Automatic recognition of the user’s emotional state in one-to-one chat

Francesco Iemma, Lorenzo Massagli, Niko Salamini, Olgerti Xhanej  
 [email@email.com](mailto:email@email.com), [email@email.com](mailto:email@email.com), [email@email.com](mailto:email@email.com), [email@email.com](mailto:email@email.com)

ABSTRACT

Emotion Detection can be performed through data that can be obtained from a Chat Application, such as text and audio messages. In this document we present a possible approach to gather and classify heterogeneous data that comes from one-to-one chats. Firstly, there is an introduction to give a proper context to the problem and the usefulness of Emotion Recognition of Users. In the second paragraph there is an in-depth review of the building blocks that made-up our system. Then there will be a description of the experimental results that have been carried out for testing the system on the task. Finally, in the conclusion, there is a general sum-up of the experimental results, with some possible improvements and takeaways. [200 words]

1 Introduction

[Here you have to make clear which is the scenario, which problem are you trying to solve, which is the main difference compared to existing approaches]

Emotion Detection of humans is a technology that can be exploited by companies to understand people perception of a particular product or can help to diagnosticate mental issues to improve quality of life of human beings.

Such task is commonly performed on images and text data. For instance, performing emotion analysis from Tweets taken from Twitter is a huge research topic.

Nowadays chat applications are wide spreading at unforeseen velocity: mobile apps such as WhatsApp counts approximately 2 billion of monthly active users all over the world. Those applications involve different sources of data that can be sent: two of the most common communication methodologies are classic text messages and audio messages. Those could be used for making emotion detection more feasible since there is available a very large dataset and probably involves a more heterogeneous set of users.

2 Architecture

The developed solution for performing Emotion Detection comprehend different modules: a Chat Application for gathering data to be classified, a Firebase project for storing users, messages and labels, and a Flask application that exposes a Rest-Api for performing the actual classification.

2.1 Chat Application

In order to make possible the gathering of data for doing the emotion detection, a chat application has been developed. The aim was to take inspiration from some popular apps available such as WhatsApp or Telegram for creating the chat interface. Some functionalities such as loading of messages through scrolls has been added in order to reduce the number of messages to a fixed size and also to reduce the actual reads in the Realtime Database of Firebase.

<<chat interface image>>

**Figure 1: Chat Interface**

Users have the possibility to send two different sources of data, as said before: text messages and audio messages. During the first phase of development a popup could appear, asking to the User to perform some manual labeling for a subset of messages in the chat. This was done to gather a sufficiently large dataset to obtain a ground truth for evaluating the models that have been adopted.  
In the right-corner of the chat interface there is an emoji which represents the ongoing perceived emotion obtained from the last messages.

2.1 Firebase

The data that is utilized for performing operations in the chat is collected through Firebase service: chat messages and users are stored in the Firebase Realtime Database, and audio messages and labels are stored by exploiting the Firebase Storage.

2.1 Models for Emotion Detection

Emotions can be modeled in many different ways, but the adopted approach consist of modeling emotions from a fixed set of labels: joy, sadness, neutral, fear, anger. The neutrality has been added, as indicated in [], for not creating biases with uncertain labels. Emotion Detection is performed by leveraging Artificial Intelligence. For what concern text messages, as indicated in [], an SVM model has been used.

<SVM image>>?

3 Experimental results (or another name for the section)

Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here.

3.1 Heading Level 2 (for subsections)

Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here.

3.2 Heading Level 2 (for subsections)

Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here.

4 Conclusion

Conclusions

REFERENCES

[1] Patricia S. Abril and Robert Plant, 2007. The patent holder's dilemma: Buy, sell, or troll? *Commun. ACM* 50, 1 (Jan, 2007), 36-44. DOI: <https://doi.org/>10.1145/1188913.1188915.

[2] Sten Andler. 1979. Predicate path expressions. In *Proceedings of the 6th. ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (POPL '79)*. ACM Press, New York, NY, 226-236. DOI:https://doi.org/10.1145/567752.567774

[3] Ian Editor (Ed.). 2007. *The title of book one* (1st. ed.). The name of the series one, Vol. 9. University of Chicago Press, Chicago. DOI:https://doi.org/10.1007/3-540-09237-4.

[4] David Kosiur. 2001. *Understanding Policy-Based Networking* (2nd. ed.). Wiley, New York, NY..

Conference Name:ACM Woodstock conference

Conference Short Name:WOODSTOCK’18

Conference Location:El Paso, Texas USA

ISBN:978-1-4503-0000-0/18/06

Year:2018

Date:June

Copyright Year:2018

Copyright Statement:rightsretained

DOI:10.1145/1234567890

RRH: F. Surname et al.

Price:$15.00