CS 598 DLH Project Presentation

(Team ID: 150, Paper ID:135)

By Ratan Bajpai and Vijay Mishra

Paper Title and Reference

- Title: A disease inference method based on symptom extraction and bidirectional Long Short Term Memory networks. Guo et al.
- Reference: https://www.sciencedirect.com/science/article/abs/pii/S1046202319301033?via%3Dihub

General Problem

1. Disease prediction using clinical text in Electronic Health Records (EHRs)
2. The paper uses patient discharge summaries in the MIMIC III dataset to predict the diseases a patient may have based on his / her symptoms.

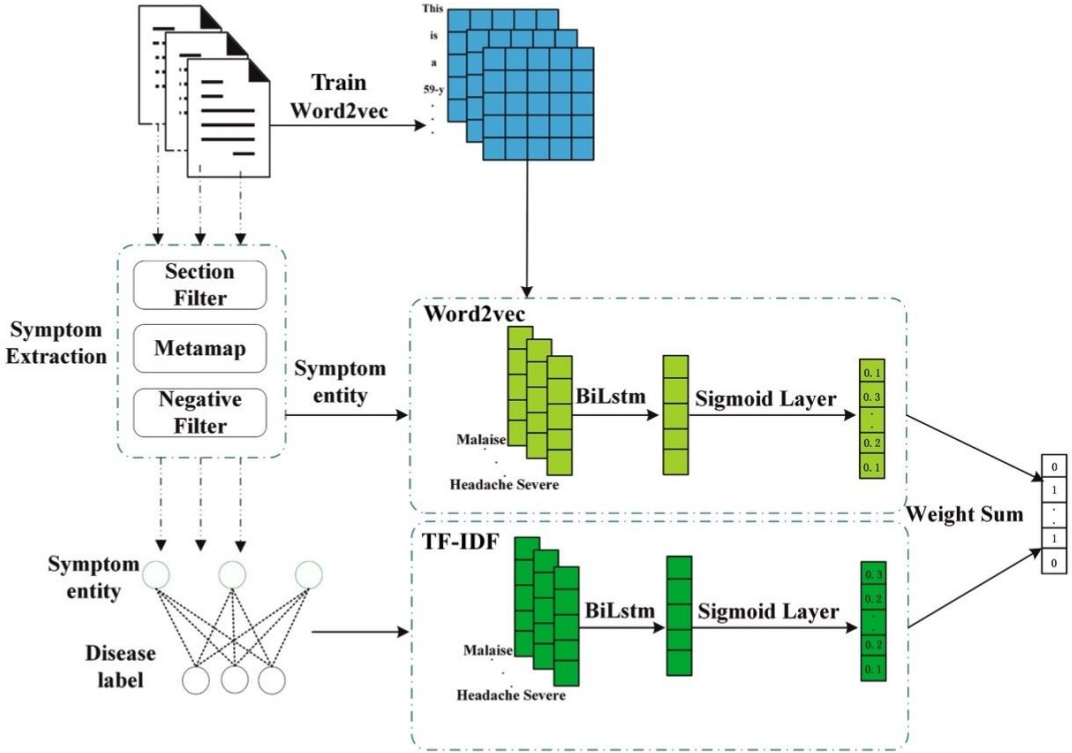
Specific Approach Taken

1. Symptoms are extracted from clinical text using MetaMap
2. Symptom embeddings are created using two approaches:
    a. First: TF-IDF based approach is used to calculate symptom vector S_i as follows:
        i. $\boldsymbol{S_i = (W_{i,1}, W_{i,2}, ..., W_{i,d})}$, where $W_{i,j}$ is defined below
        ii. $\boldsymbol{W_{i,j} = TF_{i,j} . log (N / D_i)}$, here $TF_{i,j}$ is the number of times symptom i is correlated with disease j in the discharge notes. N is total number of diseases considered, and D_i is the number of diseases associated with symptom i.
    b. Second: Word2Vec based approach in which first a Word2Vec model is trained using all the raw discharge summary notes. Then symptom vectors are generated using this trained model.

Specific Approach Taken Contd.

3. Two BiLSTM models are trained, one for each approach indicated in the previou slide. Inputs are symptom vector sequences generated using each approach, and output is the diseases diagnosed for the specific discharge note.
4. A weighted sum of both BiLSTM outputs is taken and disease output values above a specific threshold is the predicted output vector.
5. The paper claims that the combined approach of correlating symptoms with disease (TF-IDF) and symptoms with other symptoms (Word2Vec) gives better results than the predictions made by the individual approaches.

System Architecture



Results Claimed

Following are the main results claimed by the paper:

1. The proposed combined mechanism (TF-IDF + Word2vec) performs better than the system based on just TF-IDF in terms of Precision, Recall, F1-score and AUC.
2. The proposed combined mechanism (TF-IDF + Word2vec) performs better than the system based on just Word2vec in terms of Precision, Recall, F1-score and AUC.

Reproduction Attempt

In our reproduction attempt, we did the following:

1. Pre-processed discharge summary notes to remove non-symptom sections, and concepts in negated contexts.
2. Extracted symptoms from discharge summaries using MetaMap.
3. Used the TF-IDF technique for generating symptom embeddings, and trained a BiLSTM model for disease prediction using these embeddings (baseline 1).
4. Used the Word2Vec technique for generating symptom embeddings, and trained a second BiLSTM model for disease prediction using embeddings from Word2Vec (baseline 2).
5. Integrated the TF-IDF and Word2Vec based techniques and implemented an overall system for disease prediction using the combination of both approaches (TF-IDF + Word2Vec).
6. Tabulated the results from both the individual, and the combined approaches.

Dataset Used

We used the following tables and their relevant columns from the MIMIC III dataset:

1. NOTEEVENTS table which contains the discharge summaries text for hospital admission of each patient in the dataset.

row_id	subject_id	hadm_id	…	category	…	text

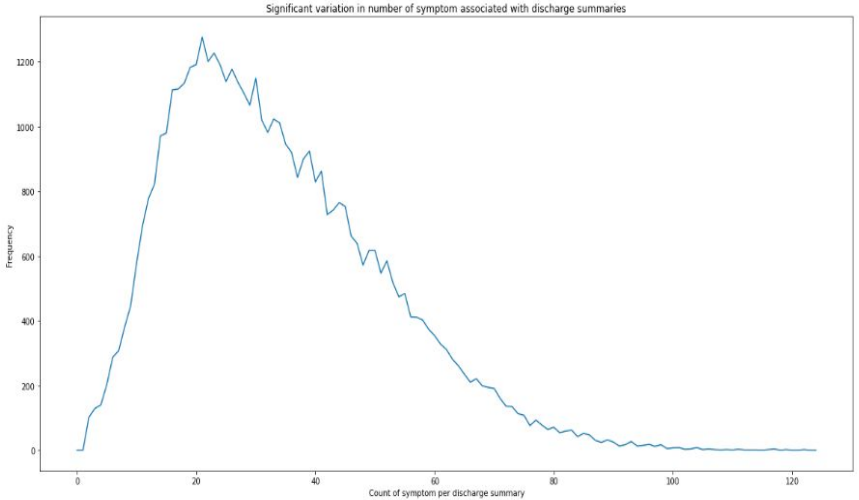
2. DIAGNOSIS_ICD table that contains the ICD codes of the diseases identified with the patient for each hospital admission.

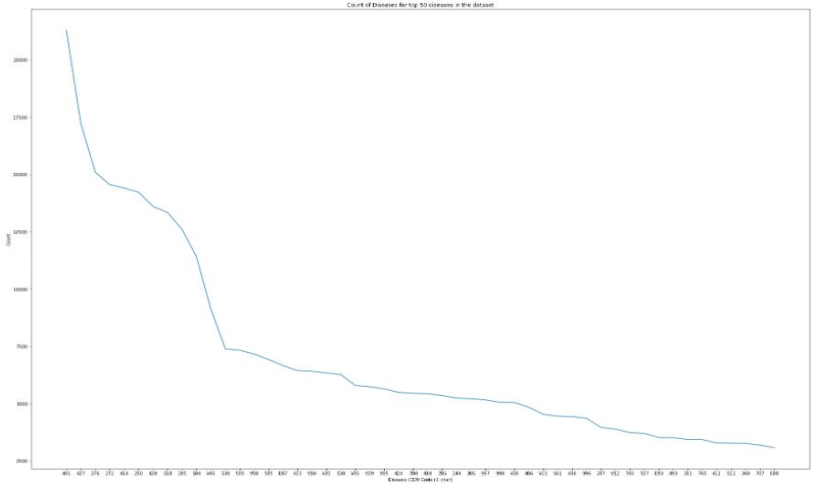
row_id	subject_id	hadm_id	seq_num	icd9_code

Dataset Details

1. In the NOTEVENTS table, the “text” column contains various types of notes. The discharge summaries text can be filtered out based on the value in “category” column.
2. The DIAGNOSIS_ICD table contains ICD9 code for each disease for a specific hospital admission.
3. The first three digits, i.e. the main disease is considered for the purpose of this paper. This results in total 931 disease codes.
4. Only the top 50 (3990 associated symptoms) and top 100 (4026 associated symptoms) diseases are considered for model training and evaluation.
5. Only notes with 2 or more symptoms are considered. Also only the first 50 symptoms are considered in a note with more than 50 symptoms.

Dataset Details Contd.





Extracting Symptoms Using MetaMap

1. MetaMap is a tool for identifying Unified Medical Language System (UMLS) constructs in medical literature.
2. MetaMap was installed and run on the Google cloud.
3. Input for MetaMap was raw discharge summaries text for each hospital admission.

Number of notes	Processing time	Min # of Symptoms	Max # of Symptoms
59651	~240 hours (8 CPU Google Cloud VM)	20	40

Symptom Embeddings Using TF-IDF

1. The symptom embeddings were created using the TF-IDF approach using the formula defined in earlier slide.
    a. $S_i = (W_{i,1}, W_{i,2}, ..., W_{i,d})$, where $W_{i,j}$ is defined below
    b. $W_{i,j} = TF_{i,j} . log (N / D_i)$, here $TF_{i,j}$ is the number of times symptom i is correlated with disease j in the discharge notes. N is total number of diseases considered, and D_i is the number of diseases associated with symptom i.
2. The total number of unique symptoms found were 3990 (for 50 diseases).
3. Top 50 and 100 diseases were considered for finding the correlation between a symptom and these diseases. This resulted in a symptom embedding vector of size 50 and 100.

Training Word2Vec Model

1. First we trained a Word2Vec model using the Gensim library.
2. The Word2Vec model was trained using raw text data from 59651 discharge summary notes. This served as the corpus for the Word2Vec model. Special characters and stop-words were filtered out from the notes.
3. Parameters for training this model were:

Window Size	Vector Size	Model Type	Negative Sampling	Down Sampling	Training Time
5	128	Skip gram	5	1e-3	~85 minutes

Symptom Embeddings Using Word2Vec

1. Once the Word2Vec model is trained, the symptoms extracted using MetaMap are used as input to this model for generating the symptom embeddings.
2. Since each symptom can have one or more words, embeddings for each word are added and then averaged.
3. Only alphanumeric characters are considered when creating symptom embeddings.
4. This process generates a symptom vector of size 128.

Training TF-IDF BiLSTM Model

1. Once we have the symptom embeddings using the TF-IDF technique, we train a BiLSTM model for disease prediction.
2. A training pair is $\{X_i, Y_i\}$, where X_i is a sequence of symptom embeddings $\{x_1, x_2, \ldots x_n\}$ in a single discharge summary note, and Y_i is a multi hot vector for diseases connected with that discharge summary note.
3. As we considered the top 50 or 100 diseases, this makes each input vector x_i of size 50 or 100.
4. Only the first fifty symptoms are considered from a single discharge summary note which makes the maximum sequence length as fifty.
5. Therefore a set of input sequence is a 2D vector of size 50x50 for the BiLSTM.

Training TF-IDF BiLSTM Model Contd.

6. BiLSTM has an input layer of size 50, two hidden layers of size 100 and an output layer of size 50.
7. We use batch size of 400, a dropout value of 0.8, learning rate of 0.001, Adam as optimizer and BCE as the loss function.

Training Word2Vec BiLSTM Model

1. After having the symptom embeddings using the Word2Vec technique, we train a BiLSTM model for disease prediction.
2. A training pair is $\{X_i, Y_i\}$, where X_i is a sequence of symptom embeddings $\{x_i, x_2, \ldots x_n\}$ in a single discharge summary note, and Y_i is a multi hot vector for diseases connected with that discharge summary note. x_is are generated using the trained Word2VecModel.
3. The size of input vector x_i is 128.
4. Only the first fifty symptoms are considered from a single discharge summary note which makes the maximum sequence length as fifty.
5. Therefore a set of input sequence is a 2D vector of size 128x50 for the BiLSTM.

Training Word2Vec BiLSTM Model Contd.

6. BiLSTM has an input layer of size 128, two hidden layers of size 256 and an output layer of size 50.
7. We use batch size of 400, a dropout value of 0.8, learning rate of 0.001, Adam as optimizer and BCE as the loss function.

TF-IDF and Word2Vec Combined Model

1. The combined model takes weighted average of the TF-IDF and Word2Vec outputs and gives the overall disease prediction. Each model's output is given weight in a range from 0 to 1.
2. When doing disease prediction using the combined output (Y), a threshold value of 0.2 is used to predict the diseases for that particular discharge summary note.

Model Hyper Parameters

1. Batch size: 400
2. Learning rate: 0.001
3. Number of epochs: 10
4. Dropout rate: 0.8
5. Optimizer: Adam
6. Loss: BCE
7. Number of hidden layers: 2
8. Hidden layer sizes: 100 (TF-IDF), 256 (Word2Vec)
9. Threshold: 0.2

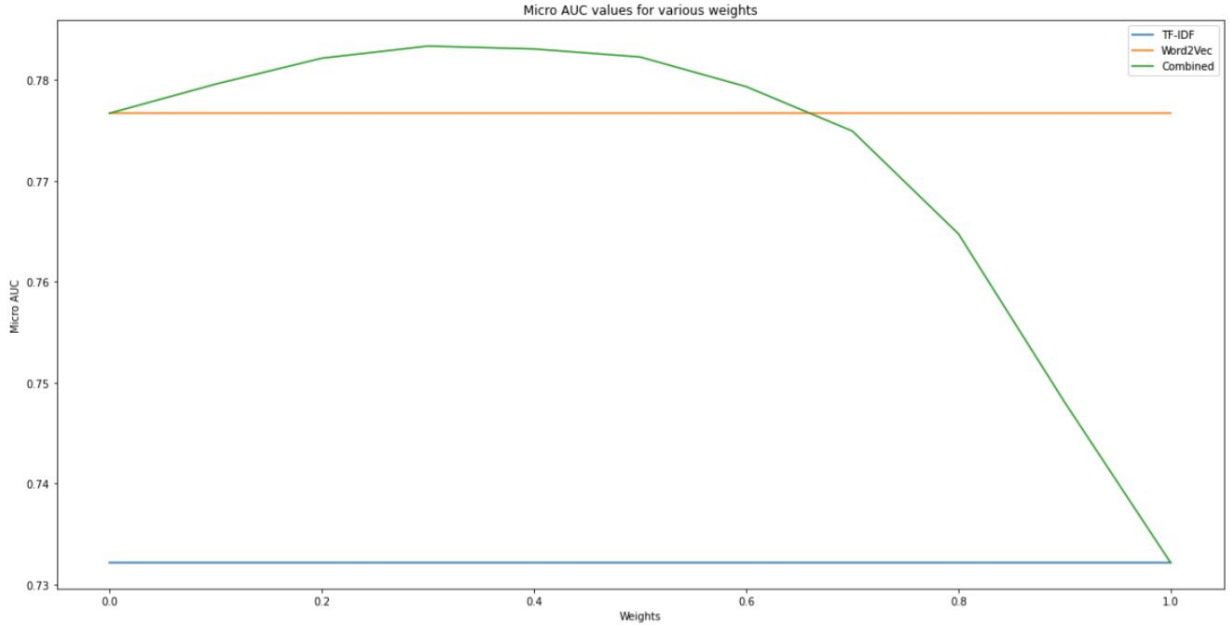
Results For Top 50 Diseases

Model	MiP	MiR	MiF1	Micro AUC
TF-IDF	0.391	0.602	0.474	0.732
Word2Vec	0.457	0.671	0.544	0.777
TF-IDF + Word2Vec	0.459	0.686	0.550	0.783

Results For Top 50 Diseases Contd.

Model	MaP	MaR	MaF1	Macro AUC
TF-IDF	0.353	0.481	0.390	0.664
Word2Vec	0.415	0.569	0.471	0.721
TF-IDF + Word2Vec	0.425	0.577	0.476	0.723

Micro AUC Score for 50 Diseases



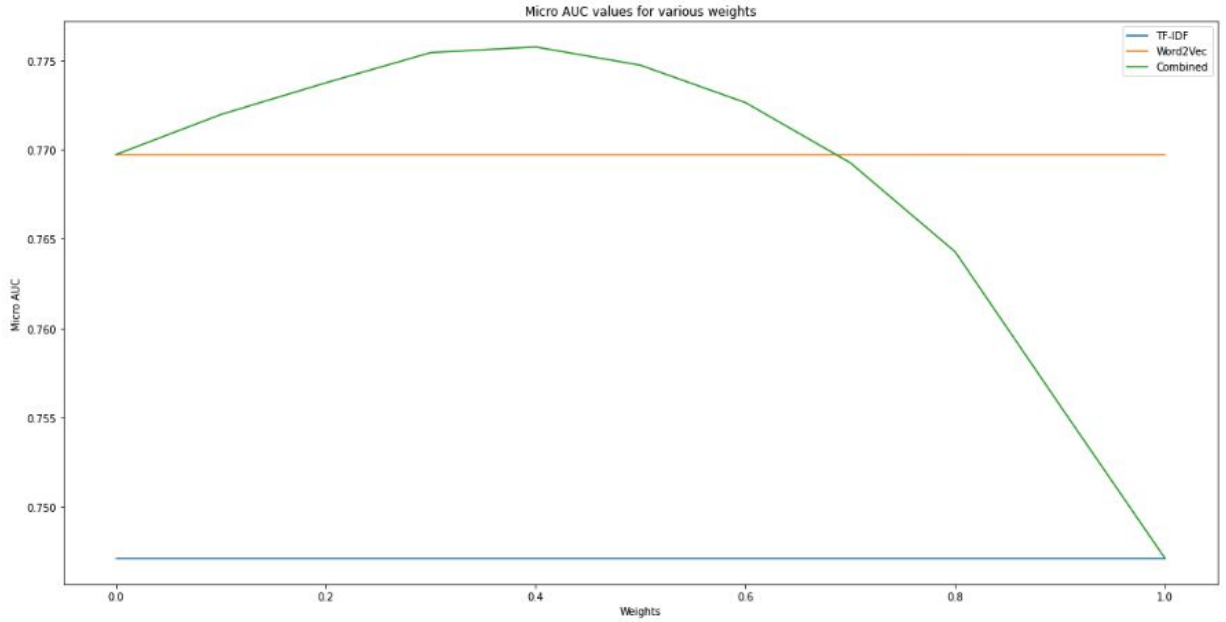
Results For Top 100 Diseases

Model	MiP	MiR	MiF1	Micro AUC
TF-IDF	0.410	0.570	0.477	0.749
Word2Vec	0.458	0.604	0.521	0.771
TF-IDF + Word2Vec	0.455	0.613	0.522	0.774

Results For Top 100 Diseases Contd.

Model	MaP	MaR	MaF1	Macro AUC
TF-IDF	0.370	0.439	0.367	0.679
Word2Vec	0.395	0.464	0.393	0.698
TF-IDF + Word2Vec	0.409	0.471	0.400	0.700

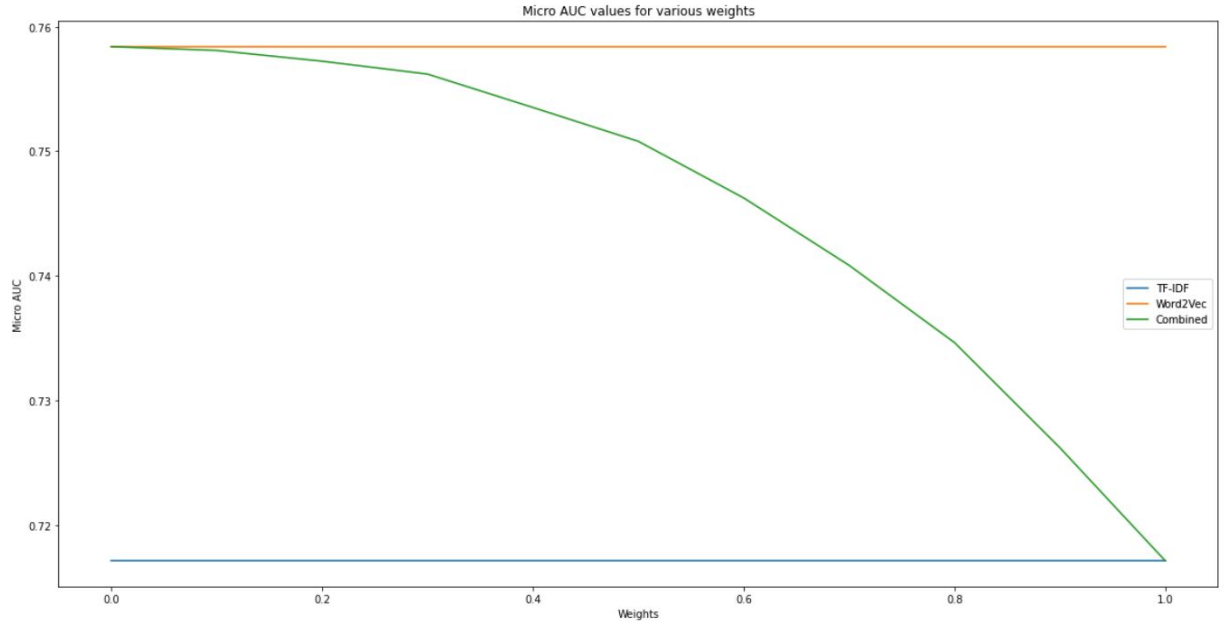
Micro AUC Score for 100 Diseases



Ablation: Using LSTM Model

Model	MiP	MiR	MiF1	Micro AUC
TF-IDF	0.40	0.50	0.44	0.72
Word2Vec	0.43	0.59	0.49	0.76
TF-IDF + Word2Vec	0.44	0.58	0.50	0.76

LSTM - Micro AUC Score for 100 Diseases



Training Loss