```
{
 "cells": [
  {
   "cell_type": "markdown",
   "id": "50680648",
   "metadata": {
    "papermill": {
     "duration": 0.016673,
     "end_time": "2022-07-04T00:12:00.987605",
     "exception": false,
     "start_time": "2022-07-04T00:12:00.970932",
     "status": "completed"
    },
    "tags": []
   },
   "source": [
    "### Bag of Words Model with Naive Bayes:"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 1,
   "id": "06cc3366",
   "metadata": {
    "execution": {
     "iopub.execute_input": "2022-07-04T00:12:01.018617Z",
     "iopub.status.busy": "2022-07-04T00:12:01.018186Z",
     "iopub.status.idle": "2022-07-04T00:12:01.029924Z",
     "shell.execute_reply": "2022-07-04T00:12:01.029140Z"
    },
    "papermill": {
```

```
  "duration": 0.029912,
  "end_time": "2022-07-04T00:12:01.032214",
  "exception": false,
  "start_time": "2022-07-04T00:12:01.002302",
  "status": "completed"
 },
 "tags": []
},
"outputs": [],
"source": [
 "import pandas as pd\n",
 "from bs4 import BeautifulSoup\n",
 "from sklearn.model_selection import train_test_split\n",
 "from sklearn.preprocessing import LabelEncoder\n",
 "from nltk.stem import WordNetLemmatizer\n",
 "from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer\n",
 "from sklearn.naive_bayes import MultinomialNB\n",
 "from sklearn.model_selection import GridSearchCV\n",
 "from sklearn.metrics import accuracy_score\n",
 "from sklearn.metrics import classification_report"
]
},
{
 "cell_type": "code",
 "execution_count": 2,
 "id": "350d8e83",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:01.064214Z",
   "iopub.status.busy": "2022-07-04T00:12:01.063527Z",
   "iopub.status.idle": "2022-07-04T00:12:02.582883Z",
```

```
  "shell.execute_reply": "2022-07-04T00:12:02.581803Z"
 },
 "papermill": {
  "duration": 1.538124,
  "end_time": "2022-07-04T00:12:02.585021",
  "exception": false,
  "start_time": "2022-07-04T00:12:01.046897",
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "data": {
   "text/html": [
    "<div>\n",
    "<style scoped>\n",
    "    .dataframe tbody tr th:only-of-type {\n",
    "        vertical-align: middle;\n",
    "    }\n",
    "\n",
    "    .dataframe tbody tr th {\n",
    "        vertical-align: top;\n",
    "    }\n",
    "\n",
    "    .dataframe thead th {\n",
    "        text-align: right;\n",
    "    }\n",
    "</style>\n",
    "<table border=\"1\" class=\"dataframe\">\n",
    "  <thead>\n",
```

```
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>review</th>\n",
"      <th>sentiment</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>One of the other reviewers has mentioned that ...</td>\n",
"      <td>positive</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>A wonderful little production. &lt;br /&gt;&lt;br /&gt;The...</td>\n",
"      <td>positive</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>I thought this was a wonderful way to spend ti...</td>\n",
"      <td>positive</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>Basically there's a family where a little boy ...</td>\n",
"      <td>negative</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>Petter Mattei's \"Love in the Time of Money\" is...</td>\n",
"      <td>positive</td>\n",
```

```
    "    </tr>\n",
    "  </tbody>\n",
    "</table>\n",
    "</div>"
   ],
   "text/plain": [
    "                                        review sentiment\n",
    "0  One of the other reviewers has mentioned that ...  positive\n",
    "1  A wonderful little production. <br /><br />The...  positive\n",
    "2  I thought this was a wonderful way to spend ti...  positive\n",
    "3  Basically there's a family where a little boy ...  negative\n",
    "4  Petter Mattei's \"Love in the Time of Money\" is...  positive"
   ]
  },
  "execution_count": 2,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "data = pd.read_csv('IMDB Dataset.csv')\n",
 "data.head()"
]
},
{
 "cell_type": "markdown",
 "id": "622104f0",
 "metadata": {
  "papermill": {
   "duration": 0.015391,
   "end_time": "2022-07-04T00:12:02.615521",
```

```
   "exception": false,
   "start_time": "2022-07-04T00:12:02.600130",
   "status": "completed"
  },
  "tags": []
 },
 "source": [
  "### Basic Statistics"
 ]
},
{
 "cell_type": "code",
 "execution_count": 3,
 "id": "dd282fb1",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:02.647396Z",
   "iopub.status.busy": "2022-07-04T00:12:02.646689Z",
   "iopub.status.idle": "2022-07-04T00:12:02.651988Z",
   "shell.execute_reply": "2022-07-04T00:12:02.651168Z"
  },
  "papermill": {
   "duration": 0.023526,
   "end_time": "2022-07-04T00:12:02.653941",
   "exception": false,
   "start_time": "2022-07-04T00:12:02.630415",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
```

```
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "Number of rows:  50000\n",
    "Number of columns:  2\n"
   ]
  }
 ],
 "source": [
  "print(\"Number of rows: \", data.shape[0])\n",
  "print(\"Number of columns: \", data.shape[1])"
 ]
},
{
 "cell_type": "code",
 "execution_count": 4,
 "id": "218b6009",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:02.685800Z",
   "iopub.status.busy": "2022-07-04T00:12:02.684758Z",
   "iopub.status.idle": "2022-07-04T00:12:02.719833Z",
   "shell.execute_reply": "2022-07-04T00:12:02.718678Z"
  },
  "papermill": {
   "duration": 0.053755,
   "end_time": "2022-07-04T00:12:02.722447",
   "exception": false,
   "start_time": "2022-07-04T00:12:02.668692",
   "status": "completed"
```

```
 },
 "tags": []
},
"outputs": [
 {
  "name": "stdout",
  "output_type": "stream",
  "text": [
   "<class 'pandas.core.frame.DataFrame'>\n",
   "RangeIndex: 50000 entries, 0 to 49999\n",
   "Data columns (total 2 columns):\n",
   " #   Column     Non-Null Count  Dtype \n",
   "---  ------     --------------  ----- \n",
   " 0   review     50000 non-null  object\n",
   " 1   sentiment  50000 non-null  object\n",
   "dtypes: object(2)\n",
   "memory usage: 781.4+ KB\n"
  ]
 }
],
"source": [
 "data.info()"
]
},
{
 "cell_type": "code",
 "execution_count": 5,
 "id": "52bab329",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:02.754832Z",
```

```
   "iopub.status.busy": "2022-07-04T00:12:02.754218Z",
   "iopub.status.idle": "2022-07-04T00:12:02.768025Z",
   "shell.execute_reply": "2022-07-04T00:12:02.767250Z"
  },
  "papermill": {
   "duration": 0.032076,
   "end_time": "2022-07-04T00:12:02.770143",
   "exception": false,
   "start_time": "2022-07-04T00:12:02.738067",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/plain": [
     "positive    25000\n",
     "negative    25000\n",
     "Name: sentiment, dtype: int64"
    ]
   },
   "execution_count": 5,
   "metadata": {},
   "output_type": "execute_result"
  }
 ],
 "source": [
  "data.sentiment.value_counts()"
 ]
},
```

```
{
 "cell_type": "markdown",
 "id": "b9d5c413",
 "metadata": {
  "papermill": {
   "duration": 0.01485,
   "end_time": "2022-07-04T00:12:02.800455",
   "exception": false,
   "start_time": "2022-07-04T00:12:02.785605",
   "status": "completed"
  },
  "tags": []
 },
 "source": [
  "from the above, we can confirm that the data is equally partioned."
 ]
},
{
 "cell_type": "markdown",
 "id": "3ef227a5",
 "metadata": {
  "papermill": {
   "duration": 0.014519,
   "end_time": "2022-07-04T00:12:02.829881",
   "exception": false,
   "start_time": "2022-07-04T00:12:02.815362",
   "status": "completed"
  },
  "tags": []
 },
 "source": [
```

```
   "### Data Cleaning and preprocessing"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": 6,
  "id": "e2ee0a23",
  "metadata": {
   "execution": {
    "iopub.execute_input": "2022-07-04T00:12:02.862410Z",
    "iopub.status.busy": "2022-07-04T00:12:02.861450Z",
    "iopub.status.idle": "2022-07-04T00:12:02.867874Z",
    "shell.execute_reply": "2022-07-04T00:12:02.867153Z"
   },
   "papermill": {
    "duration": 0.025093,
    "end_time": "2022-07-04T00:12:02.869881",
    "exception": false,
    "start_time": "2022-07-04T00:12:02.844788",
    "status": "completed"
   },
   "tags": []
  },
  "outputs": [
   {
    "data": {
     "text/plain": [
      "\"'A wonderful little production. <br /><br />The filming technique is very unassuming- very old-time-BBC fashion and gives a comforting, and sometimes discomforting, sense of realism to the entire piece. <br /><br />The actors are extremely well chosen- Michael Sheen not only \\"has got all the polari\\" but he has all the voices down pat too! You can truly see the seamless editing guided by the references to Williams\\' diary entries, not only is it well worth the watching but it is a terrificly written and performed piece. A masterful production about one of the great master\\'s of comedy
```

and his life.

The realism really comes home with the little things: the fantasy of the guard which, rather than use the traditional \\'dream\\' techniques remains solid then disappears. It plays on our knowledge and our senses, particularly with the scenes concerning Orton and Halliwell and the sets (particularly of their flat with Halliwell\\'s murals decorating every surface) are terribly well done.'"

]

},

"execution_count": 6,

"metadata": {},

"output_type": "execute_result"

}

],

"source": [

"data['review'][1]"

]

},

{

"cell_type": "markdown",

"id": "337bd6cd",

"metadata": {

"papermill": {

"duration": 0.01472,

"end_time": "2022-07-04T00:12:02.899462",

"exception": false,

"start_time": "2022-07-04T00:12:02.884742",

"status": "completed"

},

"tags": []

},

"source": [

"In the above data we can see \\
\\ break tags. We need to remove them before using this data. "

]

```
},
{
 "cell_type": "code",
 "execution_count": 7,
 "id": "b0a73a53",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:02.931216Z",
   "iopub.status.busy": "2022-07-04T00:12:02.930805Z",
   "iopub.status.idle": "2022-07-04T00:12:03.127057Z",
   "shell.execute_reply": "2022-07-04T00:12:03.125962Z"
  },
  "papermill": {
   "duration": 0.214966,
   "end_time": "2022-07-04T00:12:03.129403",
   "exception": false,
   "start_time": "2022-07-04T00:12:02.914437",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "cleantext = BeautifulSoup(data[\"review\"][1], 'lxml').text"
 ]
},
{
 "cell_type": "markdown",
 "id": "5047625b",
 "metadata": {
  "papermill": {
```

```
  "duration": 0.015912,
  "end_time": "2022-07-04T00:12:03.160578",
  "exception": false,
  "start_time": "2022-07-04T00:12:03.144666",
  "status": "completed"
 },
 "tags": []
},
"source": [
 "We need to remove the slash "
]
},
{
"cell_type": "code",
"execution_count": 8,
"id": "028e10e0",
"metadata": {
 "execution": {
  "iopub.execute_input": "2022-07-04T00:12:03.201650Z",
  "iopub.status.busy": "2022-07-04T00:12:03.201048Z",
  "iopub.status.idle": "2022-07-04T00:12:03.209730Z",
  "shell.execute_reply": "2022-07-04T00:12:03.208599Z"
 },
 "papermill": {
  "duration": 0.034491,
  "end_time": "2022-07-04T00:12:03.212060",
  "exception": false,
  "start_time": "2022-07-04T00:12:03.177569",
  "status": "completed"
 },
 "tags": []
```

```
  },
  "outputs": [
   {
    "data": {
     "text/plain": [
      "'A wonderful little production The filming technique is very unassuming very oldtimeBBC fashion and gives a comforting and sometimes discomforting sense of realism to the entire piece The actors are extremely well chosen Michael Sheen not only has got all the polari but he has all the voices down pat too You can truly see the seamless editing guided by the references to Williams diary entries not only is it well worth the watching but it is a terrificly written and performed piece A masterful production about one of the great masters of comedy and his life The realism really comes home with the little things the fantasy of the guard which rather than use the traditional dream techniques remains solid then disappears It plays on our knowledge and our senses particularly with the scenes concerning Orton and Halliwell and the sets particularly of their flat with Halliwells murals decorating every surface are terribly well done'"
     ]
    },
    "execution_count": 8,
    "metadata": {},
    "output_type": "execute_result"
   }
  ],
  "source": [
   "import re\n",
   "cleantext = re.sub(r'[^\\w\\s]', '', cleantext)\n",
   "cleantext"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": 9,
  "id": "6678cf9a",
  "metadata": {
   "execution": {
```

```
   "iopub.execute_input": "2022-07-04T00:12:03.247551Z",
   "iopub.status.busy": "2022-07-04T00:12:03.246701Z",
   "iopub.status.idle": "2022-07-04T00:12:04.775088Z",
   "shell.execute_reply": "2022-07-04T00:12:04.774016Z"
  },
  "papermill": {
   "duration": 1.549315,
   "end_time": "2022-07-04T00:12:04.777822",
   "exception": false,
   "start_time": "2022-07-04T00:12:03.228507",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "import nltk\n",
  "from nltk.corpus import stopwords"
 ]
},
{
 "cell_type": "code",
 "execution_count": 10,
 "id": "5bddcf65",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:04.812121Z",
   "iopub.status.busy": "2022-07-04T00:12:04.811313Z",
   "iopub.status.idle": "2022-07-04T00:12:04.957613Z",
   "shell.execute_reply": "2022-07-04T00:12:04.956073Z"
  },
```

```
 "papermill": {
  "duration": 0.165981,
  "end_time": "2022-07-04T00:12:04.960285",
  "exception": false,
  "start_time": "2022-07-04T00:12:04.794304",
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "name": "stderr",
  "output_type": "stream",
  "text": [
   "[nltk_data] Error loading stopwords: <urlopen error [WinError 10060] A\n",
   "[nltk_data]     connection attempt failed because the connected party\n",
   "[nltk_data]     did not properly respond after a period of time, or\n",
   "[nltk_data]     established connection failed because connected host\n",
   "[nltk_data]     has failed to respond>\n"
  ]
 },
 {
  "data": {
   "text/plain": [
    "['i',\n",
    " 'me',\n",
    " 'my',\n",
    " 'myself',\n",
    " 'we',\n",
    " 'our',\n",
    " 'ours',\n",
```

" 'ourselves',\n",

" 'you',\n",

" \"you're\",\n",

" \"you've\",\n",

" \"you'll\",\n",

" \"you'd\",\n",

" 'your',\n",

" 'yours',\n",

" 'yourself',\n",

" 'yourselves',\n",

" 'he',\n",

" 'him',\n",

" 'his',\n",

" 'himself',\n",

" 'she',\n",

" \"she's\",\n",

" 'her',\n",

" 'hers',\n",

" 'herself',\n",

" 'it',\n",

" \"it's\",\n",

" 'its',\n",

" 'itself',\n",

" 'they',\n",

" 'them',\n",

" 'their',\n",

" 'theirs',\n",

" 'themselves',\n",

" 'what',\n",

" 'which',\n",

" 'who',\n",

" 'whom',\n",

" 'this',\n",

" 'that',\n",

" \"that'll\",\n",

" 'these',\n",

" 'those',\n",

" 'am',\n",

" 'is',\n",

" 'are',\n",

" 'was',\n",

" 'were',\n",

" 'be',\n",

" 'been',\n",

" 'being',\n",

" 'have',\n",

" 'has',\n",

" 'had',\n",

" 'having',\n",

" 'do',\n",

" 'does',\n",

" 'did',\n",

" 'doing',\n",

" 'a',\n",

" 'an',\n",

" 'the',\n",

" 'and',\n",

" 'but',\n",

" 'if',\n",

" 'or',\n",

" 'because',\n",

" 'as',\n",

" 'until',\n",

" 'while',\n",

" 'of',\n",

" 'at',\n",

" 'by',\n",

" 'for',\n",

" 'with',\n",

" 'about',\n",

" 'against',\n",

" 'between',\n",

" 'into',\n",

" 'through',\n",

" 'during',\n",

" 'before',\n",

" 'after',\n",

" 'above',\n",

" 'below',\n",

" 'to',\n",

" 'from',\n",

" 'up',\n",

" 'down',\n",

" 'in',\n",

" 'out',\n",

" 'on',\n",

" 'off',\n",

" 'over',\n",

" 'under',\n",

" 'again',\n",

" 'further',\n",

" 'then',\n",

" 'once',\n",

" 'here',\n",

" 'there',\n",

" 'when',\n",

" 'where',\n",

" 'why',\n",

" 'how',\n",

" 'all',\n",

" 'any',\n",

" 'both',\n",

" 'each',\n",

" 'few',\n",

" 'more',\n",

" 'most',\n",

" 'other',\n",

" 'some',\n",

" 'such',\n",

" 'no',\n",

" 'nor',\n",

" 'not',\n",

" 'only',\n",

" 'own',\n",

" 'same',\n",

" 'so',\n",

" 'than',\n",

" 'too',\n",

" 'very',\n",

" 's',\n",

" 't',\n",

" 'can',\n",

" 'will',\n",

" 'just',\n",

" 'don',\n",

" \"don't\",\n",

" 'should',\n",

" \"should've\",\n",

" 'now',\n",

" 'd',\n",

" 'll',\n",

" 'm',\n",

" 'o',\n",

" 're',\n",

" 've',\n",

" 'y',\n",

" 'ain',\n",

" 'aren',\n",

" \"aren't\",\n",

" 'couldn',\n",

" \"couldn't\",\n",

" 'didn',\n",

" \"didn't\",\n",

" 'doesn',\n",

" \"doesn't\",\n",

" 'hadn',\n",

" \"hadn't\",\n",

" 'hasn',\n",

" \"hasn't\",\n",

" 'haven',\n",

" \"haven't\",\n",

" 'isn',\n",

" \"isn't\",\n",

" 'ma',\n",

" 'mightn',\n",

```
    " \"mightn't\",\n",
    " 'mustn',\n",
    " \"mustn't\",\n",
    " 'needn',\n",
    " \"needn't\",\n",
    " 'shan',\n",
    " \"shan't\",\n",
    " 'shouldn',\n",
    " \"shouldn't\",\n",
    " 'wasn',\n",
    " \"wasn't\",\n",
    " 'weren',\n",
    " \"weren't\",\n",
    " 'won',\n",
    " \"won't\",\n",
    " 'wouldn',\n",
    " \"wouldn't\"]"
   ]
  },
  "execution_count": 10,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "nltk.download('stopwords')\n",
 "stopwords.words('english')"
]
},
{
 "cell_type": "code",
```

```
 "execution_count": 11,
 "id": "db56398d",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:04.994513Z",
   "iopub.status.busy": "2022-07-04T00:12:04.993609Z",
   "iopub.status.idle": "2022-07-04T00:12:05.000059Z",
   "shell.execute_reply": "2022-07-04T00:12:04.999202Z"
  },
  "papermill": {
   "duration": 0.025915,
   "end_time": "2022-07-04T00:12:05.002380",
   "exception": false,
   "start_time": "2022-07-04T00:12:04.976465",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "token = cleantext.lower().split()\n",
  "stopword = set(stopwords.words('english'))\n",
  "token_list = [ word for word in token if word.lower() not in stopword ]"
 ]
},
{
 "cell_type": "code",
 "execution_count": 12,
 "id": "9a9f9e96",
 "metadata": {
  "execution": {
```

```
    "iopub.execute_input": "2022-07-04T00:12:05.035709Z",
    "iopub.status.busy": "2022-07-04T00:12:05.034916Z",
    "iopub.status.idle": "2022-07-04T00:12:05.041732Z",
    "shell.execute_reply": "2022-07-04T00:12:05.040911Z"
   },
   "papermill": {
    "duration": 0.025873,
    "end_time": "2022-07-04T00:12:05.043887",
    "exception": false,
    "start_time": "2022-07-04T00:12:05.018014",
    "status": "completed"
   },
   "tags": []
  },
  "outputs": [
   {
    "data": {
     "text/plain": [
      "'"'wonderful little production filming technique unassuming oldtimebbc fashion gives comforting sometimes discomforting sense realism entire piece actors extremely well chosen michael sheen got polari voices pat truly see seamless editing guided references williams diary entries well worth watching terrificly written performed piece masterful production one great masters comedy life realism really comes home little things fantasy guard rather use traditional dream techniques remains solid disappears plays knowledge senses particularly scenes concerning orton halliwell sets particularly flat halliwells murals decorating every surface terribly well done'"
     ]
    },
    "execution_count": 12,
    "metadata": {},
    "output_type": "execute_result"
   }
  ],
  "source": [
```

```
  "\" \".join(token_list)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 13,
 "id": "8b3755f4",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:05.118940Z",
   "iopub.status.busy": "2022-07-04T00:12:05.118201Z",
   "iopub.status.idle": "2022-07-04T00:12:05.123096Z",
   "shell.execute_reply": "2022-07-04T00:12:05.122218Z"
  },
  "papermill": {
   "duration": 0.024975,
   "end_time": "2022-07-04T00:12:05.125387",
   "exception": false,
   "start_time": "2022-07-04T00:12:05.100412",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "lemmatizer = WordNetLemmatizer()"
 ]
},
{
 "cell_type": "code",
 "execution_count": 14,
```

```
 "id": "d99258d1",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:06.391229Z",
   "iopub.status.busy": "2022-07-04T00:12:06.390833Z",
   "iopub.status.idle": "2022-07-04T00:12:08.545379Z",
   "shell.execute_reply": "2022-07-04T00:12:08.544192Z"
  },
  "papermill": {
   "duration": 2.174264,
   "end_time": "2022-07-04T00:12:08.547796",
   "exception": false,
   "start_time": "2022-07-04T00:12:06.373532",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/plain": [
     "'wonderful little production filming technique unassuming oldtimebbc fashion gives comforting sometimes discomforting sense realism entire piece actors extremely well chosen michael sheen got polari voices pat truly see seamless editing guided references williams diary entries well worth watching terrificly written performed piece masterful production one great masters comedy life realism really comes home little things fantasy guard rather use traditional dream techniques remains solid disappears plays knowledge senses particularly scenes concerning orton halliwell sets particularly flat halliwells murals decorating every surface terribly well done'"
    ]
   },
   "execution_count": 14,
   "metadata": {},
   "output_type": "execute_result"
```

```
  }
 ],
 "source": [
  "lemmatizer.lemmatize(\" \".join(token_list))"
 ]
},
{
 "cell_type": "code",
 "execution_count": 15,
 "id": "ba40231d",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:08.582587Z",
   "iopub.status.busy": "2022-07-04T00:12:08.582217Z",
   "iopub.status.idle": "2022-07-04T00:12:08.589272Z",
   "shell.execute_reply": "2022-07-04T00:12:08.588117Z"
  },
  "papermill": {
   "duration": 0.02736,
   "end_time": "2022-07-04T00:12:08.591670",
   "exception": false,
   "start_time": "2022-07-04T00:12:08.564310",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "data": {
    "text/plain": [
     "Index(['review', 'sentiment'], dtype='object')"
```

```
   ]
  },
  "execution_count": 15,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "data.keys()"
]
},
{
 "cell_type": "code",
 "execution_count": 16,
 "id": "da4e1a67",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:08.627045Z",
   "iopub.status.busy": "2022-07-04T00:12:08.626255Z",
   "iopub.status.idle": "2022-07-04T00:12:08.633628Z",
   "shell.execute_reply": "2022-07-04T00:12:08.632895Z"
  },
  "papermill": {
   "duration": 0.027571,
   "end_time": "2022-07-04T00:12:08.635875",
   "exception": false,
   "start_time": "2022-07-04T00:12:08.608304",
   "status": "completed"
  },
  "tags": []
 },
```

```
 "outputs": [],
 "source": [
  "from tqdm import tqdm\n",
  "def data_cleaner(data):\n",
  "    clean_data = []\n",
  "    for review in tqdm(data):\n",
  "        cleantext = BeautifulSoup(review, \"lxml\").text\n",
  "        cleantext = re.sub(r'[^\\w\\s]', '', cleantext)\n",
  "        cleantext = [ token for token in cleantext.lower().split() if token not in stopword ]\n",
  "        cleantext = lemmatizer.lemmatize(\" \".join(cleantext))\n",
  "        clean_data.append(cleantext.strip())\n",
  "    return clean_data"
 ]
},
{
 "cell_type": "code",
 "execution_count": null,
 "id": "acc08cad",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:08.671005Z",
   "iopub.status.busy": "2022-07-04T00:12:08.670244Z",
   "iopub.status.idle": "2022-07-04T00:12:32.481120Z",
   "shell.execute_reply": "2022-07-04T00:12:32.480128Z"
  },
  "papermill": {
   "duration": 23.831186,
   "end_time": "2022-07-04T00:12:32.483548",
   "exception": false,
   "start_time": "2022-07-04T00:12:08.652362",
   "status": "completed"
```

```
 },
 "tags": []
},
"outputs": [],
"source": [
 "clean_data = data_cleaner(data.review.values)"
]
},
{
"cell_type": "code",
"execution_count": 18,
"id": "e5a5f6a5",
"metadata": {
 "execution": {
  "iopub.execute_input": "2022-07-04T00:12:32.552077Z",
  "iopub.status.busy": "2022-07-04T00:12:32.551240Z",
  "iopub.status.idle": "2022-07-04T00:12:32.557342Z",
  "shell.execute_reply": "2022-07-04T00:12:32.556511Z"
 },
 "papermill": {
  "duration": 0.042609,
  "end_time": "2022-07-04T00:12:32.559350",
  "exception": false,
  "start_time": "2022-07-04T00:12:32.516741",
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "data": {
```

```
    "text/plain": [
     "'one reviewers mentioned watching 1 oz episode youll hooked right exactly happened methe first thing struck oz brutality unflinching scenes violence set right word go trust show faint hearted timid show pulls punches regards drugs sex violence hardcore classic use wordit called oz nickname given oswald maximum security state penitentary focuses mainly emerald city experimental section prison cells glass fronts face inwards privacy high agenda em city home manyaryans muslims gangstas latinos christians italians irish moreso scuffles death stares dodgy dealings shady agreements never far awayi would say main appeal show due fact goes shows wouldnt dare forget pretty pictures painted mainstream audiences forget charm forget romanceoz doesnt mess around first episode ever saw struck nasty surreal couldnt say ready watched developed taste oz got accustomed high levels graphic violence violence injustice crooked guards wholl sold nickel inmates wholl kill order get away well mannered middle class inmates turned prison bitches due lack street skills prison experience watching oz may become comfortable uncomfortable viewingthats get touch darker side'"
    ]
   },
   "execution_count": 18,
   "metadata": {},
   "output_type": "execute_result"
  }
 ],
 "source": [
  "clean_data[0]"
 ]
},
{
 "cell_type": "markdown",
 "id": "eb84b53d",
 "metadata": {
  "papermill": {
   "duration": 0.032865,
   "end_time": "2022-07-04T00:12:32.625067",
   "exception": false,
   "start_time": "2022-07-04T00:12:32.592202",
   "status": "completed"
```

```
   },
   "tags": []
  },
  "source": [
   "### Train test split"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": 19,
  "id": "be819337",
  "metadata": {
   "execution": {
    "iopub.execute_input": "2022-07-04T00:12:32.693524Z",
    "iopub.status.busy": "2022-07-04T00:12:32.692860Z",
    "iopub.status.idle": "2022-07-04T00:12:32.780398Z",
    "shell.execute_reply": "2022-07-04T00:12:32.779247Z"
   },
   "papermill": {
    "duration": 0.124476,
    "end_time": "2022-07-04T00:12:32.783149",
    "exception": false,
    "start_time": "2022-07-04T00:12:32.658673",
    "status": "completed"
   },
   "tags": []
  },
  "outputs": [],
  "source": [
   "X_train, X_test, y_train, y_test = train_test_split(data, data.sentiment, test_size=0.2,
random_state=42, stratify=data.sentiment)"
```

```
 ]
},
{
 "cell_type": "code",
 "execution_count": 20,
 "id": "9c517da5",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:32.853310Z",
   "iopub.status.busy": "2022-07-04T00:12:32.852310Z",
   "iopub.status.idle": "2022-07-04T00:12:32.873186Z",
   "shell.execute_reply": "2022-07-04T00:12:32.871984Z"
  },
  "papermill": {
   "duration": 0.059478,
   "end_time": "2022-07-04T00:12:32.875701",
   "exception": false,
   "start_time": "2022-07-04T00:12:32.816223",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "le = LabelEncoder()\n",
  "y_train = le.fit_transform(y_train)\n",
  "le_test = LabelEncoder()\n",
  "y_test = le_test.fit_transform(y_test)"
 ]
},
{
```

```
"cell_type": "code",
"execution_count": 21,
"id": "420e7718",
"metadata": {
 "execution": {
  "iopub.execute_input": "2022-07-04T00:12:32.943826Z",
  "iopub.status.busy": "2022-07-04T00:12:32.943420Z",
  "iopub.status.idle": "2022-07-04T00:12:32.948883Z",
  "shell.execute_reply": "2022-07-04T00:12:32.947807Z"
 },
 "papermill": {
  "duration": 0.042176,
  "end_time": "2022-07-04T00:12:32.951190",
  "exception": false,
  "start_time": "2022-07-04T00:12:32.909014",
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "name": "stdout",
  "output_type": "stream",
  "text": [
   "(40000, 2) (40000,)\n",
   "(10000, 2) (10000,)\n"
  ]
 }
],
"source": [
 "print(X_train.shape, y_train.shape)\n",
```

```
  "print(X_test.shape, y_test.shape)"
 ]
},
{
 "cell_type": "code",
 "execution_count": null,
 "id": "207cd78e",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:12:33.019027Z",
   "iopub.status.busy": "2022-07-04T00:12:33.018586Z",
   "iopub.status.idle": "2022-07-04T00:12:52.206112Z",
   "shell.execute_reply": "2022-07-04T00:12:52.204980Z"
  },
  "papermill": {
   "duration": 19.225122,
   "end_time": "2022-07-04T00:12:52.209201",
   "exception": false,
   "start_time": "2022-07-04T00:12:32.984079",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "clean_data_train_data = data_cleaner(X_train.review.values)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 23,
```

```
"id": "802dd12f",
"metadata": {
 "execution": {
  "iopub.execute_input": "2022-07-04T00:12:52.304375Z",
  "iopub.status.busy": "2022-07-04T00:12:52.303774Z",
  "iopub.status.idle": "2022-07-04T00:12:52.320457Z",
  "shell.execute_reply": "2022-07-04T00:12:52.319317Z"
 },
 "papermill": {
  "duration": 0.066291,
  "end_time": "2022-07-04T00:12:52.322906",
  "exception": false,
  "start_time": "2022-07-04T00:12:52.256615",
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "data": {
   "text/html": [
    "<div>\n",
    "<style scoped>\n",
    "    .dataframe tbody tr th:only-of-type {\n",
    "        vertical-align: middle;\n",
    "    }\n",
    "\n",
    "    .dataframe tbody tr th {\n",
    "        vertical-align: top;\n",
    "    }\n",
    "\n",
```

```
"    .dataframe thead th {\n",
"        text-align: right;\n",
"    }\n",
"</style>\n",
"<table border=\"1\" class=\"dataframe\">\n",
"  <thead>\n",
"    <tr style=\"text-align: right;\">\n",
"      <th></th>\n",
"      <th>review</th>\n",
"      <th>sentiment</th>\n",
"      <th>cleaned_text</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>47808</th>\n",
"      <td>I caught this little gem totally by accident b...</td>\n",
"      <td>positive</td>\n",
"      <td>caught little gem totally accident back 1980 8...</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>20154</th>\n",
"      <td>I can't believe that I let myself into this mo...</td>\n",
"      <td>negative</td>\n",
"      <td>cant believe let movie accomplish favor friend...</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>43069</th>\n",
"      <td>*spoiler alert!* it just gets to me the nerve ...</td>\n",
"      <td>negative</td>\n",
"      <td>spoiler alert gets nerve people remake use ter...</td>\n",
```

```
"    </tr>\n",
"    <tr>\n",
"      <th>19413</th>\n",
"      <td>If there's one thing I've learnt from watching...</td>\n",
"      <td>negative</td>\n",
"      <td>theres one thing ive learnt watching george ro...</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>13673</th>\n",
"      <td>I remember when this was in theaters, reviews ...</td>\n",
"      <td>negative</td>\n",
"      <td>remember theaters reviews said horrible well d...</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>"
],
"text/plain": [
"                                      review sentiment  \\\n",
"47808  I caught this little gem totally by accident b...  positive   \n",
"20154  I can't believe that I let myself into this mo...  negative   \n",
"43069  *spoiler alert!* it just gets to me the nerve ...  negative   \n",
"19413  If there's one thing I've learnt from watching...  negative   \n",
"13673  I remember when this was in theaters, reviews ...  negative   \n",
"\n",
"                                    cleaned_text  \n",
"47808  caught little gem totally accident back 1980 8...  \n",
"20154  cant believe let movie accomplish favor friend...  \n",
"43069  spoiler alert gets nerve people remake use ter...  \n",
"19413  theres one thing ive learnt watching george ro...  \n",
"13673  remember theaters reviews said horrible well d...  "
```

```
   ]
  },
  "execution_count": 23,
  "metadata": {},
  "output_type": "execute_result"
 }
],
"source": [
 "X_train['cleaned_text'] = clean_data_train_data\n",
 "X_train.head()"
]
},
{
"cell_type": "code",
"execution_count": null,
"id": "ce1a0a0e",
"metadata": {
 "execution": {
  "iopub.execute_input": "2022-07-04T00:12:52.417414Z",
  "iopub.status.busy": "2022-07-04T00:12:52.417037Z",
  "iopub.status.idle": "2022-07-04T00:12:57.191242Z",
  "shell.execute_reply": "2022-07-04T00:12:57.190226Z"
 },
 "papermill": {
  "duration": 4.82472,
  "end_time": "2022-07-04T00:12:57.194006",
  "exception": false,
  "start_time": "2022-07-04T00:12:52.369286",
  "status": "completed"
 },
 "tags": []
```

```
 },
 "outputs": [],
 "source": [
  "clean_data_test_data = data_cleaner(X_test.review.values)\n",
  "X_test['cleaned_text'] = clean_data_test_data\n",
  "X_test.head()"
 ]
},
{
 "cell_type": "markdown",
 "id": "4bd2ceb5",
 "metadata": {
  "papermill": {
   "duration": 0.049422,
   "end_time": "2022-07-04T00:12:57.293654",
   "exception": false,
   "start_time": "2022-07-04T00:12:57.244232",
   "status": "completed"
  },
  "tags": []
 },
 "source": [
  "### Vectorizer"
 ]
},
{
 "cell_type": "code",
 "execution_count": 25,
 "id": "0f4118fb",
 "metadata": {},
 "outputs": [],
```

```
 "source": [
  "vec = CountVectorizer()\n",
  "vec = vec.fit(X_train.cleaned_text)\n",
  "train_x_bow = vec.transform(X_train.cleaned_text)\n",
  "test_x_bow = vec.transform(X_test.cleaned_text)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 26,
 "id": "28388b55",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:13:11.770299Z",
   "iopub.status.busy": "2022-07-04T00:13:11.769917Z",
   "iopub.status.idle": "2022-07-04T00:13:11.775205Z",
   "shell.execute_reply": "2022-07-04T00:13:11.774026Z"
  },
  "papermill": {
   "duration": 0.060328,
   "end_time": "2022-07-04T00:13:11.778609",
   "exception": false,
   "start_time": "2022-07-04T00:13:11.718281",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
```

```
   "text": [
    "(40000, 192139)\n",
    "(10000, 192139)\n"
   ]
  }
 ],
 "source": [
  "print(train_x_bow.shape)\n",
  "print(test_x_bow.shape)"
 ]
},
{
 "cell_type": "markdown",
 "id": "70ffa443",
 "metadata": {
  "papermill": {
   "duration": 0.049352,
   "end_time": "2022-07-04T00:13:11.877634",
   "exception": false,
   "start_time": "2022-07-04T00:13:11.828282",
   "status": "completed"
  },
  "tags": []
 },
 "source": [
  "### Naive Bayes with Hyperparameter Tuning"
 ]
},
{
 "cell_type": "code",
 "execution_count": 27,
```

```
 "id": "e3378a17",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:13:12.090182Z",
   "iopub.status.busy": "2022-07-04T00:13:12.089768Z",
   "iopub.status.idle": "2022-07-04T00:13:12.094245Z",
   "shell.execute_reply": "2022-07-04T00:13:12.093209Z"
  },
  "papermill": {
   "duration": 0.057588,
   "end_time": "2022-07-04T00:13:12.096442",
   "exception": false,
   "start_time": "2022-07-04T00:13:12.038854",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "classifier = MultinomialNB()"
 ]
},
{
 "cell_type": "code",
 "execution_count": 28,
 "id": "52e0ab32",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:13:12.198968Z",
   "iopub.status.busy": "2022-07-04T00:13:12.197794Z",
   "iopub.status.idle": "2022-07-04T00:13:12.202897Z",
```

```
   "shell.execute_reply": "2022-07-04T00:13:12.202002Z"
  },
  "papermill": {
   "duration": 0.058692,
   "end_time": "2022-07-04T00:13:12.205045",
   "exception": false,
   "start_time": "2022-07-04T00:13:12.146353",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "alpha_ranges = {\"alpha\": [0.001, 0.01, 0.1, 1, 10.0, 100]}"
 ]
},
{
 "cell_type": "code",
 "execution_count": null,
 "id": "ba4aca61",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:13:12.306546Z",
   "iopub.status.busy": "2022-07-04T00:13:12.306164Z",
   "iopub.status.idle": "2022-07-04T00:13:14.404040Z",
   "shell.execute_reply": "2022-07-04T00:13:14.402890Z"
  },
  "papermill": {
   "duration": 2.151635,
   "end_time": "2022-07-04T00:13:14.406755",
   "exception": false,
```

```
    "start_time": "2022-07-04T00:13:12.255120",
    "status": "completed"
   },
   "tags": []
  },
  "outputs": [],
  "source": [
   "grid_search = GridSearchCV(classifier, param_grid=alpha_ranges, scoring='accuracy', cv=3, return_train_score=True)\n",
   "grid_search.fit(train_x_bow, y_train)"
  ]
 },
 {
  "cell_type": "code",
  "execution_count": 30,
  "id": "77423d56",
  "metadata": {
   "execution": {
    "iopub.execute_input": "2022-07-04T00:13:14.509627Z",
    "iopub.status.busy": "2022-07-04T00:13:14.509102Z",
    "iopub.status.idle": "2022-07-04T00:13:14.516013Z",
    "shell.execute_reply": "2022-07-04T00:13:14.514737Z"
   },
   "papermill": {
    "duration": 0.060428,
    "end_time": "2022-07-04T00:13:14.518324",
    "exception": false,
    "start_time": "2022-07-04T00:13:14.457896",
    "status": "completed"
   },
   "tags": []
```

```
 },
 "outputs": [],
 "source": [
  "alpha = [0.001, 0.01, 0.1, 1, 10.0, 100]\n",
  "train_acc = grid_search.cv_results_['mean_train_score']\n",
  "train_std = grid_search.cv_results_['std_train_score']\n",
  "\n",
  "test_acc = grid_search.cv_results_['mean_test_score']\n",
  "test_std = grid_search.cv_results_['std_test_score']\n"
 ]
},
{
 "cell_type": "code",
 "execution_count": null,
 "id": "98dac97e",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:13:15.136873Z",
   "iopub.status.busy": "2022-07-04T00:13:15.136170Z",
   "iopub.status.idle": "2022-07-04T00:13:15.142118Z",
   "shell.execute_reply": "2022-07-04T00:13:15.141293Z"
  },
  "papermill": {
   "duration": 0.059743,
   "end_time": "2022-07-04T00:13:15.144358",
   "exception": false,
   "start_time": "2022-07-04T00:13:15.084615",
   "status": "completed"
  },
  "tags": []
 },
```

```
 "outputs": [],
 "source": [
  "grid_search.best_estimator_"
 ]
},
{
 "cell_type": "code",
 "execution_count": null,
 "id": "ca85f1aa",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:13:15.248152Z",
   "iopub.status.busy": "2022-07-04T00:13:15.247451Z",
   "iopub.status.idle": "2022-07-04T00:13:15.304104Z",
   "shell.execute_reply": "2022-07-04T00:13:15.303005Z"
  },
  "papermill": {
   "duration": 0.110919,
   "end_time": "2022-07-04T00:13:15.306621",
   "exception": false,
   "start_time": "2022-07-04T00:13:15.195702",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "classifier = MultinomialNB(alpha=1)\n",
  "classifier.fit(train_x_bow, y_train)"
 ]
},
```

```
{
 "cell_type": "code",
 "execution_count": 33,
 "id": "025c020c",
 "metadata": {
  "execution": {
   "iopub.execute_input": "2022-07-04T00:13:15.409568Z",
   "iopub.status.busy": "2022-07-04T00:13:15.409155Z",
   "iopub.status.idle": "2022-07-04T00:13:15.428507Z",
   "shell.execute_reply": "2022-07-04T00:13:15.427433Z"
  },
  "papermill": {
   "duration": 0.073655,
   "end_time": "2022-07-04T00:13:15.430884",
   "exception": false,
   "start_time": "2022-07-04T00:13:15.357229",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [],
 "source": [
  "predict = classifier.predict(test_x_bow)"
 ]
},
{
 "cell_type": "code",
 "execution_count": 34,
 "id": "6a5dd33c",
 "metadata": {
  "execution": {
```

```
   "iopub.execute_input": "2022-07-04T00:13:15.534091Z",
   "iopub.status.busy": "2022-07-04T00:13:15.533610Z",
   "iopub.status.idle": "2022-07-04T00:13:15.541678Z",
   "shell.execute_reply": "2022-07-04T00:13:15.539910Z"
  },
  "papermill": {
   "duration": 0.062558,
   "end_time": "2022-07-04T00:13:15.544101",
   "exception": false,
   "start_time": "2022-07-04T00:13:15.481543",
   "status": "completed"
  },
  "tags": []
 },
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "Accuracy is  0.8599\n"
   ]
  }
 ],
 "source": [
  "print(\"Accuracy is \", accuracy_score(y_test, predict))"
 ]
},
{
 "cell_type": "code",
 "execution_count": 35,
 "id": "5e88c8e4",
```

```
"metadata": {
 "execution": {
  "iopub.execute_input": "2022-07-04T00:13:15.647895Z",
  "iopub.status.busy": "2022-07-04T00:13:15.647176Z",
  "iopub.status.idle": "2022-07-04T00:13:15.674416Z",
  "shell.execute_reply": "2022-07-04T00:13:15.672988Z"
 },
 "papermill": {
  "duration": 0.082002,
  "end_time": "2022-07-04T00:13:15.676889",
  "exception": false,
  "start_time": "2022-07-04T00:13:15.594887",
  "status": "completed"
 },
 "tags": []
},
"outputs": [
 {
  "name": "stdout",
  "output_type": "stream",
  "text": [
   "Accuracy is              precision    recall  f1-score   support\n",
   "\n",
   "           0      0.85      0.88      0.86      5000\n",
   "           1      0.87      0.84      0.86      5000\n",
   "\n",
   "    accuracy                           0.86     10000\n",
   "   macro avg      0.86      0.86      0.86     10000\n",
   "weighted avg      0.86      0.86      0.86     10000\n",
   "\n"
  ]
```

```
  }
 ],
 "source": [
  "print(\"Accuracy is \", classification_report(y_test, predict))"
 ]
},
{
 "cell_type": "markdown",
 "id": "4831cf04",
 "metadata": {},
 "source": [
  "### TF-IDF Model with Naive Bayes:"
 ]
},
{
 "cell_type": "code",
 "execution_count": 36,
 "id": "2c551fd6",
 "metadata": {},
 "outputs": [
  {
   "name": "stdout",
   "output_type": "stream",
   "text": [
    "Accuracy using TF-IDF model: 0.867\n"
   ]
  }
 ],
 "source": [
  "# Vectorize the text using TF-IDF model\n",
  "tfidf_vectorizer = TfidfVectorizer()\n",
```

```
   "X_train_tfidf = tfidf_vectorizer.fit_transform(X_train.cleaned_text)\n",
   "X_test_tfidf = tfidf_vectorizer.transform(X_test.cleaned_text)\n",
   "\n",
   "# Train a Naive Bayes classifier on the TF-IDF features\n",
   "nb_classifier_tfidf = MultinomialNB()\n",
   "nb_classifier_tfidf.fit(X_train_tfidf, y_train)\n",
   "\n",
   "# Predict and calculate accuracy\n",
   "predictions_tfidf = nb_classifier_tfidf.predict(X_test_tfidf)\n",
   "accuracy_tfidf = accuracy_score(y_test, predictions_tfidf)\n",
   "print(\"Accuracy using TF-IDF model:\", accuracy_tfidf)"
  ]
 }
],
"metadata": {
 "kernelspec": {
  "display_name": "Python 3 (ipykernel)",
  "language": "python",
  "name": "python3"
 },
 "language_info": {
  "codemirror_mode": {
   "name": "ipython",
   "version": 3
  },
  "file_extension": ".py",
  "mimetype": "text/x-python",
  "name": "python",
  "nbconvert_exporter": "python",
  "pygments_lexer": "ipython3",
  "version": "3.10.9"
```

```
 },
 "papermill": {
  "default_parameters": {},
  "duration": 88.305111,
  "end_time": "2022-07-04T00:13:19.925401",
  "environment_variables": {},
  "exception": null,
  "input_path": "__notebook__.ipynb",
  "output_path": "__notebook__.ipynb",
  "parameters": {},
  "start_time": "2022-07-04T00:11:51.620290",
  "version": "2.3.4"
 }
},
"nbformat": 4,
"nbformat_minor": 5
}
```