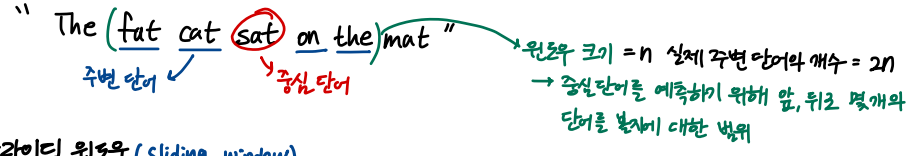
Background

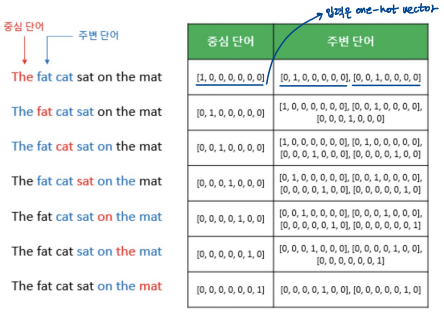
- 원-핫 벡터의 한계
  → 단어 벡터 간 유의미한 유사도를 계산할 수 없음
- 단어 벡터 간 유사도를 반영할 수 있도록 단어 의미를 수치화할 수 있는 방법 필요 =) 임베딩
- 희소표현은 고차원에 각 차원이 분리된 표현방법이고,
  분산표현은 저차원에 단어의 의미를 여러 차원에다 분산하는 표현방법 =) 분산표현을 사용해서 단어 벡터 간 유의미한 유사도 계산
  ↳ 대표적 방법이 Word2vec
- W2V 으로 무엇을 할 수 있을까?
  (한국 - 서울 + 도쿄 = 일본
  박찬호 - 야구 + 축구 = 흐나우드
  → 위와 같은 연산이 가능한 이유는 각 단어 벡터가 단어 벡터 간 유사도를 반영한 값을 가지고 있기 때문

W2V 학습방식

- CBow
  → 주변에 있는 단어들을 입력으로 중간에 있는 단어들을 예측하는 방법
  → 메커니즘

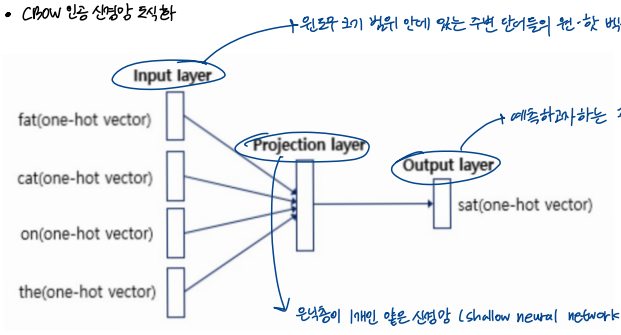


- 슬라이딩 윈도우 (Sliding Window)



→ 윈도우를 옆으로 움직여서 주변 단어와 중심단어의 선택을 변경하며 학습 데이터 셋을 만드는 방법

- CBOW 인공 신경망 모식화



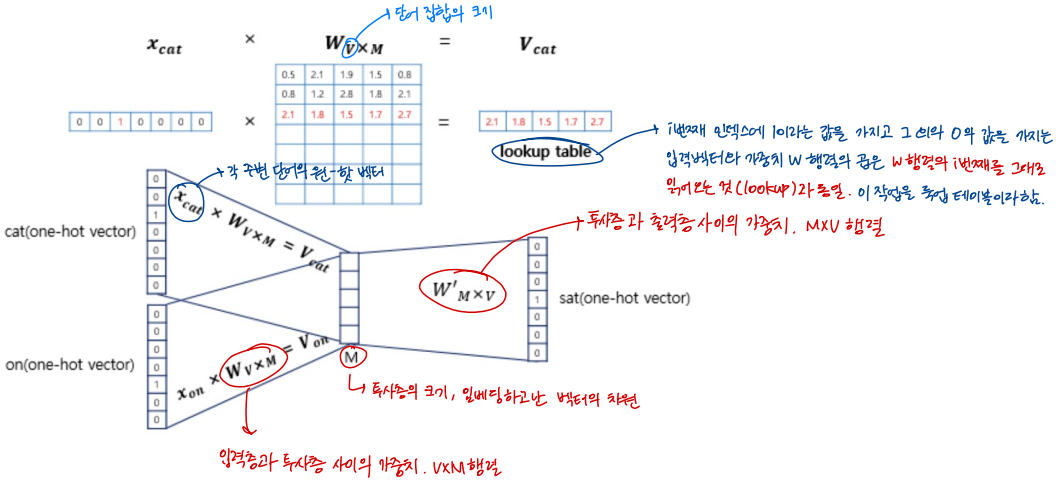
→ 윈도우 크기 범위 안에 있는 주변 단어들의 원-핫 벡터가 입력으로 들어감

→ 예측하고자 하는 중앙 단어의 원-핫 벡터가 레이블로 필요

→ 은닉층이 1개인 얕은 신경망 (shallow neural network)

→ W2V 은닉층은 활성화 함수가 존재하지 않고,
룩업 테이블이라는 연산을 담당하는 층으로 투사층(projection layer) 라고 부름.

- 인공 신경망 학습 및 동작 메커니즘



→ i번째 인덱스에 1이라는 값을 가지고 그 외의 0의 값을 가지는 입력벡터와 가중치 W 행렬의 곱은 W 행렬의 i번째 행을 그대로 읽어오는 것(lookup)과 같음. 이 작업을 룩업 테이블이라 함.

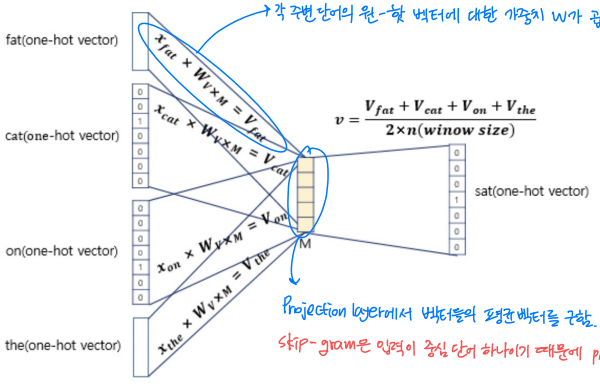
→ 투사층과 출력층 사이의 가중치, MXV 행렬

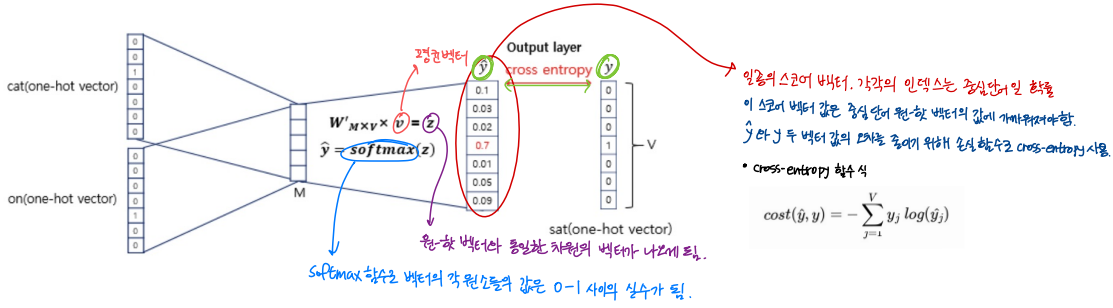
→ 투사층의 크기, 임베딩하고자 하는 벡터의 차원

→ 입력층과 투사층 사이의 가중치, VXM 행렬

→ 인공 신경망 훈련 전, 가중치 행렬 W와 W'는 랜덤 값을 가지게 되고,
주변 단어로 중심 단어를 더 정확히 맞추기 위해 계속해서 W와 W'을 학습해가는 구조.

→ lookup 해온 W의 각 행 벡터가 W2V 학습 후, 각 단어의 M차원의 임베딩 벡터로 간주됨.



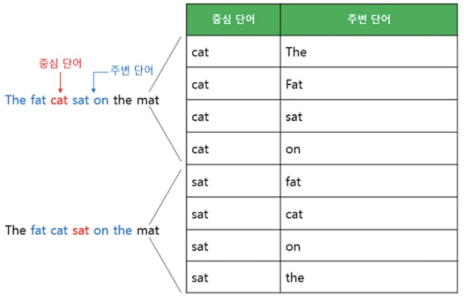


→ 역전파를 수행하면 W와 W'가 학습되고
M차원의 크기를 갖는 W의 행렬의 행을 각 단어의 임베딩 벡터로 사용하거나
W와 W' 행렬 두 가지 모두를 가지고 임베딩 벡터를 사용할 수 있음

- **Skip-gram**

  → 중심 단어에서 주변 단어를 예측함

  → 중심 단어 하나에서 주변 단어를 예측하므로 투사층에서 벡터의 평균을 구하지 않음



중심 단어	주변 단어
cat	The
cat	Fat
cat	sat
cat	on
sat	fat
sat	cat
sat	on
sat	the