# Latent Dirichlet Allocation (LDA) README: Text Mining Replication Steps

## Overview

This project uses **Latent Dirichlet Allocation (LDA)** for topic modeling on complaint data. The pipeline includes lemmatization, seed generation, LDA modeling, wordcloud creation, perplexity scoring, and merging outputs.

Before starting, we must download the topicmodels package from Dropbox and replace the default `topicmodels` stopwords list with a custom list.

Please also note that the way in which wordclouds are generated from data is purely random, and words, though the same across many iterations, may appear larger or smaller than they do in each wordcluod in the manuscript. Howver, the data output behind the wordclouds is always the same, regardless of iteration.

## NOTE: topicmodels package steps

- The topicmodels package is deprecated. To correctly download and pip install topicmodels, please do the following:
    1. Download the topicmodels package from Dropbox at the following link: https://www.dropbox.com/scl/fi/9mafn99bs173du0gkqwy8/topic-modelling-tools-master.zip?rlkey=bgs3d08klv3mxrhnjv3tvforv&e=1&dl=0. It is also provided in the replication package in ./Text_Mining/topic-modelling-tools-master/
    2. Inside downloaded package, go into the samplers/samplers_lda.pyx and make the following changes:
        - Replace xrange with range in every place xrange appears
        - Navigate to the places where instructions say to "uncomment" the code to use numpy instead of gsl. This is in the mulitnomial section of the code. Uncomment the line used for numpy randomization.
    3. Copy directory path to the topicmodels package
    4. Open a terminal and run the command: pip install <path_to_topicmodels>

## NOTE: Stopwords Replacement

- Custom stopwords file: `stopwords-long-new`
- Location: `./Text Mining/topic-modelling-tools-master/topicmodels`
- Replace the original `stopwords-long` file in your installed `topicmodels` package directory with this file and rename it to match the original filename.

## Requirements for topicmodels:

- Python 3.9.23
    - nltk 3.7
    - matplotlib 3.9.4
    - matplot 0.1.9
    - pandas 2.3.1
    - wordcloud 1.9.4
    - numpy 1.23.5
    - collection 0.1.6

---

## Descriptions of programs

### 1.1 Lemmatization

**Script:** `<subset>/Python Code/LDA Preprocess Lemmatize.py`
**Inputs:** `<subset>/Raw Data/<subset>complaints with year.txt`
**Output:** `<subset>/Processed Data/<subset>Complaints lemmatized.csv`

**Process:**

- Convert complaints to `RawDocs` object.
- Count total and unique tokens at various stages.
- Remove 1-character tokens and stopwords.
- POS tagging and lemmatization using WordNet.
- Save lemmatized output to CSV.

---

### 1.2 Seed Generation

**Script:** `<subset>/Python Code/LDA Seeds.py`
**Inputs:** `<subset>/Processed Data/<subset>Complaints lemmatized.csv`
**Outputs:** `./<subset>/LDA Outputs/<iteration>/seed.csv`

**Process:**

- Convert lemmatized complaints to `RawDocs`.

- Clean tokens (remove 1-character words, stopwords, and low tf-idf words).
- Create `LDA.Gibbs` object for each iteration.
- Extract and save initial topic allocation (seeds).

---

## 1.3 LDA and Wordclouds

### 1.3.1 LDA

**Script:** `<subset>/Python Code/LDA Lemmas <number of topics>Topics <seed>.py`
**Inputs:**

- `./<subset>/Processed Data/<subset>Complaints lemmatized.csv`
- `./<subset>/LDA Outputs/<iteration>/seed.csv`

**Outputs:**
Saved to `./<subset>/LDA Outputs/<iteration>/`

- `topic description.csv`
- `tt.csv`
- `dt.csv`
- `dict.csv`
- `lda final output.csv`
- `lda final output.dta`
  Common outputs (for all iterations):
- `df ranking.csv`, `df ranking.png`
- `tfidf ranking.csv`, `tfidf ranking.png`
- `Perplexity.csv`

**Process:**

- Convert complaints to `RawDocs` and count words.
- Clean data based on tf-idf thresholds.
- Create `LDA.Gibbs` object with fixed seed (2001), 4000 burns, thinning interval of 50, and 80 samples.
- Generate and save various outputs including topic descriptions and perplexity.

---

### 1.3.2 Wordclouds

**Script:** `<subset>/Python Code/LDA Lemmas <number of topics>Topics <seed>.py`
**Inputs:**

- `df ranking.csv`
- `topic description.csv`
- `lmroman10-regular.otf` (Font for rendering)

**Outputs:**

- `./<subset>/wordclouds/<subset>.png`
- `./<subset>/wordclouds/<iteration>/<subset> <topic number>.png`

Process:

- Generate wordclouds from document frequency and topic descriptions.
- Convert topic probabilities to frequency-like values.
- Word size indicates frequency/probability; color is randomized.

---

## 1.4 Perplexity Scores

**Script:** `./Air Subset/Python Code/Perplexity Scores.py`
**Inputs:**

- `./Air Subset/LDA Outputs/<iteration>/Perplexity.csv`

**Output:**

- `./Air Subset/LDA Outputs/10 Topics Perplexities.csv`

**Process:**

- Combine perplexity chains from different seeds.

- Label each chain and compute mean/std. dev.

---

### 1.5 Merging Outputs

**Script:** `./Air Subset/Python Code/Air 10 Topics Final Output.py`
**Inputs:**

- `lda final output.csv` from `./Air Subset/LDA Outputs/6000 tfidf 10 topics seed 3/`
- `Air Complaints Categorized By Year.csv` from `./Air Subset/Processed Data/`

**Outputs:**

- `Final Output Merged.csv`
- `Final Output Merged.dta`
  Saved to: `./Air Subset/LDA Outputs/6000 tfidf 10 topics seed 3/`

**Process:**

- Merge datasets using identifiers (`year`, `ID`, `CIN`, `RN`)
- Count words and compute topic density variance for each complaint.

---

## Replication Steps

1. Open the paths.py script in the ./Full_Subset path. Redirect the directories to the replicator's directories
2. Open main_text_mining_LDA.py. Run this script.

## Notes

- All file paths are relative and follow the `<subset>` naming convention.
- Only the **Air** subset with **10 topics** and **seed 3** is used for final output merging and evaluation.

## Mapping of tables and figures to programs

> [NOTE] Table E1 is manually generated and encompasses print statments outputted from the following scripts: Air_LDA_Preprocess_Lemmatize, Air_LDA_Lemmatize_Topics_10_seed_3,Water_LDA_Preprocess_Lemmatize, Water_LDA_Lemmatize_Topics, LDA_Preprocess_Lemmatize, LDA_Lemmatize_Topics_10, Waste_LDA_Preprocess_Lemmatize, Waste_LDA_Lemmatize_Topics_10_seed_3

| Name of Figure/Table | Name(s) and Location(s) for Figure/Table | Do File(s) that Create Figure/Table |
|---|---|---|
| Figure C7 | text_mining/Environmental_Complaints/Air_Subset/wordclouds<br>text_mining/Environmental_Complaints/Waste_Subset/wordclouds<br>text_mining/Environmental_Complaints/Water_Subset/wordclouds | text_mining/Environmental_Complaints/Air_Subset/Python_Code/Air_LDA_Lemmatize_Topics_10_seed_3.py<br>text_mining/Environmental_Complaints/Waste_Subset/Python_Code/Waste_LDA_Lemmatize_Topics_10_seed_3.py<br>text_mining/Environmental_Complaints/Water_Subset/Python_Code/Water_LDA_Lemmatize_Topics_10.py |
| Table E1 | Manually generated | text_mining/Environmental_Complaints/Air_Subset/Python_Code/Air_LDA_Preprocess_Lemmatize.py<br>text_mining/Environmental_Complaints/Air_Subset/Python_Code/Air_LDA_Lemmatize_Topics_10_seed_3.py<br>text_mining/Environmental_Complaints/Waste_Subset/Python_Code/Waste_LDA_Preprocess_Lemmatize.py<br>text_mining/Environmental_Complaints/Waste_Subset/Python_Code/Waste_LDA_Lemmatize_Topics_10_seed_3.py<br>text_mining/Environmental_Complaints/Water_Subset/Python_Code/Water_LDA_Preprocess_Lemmatize.py<br>text_mining/Environmental_Complaints/Water_Subset/Python_Code/Water_LDA_Lemmatize_Topics_10.py |

| Name of Figure/Table | Name(s) and Location(s) for Figure/Table | Do File(s) that Create Figure/Table |
|---|---|---|
| Figure E1 | text_mining/Environmental_Complaints/Air_Subset/wordclouds/10_Topics_seed_3 | text_mining/Environmental_Complaints/Air_Subset/Python_Code/Air_LDA_Lemmatize_Topics_10_seed_3.py |
| Figure E2 | text_mining/Environmental_Complaints/Waste_Subset/wordclouds/10_Topics_seed_3 | text_mining/Environmental_Complaints/Waste_Subset/Python_Code/Waste_LDA_Lemmatize_Topics_10_seed_3.py |
| Figure E3 | text_mining/Environmental_Complaints/Water_Subset/worclouds/10_Topics | text_mining/Environmental_Complaints/Water_Subset/Python_Code/Water_LDA_Lemmatize_Topics_10.py |

## References

Hansen, S. (2024). *Topic Modelling with Latent Dirichlet Allocation using Gibbs sampling*. https://github.com/sekhansen/topic-modelling-tools