italicized text## 1. Install and Import the Required Libraries

1. List item
2. List item

```
# Install all the required libraries
# !pip install -U -q pdfplumber tiktoken openai chromadb sentence-
transformers
import sys
!{sys.executable} -m pip install pdfplumber tiktoken openai chromadb
sentence-transformers
```

```
Defaulting to user installation because normal site-packages is not
writeable
Requirement already satisfied: pdfplumber in c:\users\kirth\appdata\
roaming\python\python313\site-packages (0.11.9)
Requirement already satisfied: tiktoken in c:\users\kirth\appdata\
roaming\python\python313\site-packages (0.12.0)
Requirement already satisfied: openai in c:\users\kirth\appdata\
roaming\python\python313\site-packages (2.15.0)
Requirement already satisfied: chromadb in c:\users\kirth\appdata\
roaming\python\python313\site-packages (1.5.0)
Requirement already satisfied: sentence-transformers in c:\users\
kirth\appdata\roaming\python\python313\site-packages (5.2.2)
Requirement already satisfied: pdfminer.six==20251230 in c:\users\
kirth\appdata\roaming\python\python313\site-packages (from pdfplumber)
(20251230)
Requirement already satisfied: Pillow>=9.1 in c:\programdata\
anaconda3\lib\site-packages (from pdfplumber) (11.1.0)
Requirement already satisfied: pypdfium2>=4.18.0 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from pdfplumber)
(5.4.0)
Requirement already satisfied: charset-normalizer>=2.0.0 in c:\
programdata\anaconda3\lib\site-packages (from pdfminer.six==20251230-
>pdfplumber) (3.3.2)
Requirement already satisfied: cryptography>=36.0.0 in c:\programdata\
anaconda3\lib\site-packages (from pdfminer.six==20251230->pdfplumber)
(44.0.1)
Requirement already satisfied: regex>=2022.1.18 in c:\programdata\
anaconda3\lib\site-packages (from tiktoken) (2024.11.6)
Requirement already satisfied: requests>=2.26.0 in c:\programdata\
anaconda3\lib\site-packages (from tiktoken) (2.32.3)
Requirement already satisfied: anyio<5,>=3.5.0 in c:\programdata\
anaconda3\lib\site-packages (from openai) (4.7.0)
Requirement already satisfied: distro<2,>=1.7.0 in c:\programdata\
anaconda3\lib\site-packages (from openai) (1.9.0)
Requirement already satisfied: httpx<1,>=0.23.0 in c:\programdata\
anaconda3\lib\site-packages (from openai) (0.28.1)
Requirement already satisfied: jiter<1,>=0.10.0 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from openai) (0.12.0)
```

```
Requirement already satisfied: pydantic<3,>=1.9.0 in c:\programdata\
anaconda3\lib\site-packages (from openai) (2.10.3)
Requirement already satisfied: sniffio in c:\programdata\anaconda3\
lib\site-packages (from openai) (1.3.0)
Requirement already satisfied: tqdm>4 in c:\programdata\anaconda3\lib\
site-packages (from openai) (4.67.1)
Requirement already satisfied: typing-extensions<5,>=4.11 in c:\
programdata\anaconda3\lib\site-packages (from openai) (4.12.2)
Requirement already satisfied: idna>=2.8 in c:\programdata\anaconda3\
lib\site-packages (from anyio<5,>=3.5.0->openai) (3.7)
Requirement already satisfied: certifi in c:\programdata\anaconda3\
lib\site-packages (from httpx<1,>=0.23.0->openai) (2025.4.26)
Requirement already satisfied: httpcore==1.* in c:\programdata\
anaconda3\lib\site-packages (from httpx<1,>=0.23.0->openai) (1.0.9)
Requirement already satisfied: h11>=0.16 in c:\programdata\anaconda3\
lib\site-packages (from httpcore==1.*->httpx<1,>=0.23.0->openai)
(0.16.0)
Requirement already satisfied: annotated-types>=0.6.0 in c:\
programdata\anaconda3\lib\site-packages (from pydantic<3,>=1.9.0-
>openai) (0.6.0)
Requirement already satisfied: pydantic-core==2.27.1 in c:\
programdata\anaconda3\lib\site-packages (from pydantic<3,>=1.9.0-
>openai) (2.27.1)
Requirement already satisfied: build>=1.0.3 in c:\users\kirth\appdata\
roaming\python\python313\site-packages (from chromadb) (1.4.0)
Requirement already satisfied: pybase64>=1.4.1 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb) (1.4.3)
Requirement already satisfied: uvicorn>=0.18.3 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from
uvicorn[standard]>=0.18.3->chromadb) (0.40.0)
Requirement already satisfied: numpy>=1.22.5 in c:\programdata\
anaconda3\lib\site-packages (from chromadb) (2.1.3)
Requirement already satisfied: posthog<6.0.0,>=2.4.0 in c:\users\
kirth\appdata\roaming\python\python313\site-packages (from chromadb)
(5.4.0)
Requirement already satisfied: onnxruntime>=1.14.1 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb)
(1.24.1)
Requirement already satisfied: opentelemetry-api>=1.2.0 in c:\users\
kirth\appdata\roaming\python\python313\site-packages (from chromadb)
(1.39.1)
Requirement already satisfied: opentelemetry-exporter-otlp-proto-
grpc>=1.2.0 in c:\users\kirth\appdata\roaming\python\python313\site-
packages (from chromadb) (1.39.1)
Requirement already satisfied: opentelemetry-sdk>=1.2.0 in c:\users\
kirth\appdata\roaming\python\python313\site-packages (from chromadb)
(1.39.1)
Requirement already satisfied: tokenizers>=0.13.2 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb)
```

```
(0.22.2)
Requirement already satisfied: pypika>=0.48.9 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb)
(0.51.1)
Requirement already satisfied: overrides>=7.3.1 in c:\programdata\
anaconda3\lib\site-packages (from chromadb) (7.4.0)
Requirement already satisfied: importlib-resources in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb) (6.5.2)
Requirement already satisfied: grpcio>=1.58.0 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb)
(1.75.1)
Requirement already satisfied: bcrypt>=4.0.1 in c:\programdata\
anaconda3\lib\site-packages (from chromadb) (4.3.0)
Requirement already satisfied: typer>=0.9.0 in c:\programdata\
anaconda3\lib\site-packages (from chromadb) (0.9.0)
Requirement already satisfied: kubernetes>=28.1.0 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb)
(35.0.0)
Requirement already satisfied: tenacity>=8.2.3 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb) (9.1.2)
Requirement already satisfied: pyyaml>=6.0.0 in c:\programdata\
anaconda3\lib\site-packages (from chromadb) (6.0.2)
Requirement already satisfied: mmh3>=4.0.1 in c:\users\kirth\appdata\
roaming\python\python313\site-packages (from chromadb) (5.2.0)
Requirement already satisfied: orjson>=3.9.12 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from chromadb)
(3.11.7)
Requirement already satisfied: rich>=10.11.0 in c:\programdata\
anaconda3\lib\site-packages (from chromadb) (13.9.4)
Requirement already satisfied: jsonschema>=4.19.0 in c:\programdata\
anaconda3\lib\site-packages (from chromadb) (4.23.0)
Requirement already satisfied: six>=1.5 in c:\programdata\anaconda3\
lib\site-packages (from posthog<6.0.0,>=2.4.0->chromadb) (1.17.0)
Requirement already satisfied: python-dateutil>=2.2 in c:\programdata\
anaconda3\lib\site-packages (from posthog<6.0.0,>=2.4.0->chromadb)
(2.9.0.post0)
Requirement already satisfied: backoff>=1.10.0 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from
posthog<6.0.0,>=2.4.0->chromadb) (2.2.1)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\programdata\
anaconda3\lib\site-packages (from requests>=2.26.0->tiktoken) (2.3.0)
Requirement already satisfied: transformers<6.0.0,>=4.41.0 in c:\
users\kirth\appdata\roaming\python\python313\site-packages (from
sentence-transformers) (5.0.0)
Requirement already satisfied: huggingface-hub>=0.20.0 in c:\users\
kirth\appdata\roaming\python\python313\site-packages (from sentence-
transformers) (1.3.7)
Requirement already satisfied: torch>=1.11.0 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from sentence-
transformers) (2.8.0)
Requirement already satisfied: scikit-learn in c:\programdata\
anaconda3\lib\site-packages (from sentence-transformers) (1.6.1)
Requirement already satisfied: scipy in c:\programdata\anaconda3\lib\
site-packages (from sentence-transformers) (1.15.3)
Requirement already satisfied: filelock in c:\programdata\anaconda3\
lib\site-packages (from transformers<6.0.0,>=4.41.0->sentence-
transformers) (3.17.0)
Requirement already satisfied: packaging>=20.0 in c:\programdata\
anaconda3\lib\site-packages (from transformers<6.0.0,>=4.41.0-
>sentence-transformers) (24.2)
Requirement already satisfied: typer-slim in c:\users\kirth\appdata\
roaming\python\python313\site-packages (from
transformers<6.0.0,>=4.41.0->sentence-transformers) (0.20.0)
Requirement already satisfied: safetensors>=0.4.3 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from
transformers<6.0.0,>=4.41.0->sentence-transformers) (0.7.0)
Requirement already satisfied: fsspec>=2023.5.0 in c:\programdata\
anaconda3\lib\site-packages (from huggingface-hub>=0.20.0->sentence-
transformers) (2025.3.2)
Requirement already satisfied: hf-xet<2.0.0,>=1.2.0 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from huggingface-
hub>=0.20.0->sentence-transformers) (1.2.0)
Requirement already satisfied: shellingham in c:\programdata\
anaconda3\lib\site-packages (from huggingface-hub>=0.20.0->sentence-
transformers) (1.5.0)
Requirement already satisfied: pyproject_hooks in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from build>=1.0.3-
>chromadb) (1.2.0)
Requirement already satisfied: colorama in c:\programdata\anaconda3\
lib\site-packages (from build>=1.0.3->chromadb) (0.4.6)
Requirement already satisfied: cffi>=1.12 in c:\programdata\anaconda3\
lib\site-packages (from cryptography>=36.0.0->pdfminer.six==20251230-
>pdfplumber) (1.17.1)
Requirement already satisfied: pycparser in c:\programdata\anaconda3\
lib\site-packages (from cffi>=1.12->cryptography>=36.0.0-
>pdfminer.six==20251230->pdfplumber) (2.21)
Requirement already satisfied: attrs>=22.2.0 in c:\programdata\
anaconda3\lib\site-packages (from jsonschema>=4.19.0->chromadb)
(24.3.0)
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in
c:\programdata\anaconda3\lib\site-packages (from jsonschema>=4.19.0-
>chromadb) (2023.7.1)
Requirement already satisfied: referencing>=0.28.4 in c:\programdata\
anaconda3\lib\site-packages (from jsonschema>=4.19.0->chromadb)
(0.30.2)
Requirement already satisfied: rpds-py>=0.7.1 in c:\programdata\
anaconda3\lib\site-packages (from jsonschema>=4.19.0->chromadb)
(0.22.3)
```

```
Requirement already satisfied: websocket-client!=0.40.0,!=0.41.*,!
=0.42.*,>=0.32.0 in c:\programdata\anaconda3\lib\site-packages (from
kubernetes>=28.1.0->chromadb) (1.8.0)
Requirement already satisfied: requests-oauthlib in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from
kubernetes>=28.1.0->chromadb) (2.0.0)
Requirement already satisfied: durationpy>=0.7 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from
kubernetes>=28.1.0->chromadb) (0.10)
Requirement already satisfied: flatbuffers in c:\users\kirth\appdata\
roaming\python\python313\site-packages (from onnxruntime>=1.14.1-
>chromadb) (25.9.23)
Requirement already satisfied: protobuf in c:\programdata\anaconda3\
lib\site-packages (from onnxruntime>=1.14.1->chromadb) (5.29.3)
Requirement already satisfied: sympy in c:\programdata\anaconda3\lib\
site-packages (from onnxruntime>=1.14.1->chromadb) (1.13.3)
Requirement already satisfied: importlib-metadata<8.8.0,>=6.0 in c:\
programdata\anaconda3\lib\site-packages (from opentelemetry-
api>=1.2.0->chromadb) (8.5.0)
Requirement already satisfied: zipp>=3.20 in c:\programdata\anaconda3\
lib\site-packages (from importlib-metadata<8.8.0,>=6.0->opentelemetry-
api>=1.2.0->chromadb) (3.21.0)
Requirement already satisfied: googleapis-common-protos~=1.57 in c:\
users\kirth\appdata\roaming\python\python313\site-packages (from
opentelemetry-exporter-otlp-proto-grpc>=1.2.0->chromadb) (1.72.0)
Requirement already satisfied: opentelemetry-exporter-otlp-proto-
common==1.39.1 in c:\users\kirth\appdata\roaming\python\python313\
site-packages (from opentelemetry-exporter-otlp-proto-grpc>=1.2.0-
>chromadb) (1.39.1)
Requirement already satisfied: opentelemetry-proto==1.39.1 in c:\
users\kirth\appdata\roaming\python\python313\site-packages (from
opentelemetry-exporter-otlp-proto-grpc>=1.2.0->chromadb) (1.39.1)
Requirement already satisfied: opentelemetry-semantic-
conventions==0.60b1 in c:\users\kirth\appdata\roaming\python\
python313\site-packages (from opentelemetry-sdk>=1.2.0->chromadb)
(0.60b1)
Requirement already satisfied: markdown-it-py>=2.2.0 in c:\
programdata\anaconda3\lib\site-packages (from rich>=10.11.0->chromadb)
(2.2.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in c:\
programdata\anaconda3\lib\site-packages (from rich>=10.11.0->chromadb)
(2.19.1)
Requirement already satisfied: mdurl~=0.1 in c:\programdata\anaconda3\
lib\site-packages (from markdown-it-py>=2.2.0->rich>=10.11.0-
>chromadb) (0.1.0)
Requirement already satisfied: networkx in c:\programdata\anaconda3\
lib\site-packages (from torch>=1.11.0->sentence-transformers) (3.4.2)
Requirement already satisfied: jinja2 in c:\programdata\anaconda3\lib\
site-packages (from torch>=1.11.0->sentence-transformers) (3.1.6)
```

```
Requirement already satisfied: setuptools in c:\programdata\anaconda3\
lib\site-packages (from torch>=1.11.0->sentence-transformers) (72.1.0)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in c:\programdata\
anaconda3\lib\site-packages (from sympy->onnxruntime>=1.14.1-
>chromadb) (1.3.0)
Requirement already satisfied: click<9.0.0,>=7.1.1 in c:\programdata\
anaconda3\lib\site-packages (from typer>=0.9.0->chromadb) (8.1.8)
Requirement already satisfied: httptools>=0.6.3 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from
uvicorn[standard]>=0.18.3->chromadb) (0.7.1)
Requirement already satisfied: python-dotenv>=0.13 in c:\programdata\
anaconda3\lib\site-packages (from uvicorn[standard]>=0.18.3->chromadb)
(1.1.0)
Requirement already satisfied: watchfiles>=0.13 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from
uvicorn[standard]>=0.18.3->chromadb) (1.1.1)
Requirement already satisfied: websockets>=10.4 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from
uvicorn[standard]>=0.18.3->chromadb) (16.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\programdata\
anaconda3\lib\site-packages (from jinja2->torch>=1.11.0->sentence-
transformers) (3.0.2)
Requirement already satisfied: oauthlib>=3.0.0 in c:\users\kirth\
appdata\roaming\python\python313\site-packages (from requests-
oauthlib->kubernetes>=28.1.0->chromadb) (3.3.1)
Requirement already satisfied: joblib>=1.2.0 in c:\programdata\
anaconda3\lib\site-packages (from scikit-learn->sentence-transformers)
(1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in c:\programdata\
anaconda3\lib\site-packages (from scikit-learn->sentence-transformers)
(3.5.0)
```

```python
# Import all the required Libraries

import pdfplumber
from pathlib import Path
import pandas as pd
from operator import itemgetter
import json
import tiktoken
import openai
import chromadb
```

```python
# Mount Google Drive
# from google.colab import drive
# drive.mount('/content/drive', force_remount=True)
```

```
Mounted at /content/drive
```

2. Read, Process, and Chunk the PDF Files

We will be using pdfplumber to read and process the PDF files.

`pdfplumber` allows for better parsing of the PDF file as it can read various elements of the PDF apart from the plain text, such as, tables, images, etc. It also offers wide functionaties and visual debugging features to help with advanced preprocessing as well.

```
# Define the path of the PDF
single_pdf_path = r"C:\Users\kirth\Downloads\PolicyDocumentsforrag\
HDFC-Life-Easy-Health-101N110V03-Policy-Bond-Single-Pay.pdf"
# single_pdf_path = "/content/drive/My Drive/HelpMate/Policy
Documents/HDFC-Life-Group-Poorna-Suraksha-101N137V02-Policy-
Document.pdf"
```

2.1 Reading a single PDF file and exploring it through pdfplumber

```
# Open the PDF file
with pdfplumber.open(single_pdf_path) as pdf:

    # Get one of the pages from the PDF and examine it
    single_page = pdf.pages[6]

    # Extract text from the first page
    text = single_page.extract_text()

    # Extract tables from the first page
    tables = single_page.extract_tables()

    # Print the extracted text
    print(text)
```

```
apart; and
ii. Requiring continuous permanent supplementary oxygen therapy for
hypoxemia;
and
iii. Arterial blood gas analysis with partial oxygen pressure of
55mmHg or less
(PaO2 < 55mmHg); and
iv. Dyspnea at rest.
Confirmation by a consultant physician acceptable to the Company of
the loss of
independent existence due to Illness or trauma, which has lasted for a
minimum
period of 6 months and results in a permanent inability to perform at
least three (3) of
the Activities of Daily Living (either with or without the use of
mechanical
equipment, special devices or other aids and adaptations in use for
disabled persons).
```

For the purpose of this benefit, the word “permanent”, shall mean beyond the hope of

recovery with current medical knowledge and technology.

Activities of Daily Living are:-

a) Washing: the ability to wash in the bath or shower (including getting into and out

of the bath or shower) or wash satisfactorily by other means.

Loss of

b) Dressing: the ability to put on, take off, secure and unfasten all garments and, as

Independe

11 appropriate, any braces, artificial limbs or other surgical appliances.

nt

c) Transferring: the ability to move from a bed or an upright chair or wheelchair and

Existence

vice versa.

d) Mobility: The ability to move indoors from room to room on level surfaces.

e) Toileting: the ability to use the lavatory or otherwise manage bowel and bladder

functions so as to maintain a satisfactory level of personal hygiene.

f) Feeding: the ability to feed oneself once food has been prepared and made

available.

The following is excluded:

Any injury or loss as a result of War, invasion, hostilities (whether war is declared or

not), civil war, rebellion, revolution or taking part in a riot or civil commotion

Total, permanent and irreversible loss of all vision in both eyes as a result of illness or

accident.

The Blindness is evidenced by:

12 Blindness i. corrected visual acuity being 3/60 or less in both eyes or ;

ii. the field of vision being less than 10 degrees in both eyes.

The diagnosis of blindness must be confirmed and must not be correctable by aids or

surgical procedure.

Third There must be third-degree burns with scarring that cover at least 20% of the body’s

13 Degree surface area. The diagnosis must confirm the total area involved using standardized,

Burns clinically accepted, body surface area charts covering 20% of the body surface area.

Accidental head injury resulting in permanent Neurological deficit to be assessed no

```
sooner than 3 months from the date of the accident. This diagnosis
must be supported
by unequivocal findings on Magnetic Resonance Imaging,
Computerized Tomography, or other reliable imaging techniques. The
accident must
be caused solely and directly by accidental, violent, external and
visible means and
Major independently of all other causes.
14 Head The Accidental Head injury must result in an inability to
perform at least three (3)of
Trauma the following Activities of Daily Living either with or without
the use of mechanical
equipment, special devices or other aids and adaptations in use for
disabled persons.
For the purpose of this benefit, the word “permanent” shall mean
beyond the scope of
recovery with current medical knowledge and technology.
The following are excluded:
i. Spinal cord injury
Motor Motor neurone disease diagnosed by a specialist medical
practitioner as spinal
Neurone muscular atrophy, progressive bulbar palsy, amyotrophic
lateral sclerosis or primary
Disease lateral sclerosis. There must be progressive degeneration of
corticospinal tracts and
15
With anterior horn cells or bulbar efferent neurons. There must be
current significant and
Permanent permanent functional neurological impairment with objective
evidence of motor
Symptoms dysfunction that has persisted for a continuous period of at
least 3 months.
16 Multiple The unequivocal diagnosis of Definite Multiple Sclerosis
confirmed and evidenced by
```

```
# View the table in the page, if any

tables[0]
```

```
[['',
  '',
  'apart; and\nii. Requiring continuous permanent supplementary oxygen
therapy for hypoxemia;\nand\niii. Arterial blood gas analysis with
partial oxygen pressure of 55mmHg or less\n(PaO2 < 55mmHg); and\niv.
Dyspnea at rest.'],
 ['11',
  'Loss of\nIndepende\nnt\nExistence',
  'Confirmation by a consultant physician acceptable to the Company of
the loss of\nindependent existence due to Illness or trauma, which has
lasted for a minimum\nperiod of 6 months and results in a permanent
inability to perform at least three (3) of\nthe Activities of Daily
Living (either with or without the use of mechanical\nequipment,
special devices or other aids and adaptations in use for disabled
persons).\nFor the purpose of this benefit, the word “permanent”,
shall mean beyond the hope of\nrecovery with current medical knowledge
and technology.\nActivities of Daily Living are:-\na) Washing: the
ability to wash in the bath or shower (including getting into and out\
nof the bath or shower) or wash satisfactorily by other means.\nb)
Dressing: the ability to put on, take off, secure and unfasten all
garments and, as\nappropriate, any braces, artificial limbs or other
surgical appliances.\nc) Transferring: the ability to move from a bed
or an upright chair or wheelchair and\nvice versa.\nd) Mobility: The
ability to move indoors from room to room on level surfaces.\ne)
Toileting: the ability to use the lavatory or otherwise manage bowel
and bladder\nfunctions so as to maintain a satisfactory level of
personal hygiene.\nf) Feeding: the ability to feed oneself once food
has been prepared and made\navailable.\nThe following is excluded:\
nAny injury or loss as a result of War, invasion, hostilities (whether
war is declared or\nnot), civil war, rebellion, revolution or taking
part in a riot or civil commotion'],
 ['12',
  'Blindness',
  'Total, permanent and irreversible loss of all vision in both eyes
as a result of illness or\naccident.\nThe Blindness is evidenced by:\
ni. corrected visual acuity being 3/60 or less in both eyes or ;\nii.
the field of vision being less than 10 degrees in both eyes.\nThe
diagnosis of blindness must be confirmed and must not be correctable
by aids or\nsurgical procedure.'],
 ['13',
  'Third\nDegree\nBurns',
  'There must be third-degree burns with scarring that cover at least
20% of the body’s\nsurface area. The diagnosis must confirm the total
area involved using standardized,\nclinically accepted, body surface
area charts covering 20% of the body surface area.'],
 ['14',
  'Major\nHead\nTrauma',
  'Accidental head injury resulting in permanent Neurological deficit
to be assessed no\nsooner than 3 months from the date of the accident.
This diagnosis must be supported\nby unequivocal findings on Magnetic
Resonance Imaging,\nComputerized Tomography, or other reliable imaging
techniques. The accident must\nbe caused solely and directly by
accidental, violent, external and visible means and\nindependently of
all other causes.\nThe Accidental Head injury must result in an
inability to perform at least three (3)of\nthe following Activities of
Daily Living either with or without the use of mechanical\nequipment,
special devices or other aids and adaptations in use for disabled
persons.\nFor the purpose of this benefit, the word “permanent” shall
mean beyond the scope of\nrecovery with current medical knowledge and
technology.\nThe following are excluded:\ni. Spinal cord injury'],
 ['15',
  'Motor\nNeurone\nDisease\nWith\nPermanent\nSymptoms',
  'Motor neurone disease diagnosed by a specialist medical
practitioner as spinal\nmuscular atrophy, progressive bulbar palsy,
amyotrophic lateral sclerosis or primary\nlateral sclerosis. There
must be progressive degeneration of corticospinal tracts and\nanterior
horn cells or bulbar efferent neurons. There must be current
significant and\npermanent functional neurological impairment with
objective evidence of motor\ndysfunction that has persisted for a
continuous period of at least 3 months.'],
 ['16',
  'Multiple',
  'The unequivocal diagnosis of Definite Multiple Sclerosis confirmed
and evidenced by']]
```

2.2 Extracting text from multiple PDFs

Let's now try and read multiple documents, extract text from them using appropriate preprocessing, and store them in a dataframe

```
# Define the path where all pdf documents are present
pdf_path = r"C:\Users\kirth\Downloads\PolicyDocumentsforrag"

# Function to check whether a word is present in a table or not for
segregation of regular text and tables

def check_bboxes(word, table_bbox):
    # Check whether word is inside a table bbox.
    l = word['x0'], word['top'], word['x1'], word['bottom']
    r = table_bbox
    return l[0] > r[0] and l[1] > r[1] and l[2] < r[2] and l[3] < r[3]

# Function to extract text from a PDF file.
# 1. Declare a variable p to store the iteration of the loop that will
help us store page numbers alongside the text
# 2. Declare an empty list 'full_text' to store all the text files
# 3. Use pdfplumber to open the pdf pages one by one
# 4. Find the tables and their locations in the page
# 5. Extract the text from the tables in the variable 'tables'
# 6. Extract the regular words by calling the function check_bboxes()
and checking whether words are present in the table or not
# 7. Use the cluster_objects utility to cluster non-table and table
words together so that they retain the same chronology as in the
original PDF
# 8. Declare an empty list 'lines' to store the page text
# 9. If a text element in present in the cluster, append it to
'lines', else if a table element is present, append the table
# 10. Append the page number and all lines to full_text, and increment
'p'
# 11. When the function has iterated over all pages, return the
```

```
'full_text' list

def extract_text_from_pdf(pdf_path):
    p = 0
    full_text = []


    with pdfplumber.open(pdf_path) as pdf:
        for page in pdf.pages:
            page_no = f"Page {p+1}"
            text = page.extract_text()

            tables = page.find_tables()
            table_bboxes = [i.bbox for i in tables]
            tables = [{'table': i.extract(), 'top': i.bbox[1]} for i
in tables]
            non_table_words = [word for word in page.extract_words()
if not any(
                [check_bboxes(word, table_bbox) for table_bbox in
table_bboxes])]
            lines = []

            for cluster in
pdfplumber.utils.cluster_objects(non_table_words + tables,
itemgetter('top'), tolerance=5):

                if 'text' in cluster[0]:
                    try:
                        lines.append(' '.join([i['text'] for i in
cluster]))
                    except KeyError:
                        pass

                elif 'table' in cluster[0]:
                    lines.append(json.dumps(cluster[0]['table']))


            full_text.append([page_no, " ".join(lines)])
            p +=1

    return full_text
```

Now that we have defined the function for extracting the text and tables from a PDF, let's iterate and call this function for all the PDFs in our drive and store them in a list.

```
# Define the directory containing the PDF files
pdf_directory = Path(pdf_path)

# Initialize an empty list to store the extracted texts and document
names
```

```python
data = []

# Loop through all files in the directory
for pdf_path in pdf_directory.glob("*.pdf"):

    # Process the PDF file
    print(f"...Processing {pdf_path.name}")

    # Call the function to extract the text from the PDF
    extracted_text = extract_text_from_pdf(pdf_path)

    # Convert the extracted list to a PDF, and add a column to store document names
    extracted_text_df = pd.DataFrame(extracted_text, columns=['Page No.', 'Page_Text'])
    extracted_text_df['Document Name'] = pdf_path.name

    # Append the extracted text and document name to the list
    data.append(extracted_text_df)

    # Print a message to indicate progress
    print(f"Finished processing {pdf_path.name}")

# Print a message to indicate all PDFs have been processed
print("All PDFs have been processed.")
```

```
...Processing HDFC-Life-Easy-Health-101N110V03-Policy-Bond-Single-Pay.pdf
Finished processing HDFC-Life-Easy-Health-101N110V03-Policy-Bond-Single-Pay.pdf
...Processing HDFC-Life-Group-Poorna-Suraksha-101N137V02-Policy-Document.pdf
Finished processing HDFC-Life-Group-Poorna-Suraksha-101N137V02-Policy-Document.pdf
...Processing HDFC-Life-Group-Term-Life-Policy.pdf
Finished processing HDFC-Life-Group-Term-Life-Policy.pdf
...Processing HDFC-Life-Sampoorna-Jeevan-101N158V04-Policy-Document (1).pdf
Finished processing HDFC-Life-Sampoorna-Jeevan-101N158V04-Policy-Document (1).pdf
...Processing HDFC-Life-Sanchay-Plus-Life-Long-Income-Option-101N134V19-Policy-Document.pdf
Finished processing HDFC-Life-Sanchay-Plus-Life-Long-Income-Option-101N134V19-Policy-Document.pdf
...Processing HDFC-Life-Smart-Pension-Plan-Policy-Document-Online.pdf
Finished processing HDFC-Life-Smart-Pension-Plan-Policy-Document-Online.pdf
...Processing HDFC-Surgicare-Plan-101N043V01.pdf
Finished processing HDFC-Surgicare-Plan-101N043V01.pdf
All PDFs have been processed.
```

```
# Concatenate all the DFs in the list 'data' together

insurance_pdfs_data = pd.concat(data, ignore_index=True)

insurance_pdfs_data
```

```
    Page No.                                          Page_Text  \
0     Page 1  Part A <<Date>> <<Policyholder’s Name>> <<Poli...
1     Page 2  Agency/Intermediary Contact Details: <<Agency/...
2     Page 3  POLICY DOCUMENT- HDFC LIFE EASY HEALTH Unique ...
3     Page 4  [[null, "<< dd/mm/yyyy >>"], ["Appointee's Add...
4     Page 5  Part B Definitions The following capitalised t...
..       ...                                                ...
212  Page 11  HDFC Standard Life Insurance Company Limited H...
213  Page 12  HDFC Standard Life Insurance Company Limited H...
214  Page 13  HDFC Standard Life Insurance Company Limited H...
215  Page 14  HDFC Standard Life Insurance Company Limited H...
216  Page 15  HDFC Standard Life Insurance Company Limited H...

                                         Document Name
0    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...
1    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...
2    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...
3    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...
4    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...
..                                                 ...
212                 HDFC-Surgicare-Plan-101N043V01.pdf
213                 HDFC-Surgicare-Plan-101N043V01.pdf
214                 HDFC-Surgicare-Plan-101N043V01.pdf
215                 HDFC-Surgicare-Plan-101N043V01.pdf
216                 HDFC-Surgicare-Plan-101N043V01.pdf

[217 rows x 3 columns]
```

```
# Check one of the extracted page texts to ensure that the text has
been correctly read

insurance_pdfs_data.Page_Text[2]
```

```
'POLICY DOCUMENT- HDFC LIFE EASY HEALTH Unique Identification Number:
<<101N110V03>> Your Policy is a Single Premium paying non
participating non linked fixed benefit health plan. This document is
the evidence of a contract between HDFC Life Insurance Company Limited
and the Policyholder as described in the Policy Schedule given below.
This Policy is based on the Proposal made by the within named
Policyholder and submitted to the Company along with the required
documents, declarations, statements, any response given to the Short
Medical Questionnaire (SMQ) by the Life Assured, and other information
received by the Company from the Policyholder, Life Assured or on
behalf of the Policyholder. This Policy is effective upon receipt and
realisation, by the Company, of the consideration payable as Premium
```

```
under the Policy. This Policy is written under and will be governed by
the applicable laws in force in India and all Premiums and Benefits
are expressed and payable in Indian Rupees. POLICY SCHEDULE Policy
Number: <<__________>> Client id:<<_________>> Policyholder Details
Address Life Assured Details [["Name", "<< >>"], ["Date of Birth", "<<
dd/mm/yyyy >>"], ["Age on the Date of Risk\\nCommencement", "<< >>
years"], ["Age Admitted", "<<Yes/No>>"]] Policy Details [["Date of
Commencement of Policy", "<<Date>>"], ["Date of Risk Commencement",
"<< Risk Commencement Date >>"], ["Date of Issue/Inception of Policy",
"<< Issue Date>>"], ["Plan Option", "<<>>"], ["Sum Insured", "<< >>"],
["Single Premium", "Rs. << >>"], ["Premium Paying Term", "Single"],
["Policy Term", "5 years"], ["Cover Ceasing Date", "<< dd/mm/yyyy
>>"]] The premium amount is exclusive of taxes, other statutory levies
and any underwriting extra premium. NOMINATION SCHEDULE [["Nominee\\
u2019s Name", "<<Nominee-1 >>", "<<Nominee-2 >>"], ["Date of Birth of
Nominee", "<< dd/mm/yyyy >>", "<< dd/mm/yyyy >>"], ["Nomination
Percentage", "<< >> %", "<< >> %"], ["Nominee\'s Address", "<< >>",
"<< >>"], ["Appointee\\u2019s Name\\n(Applicable where the nominee is
a\\nminor)", "<< >>", null]]'
```

```
# Let's also check the length of all the texts as there might be some
empty pages or pages with very few words that we can drop

insurance_pdfs_data['Text_Length'] =
insurance_pdfs_data['Page_Text'].apply(lambda x: len(x.split(' ')))

insurance_pdfs_data['Text_Length']
```

```
0      508
1       85
2      298
3       63
4      514
      ... 
212    316
213    353
214    348
215    496
216    171
Name: Text_Length, Length: 217, dtype: int64
```

```
# Retain only the rows with a text length of at least 10

insurance_pdfs_data =
insurance_pdfs_data.loc[insurance_pdfs_data['Text_Length'] >= 10]
insurance_pdfs_data
```

```
    Page No.                                                 Page_Text  \
0     Page 1  Part A <<Date>> <<Policyholder’s Name>> <<Poli...
1     Page 2  Agency/Intermediary Contact Details: <<Agency/...
2     Page 3  POLICY DOCUMENT- HDFC LIFE EASY HEALTH Unique ...
```

```
3     Page 4  [[null, "<< dd/mm/yyyy >>"], ["Appointee's Add...
4     Page 5  Part B Definitions The following capitalised t...
..       ...                                                ...
212  Page 11  HDFC Standard Life Insurance Company Limited H...
213  Page 12  HDFC Standard Life Insurance Company Limited H...
214  Page 13  HDFC Standard Life Insurance Company Limited H...
215  Page 14  HDFC Standard Life Insurance Company Limited H...
216  Page 15  HDFC Standard Life Insurance Company Limited H...

                                         Document Name  Text_Length
0    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...          508
1    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...           85
2    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...          298
3    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...           63
4    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...          514
..                                                 ...          ...
212                 HDFC-Surgicare-Plan-101N043V01.pdf          316
213                 HDFC-Surgicare-Plan-101N043V01.pdf          353
214                 HDFC-Surgicare-Plan-101N043V01.pdf          348
215                 HDFC-Surgicare-Plan-101N043V01.pdf          496
216                 HDFC-Surgicare-Plan-101N043V01.pdf          171

[210 rows x 4 columns]
```

```
# Store the metadata for each page in a separate column

insurance_pdfs_data['Metadata'] = insurance_pdfs_data.apply(lambda x:
{'Policy_Name': x['Document Name'][:-4], 'Page_No.': x['Page No.']},
axis=1)
```

```
C:\Users\kirth\AppData\Local\Temp\ipykernel_32896\1081778557.py:3:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  insurance_pdfs_data['Metadata'] = insurance_pdfs_data.apply(lambda
x: {'Policy_Name': x['Document Name'][:-4], 'Page_No.': x['Page
No.']}, axis=1)
```

```
insurance_pdfs_data
```

```
    Page No.                                          Page_Text  \
0     Page 1  Part A <<Date>> <<Policyholder’s Name>> <<Poli...
1     Page 2  Agency/Intermediary Contact Details: <<Agency/...
2     Page 3  POLICY DOCUMENT- HDFC LIFE EASY HEALTH Unique ...
3     Page 4  [[null, "<< dd/mm/yyyy >>"], ["Appointee's Add...
4     Page 5  Part B Definitions The following capitalised t...
..       ...                                                ...
```

```
212  Page 11  HDFC Standard Life Insurance Company Limited H...
213  Page 12  HDFC Standard Life Insurance Company Limited H...
214  Page 13  HDFC Standard Life Insurance Company Limited H...
215  Page 14  HDFC Standard Life Insurance Company Limited H...
216  Page 15  HDFC Standard Life Insurance Company Limited H...

                                         Document Name  Text_Length  \
0    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...          508   
1    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...           85   
2    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...          298   
3    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...           63   
4    HDFC-Life-Easy-Health-101N110V03-Policy-Bond-S...          514   
..                                                 ...          ...   
212                 HDFC-Surgicare-Plan-101N043V01.pdf          316   
213                 HDFC-Surgicare-Plan-101N043V01.pdf          353   
214                 HDFC-Surgicare-Plan-101N043V01.pdf          348   
215                 HDFC-Surgicare-Plan-101N043V01.pdf          496   
216                 HDFC-Surgicare-Plan-101N043V01.pdf          171   

                                              Metadata  
0    {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...  
1    {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...  
2    {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...  
3    {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...  
4    {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...  
..                                                 ...  
212  {'Policy_Name': 'HDFC-Surgicare-Plan-101N043V0...  
213  {'Policy_Name': 'HDFC-Surgicare-Plan-101N043V0...  
214  {'Policy_Name': 'HDFC-Surgicare-Plan-101N043V0...  
215  {'Policy_Name': 'HDFC-Surgicare-Plan-101N043V0...  
216  {'Policy_Name': 'HDFC-Surgicare-Plan-101N043V0...  

[210 rows x 5 columns]
```

This concludes the chunking aspect also, as we can see that mostly the pages contain few hundred words, maximum going upto 1000. So, we don't need to chunk the documents further; we can perform the embeddings on individual pages. This strategy makes sense for 2 reasons:

1. The way insurance documents are generally structured, you will not have a lot of extraneous information in a page, and all the text pieces in that page will likely be interrelated.
2. We want to have larger chunk sizes to be able to pass appropriate context to the LLM during the generation layer.

```
# This is formatted as code
```

3. Generate and Store Embeddings using OpenAI and ChromaDB

In this section, we will embed the pages in the dataframe through OpenAI's `text-embedding-ada-002` model, and store them in a ChromaDB collection.

```
# Set the API key


# sk-proj-VSH9yPRbDO_YpmigvvLo6kxGNsVsz7yISsEaYS55Nkw1OwYwjp8-
cK1Ws_YkWGNzsP-MWqE-h0T3BlbkFJhJFDJm3MQOtJxdKPKvbYE-
QiBoJc_95dxQTb2ea_VgtcIjcloHdmBKGbp-5GTpzG-lixSN29oA


# filepath = "/content/drive/MyDrive/HelpMate/"

# with open(filepath + "OpenAI_API_Key.txt", "r") as f:
  # openai.api_key = ' '.join(f.readlines())

from openai import OpenAI
client=OpenAI(api_key="sk-proj-VSH9yPRbKPKvbYE-
QiBoJc_95dxQTb2ea_VgtcIjcloHdmBKGbp-5GTpzG-lixSN29oA")

# Import the OpenAI Embedding Function into chroma

from chromadb.utils.embedding_functions import OpenAIEmbeddingFunction

# Define the path where chroma collections will be stored

chroma_data_path = '/content/drive/MyDrive/HelpMate/ChromaDB_Data'

import chromadb

# Call PersistentClient()

client = chromadb.PersistentClient()

# Set up the embedding function using the OpenAI embedding model

model = "text-embedding-ada-002"
# embedding_function = OpenAIEmbeddingFunction(api_key=openai.api_key,
model_name=model)
KEY = "sk-proj-VSH9yPRbDO_Ypmigvv-QiBoJc_95dxQTb2ea_VgtcIjcloHdmBKGbp-
5GTpzG-lixSN29oA"


embedding_function = OpenAIEmbeddingFunction(
    api_key=KEY,
    model_name=model
)
```

```
# Initialise a collection in chroma and pass the embedding_function to
it so that it used OpenAI embeddings to embed the documents

insurance_collection =
client.get_or_create_collection(name='RAG_on_Insurance',
embedding_function=embedding_function)

# Convert the page text and metadata from your dataframe to lists to
be able to pass it to chroma

documents_list = insurance_pdfs_data["Page_Text"].tolist()
metadata_list = insurance_pdfs_data['Metadata'].tolist()

# Add the documents and metadata to the collection alongwith generic
integer IDs. You can also feed the metadata information as IDs by
combining the policy name and page no.

insurance_collection.add(
    documents= documents_list,
    ids = [str(i) for i in range(0, len(documents_list))],
    metadatas = metadata_list
)

# Let's take a look at the first few entries in the collection

insurance_collection.get(
    ids = ['0','1','2'],
    include = ['embeddings', 'documents', 'metadatas']
)
```

```
{'ids': ['0', '1', '2'],
 'embeddings': array([[ 0.00618955,  0.01557669, -0.00219752, ..., -
0.0076225 ,
         -0.01586859, -0.04749963],
        [-0.00155362,  0.00827508, -0.02308898, ..., -0.01373283,
         -0.00755437, -0.04759317],
        [-0.01620748,  0.00742617,  0.00622009, ..., -0.01230467,
         -0.00234101, -0.03610094]]),
 'documents': ['Part A <<Date>> <<Policyholder’s Name>>
<<Policyholder’s Address>> <<Policyholder’s Contact Number>> Dear
<<Policyholder’s Name>>, Sub: Your Policy no. << >> We are glad to
inform you that your proposal has been accepted and the HDFC Life Easy
Health (“Policy”) being this document, has been issued. We have made
every effort to design your Policy in a simple format. We have
highlighted items of importance so that you may recognize them easily.
Policy document: As an evidence of the insurance contract between HDFC
Life Insurance Company Limited and you, the Policy is enclosed
herewith. Please preserve this document safely and also inform your
nominees about the same. A copy of your proposal form and other
relevant documents submitted by you is also enclosed for your
information and record. Cancellation in the Free-Look Period: << In
```

```
case you are not agreeable to any of the terms and conditions stated
in the Policy, you have the option to return the Policy to us for
cancellation stating the reasons thereof, within 30 days from the date
of receipt of the Policy as your Policy is an electronic Policy /
purchased through Distance Marketing mode. On receipt of your letter
along with the original Policy (original Policy Document is not
required for policies in dematerialised form), we shall arrange to
refund the Premium paid by you, subject to deduction of the
proportionate risk Premium for the period of cover and the expenses
incurred by us for medical examination (if any) and stamp duty
charges. / In case you are not agreeable to any of the terms and
conditions stated in the Policy, you have the option to return the
Policy to us for cancellation stating the reasons thereof, within 15
days from the date of receipt of the Policy. On receipt of your letter
along with the original Policy (original Policy Document is not
required for policies in dematerialised form), we shall arrange to
refund the Premium paid by you, subject to deduction of the
proportionate risk Premium for the period of cover and the expenses
incurred by us for medical examination (if any) and stamp duty
charges. >> Contacting us: The address for correspondence is specified
below. To enable us to serve you better, you are requested to quote
your Policy number in all future correspondence. In case you are keen
to know more about our products and services, we would request you to
talk to our Certified Financial Consultant (Insurance Agent) who has
advised you while taking this Policy. The details of your Certified
Financial Consultant including contact details are listed below. To
contact us in case of any grievance, please refer to Part G. In case
you are not satisfied with our response, you can also approach the
Insurance Ombudsman in your region. Thanking you for choosing HDFC
Life Insurance Company Limited and looking forward to serving you in
the years ahead, Yours sincerely, << Designation of the Authorised
Signatory >> Branch Address: <<Branch Address>> Agency/Intermediary
Code: <<Agency/Intermediary Code>> Agency/Intermediary Name:
<<Agency/Intermediary Name>> Agency/Intermediary Telephone Number:
<<Agency/Intermediary mobile & landline number>>',
  'Agency/Intermediary Contact Details: <<Agency/Intermediary
address>> th Address for Correspondence: HDFC Life Insurance Company
Limited, 11 Floor Lodha Excelus, Apollo Mills Compound, N.M. Joshi
Marg, Mahalaxmi, Mumbai-400011. th Regd. Off: Lodha Excelus, 13 Floor,
Apollo Mills Compound, N. M. Joshi Marg, Mahalaxmi, Mumbai - 400 011.
Call 1860-267-9999 (local charges apply). DO NOT prefix any country
code e.g. +91 or 00. Available Mon- Sat from 10 am to 7 pm | Email –
service@hdfclife.com | NRIservice@hdfclife.com (For NRI customers
only) Visit – www.hdfclife.com . CIN: L65110MH2000PLC128245.',
  'POLICY DOCUMENT- HDFC LIFE EASY HEALTH Unique Identification
Number: <<101N110V03>> Your Policy is a Single Premium paying non
participating non linked fixed benefit health plan. This document is
the evidence of a contract between HDFC Life Insurance Company Limited
and the Policyholder as described in the Policy Schedule given below.
```

```
This Policy is based on the Proposal made by the within named
Policyholder and submitted to the Company along with the required
documents, declarations, statements, any response given to the Short
Medical Questionnaire (SMQ) by the Life Assured, and other information
received by the Company from the Policyholder, Life Assured or on
behalf of the Policyholder. This Policy is effective upon receipt and
realisation, by the Company, of the consideration payable as Premium
under the Policy. This Policy is written under and will be governed by
the applicable laws in force in India and all Premiums and Benefits
are expressed and payable in Indian Rupees. POLICY SCHEDULE Policy
Number: <<__________>> Client id:<<_________>> Policyholder Details
Address Life Assured Details [["Name", "<< >>"], ["Date of Birth", "<<
dd/mm/yyyy >>"], ["Age on the Date of Risk\\nCommencement", "<< >>
years"], ["Age Admitted", "<<Yes/No>>"]] Policy Details [["Date of
Commencement of Policy", "<<Date>>"], ["Date of Risk Commencement",
"<< Risk Commencement Date >>"], ["Date of Issue/Inception of Policy",
"<< Issue Date>>"], ["Plan Option", "<<>>"], ["Sum Insured", "<< >>"],
["Single Premium", "Rs. << >>"], ["Premium Paying Term", "Single"],
["Policy Term", "5 years"], ["Cover Ceasing Date", "<< dd/mm/yyyy
>>"]] The premium amount is exclusive of taxes, other statutory levies
and any underwriting extra premium. NOMINATION SCHEDULE [["Nominee\\
u2019s Name", "<<Nominee-1 >>", "<<Nominee-2 >>"], ["Date of Birth of
Nominee", "<< dd/mm/yyyy >>", "<< dd/mm/yyyy >>"], ["Nomination
Percentage", "<< >> %", "<< >> %"], ["Nominee\'s Address", "<< >>",
"<< >>"], ["Appointee\\u2019s Name\\n(Applicable where the nominee is
a\\nminor)", "<< >>", null]]'],
 'uris': None,
 'included': ['embeddings', 'documents', 'metadatas'],
 'data': None,
 'metadatas': [{'Policy_Name': 'HDFC-Life-Easy-Health-101N110V03-
Policy-Bond-Single-Pay',
   'Page_No.': 'Page 1'},
  {'Page_No.': 'Page 2',
   'Policy_Name': 'HDFC-Life-Easy-Health-101N110V03-Policy-Bond-
Single-Pay'},
  {'Page_No.': 'Page 3',
   'Policy_Name': 'HDFC-Life-Easy-Health-101N110V03-Policy-Bond-
Single-Pay'}]}
```

```
cache_collection =
client.get_or_create_collection(name='Insurance_Cache',
embedding_function=embedding_function)
```

```
cache_collection.peek()
```

```
{'ids': [],
 'embeddings': array([], dtype=float64),
 'documents': [],
 'uris': None,
 'included': ['metadatas', 'documents', 'embeddings'],
 'data': None,
 'metadatas': []}
```

4. Semantic Search with Cache

In this section, we will perform a semantic search of a query in the collections embeddings to get several top semantically similar results.

```
# Read the user query

query = input()
```

```
 WHAT IS THE POLICY ON EYE ISSUES?
```

```
# Searh the Cache collection first
# Query the collection against the user query and return the top 20
results

cache_results = cache_collection.query(
    query_texts=query,
    n_results=1
)

cache_results
```

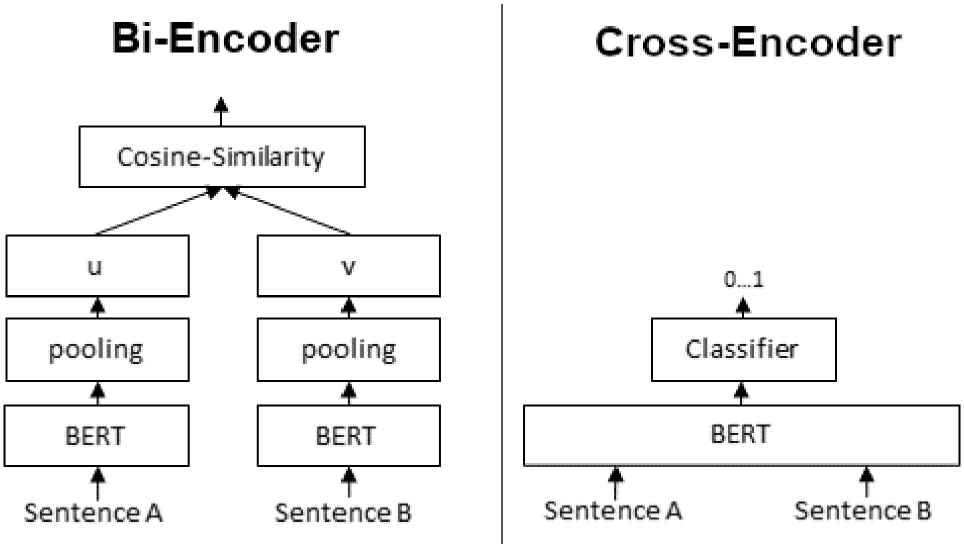
```
{'ids': [[]],
 'embeddings': None,
 'documents': [[]],
 'uris': None,
 'included': ['metadatas', 'documents', 'distances'],
 'data': None,
 'metadatas': [[]],
 'distances': [[]]}
```

```
results = insurance_collection.query(
query_texts=query,
n_results=10
)
# results.items()

# Implementing Cache in Semantic Search

import json
import pandas as pd

threshold = 0.2
results_df = pd.DataFrame()

# -------------------------------
# CHECK CACHE MISS
```

```python
# -------------------------------
if (
    len(cache_results["distances"]) == 0 or
    len(cache_results["distances"][0]) == 0 or
    cache_results["distances"][0][0] > threshold
):

    # Query main collection
    results = insurance_collection.query(
        query_texts=[query],
        n_results=10
    )

    print("Not found in cache. Found in main collection.")

    # Store safely in cache (serialize complex objects)
    cache_collection.add(
        documents=[query],
        ids=[query],
        metadatas=[{
            "ids": json.dumps(results["ids"][0]),
            "documents": json.dumps(results["documents"][0]),
            "distances": json.dumps(results["distances"][0]),
            "metadatas": json.dumps(results["metadatas"][0])
        }]
    )

    # Create DataFrame from main results
    results_df = pd.DataFrame({
        "IDs": results["ids"][0],
        "Documents": results["documents"][0],
        "Distances": results["distances"][0],
        "Metadatas": results["metadatas"][0]
    })


# -------------------------------
# CACHE HIT
# -------------------------------
else:

    print("Found in cache!")

    cached_data = cache_results["metadatas"][0][0]

    results_df = pd.DataFrame({
        "IDs": json.loads(cached_data["ids"]),
        "Documents": json.loads(cached_data["documents"]),
        "Distances": json.loads(cached_data["distances"]),
        "Metadatas": json.loads(cached_data["metadatas"])
    })


results_df

Not found in cache. Found in main collection.

   IDs                                          Documents  
Distances  \
0   15  7. Routine eye tests, any Dental Treatment or ...   0.457204   

1   29  Annexure III Provisions regarding Policy not b...   0.459669   

2   57  Annexure III Section 45 – Policy shall not be ...   0.466973   

3  156  HDFC Life Sanchay Plus (UIN – 101N134V19) – Ap...   0.471183   

4  103  PART D Policy Servicing Related Aspects D.1. F...   0.473182   

5   12  In case you are not agreeable to any of the pr...   0.474492   

6   17  (2) We reserve the right to change any of thes...   0.475405   

7  190  HDFC Life Smart Pension Plan 101L164V02 – Term...   0.475863   

8  117  ANNEXURE – I Section 45 – Policy shall not be ...   0.477507   

9   90  ANNEXURE - B Section 45 – Policy shall not be ...   0.479387   


                                           Metadatas  
0  {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...  
1  {'Page_No.': 'Page 31', 'Policy_Name': 'HDFC-L...  
2  {'Page_No.': 'Page 26', 'Policy_Name': 'HDFC-L...  
3  {'Page_No.': 'Page 26', 'Policy_Name': 'HDFC-L...  
4  {'Policy_Name': 'HDFC-Life-Sampoorna-Jeevan-10...  
5  {'Page_No.': 'Page 14', 'Policy_Name': 'HDFC-L...  
6  {'Page_No.': 'Page 19', 'Policy_Name': 'HDFC-L...  
7  {'Policy_Name': 'HDFC-Life-Smart-Pension-Plan-...  
8  {'Policy_Name': 'HDFC-Life-Sampoorna-Jeevan-10...  
9  {'Policy_Name': 'HDFC-Life-Group-Term-Life-Pol...  
```

5. Re-Ranking with a Cross Encoder

Re-ranking the results obtained from your semantic search can sometime significantly improve the relevance of the retrieved results. This is often done by passing the query paired with each of the retrieved responses into a cross-encoder to score the relevance of the response w.r.t. the query.



```
# Import the CrossEncoder library from sentence_transformers

from sentence_transformers import CrossEncoder, util

# Initialise the cross encoder model

cross_encoder = CrossEncoder('cross-encoder/ms-marco-MiniLM-L-6-v2')
```

{"model_id":"61338d5683ef4dcaa691375fef5fa41f","version_major":2,"version_minor":0}

```
C:\Users\kirth\AppData\Roaming\Python\Python313\site-packages\
huggingface_hub\file_download.py:130: UserWarning: `huggingface_hub`
cache-system uses symlinks by default to efficiently store duplicated
files but your machine does not support them in C:\Users\kirth\.cache\
huggingface\hub\models--cross-encoder--ms-marco-MiniLM-L-6-v2. Caching
files will still work but in a degraded version that might require
more space on your disk. This warning can be disabled by setting the
`HF_HUB_DISABLE_SYMLINKS_WARNING` environment variable. For more
details, see https://huggingface.co/docs/huggingface_hub/how-to-
cache#limitations.
To support symlinks on Windows, you either need to activate Developer
Mode or to run Python as an administrator. In order to activate
developer mode, see this article:
https://docs.microsoft.com/en-us/windows/apps/get-started/enable-your-
device-for-development
  warnings.warn(message)
```

{"model_id":"e644ccc0f21d4c6989741e7793217b5d","version_major":2,"version_minor":0}

{"model_id":"e61e85d4751f4936a1dcd2ece14963b9","version_major":2,"version_minor":0}

```
BertForSequenceClassification LOAD REPORT from: cross-encoder/ms-marco-MiniLM-L-6-v2
Key                          | Status     |  |
-----------------------------+------------+--+-
bert.embeddings.position_ids | UNEXPECTED |  |

Notes:
- UNEXPECTED    :can be ignored when loading from different task/architecture; not ok if you expect identical arch.
```

{"model_id":"bc3b6822fe864a16823fe1938e48a786","version_major":2,"version_minor":0}

{"model_id":"c351900c372341f2bd6c28f7cf06d104","version_major":2,"version_minor":0}

{"model_id":"fc771ae92697407f826fb2ab49871dbb","version_major":2,"version_minor":0}

{"model_id":"fa9fa8db74af4be1bc96cef2bbec562a","version_major":2,"version_minor":0}

{"model_id":"352c3aa07c854877bddf0b6f598af1e5","version_major":2,"version_minor":0}

```python
# Test the cross encoder model

scores = cross_encoder.predict([['Does the insurance cover diabetic patients?', 'The insurance policy covers some pre-existing conditions including diabetes, heart diseases, etc. The policy does not howev'],
                                ['Does the insurance cover diabetic patients?', 'The premium rates for various age groups are given as follows. Age group (<18 years): Premium rate']])

scores
```

```
array([  3.8467622, -11.252879 ], dtype=float32)
```

```python
# Input (query, response) pairs for each of the top 20 responses received from the semantic search to the cross encoder
# Generate the cross_encoder scores for these pairs

cross_inputs = [[query, response] for response in results_df['Documents']]
cross_rerank_scores = cross_encoder.predict(cross_inputs)

cross_rerank_scores
```

```
array([ -6.3455553,  -8.56617  ,  -8.375369 ,  -9.081698 ,  -
8.395123 ,
        -10.108143 , -10.240528 ,  -8.726196 ,  -8.349312 ,  -
8.328604 ],
      dtype=float32)

# Store the rerank_scores in results_df

results_df['Reranked_scores'] = cross_rerank_scores

results_df

   IDs                                          Documents
Distances  \
0   15  7. Routine eye tests, any Dental Treatment or ...   0.457204

1   29  Annexure III Provisions regarding Policy not b...   0.459669

2   57  Annexure III Section 45 – Policy shall not be ...   0.466973

3  156  HDFC Life Sanchay Plus (UIN – 101N134V19) – Ap...   0.471183

4  103  PART D Policy Servicing Related Aspects D.1. F...   0.473182

5   12  In case you are not agreeable to any of the pr...   0.474492

6   17  (2) We reserve the right to change any of thes...   0.475405

7  190  HDFC Life Smart Pension Plan 101L164V02 – Term...   0.475863

8  117  ANNEXURE – I Section 45 – Policy shall not be ...   0.477507

9   90  ANNEXURE - B Section 45 – Policy shall not be ...   0.479387


                                           Metadatas  Reranked_scores

0  {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...        -6.345555

1  {'Page_No.': 'Page 31', 'Policy_Name': 'HDFC-L...        -8.566170

2  {'Page_No.': 'Page 26', 'Policy_Name': 'HDFC-L...        -8.375369

3  {'Page_No.': 'Page 26', 'Policy_Name': 'HDFC-L...        -9.081698

4  {'Policy_Name': 'HDFC-Life-Sampoorna-Jeevan-10...        -8.395123

5  {'Page_No.': 'Page 14', 'Policy_Name': 'HDFC-L...       -10.108143

6  {'Page_No.': 'Page 19', 'Policy_Name': 'HDFC-L...       -10.240528

7  {'Policy_Name': 'HDFC-Life-Smart-Pension-Plan-...        -8.726196
```

```
8  {'Policy_Name': 'HDFC-Life-Sampoorna-Jeevan-10...        -8.349312

9  {'Policy_Name': 'HDFC-Life-Group-Term-Life-Pol...        -8.328604

# Return the top 3 results from semantic search

top_3_semantic = results_df.sort_values(by='Distances')
top_3_semantic[:3]

  IDs                                          Documents  Distances  \
0  15  7. Routine eye tests, any Dental Treatment or ...   0.457204
1  29  Annexure III Provisions regarding Policy not b...   0.459669
2  57  Annexure III Section 45 – Policy shall not be ...   0.466973

                                           Metadatas  Reranked_scores

0  {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...        -6.345555

1  {'Page_No.': 'Page 31', 'Policy_Name': 'HDFC-L...        -8.566170

2  {'Page_No.': 'Page 26', 'Policy_Name': 'HDFC-L...        -8.375369

# Return the top 3 results after reranking

top_3_rerank = results_df.sort_values(by='Reranked_scores',
ascending=False)
top_3_rerank[:3]

   IDs                                          Documents
Distances  \
0   15  7. Routine eye tests, any Dental Treatment or ...   0.457204

9   90  ANNEXURE - B Section 45 – Policy shall not be ...   0.479387

8  117  ANNEXURE – I Section 45 – Policy shall not be ...   0.477507


                                           Metadatas  Reranked_scores

0  {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...        -6.345555

9  {'Policy_Name': 'HDFC-Life-Group-Term-Life-Pol...        -8.328604

8  {'Policy_Name': 'HDFC-Life-Sampoorna-Jeevan-10...        -8.349312

top_3_RAG = top_3_rerank[["Documents", "Metadatas"]][:3]

top_3_RAG

                                           Documents  \
0  7. Routine eye tests, any Dental Treatment or ...
```

```
9  ANNEXURE - B Section 45 – Policy shall not be ...
8  ANNEXURE – I Section 45 – Policy shall not be ...

                                           Metadatas
0  {'Policy_Name': 'HDFC-Life-Easy-Health-101N110...
9  {'Policy_Name': 'HDFC-Life-Group-Term-Life-Pol...
8  {'Policy_Name': 'HDFC-Life-Sampoorna-Jeevan-10...
```

6. Retrieval Augmented Generation

Now that we have the final top search results, we can pass it to an GPT 3.5 along with the user query and a well-engineered prompt, to generate a direct answer to the query along with citations, rather than returning whole pages/chunks.

```
# Define the function to generate the response. Provide a
comprehensive prompt that passes the user query and the top 3 results
to the model
from openai import OpenAI
client=OpenAI(api_key="sk-proj-VSH9yPRxdKPKvbYE-
QiBoJc_95dxQTb2ea_VgtcIjcloHdmBKGbp-5GTpzG-lixSN29oA")

def generate_response(query, results_df):
    """
    Generate a response using GPT-3.5's ChatCompletion based on the
user query and retrieved information.
    """
    messages = [
                {"role": "system", "content":  "You are a helpful
assistant in the insurance domain who can effectively answer user
queries about insurance policies and documents."},
                {"role": "user", "content": f"""You are a helpful
assistant in the insurance domain who can effectively answer user
queries about insurance policies and documents.
                                                You have a question
asked by the user in '{query}' and you have some search results from a
corpus of insurance documents in the dataframe '{top_3_RAG}'. These
search results are essentially one page of an insurance document that
may be relevant to the user query.

                                                The column 'documents'
inside this dataframe contains the actual text from the policy
document and the column 'metadata' contains the policy name and source
page. The text inside the document may also contain tables in the
format of a list of lists where each of the nested lists indicates a
row.

                                                Use the documents in
'{top_3_RAG}' to answer the query '{query}'. Frame an informative
answer and also, use the dataframe to return the relevant policy names
and page numbers as citations.

                                                Follow the guidelines
below when performing the task.
                                                1. Try to provide
relevant/accurate numbers if available.
                                                2. You don’t have to
necessarily use all the information in the dataframe. Only choose
information that is relevant.
                                                3. If the document
text has tables with relevant information, please reformat the table
and return the final information in a tabular in format.
                                                3. Use the Metadatas
columns in the dataframe to retrieve and cite the policy name(s) and
page numbers(s) as citation.
                                                4. If you can't
provide the complete answer, please also provide any information that
will help the user to search specific sections in the relevant cited
documents.
                                                5. You are a customer
facing assistant, so do not provide any information on internal
workings, just answer the query directly.

                                                The generated response
should answer the query directly addressing the user and avoiding
additional information. If you think that the query is not relevant to
the document, reply that the query is irrelevant. Provide the final
response as a well-formatted and easily readable text along with the
citation. Provide your complete response first with all information,
and then provide the citations.
                                                """},
              ]

    response = client.chat.completions.create(
        model="gpt-3.5-turbo",
        messages=messages
    )

    return response.choices[0].message.content.split('\n')

# Generate the response

response = generate_response(query, top_3_RAG)

# Print the response

print("\n".join(response))
```

```
The policy on eye issues states that routine eye tests are covered,
but dental treatment or glasses are not covered under the policy
mentioned in the document. For detailed information, please refer to
```

```
Annexure – B and Annexure – I under Section 45 of the respective
policies.

Here is the information in a tabular format for quick reference:

| Policy Name                    | Page Number |
|-------------------------------|--------------|
| HDFC-Life-Easy-Health-101N110 | Annexure – B |
| HDFC-Life-Group-Term-Life-Policy | Annexure – I |
| HDFC-Life-Sampoorna-Jeevan-10 | Annexure – I |

Please refer to the mentioned sections in the respective policies for
more details regarding the coverage of eye issues.
```