Predicting ICD-9 Codes from ICU Discharge Notes

Abhijith Asok
Chris Hilger
Liam Loscalzo
Katherine Wang

Overview and Motivation

Background:

- Medical coding is a multibillion dollar industry which is highly labor intensive and prone to error. This presents an opportunity for Natural Language Processing (NLP).
- ICD-9 codes were created by CMS (Centers for Medicare and Medicaid Services) to standardize the way in which patients health outcomes were categorized and tracked over time. They can be entered into a patient's electronic health record and can be used for diagnostic, billing and reporting purposes.

Motivation: Currently identifying an ICD-9 code is an manual process, which is slow, expensive, and error-prone. Creating a model that could automate the prediction of ICD-9 codes off of doctor's notes would be very beneficial.

Project Objective: Predict the 5 most common ICD-9 codes from doctor discharge notes collected from ICU stays between 2001 and 2012 at the Beth Israel Deaconess Medical Center

Related Work

- Automated ICD-9 codes assignment has been studied since 1990
- Focused on pattern matching, rule base systems, or supervised classification methods [1], [2], [3], [4]
- Shown good performance for specific sets, but do not generalize well
- Deep learning has potential to overcome the limitations by eliminating the task of describing explicit features or rules
- To date, deep learning models have achieved low performance (F1 score: 0.37) [5]

Data overview

- MIMIC-III (Medical Information Mart for Intensive Care III)
- Patients with stays in critical care units of Beth Israel Deaconess Medical Center, 2001-2012
- ~40,000 patients, ~60,000 ICU admissions
- Records include:
    - Demographics
    - Vital signs measurements
    - Lab results
    - Procedure and **diagnostic codes** (ICD-9)
    - Waveforms
    - Outcomes
    - **Patient reports and notes**

Input data pre-processing: filtering

NOTEEVENTS table
2,083,180 notes
58,361 admissions

	ROW_ID	SUBJECT_ID	HADM_ID	CHARTDATE	CHARTTIME	STORETIME	CATEGORY	DESCRIPTION	CGID	ISERROR	TEXT
0	174	22532	167853.0	2151-08-04	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2151-7-16**] Dischar...
1	175	13702	107527.0	2118-06-14	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2118-6-2**] Discharg...
2	176	13702	167118.0	2119-05-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2119-5-4**] D...
3	177	13702	196489.0	2124-08-18	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2124-7-21**] ...
4	178	26880	135453.0	2162-03-25	NaN	NaN	Discharge summary	Report	NaN	NaN	Admission Date: [**2162-3-3**] D...

Remove caregiver notes, nurse's notes, radiology reports, ECG reports, etc.

All discharge notes
59,652 notes
52,726 admissions

Remove all admissions with > 1 discharge note

Discharge notes input
47,006 notes/admissions

Input data pre-processing: text cleaning/tokenization

Replace PHI with [phi] tags

```
"Admission Date:   [**2112-12-8**]             Discharge Date:   [**2112-12-10**]\n\n\nS
ervice: MEDICINE\n\nAllergies:\nSulfonamides\n\nAttending:[**First Name3 (LF) 1850**]\nC
hief Complaint:\nHypoxia\n\nMajor Surgical or Invasive Procedure:\nnone\n\nHistory of Pr
esent Illness:\n82 yo F with CAD, CHF, HTN, recent PE ([**10-17**]), who presents from\n
rehab with hypoxia and SOB despite Abx treatment for PNA x 3\ndays. The patient was in r
ehab after being discharged from here\nfor PE. She was scheduled to be discharged on [**
12-6**]; on the day\nprior to discharge she deveoped fever, hypoxia, and SOB. CXR\nshowe
d b/t lower lobe infiltrates. She was started on levoflox\nand ceftriaxone on [**12-5*
*]. When she became hypoxic on NC they\nbrought her in to the ED.\n.\nIn the [**Hospital
1 18**] ED she was febrile to 102.7, P 109 BP 135/56 R 34\nO2 90% on 3L. She was started
on vanc and zosyn for broader\ncoverage, tylenol, and 2L NS.\n.\nThe patient reports hav
ing sweats and cough before admission.\nShe complains of SOB and some upper back pain. S
he denies chest\npain, URI sx, nausea/vomiting, diarrhea, or dysuria. Of note she\nhad h
```

Replace numbers with [num] tags

```
['admission',
 'date',
 '[phi]',
 'discharge',
 'date',
 '[phi]',
 'service',
 'medicine',
 'allergies',
 'sulfonamides',
 'attending[phi]',
 'chief',
 'complaint',
 'hypoxia',
 'major',
 'surgical',
 'or',
 'invasive',
 'procedure',
 'none',
 'history',
 'of',
 'present',
 'illness',
 '[num]',
 'yo',
 'f',
 'with',
 'cad',
 'chf',
```

Additionally:

- Remove extraneous punctuation, carriage returns, and whitespace

- All characters to lowercase

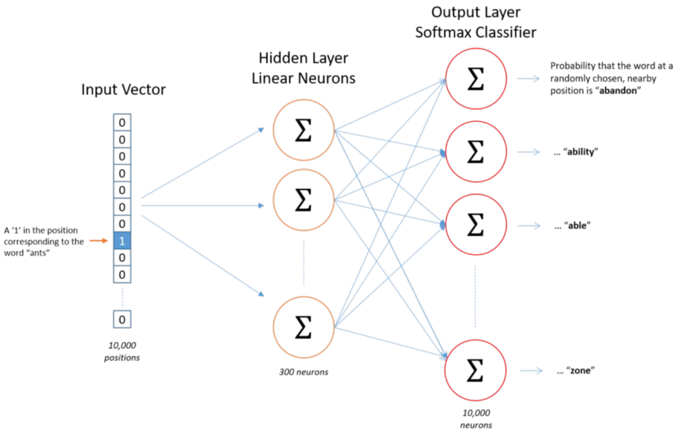
- Split string on whitespace

Input data pre-processing: word embeddings

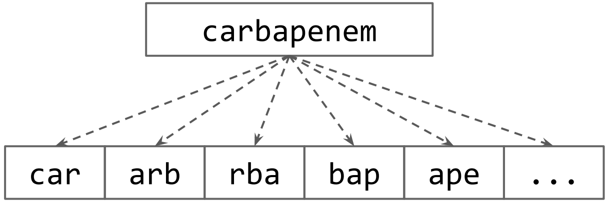
Two related approaches using `gensim` library

Word2Vec

FastText



https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78614e4d6e0b




```
In [35]: word_vectors.wv.most_similar('carbapenem')

Out[35]: [('carbapenems', 0.8965969681739807),
          ('carbapenemase', 0.8890013098716736),
          ('carbapenase', 0.831066370010376),
          ('carbapnem', 0.8257248401641846),
          ('carbepenem', 0.7826012372970581),
```

Label preparation

DIAGNOSES_ICD table
651,047 rows
6,984 ICD-9 codes

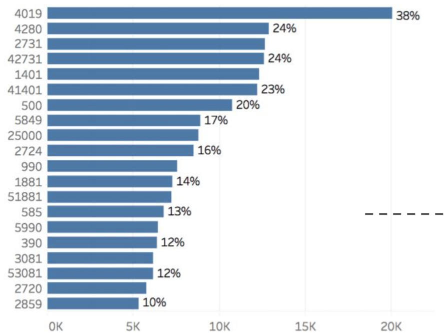
Remove codes with “V” prefix, sort remainder by frequency and take top five

Top 5 ICD-9 codes
58,976 rows

Match diagnoses to admission IDs from input data

Encoded output labels
47,006 x 5 matrix

	ROW_ID	SUBJECT_ID	HADM_ID	SEQ_NUM	ICD9_CODE
0	1297	109	172335	1.0	40301
1	1298	109	172335	2.0	486
2	1299	109	172335	3.0	58281
3	1300	109	172335	4.0	5855
4	1301	109	172335	5.0	4254



https://arxiv.org/pdf/1802.00382.pdf

4019: Hypertension
4280: Congestive heart failure
42731: Atrial fibrillation
41401: Coronary atherosclerosis of native coronary artery
5849: Acute kidney failure

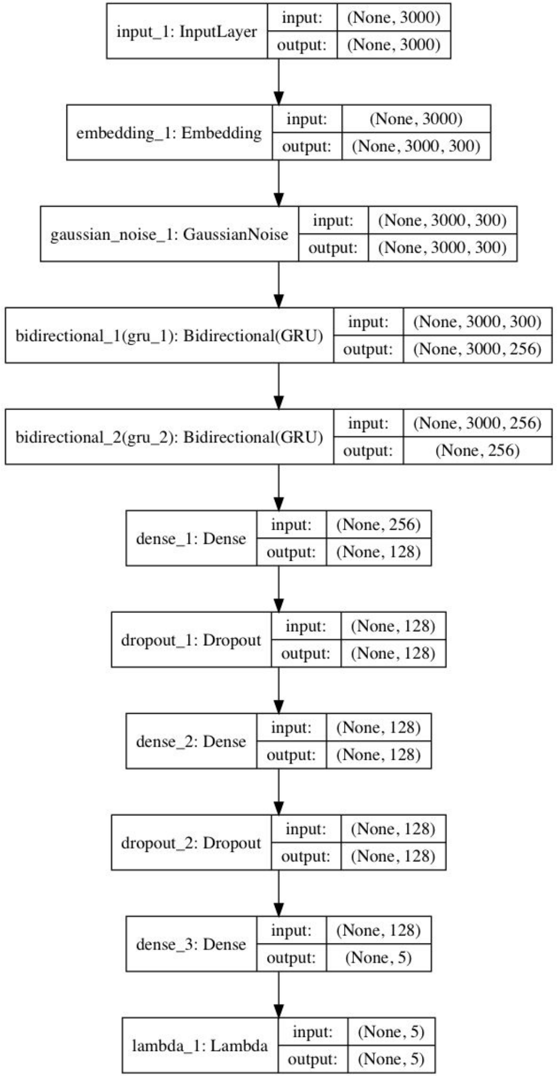
Gated Neural Network (GRU)

Why GRU?

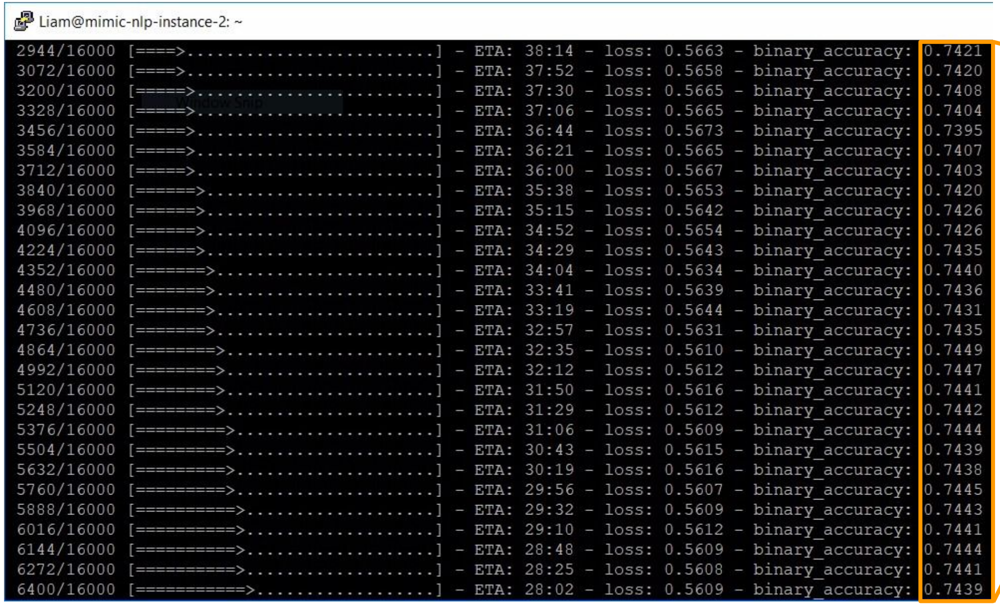
- GRUs are good for NLP because it solves the vanishing gradient problem that exists in standard RNNs
- They have similar performance to LSTM’s, but are faster to train

Model Architecture

- Embedding layer to leverage the word embeddings from the pre-processing stage
- Noise for regularization
- GRU layers for feature extraction
- Dense layers to reduce dimensionality
- Dropout to control overfitting
- Clipping layer: cut probabilities at 0.99 and 0.01



Model results



Accuracy = 0.74

But…

- Converges very quickly
- Does not improve with additional training
- Does not change with larger model/more data

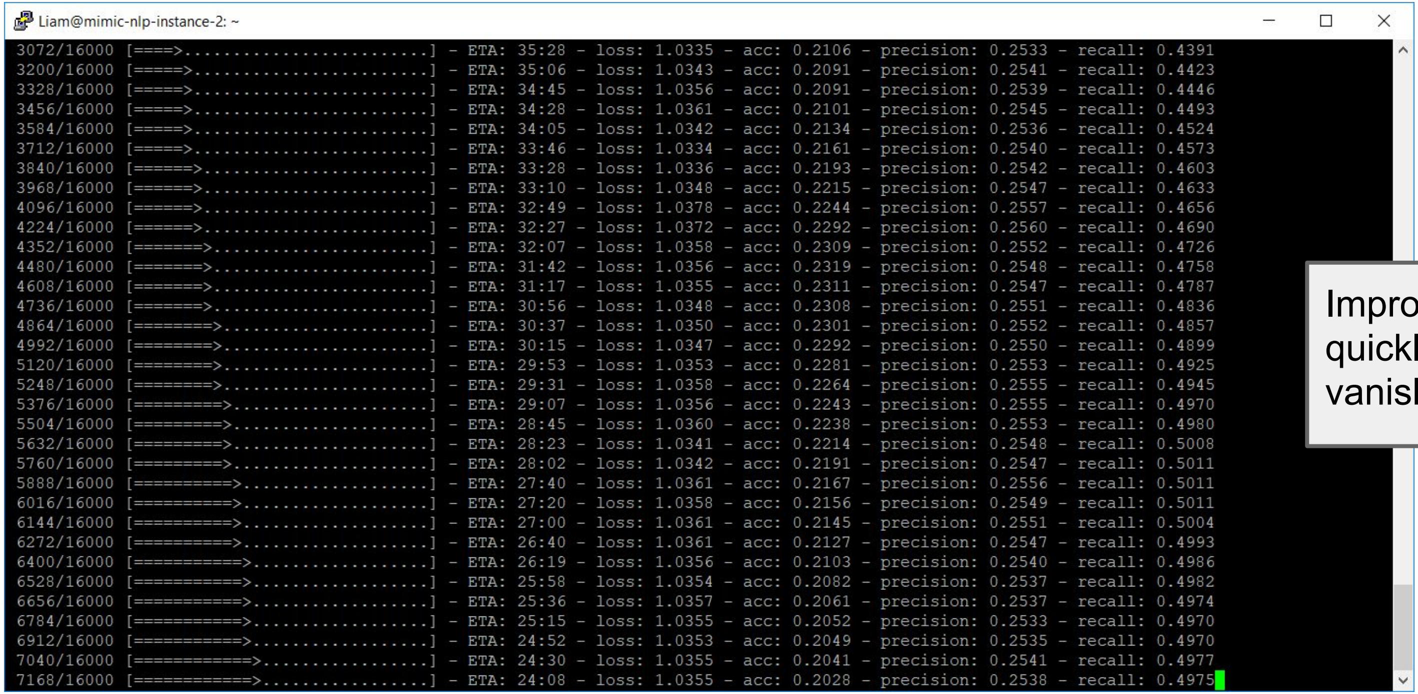
Model results

Liam@mimic-nlp-instance-2: ~

```
2944/16000 [====>..........................] - ETA: 38:14 - loss: 0.5663 - binary_accuracy: 0.7421 - precision: 0.4605 - recall: 0.0119
3072/16000 [====>..........................] - ETA: 37:52 - loss: 0.5658 - binary_accuracy: 0.7420 - precision: 0.4517 - recall: 0.0117
3200/16000 [=====>.........................] - ETA: 37:30 - loss: 0.5665 - binary_accuracy: 0.7408 - precision: 0.4508 - recall: 0.0119
3328/16000 [=====>.........................] - ETA: 37:06 - loss: 0.5665 - binary_accuracy: 0.7404 - precision: 0.4489 - recall: 0.0118
3456/16000 [=====>.........................] - ETA: 36:44 - loss: 0.5673 - binary_accuracy: 0.7395 - precision: 0.4446 - recall: 0.0118
3584/16000 [=====>.........................] - ETA: 36:21 - loss: 0.5665 - binary_accuracy: 0.7407 - precision: 0.4466 - recall: 0.0119
3712/16000 [=====>.........................] - ETA: 36:00 - loss: 0.5667 - binary_accuracy: 0.7403 - precision: 0.4410 - recall: 0.0119
3840/16000 [======>........................] - ETA: 35:38 - loss: 0.5653 - binary_accuracy: 0.7420 - precision: 0.4396 - recall: 0.0120
3968/16000 [======>........................] - ETA: 35:15 - loss: 0.5642 - binary_accuracy: 0.7426 - precision: 0.4416 - recall: 0.0124
4096/16000 [======>........................] - ETA: 34:52 - loss: 0.5654 - binary_accuracy: 0.7426 - precision: 0.4403 - recall: 0.0124
4224/16000 [======>........................] - ETA: 34:29 - loss: 0.5643 - binary_accuracy: 0.7435 - precision: 0.4330 - recall: 0.0123
4352/16000 [=======>.......................] - ETA: 34:04 - loss: 0.5634 - binary_accuracy: 0.7440 - precision: 0.4203 - recall: 0.0119
4480/16000 [=======>.......................] - ETA: 33:41 - loss: 0.5639 - binary_accuracy: 0.7436 - precision: 0.4226 - recall: 0.0117
4608/16000 [=======>.......................] - ETA: 33:19 - loss: 0.5644 - binary_accuracy: 0.7431 - precision: 0.4201 - recall: 0.0116
4736/16000 [=======>.......................] - ETA: 32:57 - loss: 0.5631 - binary_accuracy: 0.7435 - precision: 0.4357 - recall: 0.0114
4864/16000 [========>......................] - ETA: 32:35 - loss: 0.5610 - binary_accuracy: 0.7449 - precision: 0.4243 - recall: 0.0111
4992/16000 [========>......................] - ETA: 32:12 - loss: 0.5612 - binary_accuracy: 0.7447 - precision: 0.4262 - recall: 0.0110
5120/16000 [========>......................] - ETA: 31:50 - loss: 0.5616 - binary_accuracy: 0.7441 - precision: 0.4156 - recall: 0.0107
5248/16000 [========>......................] - ETA: 31:29 - loss: 0.5612 - binary_accuracy: 0.7442 - precision: 0.4054 - recall: 0.0105
5376/16000 [=========>.....................] - ETA: 31:06 - loss: 0.5609 - binary_accuracy: 0.7444 - precision: 0.3958 - recall: 0.0102
5504/16000 [=========>.....................] - ETA: 30:43 - loss: 0.5615 - binary_accuracy: 0.7439 - precision: 0.4098 - recall: 0.0101
5632/16000 [=========>.....................] - ETA: 30:19 - loss: 0.5616 - binary_accuracy: 0.7438 - precision: 0.4119 - recall: 0.0100
5760/16000 [=========>.....................] - ETA: 29:56 - loss: 0.5607 - binary_accuracy: 0.7445 - precision: 0.4027 - recall: 0.0098
5888/16000 [==========>....................] - ETA: 29:32 - loss: 0.5609 - binary_accuracy: 0.7443 - precision: 0.4157 - recall: 0.0098
6016/16000 [==========>....................] - ETA: 29:10 - loss: 0.5612 - binary_accuracy: 0.7441 - precision: 0.4281 - recall: 0.0099
6144/16000 [==========>....................] - ETA: 28:48 - loss: 0.5609 - binary_accuracy: 0.7444 - precision: 0.4192 - recall: 0.0097
6272/16000 [==========>....................] - ETA: 28:25 - loss: 0.5608 - binary_accuracy: 0.7441 - precision: 0.4107 - recall: 0.0095
6400/16000 [===========>...................] - ETA: 28:02 - loss: 0.5609 - binary_accuracy: 0.7439 - precision: 0.4225 - recall: 0.0094
```

Recall metric < 0.01 indicates that our model is simply predicting “0” for each diagnosis for nearly every patient

Attempted fix: weighted loss function



Liam@mimic-nlp-instance-2: ~

```
3072/16000 [====>.........................] - ETA: 35:28 - loss: 1.0335 - acc: 0.2106 - precision: 0.2533 - recall: 0.4391
3200/16000 [=====>........................] - ETA: 35:06 - loss: 1.0343 - acc: 0.2091 - precision: 0.2541 - recall: 0.4423
3328/16000 [=====>........................] - ETA: 34:45 - loss: 1.0356 - acc: 0.2091 - precision: 0.2539 - recall: 0.4446
3456/16000 [=====>........................] - ETA: 34:28 - loss: 1.0361 - acc: 0.2101 - precision: 0.2545 - recall: 0.4493
3584/16000 [=====>........................] - ETA: 34:05 - loss: 1.0342 - acc: 0.2134 - precision: 0.2536 - recall: 0.4524
3712/16000 [=====>........................] - ETA: 33:46 - loss: 1.0334 - acc: 0.2161 - precision: 0.2540 - recall: 0.4573
3840/16000 [======>.......................] - ETA: 33:28 - loss: 1.0336 - acc: 0.2193 - precision: 0.2542 - recall: 0.4603
3968/16000 [======>.......................] - ETA: 33:10 - loss: 1.0348 - acc: 0.2215 - precision: 0.2547 - recall: 0.4633
4096/16000 [======>.......................] - ETA: 32:49 - loss: 1.0378 - acc: 0.2244 - precision: 0.2557 - recall: 0.4656
4224/16000 [======>.......................] - ETA: 32:27 - loss: 1.0372 - acc: 0.2292 - precision: 0.2560 - recall: 0.4690
4352/16000 [=======>......................] - ETA: 32:07 - loss: 1.0358 - acc: 0.2309 - precision: 0.2552 - recall: 0.4726
4480/16000 [=======>......................] - ETA: 31:42 - loss: 1.0356 - acc: 0.2319 - precision: 0.2548 - recall: 0.4758
4608/16000 [=======>......................] - ETA: 31:17 - loss: 1.0355 - acc: 0.2311 - precision: 0.2547 - recall: 0.4787
4736/16000 [=======>......................] - ETA: 30:56 - loss: 1.0348 - acc: 0.2308 - precision: 0.2551 - recall: 0.4836
4864/16000 [========>.....................] - ETA: 30:37 - loss: 1.0350 - acc: 0.2301 - precision: 0.2552 - recall: 0.4857
4992/16000 [========>.....................] - ETA: 30:15 - loss: 1.0347 - acc: 0.2292 - precision: 0.2550 - recall: 0.4899
5120/16000 [========>.....................] - ETA: 29:53 - loss: 1.0353 - acc: 0.2281 - precision: 0.2553 - recall: 0.4925
5248/16000 [========>.....................] - ETA: 29:31 - loss: 1.0358 - acc: 0.2264 - precision: 0.2555 - recall: 0.4945
5376/16000 [=========>....................] - ETA: 29:07 - loss: 1.0356 - acc: 0.2243 - precision: 0.2555 - recall: 0.4970
5504/16000 [=========>....................] - ETA: 28:45 - loss: 1.0360 - acc: 0.2238 - precision: 0.2553 - recall: 0.4980
5632/16000 [=========>....................] - ETA: 28:23 - loss: 1.0341 - acc: 0.2214 - precision: 0.2548 - recall: 0.5008
5760/16000 [=========>....................] - ETA: 28:02 - loss: 1.0342 - acc: 0.2191 - precision: 0.2547 - recall: 0.5011
5888/16000 [==========>...................] - ETA: 27:40 - loss: 1.0361 - acc: 0.2167 - precision: 0.2556 - recall: 0.5011
6016/16000 [==========>...................] - ETA: 27:20 - loss: 1.0358 - acc: 0.2156 - precision: 0.2549 - recall: 0.5011
6144/16000 [==========>...................] - ETA: 27:00 - loss: 1.0361 - acc: 0.2145 - precision: 0.2551 - recall: 0.5004
6272/16000 [==========>...................] - ETA: 26:40 - loss: 1.0361 - acc: 0.2127 - precision: 0.2547 - recall: 0.4993
6400/16000 [===========>..................] - ETA: 26:19 - loss: 1.0356 - acc: 0.2103 - precision: 0.2540 - recall: 0.4986
6528/16000 [===========>..................] - ETA: 25:58 - loss: 1.0354 - acc: 0.2082 - precision: 0.2537 - recall: 0.4982
6656/16000 [===========>..................] - ETA: 25:36 - loss: 1.0357 - acc: 0.2061 - precision: 0.2537 - recall: 0.4974
6784/16000 [===========>..................] - ETA: 25:15 - loss: 1.0355 - acc: 0.2052 - precision: 0.2533 - recall: 0.4970
6912/16000 [===========>..................] - ETA: 24:52 - loss: 1.0353 - acc: 0.2049 - precision: 0.2535 - recall: 0.4970
7040/16000 [============>.................] - ETA: 24:30 - loss: 1.0355 - acc: 0.2041 - precision: 0.2541 - recall: 0.4977
7168/16000 [============>.................] - ETA: 24:08 - loss: 1.0355 - acc: 0.2028 - precision: 0.2538 - recall: 0.4975
```

Improved recall, but loss quickly stalls at ~1.03 → vanishing gradient?

Final Analysis

What did you learn about your model? Be sure to compare performance of the training, validation and test sets. Also discuss any limitations or future work for this project.

Further exploration : 1D CNN - Alternate approach compared to GRUs. Might be able to extract more out of keywords. Focus more on learning patterns across space than time/ordering. A CNN model was attempted and can be explored further with higher computational resources and time.

```
model = Sequential()
input = Input(shape=(seq_len,))
x = Embedding(input_dim=vocab_size, output_dim=embedding_dim, weights=[embedding_matrix], trainable=False)(input)
x = GaussianNoise(0.75)(x)
x = Conv1D(32,7, activation='relu')(x)
x = MaxPooling1D(5)(x)
x = Conv1D(32,7, activation='relu')(x)
x = MaxPooling1D(5)(x)
x = Dense(32, activation='relu')(x)
x = Dropout(0.5)(x)
x = Dense(32, activation='relu')(x)
x = Dropout(0.5)(x)
x = Dense(num_classes, activation='sigmoid')(x)
x = ClipLayer(x)

model = Model(input, x)
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

model.fit(X, y_train, epochs=10, batch_size=64)
```

- Had to limit the vector length due to computational resource and time limitations. A less restrictive trimming of word vectors for each discharge note could have possibly improved the results.
- Choice of more ICD-9 codes as potential prediction classes to tend towards the original data distribution

References

[1] Koby Crammer, Mark Dredze and Kuzman Ganchev and Partha Pratim Talukdar Automatic Code Assignment to Medical Text

[2] Ira Goldstein, M.B.A., Anna Arzumtsyan, M.L.S., and ozlem Uzuner, Ph.D Three Approaches to Automatic Assignment of ICD-9-CM Codes to Radiology Reports. AMIA 2007

[3] Alan R. Aronson1, Olivier Bodenreider1, Dina Demner-Fushman1, Kin Wah Fung1, Vivian K. Lee1,2, James G. Mork1, Aurelie Neveol1, Lee Peters1, Willie J. Rogers From Indexing the Biomedical Literature to Coding Clinical Text: Experience with MTI and Machine Learning Approaches. BioNLP 2007: Biological, translational, and clinical language processing, pages 105–112

[4] Perotte, Adler, Rimma Pivovarov, Karthik Natarajan, Nicole Weiskopf, Frank Wood, and Noaomie Elhadad. "Diagnosis Code Assignment: Models and Evaluation Metrics." Journal of the American Medical Informatics Association 21.2 (2014): 231-37. Web.

[5] Priyanka Nigam Applying Deep Learning to ICD-9 Multi-label Classification from Medical Records cs224d Class paper presentation. 2015