토픽모델링

- 목적: 문서 내에서 논의된 주요 주제를 자동으로 식별합니다.
- 방법: LDA(Latent Dirichlet Allocation)와 같은 주제 모델링 기법을 사용하여 텍스트에서 발견되는 주요 주제를 분석합니다. 이를 통해 회사와 관련된 다양한 논의의 축을 파악할 수 있습니다.

In [1]:
```
!pip install pyLDAvis
```

```
Collecting pyLDAvis
  Downloading pyLDAvis-3.4.1-py3-none-any.whl.metadata (4.2 kB)
Requirement already satisfied: numpy>=1.24.2 in /usr/local/lib/python3.11/dist-pa
ckages (from pyLDAvis) (2.0.2)
Requirement already satisfied: scipy in /usr/local/lib/python3.11/dist-packages
(from pyLDAvis) (1.14.1)
Requirement already satisfied: pandas>=2.0.0 in /usr/local/lib/python3.11/dist-pa
ckages (from pyLDAvis) (2.2.2)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.11/dist-pa
ckages (from pyLDAvis) (1.4.2)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.11/dist-packages
(from pyLDAvis) (3.1.6)
Requirement already satisfied: numexpr in /usr/local/lib/python3.11/dist-packages
(from pyLDAvis) (2.10.2)
Collecting funcy (from pyLDAvis)
  Downloading funcy-2.0-py2.py3-none-any.whl.metadata (5.9 kB)
Requirement already satisfied: scikit-learn>=1.0.0 in /usr/local/lib/python3.11/d
ist-packages (from pyLDAvis) (1.6.1)
Collecting gensim (from pyLDAvis)
  Downloading gensim-4.3.3-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_6
4.whl.metadata (8.1 kB)
Requirement already satisfied: setuptools in /usr/local/lib/python3.11/dist-packa
ges (from pyLDAvis) (75.2.0)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.1
1/dist-packages (from pandas>=2.0.0->pyLDAvis) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-pac
kages (from pandas>=2.0.0->pyLDAvis) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-p
ackages (from pandas>=2.0.0->pyLDAvis) (2025.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.11/
dist-packages (from scikit-learn>=1.0.0->pyLDAvis) (3.6.0)
Collecting numpy>=1.24.2 (from pyLDAvis)
  Downloading numpy-1.26.4-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_6
4.whl.metadata (61 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 61.0/61.0 kB 1.3 MB/s eta 0:00:00
Collecting scipy (from pyLDAvis)
  Downloading scipy-1.13.1-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_6
4.whl.metadata (60 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 60.6/60.6 kB 2.0 MB/s eta 0:00:00
Requirement already satisfied: smart-open>=1.8.1 in /usr/local/lib/python3.11/dis
t-packages (from gensim->pyLDAvis) (7.1.0)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-
packages (from jinja2->pyLDAvis) (3.0.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-package
s (from python-dateutil>=2.8.2->pandas>=2.0.0->pyLDAvis) (1.17.0)
Requirement already satisfied: wrapt in /usr/local/lib/python3.11/dist-packages
(from smart-open>=1.8.1->gensim->pyLDAvis) (1.17.2)
Downloading pyLDAvis-3.4.1-py3-none-any.whl (2.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.6/2.6 MB 20.8 MB/s eta 0:00:00
Downloading funcy-2.0-py2.py3-none-any.whl (30 kB)
Downloading gensim-4.3.3-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.w
hl (26.7 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 26.7/26.7 MB 37.0 MB/s eta 0:00:00
Downloading numpy-1.26.4-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.w
hl (18.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 18.3/18.3 MB 39.0 MB/s eta 0:00:00
Downloading scipy-1.13.1-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.w
hl (38.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 38.6/38.6 MB 8.7 MB/s eta 0:00:00
Installing collected packages: funcy, numpy, scipy, gensim, pyLDAvis
```

```
  Attempting uninstall: numpy
    Found existing installation: numpy 2.0.2
    Uninstalling numpy-2.0.2:
      Successfully uninstalled numpy-2.0.2
  Attempting uninstall: scipy
    Found existing installation: scipy 1.14.1
    Uninstalling scipy-1.14.1:
      Successfully uninstalled scipy-1.14.1
Successfully installed funcy-2.0 gensim-4.3.3 numpy-1.26.4 pyLDAvis-3.4.1 scipy-
1.13.1
```

In [5]:
```python
## 처음 한번 다운로드 필요
import nltk
nltk.download('punkt')
nltk.download('punkt_tab')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Package punkt_tab is already up-to-date!
```

Out[5]: True

In [3]:
```python
import os
import re
from gensim import corpora
from gensim.models.ldamodel import LdaModel
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
import pyLDAvis.gensim as gensimvis  # 수정된 부분
import pyLDAvis
import matplotlib.pyplot as plt

# 수동으로 정의한 한국어 불용어 리스트
korean_stopwords = {
    '의', '가', '이', '은', '들', '는', '좀', '잘', '걍', '과', '도', '를', '으로
    '자', '에', '와', '한', '하다', '에서', '것', '및', '위해', '그', '되다'
}

# 불용어 추가 (분석에 불필요한 단어 추가)
additional_stopwords = {'강점', '약점', '경쟁사'}
korean_stopwords.update(additional_stopwords)

# 텍스트 파일 경로
file_paths = [
    "01_다른경쟁사와간단비교.txt",
    "02_기업리서치관련정리.txt",
    "03_생성AI분석.txt"
]

# 파일 내용을 하나로 결합
combined_text = ""

for file_path in file_paths:
    with open(file_path, 'r', encoding='utf-8') as file:
        combined_text += file.read() + "\n"

# 텍스트 전처리 및 토큰화
def preprocess(text):
    # 소문자 변환, 특수 문자 제거, 토큰화
```

```
    text = re.sub(r'[^\w\s]', '', text.lower())
    tokens = word_tokenize(text)
    tokens = [word for word in tokens if word not in korean_stopwords and len(wo
    return tokens

# 전처리된 문서 리스트 생성
documents = preprocess(combined_text)

# 단어 사전 생성
dictionary = corpora.Dictionary([documents])

# 코퍼스 생성 (문서를 BOW(Bag of Words)로 변환)
corpus = [dictionary.doc2bow(documents)]

# LDA 모델 생성
lda_model = LdaModel(corpus, num_topics=3, id2word=dictionary, passes=15)

# pyLDAvis를 이용한 시각화
vis_data = gensimvis.prepare(lda_model, corpus, dictionary)
pyLDAvis.display(vis_data)

# 필요 시, HTML 파일로 저장
pyLDAvis.save_html(vis_data, 'lda_visualization.html')
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
<ipython-input-3-9ed4f0182825> in <cell line: 0>()
      1 import os
      2 import re
----> 3 from gensim import corpora
      4 from gensim.models.ldamodel import LdaModel
      5 from nltk.tokenize import word_tokenize

/usr/local/lib/python3.11/dist-packages/gensim/__init__.py in <module>
      9 import logging
     10
---> 11 from gensim import parsing, corpora, matutils, interfaces, models, simila
rities, utils  # noqa:F401
     12
     13

/usr/local/lib/python3.11/dist-packages/gensim/parsing/__init__.py in <module>
      2
      3 from .porter import PorterStemmer  # noqa:F401
----> 4 from .preprocessing import (  # noqa:F401
      5     preprocess_documents,
      6     preprocess_string,

/usr/local/lib/python3.11/dist-packages/gensim/parsing/preprocessing.py in <modul
e>
     24 import glob
     25
---> 26 from gensim import utils
     27 from gensim.parsing.porter import PorterStemmer
     28

/usr/local/lib/python3.11/dist-packages/gensim/utils.py in <module>
     33
     34 import numpy as np
---> 35 import scipy.sparse
     36 from smart_open import open
     37

/usr/local/lib/python3.11/dist-packages/scipy/sparse/__init__.py in <module>
    292 import warnings as _warnings
    293
--> 294 from ._base import *
    295 from ._csr import *
    296 from ._csc import *

/usr/local/lib/python3.11/dist-packages/scipy/sparse/_base.py in <module>
      3
      4 import numpy as np
----> 5 from scipy._lib._util import VisibleDeprecationWarning
      6
      7 from ._sputils import (asmatrix, check_reshape_kwargs, check_shape,

/usr/local/lib/python3.11/dist-packages/scipy/_lib/_util.py in <module>
     16
     17 import numpy as np
---> 18 from scipy._lib._array_api import array_namespace
     19
     20
```

```
/usr/local/lib/python3.11/dist-packages/scipy/_lib/_array_api.py in <module>
     15
     16 from scipy._lib import array_api_compat
---> 17 from scipy._lib.array_api_compat import (
     18     is_array_api_obj,
     19     size,

/usr/local/lib/python3.11/dist-packages/scipy/_lib/array_api_compat/numpy/__init_
_.py in <module>
----> 1 from numpy import *
      2
      3 # from numpy import * doesn't overwrite these builtin names
      4 from numpy import abs, max, min, round
      5

/usr/local/lib/python3.11/dist-packages/numpy/__init__.py in __getattr__(attr)
    362         try:
    363             x = ones(2, dtype=float32)
--> 364             if not abs(x.dot(x) - float32(2.0)) < 1e-5:
    365                 raise AssertionError()
    366         except AssertionError:

ModuleNotFoundError: No module named 'numpy.rec'

--------------------------------------------------------------------------
NOTE: If your import is failing due to a missing package, you can
manually install dependencies using either !pip or !apt.

To view examples of installing some common dependencies, click the
"Open Examples" button below.
--------------------------------------------------------------------------
```

In [5]:
```
# 25/04/13 설치 후, 적용을 위해 런타임 세션 다시 시작
!pip install --upgrade --force-reinstall gensim
```

```
Collecting gensim
  Using cached gensim-4.3.3-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_6
4.whl.metadata (8.1 kB)
Collecting numpy<2.0,>=1.18.5 (from gensim)
  Using cached numpy-1.26.4-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_6
4.whl.metadata (61 kB)
Collecting scipy<1.14.0,>=1.7.0 (from gensim)
  Using cached scipy-1.13.1-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_6
4.whl.metadata (60 kB)
Collecting smart-open>=1.8.1 (from gensim)
  Using cached smart_open-7.1.0-py3-none-any.whl.metadata (24 kB)
Collecting wrapt (from smart-open>=1.8.1->gensim)
  Using cached wrapt-1.17.2-cp311-cp311-manylinux_2_5_x86_64.manylinux1_x86_64.ma
nylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (6.4 kB)
Using cached gensim-4.3.3-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.
whl (26.7 MB)
Using cached numpy-1.26.4-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.
whl (18.3 MB)
Using cached scipy-1.13.1-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.
whl (38.6 MB)
Using cached smart_open-7.1.0-py3-none-any.whl (61 kB)
Using cached wrapt-1.17.2-cp311-cp311-manylinux_2_5_x86_64.manylinux1_x86_64.many
linux_2_17_x86_64.manylinux2014_x86_64.whl (83 kB)
Installing collected packages: wrapt, numpy, smart-open, scipy, gensim
  Attempting uninstall: wrapt
    Found existing installation: wrapt 1.17.2
    Uninstalling wrapt-1.17.2:
      Successfully uninstalled wrapt-1.17.2
  Attempting uninstall: numpy
    Found existing installation: numpy 1.26.4
    Uninstalling numpy-1.26.4:
      Successfully uninstalled numpy-1.26.4
  Attempting uninstall: smart-open
    Found existing installation: smart-open 7.1.0
    Uninstalling smart-open-7.1.0:
      Successfully uninstalled smart-open-7.1.0
  Attempting uninstall: scipy
    Found existing installation: scipy 1.13.1
    Uninstalling scipy-1.13.1:
      Successfully uninstalled scipy-1.13.1
  Attempting uninstall: gensim
    Found existing installation: gensim 4.3.3
    Uninstalling gensim-4.3.3:
      Successfully uninstalled gensim-4.3.3
Successfully installed gensim-4.3.3 numpy-1.26.4 scipy-1.13.1 smart-open-7.1.0 wr
apt-1.17.2
```

In [6]:

```python
import os
import re
from gensim import corpora
from gensim.models.ldamodel import LdaModel
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
import pyLDAvis.gensim as gensimvis  # 수정된 부분
import pyLDAvis
import matplotlib.pyplot as plt

# 수동으로 정의한 한국어 불용어 리스트
korean_stopwords = {
    '의', '가', '이', '은', '들', '는', '좀', '잘', '강', '과', '도', '를', '으로
```

```
    '자', '에', '와', '한', '하다', '에서', '것', '및', '위해', '그', '되다'
}

# 불용어 추가 (분석에 불필요한 단어 추가)
additional_stopwords = {'강점', '약점', '경쟁사'}
korean_stopwords.update(additional_stopwords)

# 텍스트 파일 경로
file_paths = [
    "01_다른경쟁사와간단비교.txt",
    "02_기업리서치관련정리.txt",
    "03_생성AI분석.txt"
]

# 파일 내용을 하나로 결합
combined_text = ""

for file_path in file_paths:
    with open(file_path, 'r', encoding='utf-8') as file:
        combined_text += file.read() + "\n"

# 텍스트 전처리 및 토큰화
def preprocess(text):
    # 소문자 변환, 특수 문자 제거, 토큰화
    text = re.sub(r'[^\w\s]', '', text.lower())
    tokens = word_tokenize(text)
    tokens = [word for word in tokens if word not in korean_stopwords and len(wo
    return tokens

# 전처리된 문서 리스트 생성
documents = preprocess(combined_text)

# 단어 사전 생성
dictionary = corpora.Dictionary([documents])

# 코퍼스 생성 (문서를 BOW(Bag of Words)로 변환)
corpus = [dictionary.doc2bow(documents)]

# LDA 모델 생성
lda_model = LdaModel(corpus, num_topics=3, id2word=dictionary, passes=15)

# pyLDAvis를 이용한 시각화
vis_data = gensimvis.prepare(lda_model, corpus, dictionary)
pyLDAvis.display(vis_data)

# 필요 시, HTML 파일로 저장
pyLDAvis.save_html(vis_data, 'lda_visualization.html')
```

In [ ]: