2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)

# Deep Learning at Your Fingertips

Jonathan Rogers
*Truman State University*
jpr8237@truman.edu

Dylan Simmons
*Saint Louis University*
dylan.simmons@slu.edu

Milesh Shah
*University of Missouri*
mas2g2@missouri.edu

Connor Rowland
*University of Missouri*
carfzf@missouri.edu

Yi Shang
*University of Missouri*
shangy@missouri.edu

***Abstract*—From SurveyMonkey to Google Forms, online surveys have become a cornerstone of modern research. However, these survey platforms lack the ability to provide advanced analysis to researchers, often requiring expensive third-party analytics software to rectify their shortcomings. We propose to solve this problem by adding data analysis capabilities onto TigerAware, an existing data collection platform. TigerAware offers a generic and customizable tool which allows researchers without technical expertise to create, manage, and deploy custom mobile surveys to participants in real-time. We seek to add data analysis functionalities to the TigerAware platform ranging from basic statistics functions to emotion recognition via deep learning. Our analysis platform uses data collected by TigerAware to give researchers real-time analytics throughout the duration of their study. Through our additions to the TigerAware platform, we present a novel all-in-one tool for researchers to facilitate effective survey creation, survey administration, data collection, and data analysis.**

***Index Terms*—mobile survey, mobile sensing, customizable survey and data collection, data analytics, emotion recognition, deep learning, cloud computing**

## I. Introduction

One of the most common methods used to gather research data is surveys. Modern analysis of big data in real-time requires extensive technical knowledge. If a researcher is looking to collect data via surveys and analyze the results of the surveys, they either have to implement their own system for collecting and analyzing data or find existing compatible suites or platforms to provide the analysis for them. We set out to address this issue by adding real-time data analysis features to TigerAware, an existing data collection platform. These analysis tools include data analysis features ranging from basic statistical analysis functions to advanced deep learning models. In doing so, we will give non-technical researchers a tool to facilitate more efficient research with a new all-in-one platform for survey creation, data collection, and data analysis. Furthermore, the ability for TigerAware to capture data via external bluetooth-equipped sensors allows researchers to ask new questions that require custom or unique data types.

## II. Related Work

### A. TigerAware

The TigerAware [1] platform, created in the Distributed and Intelligent Computing Lab at the University of Missouri, is an innovative mobile survey and sensor data collection and analytics system, designed for use by researchers across all disciplines. TigerAware gives researchers the technical tools they need to facilitate data collection. The TigerAware architecture consists of three main components: a survey creation and administration dashboard, a Firebase server for storing data and distributing surveys, and Android and iOS mobile applications for data collection. We set out to add data analysis functionalities to the existing TigerAware platform, ranging from basic statistical analysis to image classification and emotion recognition.

The work is supported in part by NSF CNS-1659134 grant.

### B. Emotion Recognition

Emotion recognition is the procedure of classifying human emotions through modalities such as facial and verbal expression. Emotion recognition plays a key role in our everyday social interactions, because the meaning in our social interactions is determined not just by what we communicate, but how we communicate it. Emotions are expressed in a multitude of measurable ways, most commonly through speech patterns and facial expressions. We are going to add functionalities to the existing TigerAware platform that use deep learning to perform emotion recognition.

Emotion recognition through speech patterns is done by the analysis of prosodic features: the intonation, intensity, and rhythm of the speech next to voice quality features [2]. Furthermore, analysis of the word content of speech such as word choice, word ordering, and word frequency can all be indicators of emotional states [3]. Emotion recognition through facial expressions is done by the analysis of facial features such as the positioning of eyebrows, lips, and cheeks [4]. To increase empathic accuracy, modern research has found that one should analyze these facial features as a whole rather than in isolation [5].

There are several obstacles with using deep neural networks to understand human emotions. In speech recognition, cultures and accents can confuse models which are not accurate enough to differentiate between emotions such as anger or happiness [6]. Facial recognition also suffers from variables such as skin tone, facial hair, existing similarities in the features of particular facial expressions, and general facial obstructions in images [4]. Due to these issues with individual emotion recognition algorithms, empathic accuracy is enhanced when using a multi-modal interface [6].

With the introduction of many digital assistants such as Alexa, Google Assistant, and Siri, human interaction with computers has become a routine part of everyday life [2]. Via natural language processing, these assistants can convert your spoken commands to actionable text, and via facial recognition can authenticate you via your individual facial characteristics. Despite the data that these digital assistants are capable of obtaining, these softwares still do not have the functionality of recognizing the emotions of the user. This data can add more meaning to the interactions between the user and the computer, in the same way that emotion adds meaning to human interactions.

### *C. Data Analysis Integration in TigerAware*

We are adding data analysis functionalities to TigerAware to enhance the experience for non-technical users. Adding data analysis to TigerAware is a natural evolution of the platform due to the increasing demand for technical analysis of data gathered from traditionally non-technical disciplines. Furthermore, by incorporating this analysis into TigerAware, the platform can allow researchers to plan their surveys in parallel with their desired form of analysis - rather than moving data between multiple platforms - saving researchers valuable time and effort.

The suite of analysis modules we are adding to the TigerAware Platform are meant to cover a variety of potential research situations and demonstrate the breadth of analyses that could be handled by the platform. The first added analyses are simple but practical - these include basic statistic operations such as mean, median, mode, and standard deviation. These analyses are added not only for data analysis but also as a starting point for the construction of the analysis suite and as a proof of concept. More technical analyses which would be difficult for researchers without a technical background to perform are added to give researchers powerful tools without the necessity to understand the methods used to create or utilize complicated algorithms. The last analyses added to TigerAware are the emotion recognition models, which demonstrate the ability of our platform to provide advanced and cutting-edge deep learning analysis to researchers in any field or discipline.

## III. System Architecture

The goal of our research is to complete an all-in-one survey creation, data collection, and real-time data analysis platform. We will accomplish this by adding real-time data analysis features to the TigerAware platform, consisting of functionality ranging from basic statistical analysis to advanced deep learning models. In completing the platform, we will allow non-technical researchers to analyze their survey data via deep learning without having to learn the technical skills required to understand and implement complex algorithms. Fig. 1 illustrates the TigerAware architecture and the additional components that we have added. We have designed the system to be modular, with Firebase being the central connection point for most of the components. We have added an Amazon Web Services (AWS) EC2 Instance [7] to run server-side Node.js [8] scripts and analysis algorithms. The Node server retrieves new survey data stored in Firebase and acts as the logic unit to pipeline the data to the appropriate statistics or

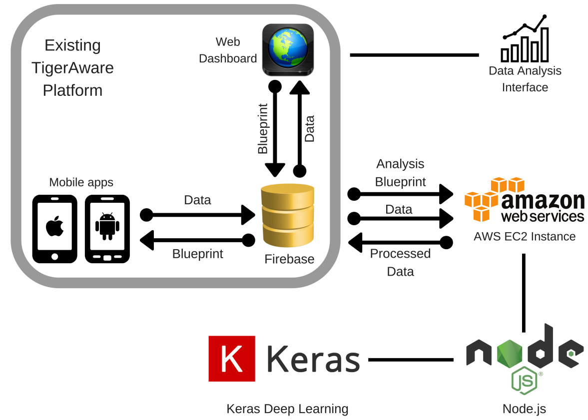


Fig. 1. TigerAware architecture.

analysis module. Additionally, a data analysis interface was added to the web dashboard. This interface allows a researcher to choose which of the available data analysis modules they would like to use for their survey. Then, these analyses are run in real-time on data collected from the deployed survey. The end goal of our work is to give non-technical researchers a tool to analyze their data via deep learning. With our platform, non-technical researchers can perform technical analytics on their research data without having to manually implement the analytics themselves or transfer data to expensive third-party analysis software. Researchers can easily create, administer, and manipulate the results of their surveys in one robust and comprehensive platform. TigerAware effectively increases accessibility of research tools and practices to non-technical fields, and these easy-to-use advanced research tools will help produce higher quality research. Our additions to the TigerAware platform support functionalities for emotion tracking of the survey participants, which has use cases such as depression prevention, addiction and withdrawal monitoring, and engagement detection.

## IV. Implementation

TigerAware supports functionality for users to create and deploy mobile surveys which survey participants can take on their mobile devices. Currently, the only way to analyze survey data is to export the survey results to a .csv file and then use a third-party platform or suite to complete the analysis. This work creates an analysis suite consisting of real-time analysis functionalities to be added directly to TigerAware.

### *A. Web Dashboard*

The front-end is built directly into the existing TigerAware platform. The front end provides researches with a graphical user interface to select different kinds of analyses while creating a survey. To select the desired analysis, a researcher picks from a list of available analyses. A new analysis can also be added to a survey post-creation on the survey display page. Once an analysis type is selected, an analysis blueprint is created and stored in parallel to the survey blueprint in

Firebase. The analysis blueprint contains the survey name, question name, type of analysis, and relevant metadata.

### *B. Server*

The Node.js server acts as the logical unit of our system and as the main point of connection between the new components and existing TigerAware architecture. An Amazon Web Services (AWS) EC2 instance is used to host all of the data analysis modules. When a user completes a survey on the mobile app, the new survey data is pushed to Firebase. The Node.js server will then pull the new data so it can be evaluated. All relevant data will be retrieved from the response along with the instructions for completing the desired analysis. The data is then distributed to the appropriate data analysis module and the analysis is computed. Once the data has been processed, the Node.js server uploads the newly processed data back to Firebase along with the analysis result. The Node.js server is written using promises, which allow for the data to be processed asynchronously in order to ensure data consistency and integrity. Since Firebase stores data in JSON format, we developed an appropriate JSON schema to store analysis instructions and data. With this schema, new analysis types can be added to the existing library quickly and easily.

### *C. Analytics Engine*

The back-end portion of this project is centered around providing the functionality of data analysis to non-technical researchers. With the new platform, researchers will not have to perform any statistical calculations or analyses on their own or purchase expensive third party software to analyze the data once it is collected. Our platform aims to perform many different types of analysis, ranging from the simplest statistical analysis on numerical datasets to cutting-edge deep learning analysis on images, audio, and other complex data types. As a proof of concept, we implemented a few existing and pre-trained models that utilize image and audio data to predict the emotional state of a user. The task of emotion recognition will be performed by taking both image and audio input from the user and passing it through trained models for either facial or speech recognition. These models are pre-trained on existing datasets.

### *D. Firebase*

Fig. 2 shows an example of the schema for storing analysis blueprints in Firebase. Each analysis blueprint is associated with a unique 32-bit survey key that is generated by Firebase at the time a survey is created. Below that, there is a zero-indexed list corresponding to each question within the survey that requires analysis. Finally, each of these questions contains all of the meta data needed to run the desired analysis, including the analysis type (`analysis`), question identifier (`questionId`), and other meta data related to the specific analysis type. There may also be another child, `data`, which indicates that the desired analysis requires data from more than one question to compute. For example, in Fig. 2, the

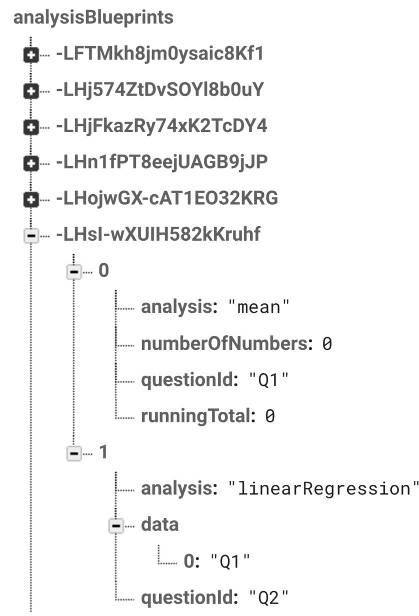


Fig. 2. An example of the analysis blueprints.

`linearRegression` analysis type requires data from two questions: `Q1` and `Q2`.

Fig. 3 shows an example of the schema for storing analysis data in Firebase. Whenever a survey participant completes a survey, their responses are automatically stored in Firebase in the `data` object. The `data` object contains a list of surveys with data. The keys for each survey object are unique 32-bit keys generated by Firebase at the time of the survey creation, used to identify each survey across Firebase. All survey response data is stored in in its respective survey object. Each survey object contains an `answers` object, which contains a list of all user responses for the given survey. The key for each response is, again, a unique 32-bit identifier created by Firebase at the time the participant completes the survey. Each of these response objects contains a `surveyData` object containing the question ID and participant response for each question in the survey, saved as a key-value pair. The response object also contains other meta data, such as the `timeStamp` of when the survey was completed and `userID` identifying which user submitted the reponse. If the survey is to have analysis run on it, then an `analysisData` object will also be added to the response once analysis is completed and stored in Firebase for the first participant response. The `analysisData` object contains a list of questions, indexed by question ID, with each question containing the relevant computed analysis data.

## V. Preliminary Results

We implemented simple scripts to compute basic statistical analysis as well as deep learning models for emotion recognition based on text sentiment analysis, facial expression recognition, and speech audio feature analysis.

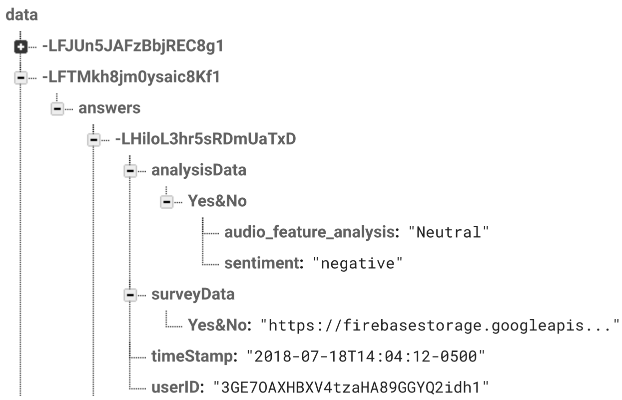


Fig. 3. An example of the analysis data schema.

### A. Statistical Analysis

The first analysis category introduced to the TigerAware platform is basic statistical functions such as mean, median, mode, standard deviation, and variance. These simple analyses show that the platform can handle traditional analysis techniques on a dynamic dataset in real-time. With our implementation, these statistics can be computed in real-time as new data is submitted by users in the field. This allows researchers to see the current mean, median, mode, etc. of their data at any point in the study. In addition, these statistical analysis methods show that the platform can handle analysis techniques which require data from multiple responses or multiple questions.

### B. Text Sentiment Analysis

For text sentiment analysis, we used the Python Software Foundation's pre-trained speech recognition module, called "Speech Recognition". We also used the following attributes from the "SciKit Learn" module: database to load the datasets to train the module, feature-extraction, naive-bayes, model-selection, and matrix. We also use the 'Natural Language Kit' module to import movie review datasets containing 2000 .txt files. Each of these reviews is labeled as 'positive' or 'negative'. Based on this movie review dataset, this module achieves an overall accuracy of 82%. This module was then tested on a dataset of 42 audio files also labeled as 'positive' or 'negative'. For this test set, we achieved a satisfactory accuracy for positive sentiment of 100% and an accuracy for negative sentiment of 77% - an overall accuracy of 88.5%. Later, we tested the module on 105 audio files, again labeled 'positive' or 'negative', and achieved an accuracy of 92% for both positive and negative sentiment.

### C. Facial Expression Recognition Analysis

For facial expression recognition analysis, we downloaded a pre-trained facial expression recognition deep neural network from a GitHub repository called Facial-Expression-Recognition [9], which used libraries for deep learning such as the Keras [10] module in Python, and its attributes models, layers, and metrics. The layers attributes were used to build each layer of a convolutional neural network to be used for extracting raw image data for image classification. The convolutional neural network used for image classification consists of four convolutional layers and two fully-connected layers. We converted the images that were used for training into two-dimensional arrays that contain RGB values. This network was trained on a dataset used in the 2013 Kaggle Challenge for facial expression recognition. This dataset contains 28,709 labeled images with dimensions 48x48. The deep neural network achieved an accuracy of 65.45%. This is a low accuracy when compared to the sentiment analysis module. We trained this module further but we were not able to improve its accuracy much. After training this module, we tested the module on 128 self-produced images from each member of our team, labeled with the corresponding emotion.

### D. Speech Audio Feature Analysis

For audio feature analysis, we used a GitHub repository called DeepSentiment [11] that contained a pre-trained module which extracts features from a speaker's voice, such as loudness and pitch, and uses these features to predict emotion. The predictive accuracy of various expressions are shown below, similar to previous work [6].

| Anger | Disgust | Fear | Happy | Neutral | Sad | Surprise |
|---|---|---|---|---|---|---|
| 0.9 | 0.27 | 0.13 | 0.27 | 0.33 | 0.07 | 0.13 |

## References

[1] W. Morrison, L. Guerdan, J. Kanugo, T. Trull, and Y. Shang, "Tigeraware: An innovative mobile survey and sensor data collection and analytics system," *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, 2018.

[2] B. W. Schuller, "Speech emotion recognition: Two decades in a nutshell, benchmarks, and ongoing trends," *Communications of the ACM*, vol. 61, no. 5, pp. 90–99, 2018.

[3] P. Vateekul and T. Koomsubha, "A study of sentiment analysis using deep learning techniques on thai twitter data," *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2016.

[4] Y. Kim and E. M. Provost, "Emotion recognition during speech using dynamics of multiple regions of the face," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 12, no. 1s, pp. 1–23, 2015.

[5] Y. Huang and H. Lu, "Deep learning driven hypergraph representation for image-based emotion recognition," *Proceedings of the 18th ACM International Conference on Multimodal Interaction - ICMI 2016*, 2016.

[6] C. Busso, Z. Deng, S. Yildirim, M. Bulut, C. M. Lee, A. Kazemzadeh, S. Lee, U. Neumann, and S. Narayanan, "Analysis of emotion recognition using facial expressions, speech and multimodal information," *Proceedings of the 6th international conference on Multimodal interfaces - ICMI 04*, 2004.

[7] Amazon, "Amazon web services," https://aws.amazon.com/ec2/, 2018.

[8] "Node.js," https://nodejs.org/en/, 2018.

[9] R. Jain, "Facial-expression-recognition," https://github.com/jrishabh96/Facial-Expression-Recognition, 2018.

[10] "Keras," https://keras.io/, 2018.

[11] V. Kovakkat, "Deepsentiment," https://github.com/vyassu/DeepSentiment, 2018.