CS230

Duolingo Shared Task on Second Language Acquisition Modeling (SLAM)

Nathaniel Goenawan
Department of Computer Science
Stanford University
nathgoh@stanford.edu

Christopher Wong
Department of Computer Science
Stanford University
cwong7@stanford.edu

Abstract

As more student data becomes more available, deep learning techniques can provide analysis of task like language acquisitions that were previously difficult and computationally time-consuming to perform. Duolingo's Second Language Acquisition Modeling (SLAM) task provides a large corpus of student data to determine how students learn a new language through three different exercises on three different langauge tracks. We propose to apply a recurrent architecture namely LSTM to SLAM at the user-level.

Our baseline is a logistic regression model our main model is a user-level LSTM model. All features that we extracted were encoded as embedding matrices. We found that performing LSTM at the user-level outperformed the baseline in accuracy and F1-score. However, our LSTM resulting AUC scores were worse than the baseline.

1 Introduction

1.1 Problem Description

Deep learning has many exciting applications in the education space. One such example is Duolingo's Second Language Acquisition Modeling (SLAM) task. In this task, we are provided a large data set of beginner-level student data to trace how students learn a new language through three different exercises. In this project, we will build deep learning models to predict students' learning performance based on their past learning data and compare our model's performance to the logistic regression baseline model's performance.

1.2 What is SLAM

In a more modernized and well-connected world, individuals are able to easily communicate with one another, so it is imperative that we improve our instructions on Second Language Acquisition (SLA). Online learning language platforms like Duolingo allow individuals from around the world to learn a second language (L2) from their native language (L1).

Duolingo's Shared Task on Second Language Acquisition Modeling (SLAM)[1] provides us with publicly available SLA data, which was released as part of a timed challenge by Duolingo. The paper referenced prior provides details on the dataset and the challenge description in detail.

We used this paper because Duolingo's SLAM dataset is one of the largest publicly available language acquisition datasets. Working with the SLAM datasets provides us with new avenues of applying

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

natural language processing (NLP) models to educational tasks that can have worldwide impact. Furthermore, having a large dataset at hand allows us to further apply deep learning techniques to our NLP modelings.

2 Related Works

Duolingo released SLAM as a challenge with a logistic regression baseline, most submission to the challenge improved on the baseline. A total of fourteen teams submitted results for all the language tracks. Most of the best performing teams employed some form of Recurrent Neural Network (RNN). The best team, SanaLabs (Nilsson et al.[2018]) who scored first place across all language tracks employed a RNN in combination with a Gradient Boosted Decision Tree (GBDT) ensembles[3].

One interesting approach done by Xu et al. [2018] was using four separate RNN encoders relative success. Their resulting model score of area under the ROC curve (AUC) of 0.861 tied first with SanaLabs in the English track[2]. Despite using relatively little feature engineering, they were able to be at the top of the leaderboard (2nd place for Spanish and French Track)[2]. Their encoders stored information on token context, linguistic information, user data, and exercise format. They found that the context encoder which represented the tokens contributed the most to the model's performance, while the linguistic encoder which represented grammatical information contributed the least[1].

The models that scored at the top of leaderboard didn't use feature engineering indicating that automated feature engineering of the models are adequate enough for the SLAM task. This analysis is noted in the Duolingo summary paper, feature engineering has a smaller impact on the system performance than the choice of the learnging algorithm[1]. Because previous works have shown that custom features won't significantly increase a model's performance, we will only used the features provided by the SLAM task rather than attempting to create new features.

3 Dataset

We will be working on the Duolingo's Second Language Acquisition Modeling (SLAM) dataset and prediction task[1]. The dataset is comprised of exercise data from more than 6,000 Duolingo students over the course of their first 30 days.

Exercises are in one of three formats: (1) translate a prompt written in L1 to L2 , (2) translate from L1 language to L2 using a provided bank of words and distractors, and (3) transcribe an utterance in L2. Each exercise contains a list of L2 words (tokens) and a binary label for whether the student answered the exercise correctly or not for the given L2 word. The dataset contains three language tracks: English, Spanish, and French. More specifically, they are organized as English students (who already speak Spanish), Spanish students (who already speak English), and French students (who already speak English).

```
# user:D2inSf5+  countries:MX  days:1.793  client:web  session:lesson  format:reverse_translate  time:16
8rgJEAPw1001  She          PRON    Case=Nom|Gender=Fem|Number=Sing|Person=3|PronType=Prs|fPOS=PRON++PRP    nsubj       4  0
8rgJEAPw1002  is           VERB    Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|fPOS=VERB++VBZ    cop         4  0
8rgJEAPw1003  my           PRON    Number=Sing|Person=1|Poss=Yes|PronType=Prs|fPOS=PRON++PRP$              nmod:poss   4  1
8rgJEAPw1004  mother       NOUN    Degree=Pos|fPOS=ADJ++JJ                                                ROOT        0  1
8rgJEAPw1005  and          CONJ    fPOS=CONJ++CC                                                          cc          4  0
8rgJEAPw1006  he           PRON    Case=Nom|Gender=Masc|Number=Sing|Person=3|PronType=Prs|fPOS=PRON++PRP   nsubj       9  0
8rgJEAPw1007  is           VERB    Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|fPOS=VERB++VBZ    cop         9  0
8rgJEAPw1008  my           PRON    Number=Sing|Person=1|Poss=Yes|PronType=Prs|fPOS=PRON++PRP$              nmod:poss   9  1
8rgJEAPw1009  father       NOUN    Number=Sing|fPOS=NOUN++NN                                              conj        4  1

# user:D2inSf5+  countries:MX  days:2.689  client:web  session:practice  format:reverse_translate  time:6
oMGsnnH/0101  When         ADV     PronType=Int|fPOS=ADV++WRB                                             advmod      4  1
oMGsnnH/0102  can          AUX     VerbForm=Fin|fPOS=AUX++MD                                              aux         4  0
oMGsnnH/0103  I            PRON    Case=Nom|Number=Sing|Person=1|PronType=Prs|fPOS=PRON++PRP              nsubj       4  1
oMGsnnH/0104  help         VERB    VerbForm=Inf|fPOS=VERB++VB                                             ROOT        0  0
```

Figure 1: Data format group into exercise sessions with one word on each line along with the features for each word.[1]

The dataset is formatted where each student is represented by a group of lines separated by a blank line: one token per line prepended with exercise-level metadata. The metadata contains information about the student such as how days of experience they have with L2, the client (iOS, Android, client) they are using, and what type of exercise they are doing. In addition, each exercise group also contains a list of instances corresponding to a L2 word (token) and its linguistic information. Alongside that is the binary label as to whether as the L2 word correctly or not for the particular exercise. Part of the dataset is shown in Figure 1, showcasing the formatting of the dataset.

4 Task & Approach

4.1 Task Description

For each instance or word token, we are going to predict whether the student got the token right or wrong. This is a binary classification task where tokens answered correctly, it is labeled '0' whereas tokens answered incorrectly are labeled as '1'. We will pass in as inputs at the user-level into the model, meaning that all exercises and their associated features for each user is passed into the model.

4.2 Data Formatting & Features

Duolingo provided us with a basic code for reading in the files and necessary features at an instance level. In addition to the provided code, we added additional functions to create a feature extraction system that will allows us to map each distinct feature to a numerical feature that be used in the model. For features, we encoded them as embedding matrices and used exercise-level and word-level features namely:

- User: a base-64 unique user id of length 8
- Country: 2-character country codes from which this user has done exercises
- Session: 3 possible sessions (lesson, practice, test)
- Format: 3 possible formats (reverse_translate, reverse_tap, listen)
- Part of Speech: 17 different parts of speech
- Token: word in sentence
- Dependency Label: Indication of what other words it depends on in the sentence

4.3 Code

`https://github.com/nathgoh/CS230-Duolingo`

4.4 Baseline

SLAM_baseline, a simple logistic regression using data set features, trained separately for each track using stochastic gradient descent (SGD) on the training set only[1]. In particular, the model is L2-regularized logistic regression trained with SGD weighted by frequency. The baseline takes in exercise-level features and will output binary result of whether or not the student answered the correct word or not. It's a 0 layer neural network that has a sigmoid activation function.

4.5 LSTM

LSTM, or long short-term memory, is the approach that we will be going with for this project.

Normal recurrent neural networks were originally considered; this is because as networks with loops in them, information is allowed to persist, which allows us to use reasoning about a previous event to infer or inform later ones. For this reason, RNNs have been particularly successful for problems pertaining to speech recognition, language modeling, and translation. In the event that we only need to look at recent information to perform a task, such RNNs would excel[5].

Consider a language model attempting to predict the next word based on the ones it previously received. If we try to predict the blank space in "The seagull flew over the __", we don't need more context to infer that this blank is likely *sea*.

However, consider that same language model attempting to predict the last word in the sentence "My favorite sport is __" when given a previous sentence such as "I've played basketball every week since I was 7". We know from the context of the rest of the original sentence that we're likely looking for the name of the sport, but we would have no idea what sport this is without further context. Our previous sentence would give us a likely guess, but we don't necessarily know how large the gap is between these two sentences (say in the case that these are sentences in a book), and the gap between these two sentences may in fact be quite large. The existence of large gaps such as these make standard RNNs less of a viable option, as they're unable to bridge or connect that information in the presence of large gaps[5]. This is especially problematic for us, as this is the project we are working on. For this reason, LSTMs are a special type of RNN that will work much better for the task that we have.

LSTMs are designed specifically to be able to properly deal with the large gaps that the aforementioned standard RNNs struggle with, and for this reason, we're approaching the project with LSTMs[5].

4.6 Learning Model

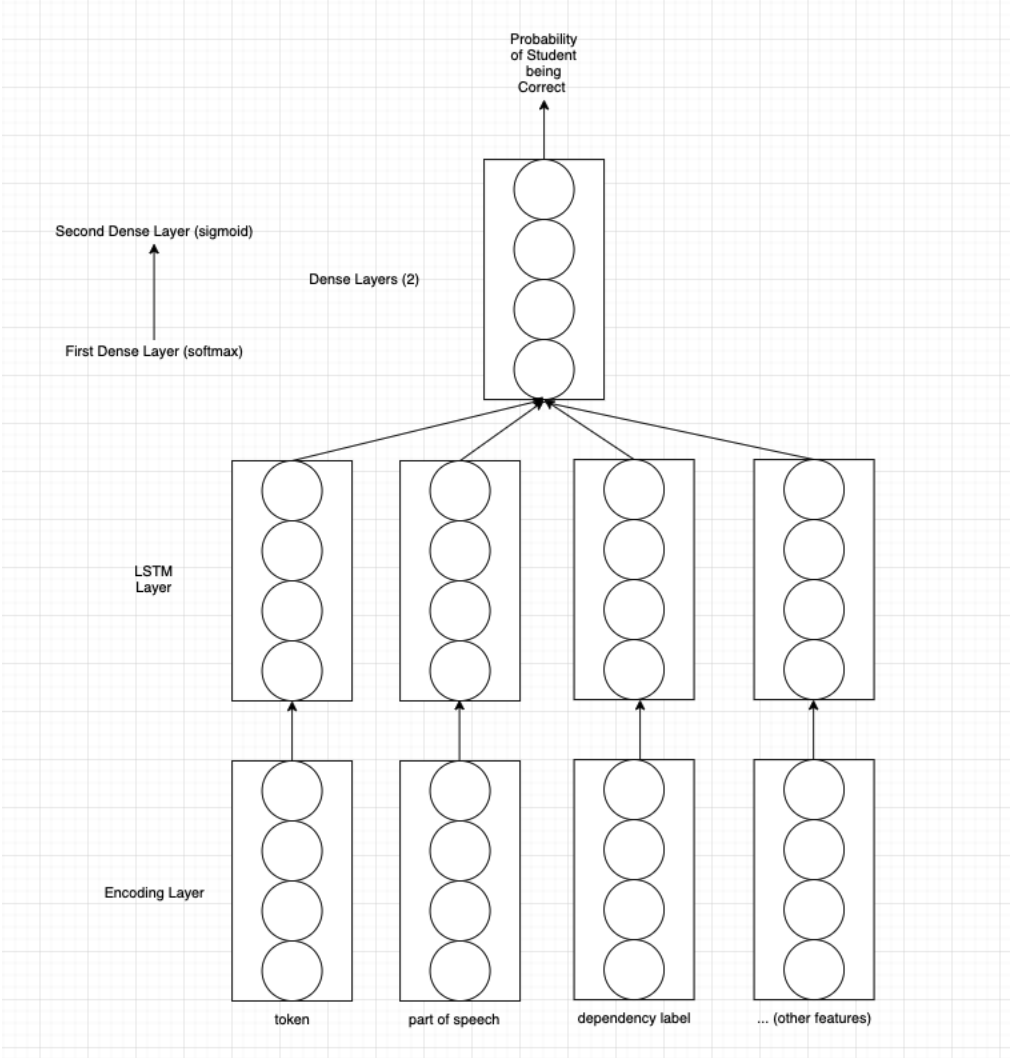

Figure 2:Architecture for the LSTM

5 Results

5.1 Evaluation

Primarily, the evaluation will be done using the the following metrics: area under the ROC curve (AUC), F1 score (with a threshold of 0.5), and accuracy. In addition to the basic metric score evaluations, we will also evaluate our model against the logistic regression baseline model.

5.2 Baseline

	Accuracy	AUC	F1
English	85.6%	0.774	0.194
Spanish	84.4%	0.746	0.179
French	83.4%	0.771	0.281

Table 1: Evaluation results of the logistic regression baseline.

5.3 LSTM

	Accuracy	AUC	F1
English	86.96%	0.501	0.930
Spanish	85.7%	0.500	0.923
French	83.77%	0.501	0.912

Table 2: Evaluation results of the LSTM model.

5.4 Analysis

The LSTM model outperforms the logistic regression baseline model in terms of accuracy in the English, Spanish, and French languages, although marginally so. However, AUC scores are effectively at 0.5, which is significantly worse than that of the logistic regression baseline model. The 0.5 scores received with our LSTM model is effectively akin to a predictor simply making random guesses. However, F1 scores are significantly higher in the LSTM model than in the logistic regression baseline model. This is indicative of both higher recall and precision levels in our model compared to that of the baseline model.

6 Discussion

Originally, the model had an Adam optimizer with a learning rate of 0.0001. However, the model's performance during training quickly plateaued with minimal signs of improvement. Thus, we decided to a faster learning rate of 0.001, which showed substantial improvements to the model's training. We decided on a binary cross-entropy loss function and a sigmoid activation function for our final Dense layer. These respective functions were chosen because the goal of the project results in binary labels - that is, whether we can predict if a student got a token right (0) or wrong (1).

Data being passed into the model is of a long sequence length (over 7500 in sequence length) due to the large amount of exercise data from many different users available for each track. Due to the long sequence length and performance constraints, we needed to cap the training sequences to a substantially shorter length of 2048. Yet, even at this length, it is known that LSTMs do not perform favorably to long sequence lengths. This could be a contributing factor to the poor classification performance measure results on the test set despite favorable training results. Future work can be done to input data at the exercise level rather than our current approach of inputting data at the user level. In the scenario where data is input at the exercise level, we would have that each user gets their exercises with their associated features passed into the model. Because exercises have a shorter sequence length (length being determined by the number of tokens) the LSTM model should perform better.

7 Contribution

Nathaniel Goenawan built the additional data formatting code to the one provided by Duolingo SLAM and debug the LSTM model. In addition, he helped write the paper and presentation for the video submission.

Christopher Wong helped lay out the LSTM model structure and feature extraction code. In addition, he helped write the paper and present the video presentation.

References

[1] B. Settles, C. Brust, E. Gustafson, M. Hagiwara, and N. Madnani. Second language acquisition modeling. In *Proceedings of the NAACL-HLT Workshop on Innovative Use of NLP for Building Educational Applications (BEA)*. ACL, 2018.

[2] Shuyao Xu, Jin Chen, and Long Qin. Cluf: a neural model for second language acquisition modeling. In Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 374–380, 2018.

[3] Susanna Nilsson, Anton Osika, Andrii Sydorchuk, Faruk Sahin, and Anders Huss. Second language acquisition modeling: An ensemble approach. CoRR, abs/1806.04525, 2018. URL http:// arxiv.org/abs/1806.04525. section*Abstract* [4] N.V. Nayak and A.R. Rao. 2018. Context based approach for second language acquisition. In Proceedings of the NAACL-HLT Workshop on Innovative Use of NLP for Building Educational Applications (BEA). ACL.

[5] Olah, Christopher. "Understanding LSTM Networks." Understanding LSTM Networks – Colah's Blog, 25 Aug. 2015, colah.github.io/posts/2015-08-Understanding-LSTMs/.