Instructions Document

- **Component Web Url:**
  Article Recommendation System Website
  - Our Article Recommendation System has been deployed to the above weblink. Follow this link to user and verify our component.

For code Test and Verification follow the following steps

- **Dataset:**
  - To test our implementation from beginning, please download the wikipedia dataset from the following link
  - **Wikipedia Dataset Link**
- **Setup**:
  - To run our algorithm implementation steps below, you need to install "**gensim"** python module. You can follow the following link for download and installation.
  - **Gensim Weblink**

- **Top 100 large file index finder**
  - **Filename: largefileindexfinder.py** under "**/code"**
  - **Description:** Choosing the top 100 large files to calculate distribution for.

- **PreProcessing Stage**
  - **Filename: make_wikicorpus** in the gensim module
  - **Description :** Does the preprocessing and generates three output files in the current directory: _bow.mm, _wordids.txt, _tfidf.mm
  - **Usage:** python -m gensim.scripts.make_wikicorpus ./enwiki-20161101-pages-articles.xml.bz2 ./

- **Generating the LDA model**
  - **Filename : LdaModelling.py** under **"/code"**
  - **Description :** Generates the LDA model and saves it in the binary file 'lda_model.out'
  - **Usage:** python LdaModelling.py

- **Generating the topic distribution across documents**
  - **Filename : GenerateTopicDis.py** under **“/code”**
  - **Description :** Generates the topic distributions in the documents and stores it in the output file ‘topic_dist.out’
  - **Usage :** python GenerateTopicDis.py

- **The MinHeap helper methods**
  - **FileName : MinHeap.py and Node.py**
  - **Description :** The MinHeap Implementation is used to keep track of the 10 nearest neighbours of the documents we want to show in our recommendation system.
  - **Usage:** Compile these two files , python Node.py MinHeap.py before proceeding to the next stage.

- **The Recommendation System**
  - **FileName : Recommend_new.py**
  - **Description :** This generates the 10 most similar documents to the given documents we want to show in the output of our recommendation system.
  - **Usage:** python Recommend_new.py

- **Wiki Extractor:**
  - **Filename: “WikiExtractor.py”** under **“code/Wikipedia Dataset XML to html page converter/”**
  - **Descriptions:** This code takes whole wikipedia dataset xml file as an input and generates the content of each topic as a separate HTML file also including the actual wikipedia link inside it
  - **Usage:** Run the code with command line arguments as follows
    - -o <Output folder path> <path to wikipedia dataset xml bz2 file>
    - Ex -
      - --html --links --sections --lists -o E:\WikipediaDataset\dataset_HTML\ E:\WikipediaDataset\enwiki-20161101-pages-articles.xml.bz2

- **Recommendation link inclusion**
  - **Filename: filereducer.py** under ”**/code**”
  - **Description:** Including the generated each 10 recommendations to the corresponding each of the 100 files as links