야 하는 단어들에 대해서 bias 요소를 제거해야하는데, project를 통해서 각 단어 벡터의 bias direction 요소를 제거한다. doctor나 babysitter등의 단어의 성별 bias를 제거하는 것이다.

## 3. Equalize pairs 작업

boy-girl / grandfather-grandmother과 같은 단어는 각 단어가 성별 요소가 있기 때문에, 이러한 단어들이 bias direction을 기준으로 같은 거리에 있도록 한다. 즉, 각 성별 요소가 있는 단어들은 bias direction과의 거리의 차이가 동일하도록 만들어주는 것이다.

감사합니다.