CS 410 Final Project Progress Report: Topic Mining Healthcare Data

Completed Tasks

- ☒ Curate Dataset
  - ☒ Exported 20,000 labelled (positive/not positive) notes summarizing telehealth care encounters
  - ☒ Automated de-identification of PHI using DEID Software Package
  - ☒ Exported Sample Of De-Identification to work on Topic Mining

Pending Tasks

- ☐ Topic Mining
  - ☐ Prep De-Identification notes for Topic Mining tool
  - ☐ Run a series of topic mining trails changing the number of topics trying to be mined
  - ☐ Perform analysis of topic coverage for positive notes and non-positive notes
- ☐ Classifier
  - ☐ Create a classifier that can determine if a telehealth note summary had a positive outcome for the patient
  - ☐ Train on half of the de-identified dataset
  - ☐ Run on other half of the de-identified dataset and compare the results

Current Challenges/Issues Faced

Due to the export of the providers from medical systems containing a lot of invalid data such as names of lab tests, diseases and specialists, the list had to be manually scrubbed to avoid the DEID tool from falsely redacting it thinking they were doctor names. This was a manual tedious process that required review of a few thousand records.

Detailed Progress Updates

Curate Dataset

1. Exporting Labelled Dataset

Two CSV files, each containing 10,000 records was exported from the TimeDoc system. One file named `positive_encounters.csv` contained only notes that were labelled as a positive outcome due to the telehealth services while another file named `no_positive_encounters.csv` only contained notes that weren't labelled as a positive outcome for the patient. The format of the exported CSV files are as follows: `<note_id>,<patient_id>,<purpose>,<duration>,<note>` The `<purpose>` is an array of attributes of the telehealth encounter, it is selected from a pre-defined list and can provide insights to the actions of the telehealth encounter. The `<duration>` is the total amount of time the telehealth encounter took, and `<note>` is the free-text nursing note summarizing the encounter that we will be performing topic mining on.

2. Automated De-Identification of Protected Health Information (PHI)

To ensure we're adhering to HIPPA [1] we have to redact Protected Health Information (PHI). This was redacted using the De-Identification (DEID) Software Package [2]. For the DEID to be effective, it required creating separate files of patient names and identifiers `pid_patientname.txt`, doc-

[1] HIPPA Privacy Guidelines, https://www.physionet.org/content/deid/1.1/

[2] De-Identification Software Package, https://www.physionet.org/content/deid/1.1/

tor first names `doctor_first_names.txt`, doctor last names `doctor_last_names.txt`, locations `local_places.txt`, and company names `company_names.txt`. The `pid_patientname.txt` was created by referencing all the patients from the two exported CSV lists and curating a file formatted with each line as `<PATIENT_ID>||||<PATIENT_FIRST_NAME>||||<PATIENT_LAST_NAME>`. The `doctor_first_names.txt` and the `doctor_last_names.txt` files were created by referencing exporting each care team member such as their Primary Care Physician (PCP), Radiologist, etc. and writing each name to a new line. Both files were scrubbed for duplicates and invalid data. The `local_places.txt` was created by taking each address related for the patient and writing the city to a town to each line. The `company_names.txt` file was created by listing out the pharmacies and local healthcare organizations that the patient utilizes and writing each to a new line.

For the DEID to perform the redaction of PHI, it required to be fed the notes in a particular format. So the exported CSV file had to be transformed to the following format:

```
START_OF_RECORD=<PATIENT_ID>||||<DOCUMENT_ID>||||
<DOCUMENT_CONTENT>
||||END_OF_RECORD
```

We accomplished this transformation for both of the CSV exported files using a ruby script located at `deid_support/convert_csv_to_text.rb` and ran the following commands:

```
# convert csv files to deid text format
ruby deid_support/convert_csv_to_text.rb positive_encounters.csv
ruby deid_support/convert_csv_to_text.rb no_positive_encounters.csv
```

The output produced two files named `positive_encounters.text` and `no_positive_encounters.text` respectively. Afterwards the new text files were copied into the DEID directory and we ran the DEID perl script to remove the PHI using the following commands:

```
# redact PHI from text files
perl deid.pl positive_encounters deid-output.config
perl deid.pl no_positive_encounters deid-output.config
```

The output produced two PHI redacted files named `positive_encounters.res` and `no_positive_encounters.res`. To convert the files back into the CSV format, we used the following script located at `deid_support/convert_res_to_csv.rb` and ran the following commands:

```
# convert redacted res files to csv
ruby deid_support/convert_res_to_csv.rb \
  positive_encounters.res \
  positive_encounters.csv
ruby deid_support/convert_res_to_csv.rb \
  no_positive_encounters.res \
  no_positive_encounters.csv
```

The output produced two files named `positive_encounters.res.csv` and `no_positive_encounters.res.csv`.

3. Sample De-Identified Notes Data

Since the DEID is an automated tool, we had to account for the possibility on not redacting all PHI data. To minimize actual PHI distributed, 50 samples were taken from both the `positive_encounters.res` and `no_positive_encounters.res` file and manually verified to not contain PHI. After the verification, the sampled data was shared with the rest of the team so to create scripts that will perform the topic mining and analysis. This was accomplished by utilizing the `deid_support/sample_res.rb` and running the following commands:

```
# sample res files per manual review
```

```
ruby deid_support/sample_res.rb positive_encounters.res 50
ruby deid_support/sample_res.rb no_positive_encounters.res 50
```

The output produced two files only containing 50 redacted PHI documents named

```
positive_encounters.res-sample-50.res
no_positive_encounters.res-sample-50.res
```

After manual verification that all PHI was redacted, the sampled files were transformed to the original CSV format by running the following commands:

```
# convert sampled res files to CSV format
ruby deid_support/convert_res_to_csv.rb \
  positive_encounters.res-sample-50.res \
  positive_encounters.csv
ruby deid_support/convert_res_to_csv.rb \
  no_positive_encounters.res-sample-50.res \
  no_positive_encounters.csv
```

The output produced two files named `positive_encounters.res-sample-50.res.csv` and `no_positive_encounters.res-sample-50.res.csv`.

The four sampled redacted PHI documents can be provided at request by emailing rsw2@illinois.edu