Root Cause Analysis and Streaming Prediction of Call KPIs in Call Center Based on Transcription and Voice Features

Students: Titus Berndt, Kevin Hsu, Ricky Zhang
Sponsor: Spoken Communications
Sponsor Mentors: Yishay Carmiel, Matthew Peters *Faculty Adviser:* Eli Shlizerman

ELECTRICAL ENGINEERING
UNIVERSITY *of* WASHINGTON

Introduction

Customer service is a critical part of business operations. Good customer service can incentivize a consumer to continue purchasing from a company. Bad customer service can destroy the customer's relationship with a company permanently.

Call-centers provide a company with a human ambassador between them and their customers. This position is critical to providing the quality customer service which retains customers. However, understanding and teaching the qualities of good customer service is a challenge.

To address this, we researched some of the root causes of call performance. Using timestamped transcripts, we analyzed and predict a key performance index (KPI), **call duration** (*most callers want to be serviced as soon as possible*).

Observations and Classification

Classification
To classify our data, we profiled *total call duration*. Four minute intervals gave us the most useful definition for simple classification *(Figure 1).*

- 0-4 minutes, 4-8 minutes, 8-12 minutes, 12+ minutes

Observations
We took observations over only the *first minute* of every conversation's transcript.
Using each word's timestamp, we analyzed the *speaking characteristics* of call participants

- **Call Metrics**.

Using the actual words, we analyzed the effect of certain *key words or phrases*

- **NLP Encoding**

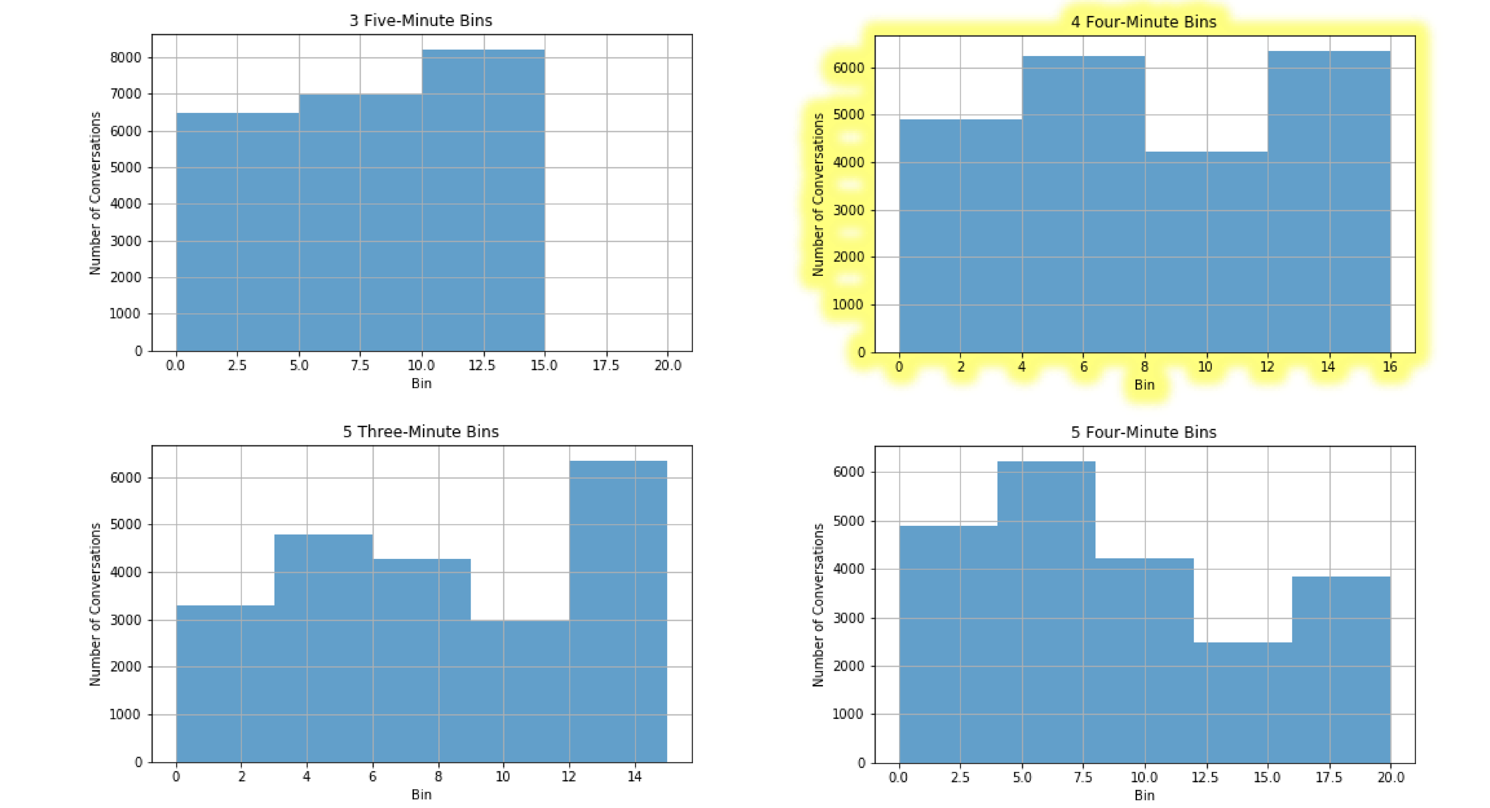

Figure 1: Histograms of conversation length in minutes

Call Metrics

To analyze speaking characteristics, we used five metrics:

- Speaking time *(time when at least one channel spoke)*
- Silence *(time when no one spoke)*
- Crosstalk *(time when both channels spoke)*
- Speaker ratio *(agent vs customer)*
- Speaker pace *(words per second)*

Each metric was analyzed as an average between 5-second intervals.

- 0-5, 5-10, …, 55-60
- 0-10, 0-15, …, 0-60
- 5-15, …, 5-60, etc.

Metric Analysis Models

We created several models to analyze the speaking characteristics. These included:

- Linearization with backwards elimination
- Random Forest
- Gradient Boosting
- K-Nearest Neighbors

However, no combination of these models and the call metrics produced meaningful predictions. The highest predictive accuracy reached was 40%. A random guess would produce at least 25% accuracy for a balanced 4-way classifier. *These metrics did not predict call duration.*

NLP Encoding

We converted the conversation text into a numerical representation to be analyzed by our machine learning models. Two different types of word embeddings were explored: **frequency-based embedding** and **prediction-based embedding**. These embeddings were analyzed using a neural network.

Prediction-based embedding
The prediction-based method provides probabilities for the words. It is optimal for tasks such as word analogies and word similarities. Google's pre-trained *word2vec* model was loaded with *gensim* and tested with our models. However, while this method is good for analyzing a small number of words, we needed to analyze hundreds of words over thousands of conversations. The size of the resultant word embedding vector was beyond our computational capability.

Frequency-based embedding
Term Frequency-Inverse Document Frequency (TF-IDF)

- Built on top of a base bag-of-words model, where each element in the Document Vector is the frequency of each word in a given conversation *(Figure 2)*
- Penalizes the most common but meaningless words by dividing the term frequency by the total document frequency
- Efficient while preserving valuable information
- Implemented with scikit-learn

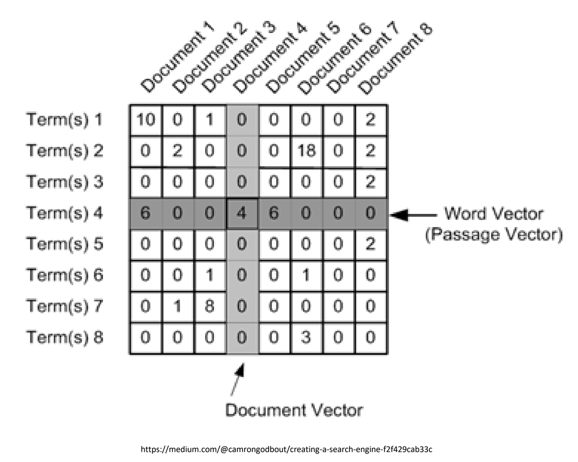

Figure 2: Illustration of Bag-of-Words model

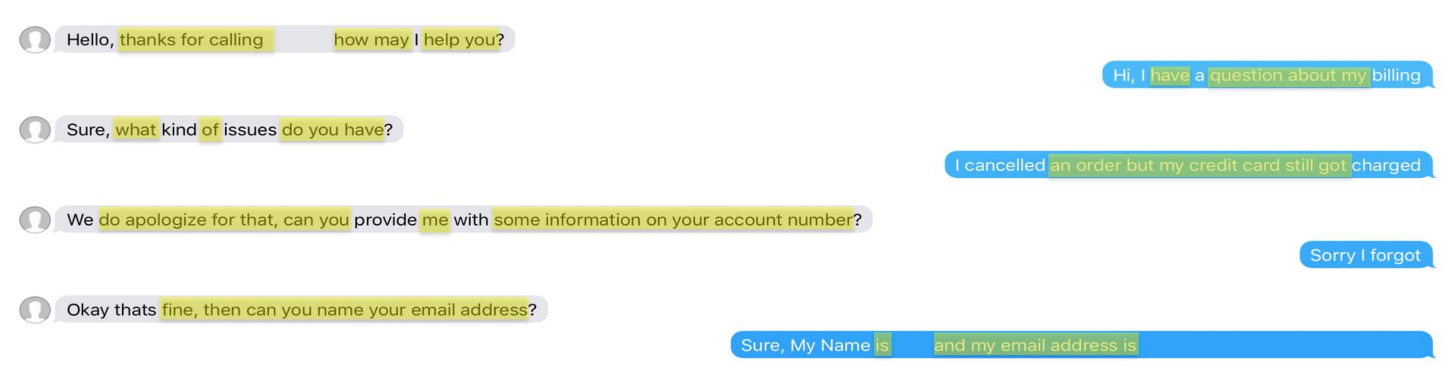

Figure 3: A sample conversation with key words highlighted

Neural Network

Neural Networks have proven to be one of the most powerful tools in both NLP and various other deep learning tasks. The theory behind a neural network is as follows.

Forward Propagation
During forward propagation, the neural network takes in several inputs (*the 300-d embedding vector in our case*). It processes the inputs through multiple neurons (*64 neurons in each layer in our model*) and in multiple hidden layers (*3 hidden layers in our model*). The operation on a single neuron is shown in Figure 4, where the weighted sum of the inputs passes through an activation function (*ReLU function in our model*). The output generated is passed to the neurons in the next hidden layer.

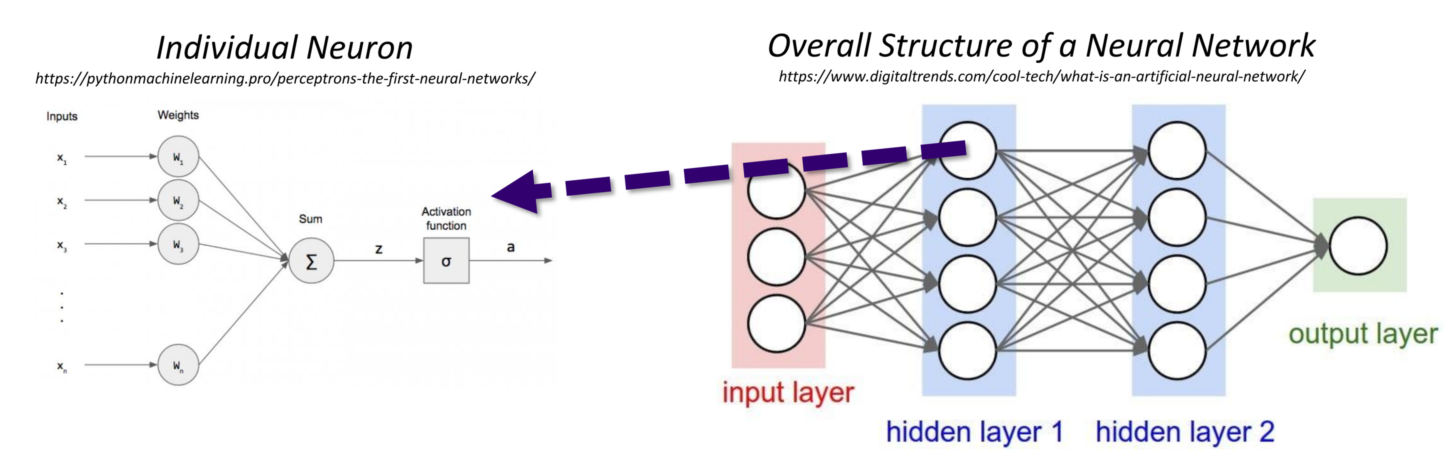

Figure 4: Illustration of forward propagation in a neural network

Backward Propagation
During backward propagation, the model tries to minimize the weights of neurons which contribute more to the error. This is done by first computing the loss function (*cross-entropy in our model*), then computing the gradient at each neuron. The model uses updating algorithms (*Adam Optimizer in our model*) to adjust the weights, following the direction of the gradient.

Iteration
Each iteration is defined by one operation of forward propagation and one of backward propagation as stated above. The training process uses multiple iterations as the loss function converges.

Data Analysis

Data Set
The data set is based on over 29,000 raw transcript files provided by Spoken. 80% of the conversations were allocated for the training set, and 10% each for the validation and test sets.

Quantitative Results
The implemented Neural Network training slowly converges to about 90% accuracy over 100,000 iterations, and the loss decreases to about 0.3 *(Figure 5).*

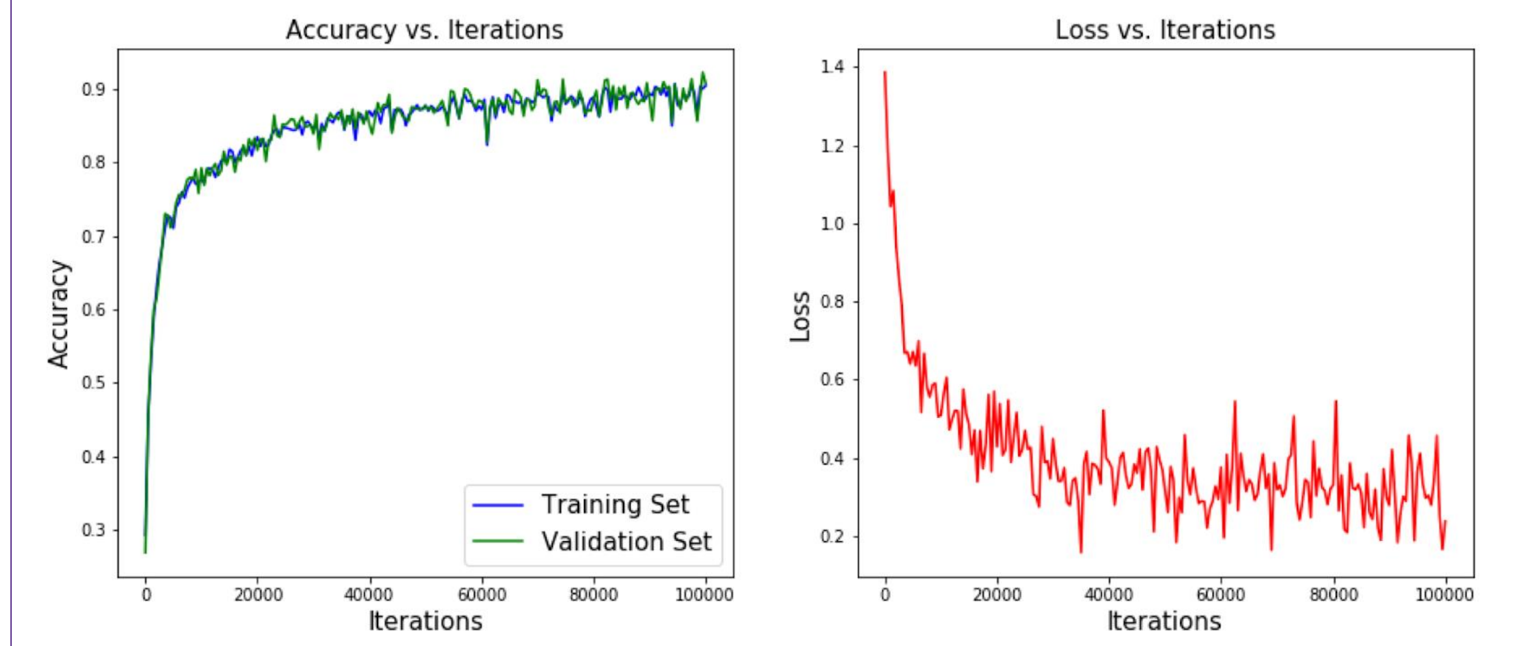

Figure 5: Accuracy and loss of our neural network on training and development data

Discussion

Academic Results
Our system uses raw transcripts to predict call duration using speaking characteristics and key words. We analyzed these with call metrics and word-encoding vectors. We observed that the metrics were ineffective, but the TF-IDF word-encoding vector showed meaningful results.

Various machine-learning models were implemented and tested. They included a Neural Network, Random Forest, and K-Nearest Neighbors. Only the neural network proved to be effective at predicting call duration.

Professional Results
Beyond profiling call duration, we set out to find ways to improve customer-service by investigating the causes of call performance. Our results showed a relationship between the words and phrases used within the first minute and the call duration, but refuted a relationship between certain speaking characteristics and the call duration.

From these results, we can infer that an individual who lacks confidence or is unfamiliar with the language can still maximize their customer-service potential by carefully selecting the words they use in a conversation.

Future Opportunities

Practical applications:

- Apply to a live transcript
  - Quick feedback to agent
  - Not designed for *direct* improvement
- Determine key words/phrases
  - Script development, training
  - Manager oversight, performance reviews

Improvements:

- Classify within 2 or 3 minutes
- Analyze after 10, 15, or 30 seconds
- Refine model once results are applied
  - Knowing the expected length may change agent behavior

Acknowledgements

The success of the project is immensely due to the excellent mentorship of Matthew Peters and Yishay Carmiel of Spoken Communications. Yishay constantly motivated us to investigate alternative solutions and expand our system. Matthew regularly provided guidance and advice, and worked with us when we struggled. Both of them encouraged us to be self-reliant, allowing us to learn for ourselves. For their sponsorship, mentorship, and encouragement we are most grateful.

Thank you Matt and Yishay!