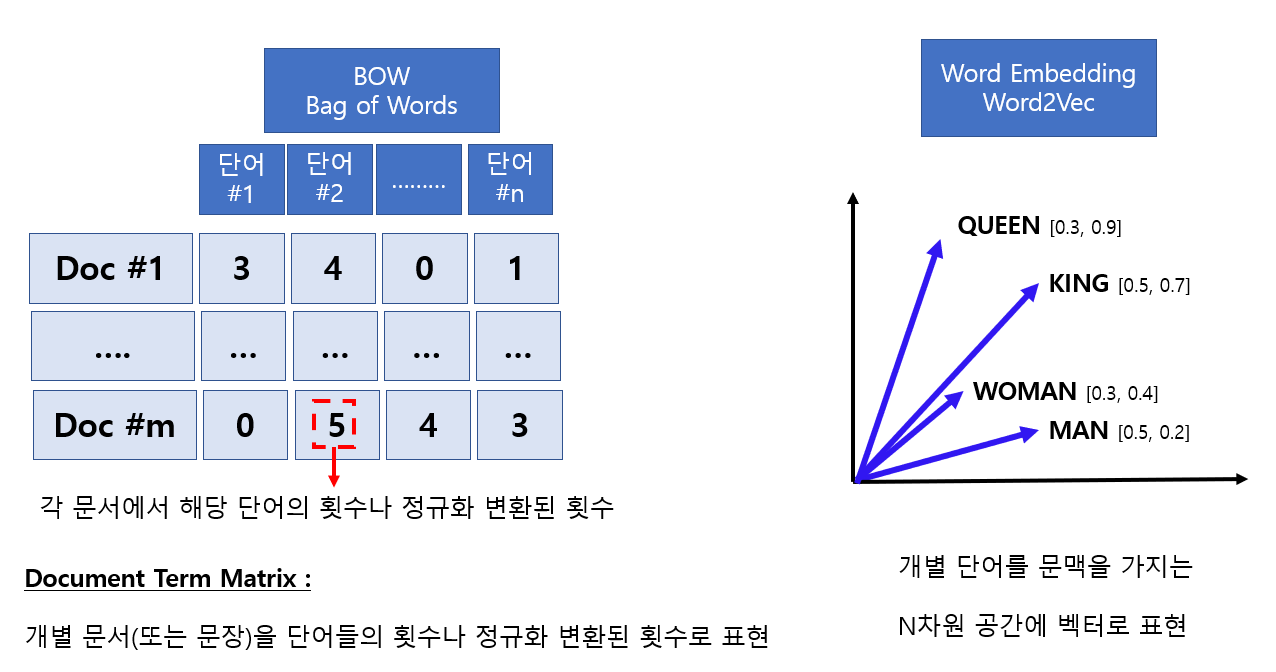
* **텍스트의 피처 벡터화 유형**



* **Bag of Words – BOW**

Bag of Words 모델은 문서가 가지는 모든 단어(Words)를 **문맥이나 순서를 무시**하고 일괄적으로 **단어에 대해 빈도 값**을 부여해 피처 값을 추출하는 모델

* 문서 내 모든 단어를 한꺼번에 봉투(Bag)안에 넣어 흔들어 섞는다는 의미로 Bag Of Words(BOW)모델이라고 함.
* **Bag of Words 구조**

1. 각 문서나 문장에 있는 모든 단어에서 **중복을 제거 후** 각 단어를 컬럼으로 지정

* 각 단어에 고유의 index 부여

1. 문장에서 **해당 단어가 나타나는 횟수(occurrence)**를 각 단어 (단어 index)에 기재

테이블이(가) 표시된 사진

자동 생성된 설명

* **BOW 장단점**

**장점 :** 쉽고 빠른 구축

예상보다 문서의 특징을 잘 나타내어 전통적으로 여러 분야에서 활용도가 높음

**단점 :** 문맥의미 (Semantic Context) 반영 문제

희소 행렬 문제: 단어 빈도수가 0인 컬럼이 너무 많이 존재..

* **BOW 피처 벡터화 유형**

**단순 카운트 기반의 벡터화 :** 단어 피처의 값을 각 문서에서 해당 단어가 나타나는 횟수.

**COUNT를 부여하는 경우를 카운트 벡터화**라고 함

카운트 벡터화에서는 **카운트 값이 높을수록 중요한 단어로 인식**됨.

**TF-IDF 벡터화 : 카운트만 부여**하면 해당 문서의 특징을 나타내기 보다는

**자주 사용되는 단어의 특징만 나타나게 됨.**

🡪 for 이러한 문제를 보완, **TF-IDF(Term Frequency Inverse Document Frequency) 벡터화** 사용

**TF-IDF**는 개별 문서에서 **자주 나타나는 단어에 높은 가중치를 주되,**

**모든 문서에서 전반적으로 자주 나타나는 단어**에 대해서는 **페널티를** 주는 방식으로 값을 부여함

* **TF-IDF(Term Frequency Inverse Document Frequency)**

**특정 단어가** 다른 문서에서는 나타나지 않고 **특정 문서에서만 자주 사용된다면,**

그 단어는 **해당 문서를 잘 특징짓는 중요 단어일 가능성이 높음**

**특정단어**가 **매우 많은 여러 문서에서 빈번히** 나타난다면

해당 단어는 **개별 문서를 특징짓는 정보 로서의 의미 상실**

**TF(Term Frequency) :** 문서에서 해당 단어가 얼마나 나왔는지를 나타내는 지표

**DF(Document Frequency) :** 해당 단어가 몇 개의 문서에서 나타났는지를 나타내는 지표

**IDF(Inverse Document Frequency) :** DF의 역수로서 전체 문서수/DF

**텍스트이(가) 표시된 사진

자동 생성된 설명**

* **사이킷런 CountVectorizer 초기화 파라미터**

**max\_df :** 전체 문서에 걸쳐서 너무 높은 빈도수를 가지는 단어 피처를 제외하기 위한 파라미터

너무 높은 빈도수를 가지는 단어는 stopword와 비슷한 문법적인 특성일 가능성이 높음.

* 따라서 이를 제거하기 위해 사용

max\_df = 100: 100개 이하로 나타나는 단어만 피처로 추출

max\_df = 0.95: 빈도수 95%까지의 단어만 피처로 추출, 나머지 상위 5%는 피처로추출 **X**

**min\_df :** 전체 문서에 걸쳐서 너무 낮은 빈도수를 가지는 단어 피처를 제외하기 위한 파라미터

min\_df보다 적은 빈도수의 단어는 크게 중요하지 않거나 garbage성 단어일 확률이 높음

min\_df = 2: 2번 이하의 단어는 피처로 추출 **X**

min\_df = 0.02: 하위 2%이하의 빈도수를 가지는 단어는 피처로 추출 **X**

**max\_features :** 피처의 개수를 제한

**stop\_words :** ‘english’로 지정하면 영어의 stop word로 지정된 단어는 추출에서 제외

**ngram\_range :** Bag of Words **모델의 단어순서를 어느 정도 보강하기** 위한 n-gram 범위 설정

튜플 형태 (범위 최솟값, 범위 최댓값)

(1, 1) 🡪 토큰화된 단어를 1개씩 피처로 추출

(1,2) 🡪 토큰화된 단어를 1개씩, 그리고 순서대로 2개씩 묶어서 피처로 추출

**analyzer :** 피처 추출을 수행한 단위 지정 **디폴트는 ‘word’**

word가 아니라 **character의 특정 범위를 피처로 만드는 특정한 경우** 사용됨

**token\_pattern :** 토큰화를 수행하는 정규 표현식 패턴 지정 **디폴트는 ‘\b\w\w+\b’**

**어근 추출** 시 외부함수를 사용할 경우 해당 외부함수를 **token\_patteren의 인자로 사용**

**lower\_case :** 모든 문자를 소문자로 변경할 것인지를 설정 **디폴트는 True**

* **CountVectorizer를 이용한 피처 벡터화**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

* **희소 행렬**

**테이블이(가) 표시된 사진

자동 생성된 설명**

* **희소 행렬의 저장 변환 형식**

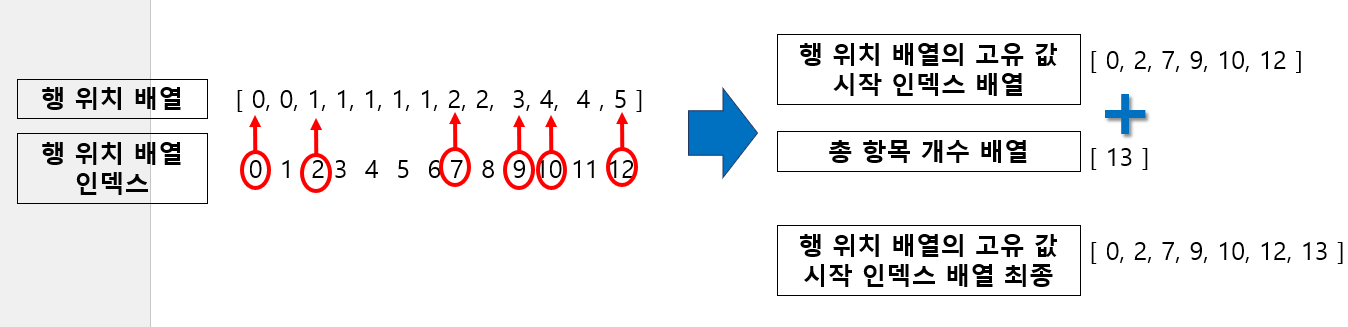
**COO 형식 :** Coordinate(좌표) 방식을 의미하며 0이 아닌 데이터만 별도의 배열(Array)에 저장하고 그 데이터를 가리키는 행과 열의 위치를 별도의 배열로 저장하는 방식

텍스트이(가) 표시된 사진

자동 생성된 설명

**CSR 형식 :** COO형식이 위치 배열 값을 중복적으로 가지는 문제를 해결한 방식

일반적으로 CSR형식이 COO보다 많이 사용됨



🡪 희소 행렬을 COO, CSR형식으로 변환하기 위해 **Scipy 의 coo\_matrix() csr\_matrix() 함수** 이용