Emotional Detection Chatboy

Savannah Michel

*#Long Island University*

*720 Northern BLVD,*

*Greenvale, NY 11548*

1savannah.michel@my.liu.edu

***Abstract*— Virtual assistants have permeated almost every aspect of our lives with the rise of artificially intelligent devices, such as Apple’s Siri and Amazon’s Alexa. Even with the wide reach, these devices lack one notable skill: emotional detection. Although these devices utilize voice capabilities, emotional detection from text remains the more challenging area in the study of emotional intelligence and detection. This paper will look at those struggles, possible solutions, and a functional emotional detection chatbot.**

***Keywords***— **Include at least 5 keywords or phrases**

1. Introduction

The rise of chatbots and virtual assistants have changed how humans and computers interact. However, all of these conversational agents are lacking an element that would take them to the next level: recognizing the emotions of the human user. Being able to identify and react to the emotional state of the user would enable the agent to answer and respond to questions and commands with much more accuracy thus ensuring a happier user experience. However, the ability for these agents to detect and understand the emotions of the user will probably not be a widespread feature for some time, especially when it comes to detecting emotion in written text.

Affective computing, the interdisciplinary research regarding systems that can recognize, interpret, process, and in some case stimulate human affections and emotions, is one of the most active topics within Human-Computer Interaction (HCI), with BusinessWire expecting a $72.52 billion increase by 2023 [1]. Chatboats, in particular, offer a lucrative potential for all sorts of businesses and these businesses have discovered many innovative ways to put them to use. The scope of businesses that employ these chatbots is not narrow by any means ranging from content delivery to food service to companionship. Not only are these chatbots beneficial to the businesses using them, but also to their customers as chatbots are available 24/7, there is no need for long wait times to speak to a service representative which saves time for the customer, and they are easy to use. However, emotional conversation agents are still in their infancy with no general framework in use to implement them for a wide audience. The process of creating and maintaining these conversational agents without the ability to detect emotions is challenging enough as the designer has to anticipate all the ways a user might input a request or change the conversation. This turns these agents into having to figure out something of a guessing game that requires a vast amount of different specialties to program the agent and to update it as necessary [3], and this is not including the boy to have the ability to understand semantics, syntax, and other cues that humans have been trained to identify since birth.

To understand the now wide reach of chatbots, it is best to look at where they started: with the famous conversational agent Eliza. Eliza was an experiment to see the natural language processing that occurred between a human and a machine created between 1964-1966 in the Artificial Intelligence Laboratory within MIT by researcher Joseph Weizenbaum, and had the ability to recognize key words or expressions before sending back questions derived from the key works and if it wasn’t able to return with a question ti would respond with “I understand” [2]. Weizenbaum created Eliza to simply be a caricature of human conversation, yet users were confiding all their thoughts, memories, and personal problems to the agent. This caused experts to declare that chatbots would be indistinguishable from humans within the matter of a few years, which was a statement that Weizenbaum rejected as he doubted machines could ever replace human intellect. His argument centering around that these devices were just tools and an extension of the human mind, and that a computers’ understanding of language remained dependent on the context of which the computer was being used, and that a computer would never have a general understanding of human language. The end of Eliza came when Weizenbaum’s secretary was caught using the agent to try and solve issues she was having with Weizenbaum at the time. However, Weizenbaum’s model have been a basepoint for almost all chatbots following Eliza and the end goal for more human-like interactions with passing the Turing test to see if the conversational talents of the bots could pass against a board of human judges with the hardest thing in that test being the lack of limits on what people can chose to discuss.

From Eliza, two different conversational agents have derived and become increasingly popular with an estimated 80% of companies using some form of artificial intelligent agents, usually a chatbot [1]. These chatbots can function within two different domains: open and closed domains. An open domain chatbot has the ability to respond to questions ranging in a great variety of topics and a closed domain chatbot can only function inside a specific domain in which they have knowledge. Outside of the domain a chatbot resides in, the type of responses a chatbot can come up with can either be retrieval-based or generative-based. Retrieval-based responses are retrieved from a dataset with a retrieval process ranging from simple rule-based sentence matching to advanced retrieval using machine learning techniques. Whereas generative-based chatbots do not give any predefined responses as they generate the responses themselves which enables them to handle questions that relate to the underlying dataset but also questions outside of that dataset. These responses tend to usually be more human-like and usually have what could be considered a personality. However, there are disadvantages to generative responses such as the generated answers tend to be full of grammatical errors and require large amounts of training data. While open domain seems to be the better option as it enables chatbots to pull from all areas, open domain retrieval-based systems are considered impossible to build and open domain generative system have yet to be successfully built, but closed domain generative and retrieval systems are currently in use and applied in practice today with the hope that when generative-based techniques have improved that generative response chatbots will be in practice then.

Sentiment analysis tends to be the strongest method for identifying the real time emotion of the human user and is one of the more common classification tools as it can analyze a statement within text and predict sentiment, emotion, or polarity from the statement. Sentiment analysis is the contextual mining of text to identify and extract information, sometimes subjective, from the source material or the conversation at hand. These systems usually reside on using lexicons to pair lists of words with the emotions, polarity, or sentiment they could convey. Sentiment extraction can be difficult as the need for preprocessing work needs to be done and some preprocessing work requires a large number of resources, some being the earlier mentioned lexicons but many other resources have to be created by the programmer. There are various methods and algorithms to approach sentiment analysis such as:

*A. Automatic Systems*

These rely on machine learning techniques and algorithms to learn from the data and ongoing inputs from users. The first step of this technique is the training of the model which starts with feature extraction from the text. The text is transformed into a numerical vector or other form of data that enables the algorithm to easily embed the words and group those with similar meanings into similar representations. Some well known models that do this are bag-of-words and bag-of-ngrams. After the text is vectorized, the next step is to classify the text through different classification algorithms and usually involves a statistical model such as Naive Bayes, Linear Regression, Support Vector Machines, and Deep Learning models.

The Naive Bayes models are a family of probabilistic algorithms that utilize Bayes’s Theorem to help predict the category or sentime of the text.

Linear regression models is a statistical algorithm that is used to predict value (Y) against a set of features (X).

Support vector machines are non-probabilistic models that use text representation of examples and use it to mark points in a multidimensional space. Sentiments are mapped into unique spaces and new text is assigned a category based on similarities to already mapped text.

Deep learning models are a diverse set of algorithms that attempt to mimic the human brain by employing artificial neural networks to process the text data.

*B. Rule-based Systems*

As the name suggests, this system performs sentiment analysis based on a previously prescribed set of rules. These rules may include natural language processing techniques such as tokenization, stemming, and parsing. These systems start with a list of words that are connected with either polarizing statements or emotions. Then the given statements are collected where the words are separated by the list of words and counted towards what is being measured, and then the sentiment is identified based on the amount of words in each category. So if there are more positive words than negative, the statement would be classified as one with positive sentiment.

These systems tend to be very naive in nature as they lack the ability to take into account how words can be combined in a sentence, unless new rules are added. However, if new rules are added it can affect how previous rules are carried out meaning these systems often require lots of fine-tuning and regular maintenance.

*C. Hybrid Systems*

This system combines the automatic systems with rule-based systems and takes the desirable attributes of both techniques to make one system.

1. The Experiment
2. *Dataset and Data Preprocessing*

The dataset I used for this emotional detection experiment was the ISEAR dataset which is labeled with emotions such as sadness, fear, guilt, anger, shame, joy, and disgust. As neural networks cannot accept words or sentences as input, all data of this type must be converted into a format that will work with the network to be able to be processed. To ensure that each sentence and word was changed to the correct vectorized format, I started by using a LancasterStemmer from the Natural Language Tool Kit (NLTK) to get to the base of all words to help the semantic analysis have an easier time grouping and understanding different words. After running the LancasterStemmer on the dataset, I ran a Tokenizer to help break down the sentences to smaller variables such as single words within a phrase. After running a couple different tokenizers on the dataset, I used OneHotEncoder to convert the categorical data to numerical data.

1. *Model Implementation*

After the data has been converted to numerical data so the neural networks can process and learn from it, the next step is setting up a model. The model I used utilized Keras with TensorFlow backend in a Jupyter Notebook running Python3. When training the model, I used an Adam optimized with 20% of the dataset being used for training. The results of the training were a Bidirectional GRU with an accuracy score of 92.08% and a Bi-directional LSTM with an accuracy score of 87.96%.

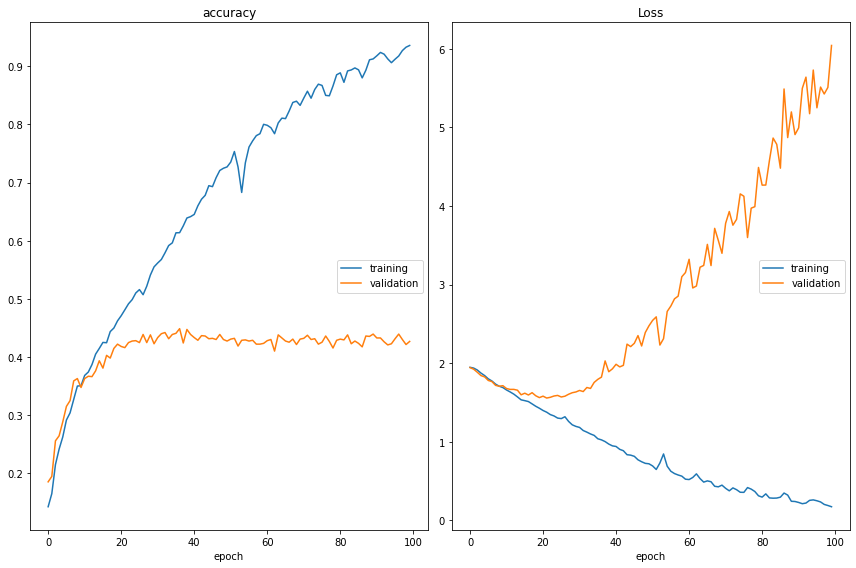


Figure 1. Bi-directional GRU accuracy and loss charts.

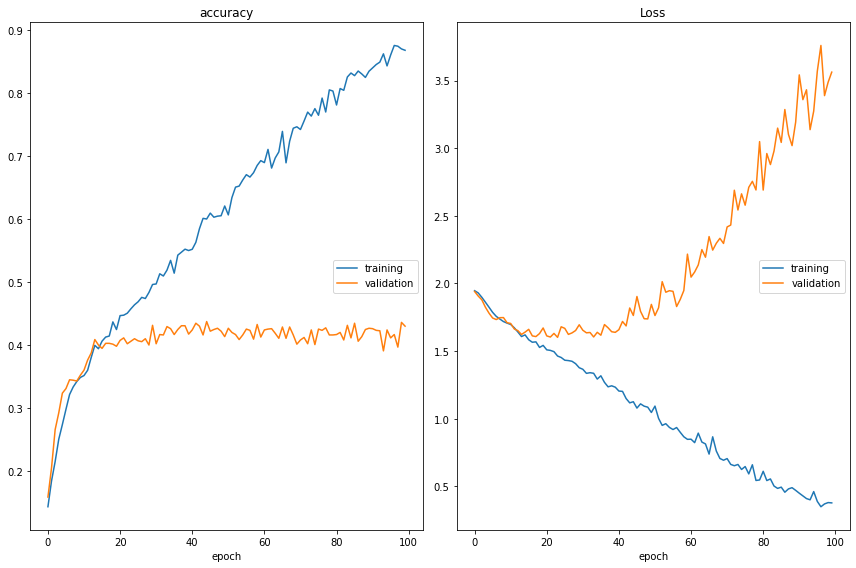


Figure 2. Bi-directional LSTM accuracy and loss charts.

1. *Results*

After running the training parameters, I ran some testing phrases to see how well the model could identify the emotion behind the phrase. Each statement was almost always correctly labeled, with some exceptions being when guilt and shame were confused. As these are similar emotions, I did not think this was an error with any parameters or the training or testing of the model and rather was just an issue with semantics or properly identifying words that have very similar meanings.

1. *Chatbot*

After running the training and a few tester phrases to see how the emotion detection was working, the next step was setting up the chatbot. The chatbot for this project was a retrieval based with no generative properties that only responded from a set list of phrases. To state, the chatbot would inquire about the user’s name before asking what is on their mind. From there the user can either agree to an analysis of their statement or decline. The chatbot will then offer to tell the user a joke or give another analysis.

1. Conclusion

Overall, the emotional detection proved to be accurate with only mild mislabeling when it came to similar emotions such as shame and guilt. To continue this project in the future, I would like to be able to use more datasets to hopefully have a wider understanding of the different emotions used and perhaps add other semantic features to be identified. The chatbot is the area where the most work could be done and added to increase its abilities. For the chatbot, I would like to add a generative method so when the bot responds it doesn’t have the same pre-programmed responses everytime used.

References

1. Chopra, Anoushka. “21 Vital Chatbot Statistics for 2020: Interactive Content on Outgrow.” Outgrow, 16 Dec. 2020, outgrow.co/blog/vital-chatbot-statistics.
2. Fainchtein, Leah. “Generative vs Retrieval Based Chatbots: A Quick Guide.” Medium, CloudBoost, 2 Dec. 2020, blog.cloudboost.io/generative-vs-retrieval-based-chatbots-a-quick-guide-8d19edb1d645.
3. Sentiment Analysis Guide. MonkeyLearn. (n.d.). Retrieved December 10, 2021, from https://monkeylearn.com/sentiment-analysis/
4. Yin, J., Chen, Z., Zhou, K., & Yu, C. (2019). A deep learning based chatbot for campus psychological therapy. arXiv preprint arXiv:1910.06707.