

# **What is KeyBERT?**

**Keyword Extraction using KeyBERT**

AI 05 Hyungjin Kim

# Text Summarization

## Approaches

Extractive Approaches

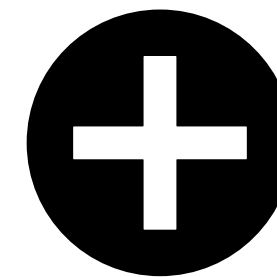
Abstractive Approaches

# KeyBERT

What it is?

BERT- embeddings

extract N-gram words



Cosine similarity

calculate similarity btw words

sub-phrases

that are the most similar to the document itself

# KeyBERT

## How we could use it?

```
pip install keybert
```

```
from keybert import KeyBERT
```

```
doc = """
```

```
    Supervised learning is the machine learning task of learning a  
    maps an input to an output based on example input-output pairs.  
    function from labeled training data consisting of a set of train  
    In supervised learning, each example is a pair consisting of an  
    (typically a vector) and a desired output value (also called the  
    A supervised learning algorithm analyzes the training data and  
    which can be used for mapping new examples. An optimal scenario  
    algorithm to correctly determine the class labels for unseen in  
    the learning algorithm to generalize from the training data to  
    'reasonable' way (see inductive bias).
```

```
    """
```

```
kw_model = KeyBERT()
```

```
keywords = kw_model.extract_keywords(doc)
```

# KeyBERT

## extract keywords

keyphrase\_ngram\_range

```
>>> kw_model.extract_keywords(doc, keyphrase_ngram_range=(1, 1), stop_words=None)
[('learning', 0.4604),
 ('algorithm', 0.4556),
 ('training', 0.4487),
 ('class', 0.4086),
 ('mapping', 0.3700)]
```

```
>>> kw_model.extract_keywords(doc, keyphrase_ngram_range=(1, 2), stop_words=None)
[('learning algorithm', 0.6978),
 ('machine learning', 0.6305),
 ('supervised learning', 0.5985),
 ('algorithm analyzes', 0.5860),
 ('learning function', 0.5850)]
```

```
keywords = kw_model.extract_keywords(doc, highlight=True)
```

highlight

Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labeled training data consisting of a set of training examples. In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal). A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances. This requires the learning algorithm to generalize from the training data to unseen situations in a 'reasonable' way (see inductive bias).

**How to get diverse keywords?**

# KeyBERT

Max sum similarity

```
>>> kw_model.extract_keywords(doc, keyphrase_ngram_range=(3, 3), stop_words='english',  
                               use_maxsum=True, nr_candidates=20, top_n=5)  
[('set training examples', 0.7504),  
 ('generalize training data', 0.7727),  
 ('requires learning algorithm', 0.5050),  
 ('supervised learning algorithm', 0.3779),  
 ('learning machine learning', 0.2891)]
```



# KeyBERT

## Maximal Marginal Relevance

```
>>> kw_model.extract_keywords(doc, keyphrase_ngram_range=(3, 3), stop_words='english',  
                               use_mmr=True, diversity=0.2)  
[('algorithm generalize training', 0.7727),  
 ('supervised learning algorithm', 0.7502),  
 ('learning machine learning', 0.7577),  
 ('learning algorithm analyzes', 0.7587),  
 ('learning algorithm generalize', 0.7514)]
```

low diversity

```
>>> kw_model.extract_keywords(doc, keyphrase_ngram_range=(3, 3), stop_words='english',  
                               use_mmr=True, diversity=0.7)  
[('algorithm generalize training', 0.7727),  
 ('labels unseen instances', 0.1649),  
 ('new examples optimal', 0.4185),  
 ('determine class labels', 0.4774),  
 ('supervised learning algorithm', 0.7502)]
```

high diversity



# Reference

- <https://github.com/MaartenGr/KeyBERT>
- <https://wikidocs.net/159467>

**Thank you**  
**Happy Lunar New Year**