Institut für Maschinelle Sprachverarbeitung

Universität Stuttgart
Pfaffenwaldring 5b
70569 Stuttgart

Bachelorarbeit

# Path to solve Conflicts in Chats with the help of NLP

Eric Oliver Hämmerle

**Studiengang:** Medieninformatik B.Sc.

**Prüfer:** Prof. Dr. Jonas Kuhn

**Betreuer:** Markus Gärtner

**begonnen am:** 01.06.2019

**beendet am:** 01.12.2019

**Erklärung (Statement of Authorship)**

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst habe und dabei keine andere als die angegebene Literatur verwendet habe. Alle Zitate und sinngemäßen Entlehnungen sind als solche unter genauer Angabe der Quelle gekennzeichnet. Die eingereichte Arbeit ist weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens gewesen. Sie ist weder vollständig noch in Teilen bereits veröffentlicht. Die beigefügte elektronische Version stimmt mit dem Druckexemplar überein.[1]

(Eric Hämmerle))

[1]Non-binding translation for convenience: This text is the result of my own work, and any material from published or unpublished work of others which is used either verbatim or indirectly in the text is credited to the author including details about the exact source in the text. This work has not been part of any other previous examination, neither completely nor in parts. It has neither completeley nor partially been published before. The submitted electronic version is identical to this print version.

# Contents

**Abstract**

This thesis concerns the development and evaluation of Chattitude, whose purpose it is to influence textual conversations. To achieve this, modern Natural-Language-Processing(NLP) systems are integrated into Chattitude. The results of these analysis approaches, namely the emotional and profane contents of a message are then visualized in the user-interface, so that users are encouraged to reflect over the content they are about to send. Through a user study the user experience of Chattitude and the effect of its signature features, the visualizations of the integrated analysis systems, is evaluated. The results show that the user experience of Chattitude is sufficient enough to have no negative influence on the study. In one metric it even closes the gap between it and other commonly used applications. The effect of the signature features is not significant on the user experience of Chattitude, but improvements and fine-tuning is still needed to achieve a notable positive influence on conversations.

**Zusammenfassung**

Diese Arbeit befasst sich mit der Entwicklung und Evaluierung eines Web-Applikations-Prototypen namens Chattitude, der die Aufgabe hat, textbasierte Gespräche zu beeinflussen. Um ein solches Ziel zu erreichen, wurden Analysen der Maschinellen Sprachverarbeitung in Chattitude integriert. Die Ergebnisse dieser Analysen, speziell Erkennung von Emotionen und Profanität in Textnachrichten, werden dann so visualisiert, dass Nutzer zu einer erneuten Reflexion über den Inhalt der zu sendenden Nachricht ermutigt werden. Durch eine Nutzerstudie wurde die Benutzererfahrung von Chattitude und der Effekt der Signatur Features, also der genannten Visualisierungen, evaluiert. Die Ergebnisse dieser Studie zeigen, dass die Benutzererfahrung von Chattitude ausreichend ist, sodass sie keinen negativen Einfluss auf Diskussionen und deren Lösungen darstellt. In einer verwendeten Metrik, erreichte Chattitude Werte, die mit häufig verwendeten Chat Anwendungen vergleichbar sind. Die Auswirkungen der signatur features sind nicht signifikant bezogen auf die Benutzererfahrung, allerdings sind noch Anpassungen nötig um einen spürbar positiven Einfluss auf Chats zu erreichen.

# 1 Introduction

In recent years, our Western society struggled with social controversies on online platforms, which included all kinds of verbal harassment. With the rise of social media more and more people were able to interact and connect with each other, which generally made new relationships possible. However, through social media a huge arena for discussions, debates, and conflict was created, posing a great challenge in today's society. Due to the many factors that come into play in social conflicts, this challenge seems to be widely unsolved. One crucial facet of the problem with online conflicts, is the anonymity and lack of negative relational feedback, which causes people to lose respect. This encourages people to go overboard and use a lot of foul language.

The classical response of the platforms was to simply censor the profane words, but this approach wasn't satisfactory. Another crucial factor in social conflicts is the number of misunderstandings between the conflict parties, which sometimes cause additional conflicts as Easterbrook et al. (1993) states.

The focus of this thesis is to propose and evaluate two new approaches for better hemming of foul language as well as reducing emotional misunderstandings. The first approach which addresses the challenge of verbal violence is applying cognitive conditioning through minimal punishments through annoyances, that add up over time. According to Mowrer (1960), who discussed various theory in the field of learning, "punishment is an effective condition of learning"[2]. Here, the basic idea is to display a popup warning that a profane message is about to be sent. The user has to confirm this warning with a click of his mouse, if he or she insists on sending a profane message. This leads to associating foul language with effort in the brain.

In the long run, we can expect people to subconsciously attempt to prevent a situation in which they are confronted with this warning and think twice whether to profane or not. This feature is supposed to help reducing

[2]Mowrer (1960) p.22

the amount of profane posts, comments or messages in the long-term. The warning furthermore prevents spamming as well, which sometimes plays a role in so-called "shitstorms". A shitstorm refers to an online situation in which a huge group of people is slandering a certain person, group, company, etc. Some online platforms attempted to solve the issue of hate speech by censoring the profane contents of a text. This approach clashes with the freedom of speech, as investigated by Kjar (2019) for example, which is why this approach is uncommon on social media platforms.

The second approach addresses the problem of misunderstandings as well as the emotional dynamics in conflict situations. Emotions tend to be a very abstract concept that difficult to express, in certain situations. Especially since the people cannot see the facial expressions of their interlocutor, detecting and taking note of them becomes even more difficult. The idea of the prototype developed in this context, is to display the emotions in a message, so the user notices them. This could improve the mutual understanding and help resolving conflicts as a consequence.

# 2 Related Work

The topic of this thesis encompasses many areas in which research or development takes place. These areas include Social Sciences and Natural Language Processing (NLP). Part of the social aspects of the thesis are definition and situation of conflicts and discussions in general as well as online settings. The section for NLP will feature an overview over the used analysis techniques and possible alternatives.

**Social Science** The inspiration for the topic of this thesis flows out of the context of Social Science and Sociology, which investigates social problems among people. With the rise and extended use of social network platforms, the use of hate speech and cyber-bullying rose as well. Out of 723 of young

(age between 15 and 18 years) social media users, 67% are exposed to hate material online and 21% fall victim to it according to Oksanen et al. (2014).

Waldron (2012) studied and discussed the harm that hate speech can have on people and society, as well as potential approaches on how to cope with it. The traditional way of solving social conflicts of hate speech is to resort to legislative instruments for the reduction the amount of it, which is also discussed by Waldron (2012). This thesis proposes a solution that avoids limiting or censoring people but to influences their behavior. Unlike concrete actions that can by encouraged by utilizing the tools of the "Dragonfly Effect" discovered by Aaker and Smith (2010), hate speech can often point toward a character trait of a person or originates from habit. In such a case, lasting change might need a lot more time and constructive influence.

Hate speech usually surfaces as a result of an underlying conflict. According to Easterbrook et al. (1993) on p.2, the term "conflict" is not easy to define, but Easterbrook et al. (1993), still gives an overview over attempted definitions. Furthermore, Easterbrook et al. (1993) mentions that even though psychological aspects of software applications are focused in Human-Computer-Interaction (HCI), the social aspects of them are understudied due to the difficulty of measuring them in a experimental setting. Easterbrook et al. (1993) however shows that there is a variety of so called Computer Supported Cooperative Work (CSCW) systems, where the idea of Chattitude falls into the category Computer Mediated Communication (CMC) of under the term textual conferencing. It is also mention that characteristics suggested byClark et al. (1991) are missing, in particular "co-presence", "visibility" and "audibility", giving further reasons to improve communication in chats, given their high usage.

An other prominnent scholar in the field of conflict resolution is Galtung and Wagner (1975), who thinks of conflict as an attribute of a system in which incompatible goals exist. Incompatible goals are present when achieving one

of them would diminish the possibility of achieving the other[3]. Identifying incompatible goals in a conflict situation like an online discussion or in the case of cyber-bullying is not yet possible for automatic detection systems.

Ben-David and Matamoros-Fernandez (2016) also suggest that hate speech resulting from conflict can be explained by the policy and technological affordances of a social media platform as well as the communicative actions by their users. Consequently these factors can play a significant role in an online conflict or discussion. Deliberately utilizing technological affordance to deescalate a discussion is the key idea of the prototype.

For the evaluation the utilization of standardized surveys that measure the quality of a relationship like for example the Positive-Negative-Relationship-Quality Scale[4] developed by Rogge et al. (2017), was initially planned.

Selecting participants, it appeared more practical to use a customized survey focusing on the evaluation of a conversation rather than a relationship. The reason behind that decision is that participants didn't know each other, which makes evaluating a relationship questionable. In the future the effect of Chattitude on relationships over longer periods of time can still be investigated.

**Natural Language Processing** Liddy (2001) defines Natural Language Processing as the "[...] theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications."[5]

In this field the detection of sentiment es preceding computational technique of the emotion detection, to what Yadollahi et al. (2017) give a good overview of. Due to the active research, novel work has surfaced, for example

[3]Mediated from Galtung and Wagner (1975) p.60

[4]`http://www.fincham.info/measures/pnrq.pdf`(last access: 13.11.2019)

[5]Liddy (2001) p.2

IEST of Klinger et al. (2018) or an improved model for social media text by Rout et al. (2018).

The emotion detection is basically an extension to the sentiment analysis according to Yadollahi et al. (2017). For this purpose a classifier that is based on the aggregated corpus from Bostan and Klinger (2018), was used in Chattitude. To our knowledge, there are no corpora for emotion detection dedicated specifically for chat-conversations available, so aggregating multiple corpora might be the next best solution, but there is still room for improvement.

# 3 Implementation

Chattitude is supposed to contribute a new approach addressing social conflicts on social media. The focus of Chattitude is on the asynchronous messaging environment, similar to products like WhatsApp [6], Telegram [7] or Facebook Messenger [8].

The objective is to help deescalating a discussion in this kind of environment by making the use of profanity cumbersome and help the discussion parties of the discussion to understand each other better emotionally. To fulfill these objectives in a manner that doesn't require human resources, Chattitude is taking advantage of NLP.

One of the NLP based approaches that was integrated into Chattitude is recognition of profane words. Using knowledge about the existence of these words, the task is to discourage the user from using them. The idea of Chattitude is to help condition the user to associate foul language with too much effort or inconvenience.

[6] `https://www.whatsapp.com/`(last accessed: 27.11.2019)

[7] `https://telegram.org/`(last accessed: 27.11.2019)

[8] `https://play.google.com/store/apps/details?id=com.facebook.orca`(last accessed 27.11.2019)

This can generally be achieved by by a technique known as "Shadow-banning" for example. The idea here is to partially block the content a user creates in such a way that only the author can see his content but other users don't react to it, or as Kjar (2019) puts it: "Shadow banning is hiding someone's comments from observers while making it appear to the author that their posts are visible online"[9]. While this may work in environments of groups or forums, applying this concept in a chat environment with one-to-one interactions is likely to irritate both users. Users would feel annoyed and rightly so because they would interpret the interruption as a connection problem rather than one caused by blocking foul language.

The other problem of this approach is that it doesn't subconsciously teach users not use bad words. Consequently, a system is needed that teaches the users, thus requiring feedback. A simple way of achieving this objective in a chat environment is to display an overlay popup with a simple task to do for the user. The task used in the final version of the prototype, created in the context of this thesis, is for the user to simply to confirm their message with the mouse or touch on the screen. This task fits well into the flow of the chat application, being nothing more than a warning. The task can be varied in either direction on the difficulty scale, from a non-obstructing warning sign to the solving of a completely automated public Turing test to tell computers and humans apart (CAPTCHA) with a success probability.

Neumann (2013) implemented this concept in such a way that regardless of the correct solution of the CAPTCHA, the system only allows the user to pass with a specific probability. In `re:Fefes` case this probability originated from the probability that a given post was written by a troll[10].

The approach pursued for Chattitude for deescalating discussions is the recognition of emotions in text and displaying them. This information can influence writers as well as readers of a message. Providing this information to the authors could give them the opportunity to first reflect on the content

[9]Kjar (2019) p.8
[10]Cambridge (2019)

to be sent and, if necessary, to edit it. The resulting messages could, therefore, then contain more intentional and reflected emotions, which would lead to fewer misunderstandings on the side of the reader. This influence is increased even more if both parties have access to the same information about the emotions of a message, as the reader would presuppose that the sender has intentionally included the particular emotion into the message. Even in the case of a misunderstanding, it would be easier to pinpoint the origin if the information about the emotion of the text message is available. Potentially it could lead to more messages where emotions are being articulated.

## 3.1 Concept

In this section the final concept of Chattitude is presented by first clarifying the use-cases of Chattitude Secondly the fundamental architecture is described and thirdly the Signature Features are introduced and explained.

**Use-Cases** The general underlying use-case of Chattitude is to be able to communicate with other people, similar to established messaging services. Different from those messaging services is the app's slight influence on the communication by displaying both the emotions and the profanity in each message. Generally, four objectives guided the choices made for the features and abilities of Chattitude. The first was to make Chattitude comparable to established messaging services which provide the user with features like chat-rooms, profiles, and contacts. A not inconsiderable part of time and focus was invested into the responsiveness, usability, and performance of the Graphical User Interface (GUI). This means that Chattitude can be used on mobile phones, has a fast analysis of the message and is supporting chat rooms.

Another feature solely dedicated to performance is the use of lazy loading for the display of messages. This could potentially break – or at least significantly slow down the client if many messages were to be displayed. In

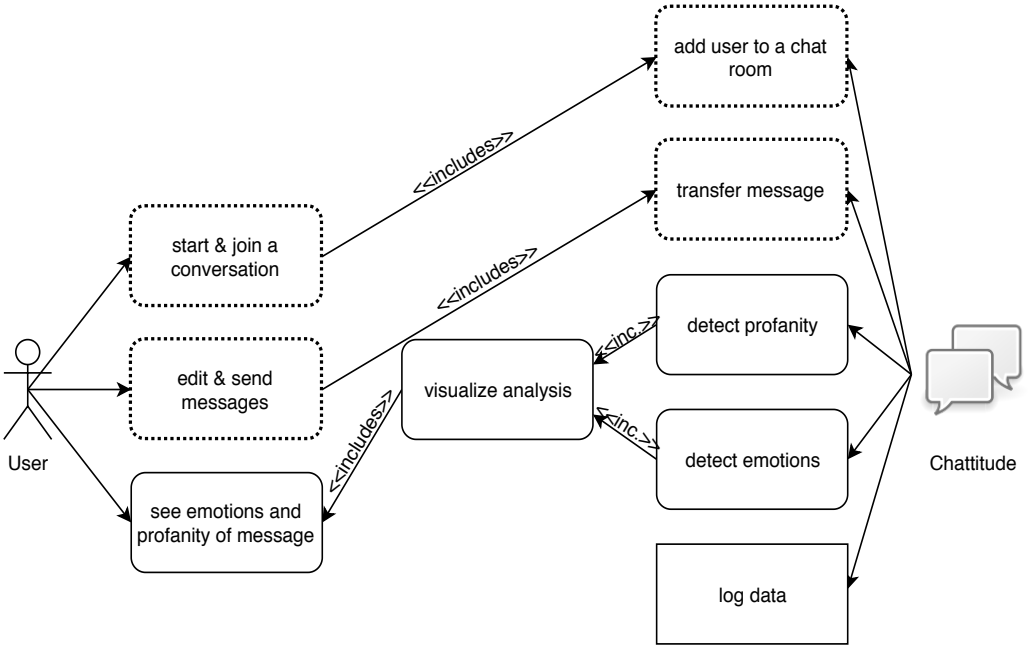



Figure 1: Use-Case diagram for the top-level use-cases of Chattitude. The dotted boxes indicate an use-case that is primarily dedicated to the standard use-cases of a normal chat-application. The continuous line with rounded corners indicates that the use-case is part of the analysis-pipeline, and the last with angular corners is a use-case dedicated to the successful execution of the user study.

the end, users should forget, that they are using a messaging service and are able to concentrate on the conversation, rather.

The second objective was dedicated to the successful execution of the study which required a setting where information from participants could easily be selected, measured and evaluated. This called for easy access to the contents inside of Chattitude. This access which was implemented via Uniform Resource Locators (URL), as it was the simplest solution to guarantee the privacy of the users, documentation of study specific data, control over the logging of the contents of messages and an easy distribution of an access possibility that can be distributed through commonly used communication systems like email. The alternative is to implement a complete separate

study environment, that is customized for a specific kind of user study. Another requirement of this objective was the recording of the analysis results, interaction with the profanity warning and Emotion-Chart, as well as the raw content of the messages of the users, when permitted. Furthermore a smooth transition between the conversation and its evaluation via surveys was required as well.

The third objective was to create an application that made use of analysis procedures taken from NLP, in particular the emotion analysis. This in turn demanded an intuitive and handy visualization of the results of each analysis.

With the first objective, the pipeline was to be able to be executed in real-time so that the user wouldn't concentrate on the analysis but on the conversation, rather. The last challenge was to create an application able to deescalate or smooth out social conflicts or discussions in a chat environment. Being the inspiration for Chattitude, this objective called for mechanisms influencing, yet not disturbing an ongoing conversation. Together with the third objective, the goal now was to visualize the information gained from the analyses in such a way so that is would encourage or discourage a certain desired or undesired behavior.

**Conceptual Architecture** To satisfy most of these use-cases, mostly those of the first objective, a web application which would be accessible on a browser with devices of variable display size was implemented. Naturally, this leads to a server-client setup in which the server needs to support messaging, analysis of messages and provision of the website. The tasks of the client, on the other hand, includes providing a GUI, handling various user events, and visualizing the information provided by the server – in particular the results of the message analysis.

**Signature Features** The Signature Features are technical features which, based on our knowledge, make Chattitude unique as a chat application. These features will also serve as the independent variable in the user study of this

thesis. Consisting of three features, the Signature Feature are called the profanity warning, Emotion-Chart, and highlighting. The profanity warning is basically an overlay panel warning the user, about the existence of profane words in the message as seen in Figure 2.

To get rid of this warning the user has to either confirm or abort sending the message, using either mouse or touch-screen. In particular it is not possible to confirm by using the "Enter"-key of the keyboard. The purpose of this feature is to discourage the usage of foul language. Another quality of this feature is a small warning symbol that will be displayed as soon as profanity is detected in the message, while the overlay will only be displayed when the user attempts to send the message.

The second feature is the so-called Emotion-Chart (Figure 3), which represents a simplified visualization of the emotions detected in a given message. The Emotion-Chart consists of two parts, one being the emoji in the center and the "doughnut" at the border. The emoji represents the overall emotional sentiment of the message. The doughnut represents the distribution of emotional words in the text and is connected to the last Signature Feature, the highlighting. In a nutshell, the doughnut shows the number of words associated with a specific emotion. Note that there are six emotions used in the app: `anger, sadness, surprise, joy, disgust, fear` and a non-emotion as a placeholder, used when no emotion is detected. The highlighting marks each word in the message associated with an emotion or with profanity (Figure 3) using color or a bold font-weight.

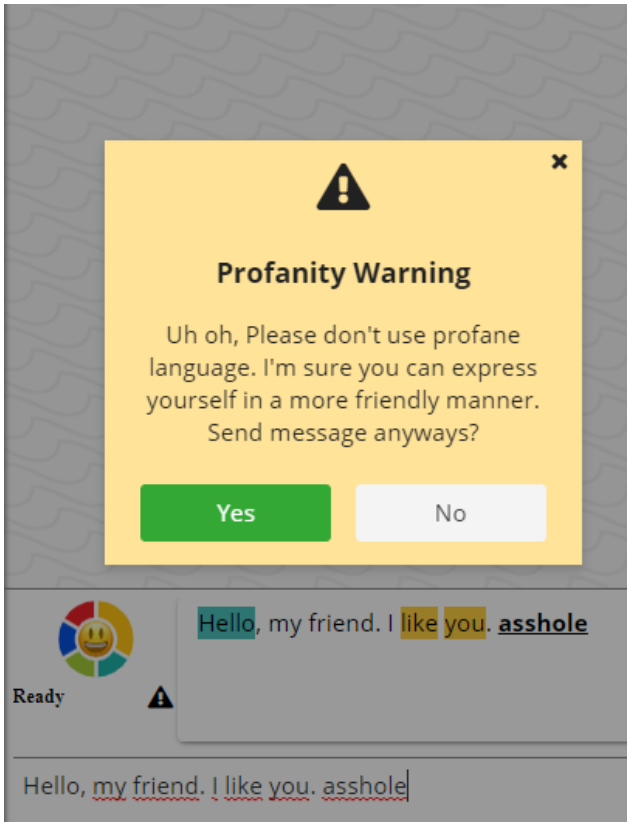



Figure 2: The profanity warning of Chattitude, displayed when attempting to send a message which contains profanity. This warning functions as an overlay meaning that the rest of the website can't be used until the user is either confirming or declining this warning. In the background you can see the Preview-Message which you can take a better look at in Figure 3. Note that if profanity exists the profane word is marked bold and underlined and a small warning indicator is displayed next to the Emotion-Chart.

## 3.2 Architecture

**Back-End:** The final version of Chattitude consists of four servers: One for the simple provision of the website for the user, one for the message exchange and two for the language conscious analysis of the contents of a user message.

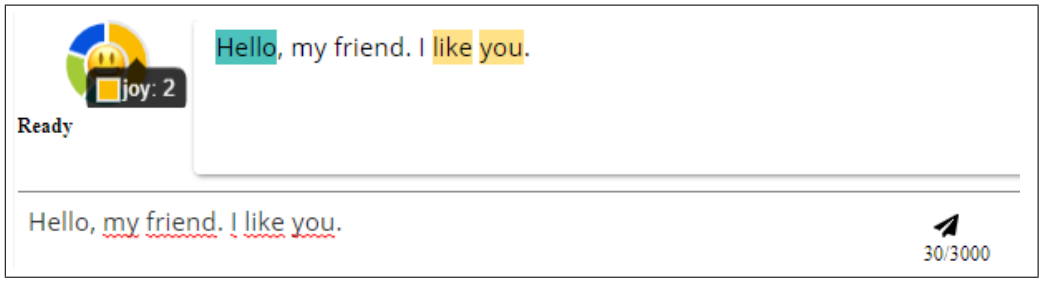



Figure 3: The Preview-Message of Chattitude. It mainly consists of the Emotion-Chart, input-field and the highlighted version of the message. When hovering or clicking on the emotion-chart, a tool-tip appears, which indicates the number of words in a message that are labeled with an particular emotion. In this case "like you" are two words that are labeled with "joy". Both are highlighted in the text with the same color. The "Ready" indicator, provides information about the progress of the real-time analysis and changes to "Analyzing..." if the analysis is still analyzing the message.

The first server doesn't have a lot of difficult tasks, but besides serving the website to the user, it still provides the environment and influences the architecture of the message-exchange server. It is using express.js, a Node JS framework for fast and scalable server applications. Its common use-case is to set up a REpresentational State Transfer (REST)-Application Programming Interface (API) which is used here to answering user requests.

The message exchange server is also written in Node JS, but uses another protocol besides the standard http-requests for handling chat messages, called Socket IO. Socket IO is optimized for applications featuring chat environments or other open connections. On top of this protocol a package Chat-Service is used to implement are the core chat functionalities, like room management and reliable message exchange.

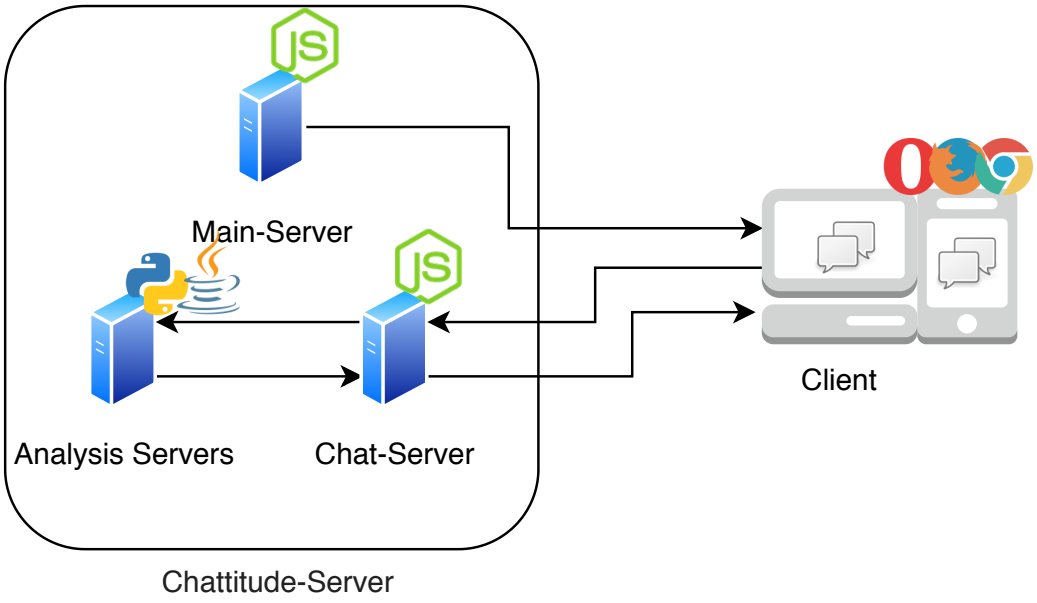



Figure 4: Overview of the top-level architecture of Chattitude. The Chattitude consists of 4 Servers. The Main-Server is the main entrance for the client and provides the context for the other servers. The Chat-Server is tasked with the message exchange and chat-management as well as being part of the analysis pipeline. The Analysis Servers are tasked with the actual linguistic analysis. One of of these analysis-servers is Stanford CoreNLP, which is a free server application, able to do various linguistic annotations. The other is a classifier, embedded into a server framework and used for the emotion and profanity analysis. The Client can be any device where the browsers Chrome, Firefox and Opera can be used. For the most part a normal sized (15-17'diameter) laptop is assumed.

Besides the message exchange, management of the real-time analysis of the messages is a big part of this server. However, the actual analysis of the messages is conducted by the two analysis servers. One is a Stanford CoreNLP server developed by Manning et al. (2014), mainly tasked with the tokenization of the message. Stanford CoreNLP is a very scalable option due to the variety of linguistic processing tools it provides, as well as due

the possibility to deploy multiple servers if needed. Thus the way the way for features like named-entity-recognition, co-referencing and the support for multiple languages is prepared. The second analysis server is tasked with the emotion and profanity analysis of the messages. Unlike Stanford CoreNLP which is implemented in Java, this analysis server is written in Python.

**Message Model:** Due to the amount of components involved in the processing of messages there was a need to define a schema-model written in Javascript Object Notation (JSON) Schema for them. Although this used up some time, it simplified the communication between servers and clients, as the model of the message was always the same. The main idea for the structure of the message is to account for general message information, token-based information and label-based information. The message information consists of everything that is needed for the message exchange (e.g. message ids, sender, receiver, raw content of the message).

The token-based information is primarily the result of the Stanford CoreNLP server. It supports a variety of information and is based on JSON-NLP[11]. In practice it mostly is used for the basic token information, like the text content, offsets, indexes and whether it is a white-space or not. The label-based information consists of the name of the label and all the spans that are associated with this specific label. This simplified example `"anger:{begin:2, end:5}"` would label each token with an index between two and five with anger. The labeling routine on client then can simply go over each span and label them by adding a background-color to each.

## 3.3 Analysis Pipeline

**General** The automatic analysis of messages using NLP is a foundational part of Chattitude. One of the two overarching challenges was to implement

[11] `https://github.com/dcavar/JSON-NLP`(last access: 12.11.2019) developed by Damir Cavar

the pipeline in such a way that it would be compatible with the rest of Chattitude and also remain performant enough to prevent irritations by slow responses. Fulfilling these requirements meant splitting the pipeline into each NLP technique and call each asynchronously. In this way one slow analysis would not slow down the others and the results of each could be presented to the user independently. In this way it is easier to improve one of the techniques.

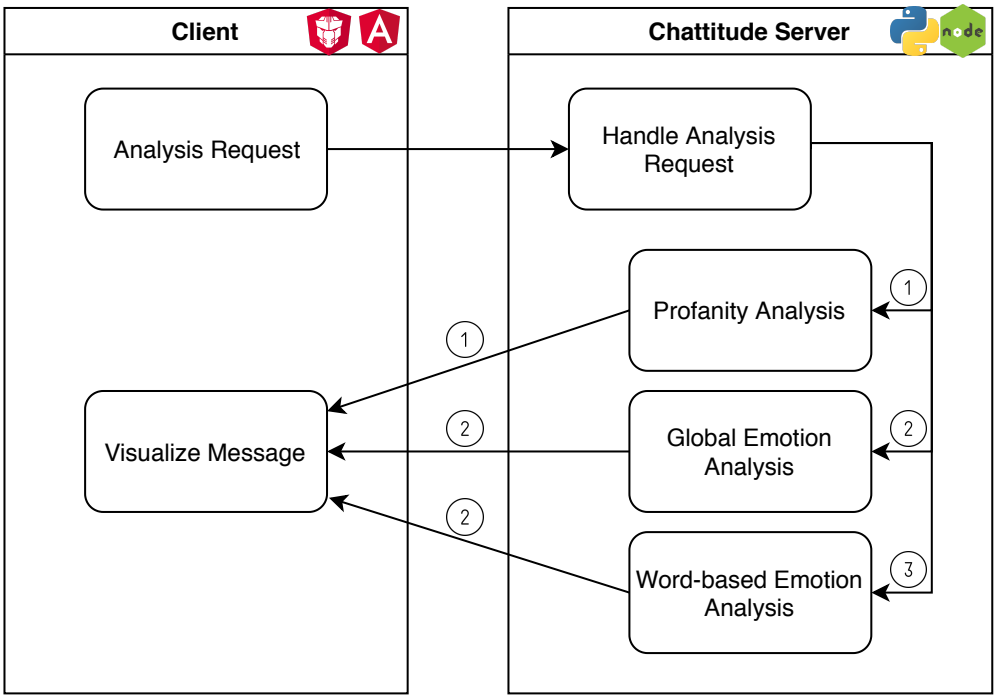



Figure 5: Overview of the analysis-pipeline. After each Analysis Request the Chattitude Server will asynchronously run the profanity- and emotion-analysis. The numbers 1,2 & 3 indicate in which order the Analysis processes are called and usually completed. The symbols on the top right corners of each component indicate which major frameworks were used for the implementation and stand for Prime NG, Angular, Python and Node JS.

The diagram in Figure 5 provides a top-level view on the pipeline. The start of the analysis pipeline is marked by an "Analysis Request" that is fired when the user is editing or sending a message. After the Chattitude

Server receives the request it calls the profanity-, and emotion-analysis asynchronously.

Normally, the analyses are completed in the order described in Figure 5, because of the time each of them takes to finish.

Whenever the last analysis is finished and the user has edited the message in the meantime the whole process is repeated. Consequently, at most one analysis pipeline is available for each user at any given time.

The second challenge of the analysis-pipeline was to trace back the analysis so that the original text could be built and the words that are relevant for the user could be highlighted. The simple approach to this task, was to tokenize the text, then run the analysis on each token separately. The more advanced approach for the future is to create a classifier able to provide details about the effects of a particular part of the input. This would be the more accurate approach to find the words that impacted the final decision of the classifier. This could possibly be achieved by implementing an attention-layer into the architecture of the AI. The primary goal in our context however, was to simply generate an indicator for the user so that the words that might have an impact on the emotion analysis result of the message can be found.

**Preprocessing** Stanford CoreNLP is mostly responsible for the preprocessing. The most important part of the preprocessing is preparing the raw content of the message text into a form that can be analyzed by the emotion- and profanity-analysis. Since both partly analyze every token in the message, the Stanford CoreNLP server is tasked with the tokenization of the message. Since there is no recognizable decline in performance CoreNLP also provides information about the Part of Speech (POS) and lemmatization. The POS is used to prevent bugs from interfering, where brackets are parsed to text snippets like `"-RRB-"`. Using the POS information these snippets were filtered out. The next part of the preprocessing was to translate the results of CoreNLP to the schema of the message and to cache the parts that were

already analyzed by the main analysis. Caching was used to speed up the token-based analysis, as the results for each single token, did not change.

**Profanity Analysis** To detect profanity in messages the python package profanity-check was used on each word. As this kind of analysis is commonly used to censor the contents in social media it is optimized and fast. There is still room for improvement, as people started to invent different notations for profane words so they would not be detected.

The package `profanity-check` used in Chattitude is based on two datasets one being the work of Davidson et al. (2017), the other being the Toxic Comment Classification Challenge from Jigsaw (2017). It is is able to analyze a text fast enough to count as real-time and was developed by Zhou (2019). According to Zhou (2019), profanity-check has a 95% test accuracy with 86% precision and 89% recall, resulting in a 0,88 F1 score. While this is better than its competitors the most important aspect here is the speed of the analysis, which needs only 0.2 ms for each prediction[12]. Speed is crucial to provide the user with a real-time analysis. The alternative packages, namely `profanity-filter` from Inflianskas (2019) and `profanity` developed by Friedland (2013) seem to be less performant. Note that the profanity-analysis is so fast, that there was no need for caching.

**Emotion Analysis** Out of multiple approaches for implementing the emotion-analysis, the one chosen for Chattitude is a classifier trained by Bostan et al. on the aggregated corpus from Bostan and Klinger (2018). It is based on a discrete emotion model and supports the labels `joy, anger, surprise, sadness, disgust and fear`. The advantage of the discrete emotion model is the simple association of an emotion with a color. This enables a simple highlighting system which is easy to grasp, understand and most of all distinguishable by the user. A continuous emotion model would require continuous

[12]Performance test conducted by Zhou (2019) in 2018, using a new 2018 Macbook Pro on the dataset of Jigsaw (2017)

colors which would have reduced the learnability of Chattitude.

Chattitude generally splits the emotion-analysis into two separate processes. One simply takes the whole raw content of the message and makes a prediction on it. From the perspective of the user this process is noticeably slower than the profanity analysis. However, since chat-messages tended to have 29,85 tokens on average the delay that would have been caused by very long messages is almost unrecognizable and thus acceptable.

The other process analyzes each single token of the message. The analysis server will therefore take a list of words as input and return the same list with the results of the analysis for each item in it. This approach is a lot slower, especially with longer lists, which keep users waiting for minutes while typing the message if the message was long enough. To resolve this issue the first naive approach was to introduce an "append" event, in which a new message part is added and analyzed separately. This approach would have come with some other challenges such as edits in the middle of the message or handling of unfinished words.

The final solution for this issue was a Least-Recently-Used (LRU) cache. This cache would save and provide the results for each token based on its usage and is limited to 2000 tokens. This accelerated the analysis enough so that the delay is not recognizable even for people typing fairly fast. The delay only returns if a huge message is pasted instantly into the application which was not the case during the user study. In this case none of the tokens in the message are stored in the cache yet and the classifier has to check the whole list of the tokens from the message.

## 3.4 Front-End

The front-end is the crucial part for users to evaluate and is therefore inevitably focused on, in the user study. In Figure 6 the user's main view of Chattitude is presented with it's main areas, as well as the Signature Feature. The structure is inspired by established chat applications so users had a basic

grasp of the application. Notable parts of the front-end are the menu-bar, a standard part of any website, the conversation list, the message history, and the so-called preview-message. The preview-message is representing the result of the real-time analysis of the message currently edited by the user. A more advanced but also complex approach that can be pursued in the future is to merge the input-field and the preview-message and so can save space. This approach requires a new Hypertext Markup Language (HTML) element, as there is no such element that supports user input and simultaneous highlighting, other visualization operations and handling of user interaction on the text in the input field. In Chattitude this is achieved by wrapping each token with a so-called `<span>` element, which enables all visualization options and the handling of specific user events like hovering or clicking.

Due to the competition of other chat-applications, users have high subconscious expectations towards chat clients. Some of these expectations are the absence of any kind of lag without feedback, an appealing design, simplicity and the same functionalities that other chat clients support. The only potential lag in Chattitude is caused by the analysis pipeline which is why a loading indicator was shown to the user. Anything else did not cause a recognizable delay, however, if Chattitude has to introduce new and slow features, some minor changes might be necessary.

## 3.5 Alternative Setups

Chattitude is a final product that resulted out of tests of different approaches. The initial idea was to simply augment an existing chat-service, in particular one of the three: WhatsApp, Telegram or Slack.

The advantage of extending one of these applications would have been the huge user communities that could have been invited for a user study. Also, these apps have an environment proven to operate very well. In particular, there is less risk of unintended behavior of the chat-application, as well as a known User Experience (UX) and user acceptance. Other advantages are

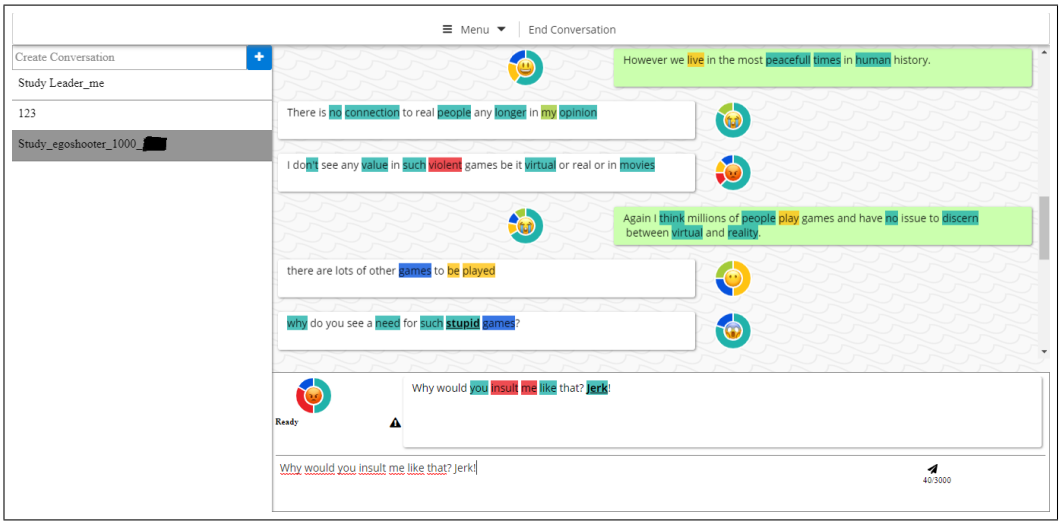



Figure 6: Screenshot of the whole view of Chattitude. On the left you can see the automatic generated conversation chat-rooms. The current active one it marked by the gray color. The menu-bar is positioned on in the top of the picture and is dedicated to the navigation of the whole website and the "End Conversation" button which leads to the Google evaluation survey. The right-hand center of the screenshot represents the history of sent messages, each with an Emotion-Chart and highlighting. At the bottom there is the Preview-Message which is currently edited by the user. Note: This is an abstract of a real study conversation (that's why the conversation name was partly blackened). However the currently edited message was not sent and was created for demonstration purposes.

the potential speed with which the extensions could have been implemented and the easy publication of the prototype. The main reason for not using these applications are the technical limitations on development. WhatsApp, for instance, does not allow any access to messages or to extensions of the software, directly excluding this otherwise best predestined application for a user study in Germany where WhatsApp is the most widely used messaging service according to Priori-Data (2019).

Slack is the application with presumably the least limitations on the API,

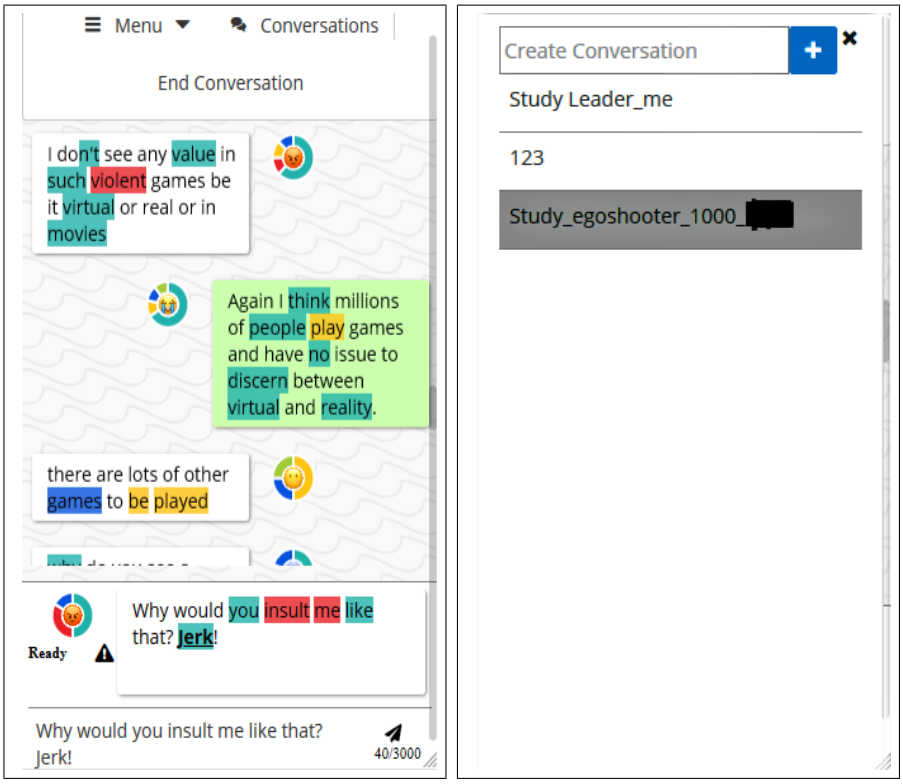



Figure 7: Screenshots of Chattitude when used with devices with small displays. The first view focuses on the display of messages. The button labeled with "Conversations" in the menu-bar enables the user to see and change the conversations. With the "x"-button on the top right, he can go back to the first view.

which is needed to create applications inside such a chat environment. It also is a service used by a community, predominantly in a professional context. Slack furthermore provides an internal store for extensions like voting or calendars. However, these extensions are very limited regarding their options for displaying anything. To bypass theses limitations the idea of a "chat-bot" was investigated. A chat-bot is a software-application that can interpret certain messages and can write messages as well.

### 3.5.1 Chat Protocols

Telegram was another eligible application, that could have been extended. Telegram provides a so-called Telegram Database Library (TDLib) [13], which represents a fully functional Telegram client that can be customized. At this point in development, however, it was decided to use a framework with the least limitations to a custom GUI. The functionality and complexity of the API also needed to converge with the abilities of the developer.

The chosen framework therefore was Angular, a web-development framework, based on TypeScript. Alternative frameworks of this kind could have been Vue.js and React.js, with many similar capabilities, limitations, and use-cases. The overhead, cause by the testing and use of TDLib in the setting of an Angular or similar application, compared with the utilization of alternative messaging protocols was too high. These alternative protocols were Internet Relay Chat (IRC), Comet Chat [14] and Chat-Service. After some research, tests, and comparisons the conclusion was that IRC was excluded because it only provides very old and unsupported libraries, which increases the risk of incompatibilities, security issues or other limitations. Comet Chat was excluded because it required a financial subscription and stored the message on servers outside the developer's direct control. This could have lead to legal and privacy problems in our context, although this option would otherwise have been the best technical solution.

# 4 User Study

**Introduction & General Idea** Since the aim of Chattitude is to improve the conflict resolution in chat environments the underlying question of this user study is whether Chattitude is in fact serving this purpose. The question specifically is whether Chattitude is able to help deescalate discussions by

[13] https://core.telegram.org/tdlib(last access: 03.11.2019)

[14] https://www.cometchat.com/ (last access: 05.11.2019)

conditioning users to use less profane language and by improving the mutual understanding of the emotional state of the chat partner. Specifically, are the Signature Features of Chattitude able to reduce the amount of negative emotions users express like anger or sadness. Since the usability of Chattitude could have a big effect on the measurements the secondary purpose of this user study is to determine the usability by using a standardized questionnaire. Moreover we ask here whether using the Signature Feature is improving the perceived usability and whether it increases the perceived workload on the user. These questions results in the following underlying hypotheses:

**Hypothesis 1 (H1):** *The use of the Signature Feature of Chattitude is reducing the number of messages labeled with anger, sadness and fear in discussions.*

**Hypothesis 2 (H2):** *Using the Signature Feature of Chattitude is either improving it's usability or has no significant effect.*

**Hypothesis 3 (H3):** *The use of the Signature Features is either reducing it's perceived workload or has no significant impact on it.*

Due to the nature of the statistical evaluation process, these hypotheses will be split into multiple smaller ones.

## 4.1 Study Design

**Overview** The user study was conducted entirely online by having discussions over Chattitude, since this is the natural way to use a chat application such as Chattitude and potentially yields more participants. Each participant was discussing a topic listed in Table 11 with a single interlocutor.

The study was following the repeated measures pattern with mainly two conditions: In Condition One (C1) both participants were using Chattitude with all its Signature Feature, namely the Emotion-Chart, highlighting and

the profanity warning. The second Condition (Condition Two (C2)) was Chattitude without any of the previously named features. In this way, the number of required participants can be reduced compared to an independent measures design and it can decrease the effect of novelty on users.

With this design, the most appropriate statistical evaluation technique is utilization of t-tests with dependent means as both sample-means depend on the same group of people. With the results of the t-tests, the difference between using the Signature Features and discarding them can be measured with regard to usability, perceived workload and the "conversation quality". Later in the execution, the study-leader had to substitute for one participant, since there was only one case where both participants showed up.

To minimize the bias, the study-leader's goal was to keep the discussions running as long as possible. Consequently, most studies were terminated by the participant with a few cases in which the study-leader was able to convince the participant of the opposite position. Note that the study-leader substituted for missing participants, so it happened quite often that the study-leader was arguing against his personally prefered stance in the discussion.

In case, the participant knew the study-leader, the study-leader attempted to play the role of the other user in such a way that the real participant would not think that he or she were actually conversing with the study-leader. This was possible due to the missing of author names above each message. In this way the participants didn't know with whom actually they were discussing.

**Study Variables** In this study multiple variables were measured by either surveys or the logging of specific information during the discussions. The first category was dedicated to the difficulty of solving a conflict or heated discussion and was split into multiple aspects, mostly measured by a custom survey (Table 1) introduced in section 4.2 "Measurement techniques & scales". The variables the survey is supposed to measure are enjoyment,

mutual understanding, likeliness to resolve conflicts, emotionality, intellectuality and difficulty of the conversation, as well as the perceived obligation to enforce one's position or opinion and the usage of foul-language.

The feeling of obligation in particular is an important factor for the difficulty of a discussion and might insert fuel into the conflict. The opposite of this factor is the enjoyment felt during an intellectual discussion, which can make the discussion less serious and thus easier to resolve.

In addition to the survey the results of the classifier for each message were logged. After the study the number of messages for each emotion were counted and presented in a Histogram. The number of messages containing profanity and insults is also measured in a custom survey and by logging. The logged information discloses whether a message contains profanity and the interaction with the profanity warning. In particular, it shows the number of edits per message due to the profanity warning.

The second category was dedicated to the UX of Chattitude. Two major factors were measured, namely the perceived usability and workload, using two standardized questionnaires.

The main independent variable for this study was the use of the Signature Features, whose activation was reflected in C1 and C2. The secondary variable was the level of disagreement, which initially maximized by pairing up participants according to viewpoints opposite each other. In the execution the study-leader took over the role of the opponent, as the meeting of the participants failed.

By way of the repeated measures design of the study and a short "Warm-up"-phase, in which the participant had time to get used to Chattitude, the effect of novelty was partly accounted for. In order to make sure that novelty is not influencing the results a long-term study is necessary. Through the invitation survey demographic information (age, gender) was accounted for.

Another factor that can only be excluded by randomization is the order of the used discussion topics. Since this experiment was already randomizing

the order of the features displayed, randomizing the order of the discussion topics was creating two new conditions, which would doubled the required number of participants. Therefore in this experiment only the display of the Signature Features was randomized.

## 4.2 Measurement Techniques And Scales

**Custom Survey – Invitation** For the user study multiple surveys were created because no standardized surveys that measure the needed information were found. Some of the custom surveys don't serve not the purpose of empirical measurement like the survey for selecting and pairing participants. This survey mainly gathers demographic data but also the opinion of the participants about certain controversial topics, as seen in Table 11. The idea was to find two topics with maximum disagreement among the participants. In this way each participant was paired with another participant of opposite opinion was supposed to lead to rather natural discussions.

In Table 1 the topics, the participants needed to state their opinion on, are listed. Note that the format was to score the agreement to a statement on a scale from 1 to 7 (7 = "Strongly Agree) with the option to skip the topic. This option was needed when the participant had a reason to not take part in the specific discussion. Possible reasons were lack of knowledge or high personal sensitivity among others.

**Evaluating the Conversations** The setup of this user study only provides information about the conversation participants were having not their relationships[15]. Multiple standardized surveys were developed for measuring the quality of relationships but they presuppose either enough time or an existing relationship which are both not given in this user study. Participants were paired and only have two conversations with a stranger which makes the use of relationship quality surveys questionable. As a result, a survey

[15]Relationship in general, not the romantic sense.

consisting of self-developed items was created (Table 1) to measure certain aspects of a discussion and conversation.

| Q | Statement | Adjacency |
|---|---|---|
| 1 | I consider the Conflict/Discussion to be solved: (You and your chat partner agree on a position/solution): | none |
| 2 | I enjoyed the discussion: | none |
| 3 | I think, my chat partner understood me: | to 4 |
| 4 | I think, I understood my chat-partner: | to 3 |
| 5 | I became emotional during the discussion: | to 6 |
| 6 | My chat partner became emotional during the discussion: | to 5 |
| 7 | My arguments were factual/intellectual: | to 8 |
| 8 | The arguments of my chat-partner were factually/intellectual: | to 7 |
| 9 | The discussion felt difficult: | none |
| 10 | I felt insulted often: | none |
| 11 | I felt obliged to enforce my position/opinion: | none |

Table 1: Content of the custom survey used to evaluate the discussions in the user study of this thesis. Items are scored from 1 to 7 (7 = "Strongly Agree") The adjacency in this case specifies whether the items are related to each other. Statements 3 & 4, for example are the same statements from a different point of view, which are each compared with the results of the counterpart scored by their chat partner.

Each item in this survey is scored by the participant in each condition on a Likert scale from 1 to 7 (7 = "Strongly Agree") and is compared afterwards. For the statistical comparison, a t-test of dependent means is used for each item, as results come from the same participants in each condition. An additional evaluation was planned for the items that are connected to another one. In particular items 3, 4, 5, 6, 7, 8 of Table 1 are affected by this connection, meaning that they have a counterpart, which should be compared with

the scores of the chat partner in the same condition, by calculating the difference. So, if a participant (A) in C1 scored "3" for item 3 ("I think my chat partner understood me:") and the this participants chat partner (B) of this participant scored "1" for item 4 ("I think, I understood my chat partner:"), the difference would be 2. These calculated differences should be sampled in each condition and compared afterwards. In this way, the perspective of both parties of the discussion can be compared, as there could be big discrepancies between them. The assumption here is that these discrepancies indicate difficulties in communication which lead to misunderstandings or conflicting views.

**System Usability Scale** The Sytem Usability Scale (SUS) is a Likert scale based survey developed by Brooke et al. (1996) in which users of a system rate its UX. The SUS is one of many established surveys whose objective is to measure the UX of an application that focuses on the interaction with people. The Table 9 has the content of this particular survey. Note that the users will score each item on a scale from 1 to 5, while 1 refers to "Strongly Disagree" and 5 to "Strongly Agree". This survey was used because of its simplicity and the little difference to other questionnaires with the same objective. The main reason for measuring the usability of Chattitude is to ensure that the quality of Chattitude is not negatively interfering with the results of the study and to find out if the augmentation of a message has any significant impact on the UX of Chattitude. The goal, according to H2 is that the Signature Features either have no or even a positive impact on Chattitude. Furthermore, the result will also be compared with the results of Kaya et al. (2019) in which the SUS scores of established applications like WhatsApp, Facebook and YouTube are compared on various Operating Sytems (OS)

**NASA Task Load Index** The purpose of the NASA Task Load Index (NASA TLX) developed by Hart and Staveland (1988) is to measure the

"perceived workload" of a given task. Users score each item of this survey shown in Table 10 from 0 ("low") to 20("high"). In the user study this survey is used to ascertain the difference of this perceived workload and some of its sub-scales between C1 and C2. In particular mental demand, frustration, and effort is of interest in this study, as in a discussion on Chattitude physical or temporal demand play no significant role. Since the task in this user study, was to only have a discussion over Chattitude, there is no intuitive source with which the "performance" can be estimated, making the results a bit questionable yet interesting in the case of big differences.

**Final Survey** The final survey consists of two parts, one dedicated to rating the application using the commonly used star rating system. The star rating system is a simple Likert scale from 1 to 5, with 5 being a good rating. The survey particularly asks the participant to rate each Signature Feature and Chattitude in general.

The second part of the survey contains three questions for qualitative feedback. The first question ask about the positive aspects of Chattitude and the second about possible improvements The last question provides the opportunity to mention anything else which might not have been covered by any of the preceding surveys.

## 4.3 Execution

The overall execution of the study consisted of five phases: `Selection`, `Warm-up`, `Online Discussions`, and a `final Survey`. In the selection phase participants were invited by distributing a prepared email in various email lists for students of the University of Stuttgart. Besides the generic invitation this email featured a link to the previously mentioned invitation survey and one for an online calendar tool[16], for organizing the meetings of the participants. After about ten participants joined the study the invitation survey

[16]Doodle: `https://www.doodle.com`(last access: 08.11.2019)

was evaluated and two topics for discussions with maximal disagreement, were selected. At this point the most promising topics were "Ego-Shooters" and "Smoking", followed by "GenderSTEM" and "E-Mobility". The decision was based on four factors:

1. The number of participants who didn't have an opinion or position on the given topic was minimal or zero. The reason behind this requirement was that participants with no opinion could not be used in the study.

2. The number of participants of weighted opposite opinion was balanced. To weigh the position, the answers on the likert scale were multiplied with 2, 1 and 0,5. Answer 4 for example which represented a balanced opinion with almost no conflict potential was excluded. In the case of the topic of Ego-Shooters 12 people generally tended to disagree with a registration requirement for Ego-shooters(Table 2). The weighted disagreement, however, was 17.5 because 6 participants scored 1, 5 scored 2 and 1 scored 3 on the Likert scale of 1-7 (1 = Strongly Disagree), resulting in this formula: $6 * 2 + 5 * 1 + 1 * 0,5 = 17.5$. The calculation for the weighted agreement towards the statement was similar to the mentioned formula.

3. The number of possible pairs was maximal. A pair were two participants that generally tended toward opposite positions or opinions on a topic.

Generally the weighted difference was the more preferred factor, as maximizing this factor results in more intense discussions. In Table 2 the calculated factors of the four most promising topics are presented.

| | Ego-Shooters | Smoking | E-Mobility | Gender-Gap |
|---|---|---|---|---|
| Sum Disagree | 6 (12) | 5 (6) | 3 (9) | 5 (12) |
| Sum Agree | 4 (8) | 3 (10) | 6 (11) | 4 (6) |
| Possible Pairs | 4 (8) | 3 (6) | 3 (9) | 4 (6) |
| Weighted Disagree | 8.5(17.5) | 6.5 (8.5) | 4.5(13) | 4,5 (13) |
| Weighted Agree | 7 (12) | 6 (14,5) | 7 (13) | 5.5(9.5) |
| Balance Weighted | 1.5 (5.5) | 0.5 (-6) | -2.5 (0) | -1 (3.5) |

Table 2: Overview over the factors that influence the decision for the used topic in the study after 10 registered participants. The number in the brackets represents the result at the end of the study with 21 participants. (Sum (Dis)Agree: Number of participants (dis)agreeing with the statement of the given topic; Possible Pairs: Minimum of Sum Agree and Disagree; Weighted (Dis)Agree: Answered are added together but each answer on the Likert scale was weighted with 2, 1 and 0.5; Balance Weighted: Difference of Weighted (Dis)Agree)

After more participants joined the topic of Gender Gap in STEM fields at universities, yielded a higher disagreement, however, because some participant already completed the study, the topics chosen initially couldn't be changed.

After that, all participant pairs with compatible dates and high disagreement levels were identified and notified by a second prepared email with detailed instructions and access URLs to the chat rooms. The first URL leads users to the Warm-up chat room, where they were able to accustom themselves with Chattitude commencing the Warm-up phase. The Warm-up phase also featured a little icebreaker game, so participants had something to chat about. After the Warm-up, the main experiment started, with either C1 or C2. Each condition consisted of a discussion and an evaluation survey with SUS, NASA TLX and the custom survey for the evaluation of the conversations. After going through both conditions on random order, the study

concluded with the final Survey previously described in section 4.2.

## 4.4 Evaluation & Results

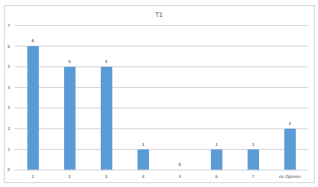
The general approach for evaluating the data from the user study was to group them for each condition and compare both conditions based on various metrics. In the case of the SUS, NASA TLX and the custom conversation survey a t-test was used. The t-test was based on the following hypotheses:

1. H0 = $\mu_{C1}$ - $\mu_{C2}$ = 0
2. H1 = $\mu_{C1}$ - $\mu_{C2} \neq 0$
3. H2 = $\mu_{C1} > \mu_{C2}$
4. H3 = $\mu_{C1} < \mu_{C2}$

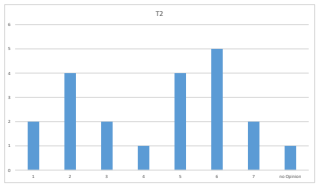
While H0 and H1 where always tested, H2 and H3 where chosen based on the means of the results. The logging information, however, did not provide data supported by the t-test, so only the means, standard deviations, as well as differences were reported.

In total 13 out of 21 registered participants fully took part in the study. One of the 13 did not fill out the final survey though. Most of the participants were selected through an invitation mail in email distribution lists of the University of Stuttgart. Consequently, the majority of participants were students of the computer science faculty. The total group of 21 registered participants were 27.29 years old on average and were 71.43% male and 28.57% female. They also estimated their English proficiency 5.76/7 on average. In Figure 8 their distributed opinion each topic noted in Table 1 is presented. Note that the last pillar in each diagram is representing the number of participants with no opinion on that topic. The higher this number, the more participant cannot take part in the study, rendering the topic less useful.

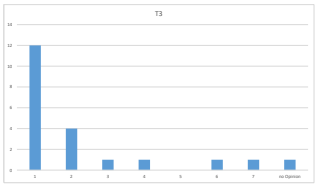
The 13 participants that fully executed the study, had similar values: 69.23% male and 30.77% female and an average English proficiency of 5.85/7.



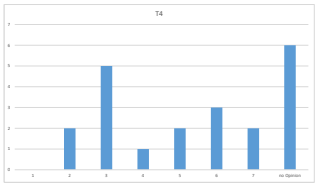
(a) T1 - DemoPupils



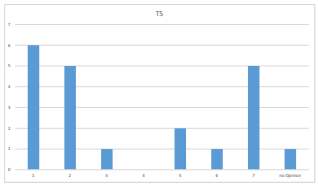
(b) T2 - Diving



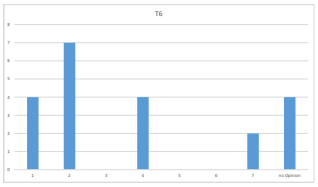
(c) T3 - VoteParent



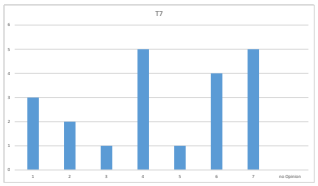
(d) T4 - GeneEditing



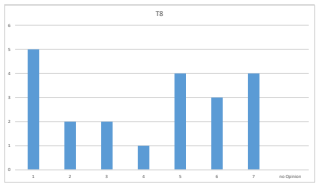
(e) T5 - EgoShooter



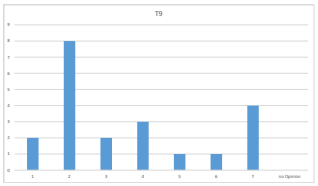
(f) T6 - Embryos



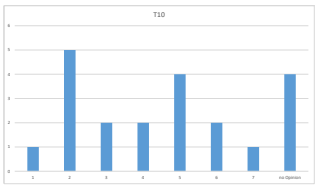
(g) T7 - Smoking



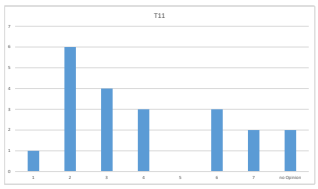
(h) T8 - E-Mobility



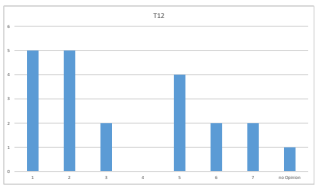
(i) T9 - GenderSTEM



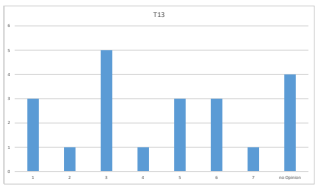
(j) T10 - RentLimit



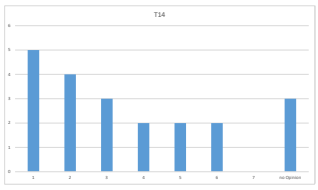
(k) T11 - Military



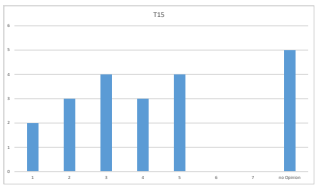
(l) T12 - Dresscode



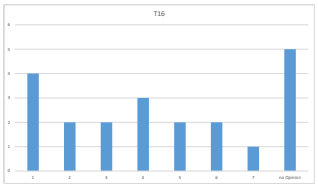
(m) T13 - AfricaAid



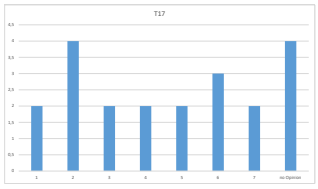
(n) T14 - ClimatTech



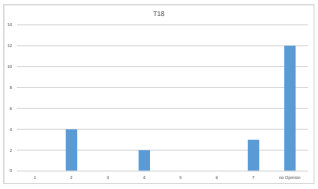
(o) T15 - China



(p) T16 - PoliceDrones



(q) T17 - Soccer



(r) T18 - Brexit

Figure 8: Opinion distributions of the participants. x-Axis: Likert-scale 1-7 + no opinion, y-Axis: Number of votes.

Over the course of the study, these graphics changed substantially, so that smoking went from a well polarized and balanced to an imbalanced topic. The selection of balanced and polarized topics is dependent of the selection of the participants and is therefore difficult. As stated in the execution the preliminary result with 10 participants resulted in the selection of the topic "EgoShooter" and "Smoking".

### 4.4.1 Evaluation Of Conversations

The evaluation of a conversation comprises two individual assessments, one being the evaluation of the surveys and the other the evaluation of the logged data, reported by Chattitude. The evaluation of the surveys follows the pattern of the t-test previously mentioned.

**Custom Conversation Evaluation** The custom survey (Table 1) yielded results, presented in Table 3 that are slightly in favor of C2, where the Signature Features were deactivated. For the items 2,5,6,7,9,10 and 11 there was no significant difference between using C1 and C2. That means that the Signature Features make no difference in enjoyment, emotionality, difficulty, or the feeling of being insulted and obligated to enforce their own position. However, when switching off the Signature Features users indicated a higher conflict resolution, mutual understanding and they perceived the messages of their chat-partner to be more factual or intellectual.

**SUS** The results of the SUS were evaluated in the same way as the results of the custom conversation survey. A t-test was applied to each item of the SUS and resulted in no significant difference between C1 and C2. Consequently, the overall SUS-score also didn't yield any significant difference upon investigation. These overall scores where calculated based on the procedure described by (Kaya et al., 2019; p. 393), where Chattitude achieved an overall score of 76.25 of 100 for 13 participants.

| Q | $\mu_{C1}$ | $\mu_{C2}$ | Diff | $s_{C1}$ | $s_{C2}$ | $s_{C2}$ | t | p1 | p2 | H |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 3.23 | 5.15 | 1.92 | 1.83 | 2.19 | 2.02 | 3.43 | 0.0025 | 0.0049 | H1 & H3 |
| 2 | 5.54 | 5.62 | 0.08 | 1.33 | 0.77 | 1.61 | 0.17 | 0.4329 | 0.8657 | H0 |
| 3 | 4.38 | 5.54 | 1.15 | 1.19 | 1.56 | 2.15 | 1.93 | 0.0387 | 0.0774 | H0 & H3 |
| 4 | 4.85 | 5.77 | 0.92 | 1.21 | 0.93 | 1.50 | 2.22 | 0.0231 | 0.0463 | H1 & H3 |
| 5 | 3.85 | 2.85 | -1.00 | 1.99 | 1.63 | 2.61 | -1.38 | 0.0965 | 0.1930 | H0 |
| 6 | 3.85 | 2.85 | -1.00 | 1.63 | 1.68 | 2.55 | -1.41 | 0.0914 | 0.1827 | H0 |
| 7 | 4.38 | 4.77 | 0.38 | 1.66 | 1.48 | 2.40 | 0.58 | 0.2870 | 0.5740 | H0 |
| 8 | 3.54 | 4.92 | 1.38 | 1.45 | 1.50 | 2.43 | 2.05 | 0.0314 | 0.0627 | H1 & H3 |
| 9 | 2.69 | 2.46 | -0.23 | 1.93 | 1.39 | 1.92 | -0.43 | 0.3363 | 0.6727 | H0 |
| 10 | 2.15 | 1.85 | -0.31 | 1.72 | 1.77 | 2.56 | -0.43 | 0.3363 | 0.6727 | H0 |
| 11 | 4.46 | 3.77 | -0.69 | 1.45 | 1.79 | 2.39 | -1.04 | 0.1588 | 0.3176 | H0 |

Table 3: Results of the evaluation of the custom questionnaire investigating a conversation. The first three columns represent the means of the results of question Q (view Table 1) and their difference for each condition. The next three columns represent the standard deviation(s) of the results and differences. The next three columns present the results of the t-test, while t is the t-value and p1 is the p-value for a one-tailed test and p2 the p-value for a two-tailed test, with a level of significance of 0,05. The last column specifies the accepted statistical hypothesis. There were four possible hypotheses that were chosen based on resulting means: H0 = $\mu_{C1}$ - $\mu_{C2}$ = 0, H1 =$\mu_{C1}$ - $\mu_{C2}$ $\neq$ 0, H2 = $\mu_{C1}$ >$\mu_{C2}$, H3 = $\mu_{C1}$ <$\mu_{C2}$

Kaya et al. (2019) analyzed commonly used social media platforms like WhatsApp, YouTube, and Facebook. Furthermore, they have compared them on different smartphone operating systems like Android and IOS using the SUS. On average these platforms achieved scores of 80.63 in the eyes of 222 Turkish participants. Bangor et al. (2009) investigated the correlation between different ways of interpretation and rating with the SUS. Following Figure 4 on page 121 of Bangor et al. (2009), Chattitude could also be rated "good" or with the school grade of "C" and is thought to be acceptable by

| Q | $\mu_{C1}$ | $\mu_{C2}$ | $Diff_{\mu}$ | $s_{C1}$ | $s_{C2}$ | $s_{Diff\mu}$ | t | p1 | p2 | H |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2.00 | 2.23 | 0.23 | 1.00 | 1.01 | 1.42 | 0.585 | 0.285 | 0.570 | H0 |
| 2 | 3.15 | 3.62 | 0.46 | 1.07 | 0.87 | 1.61 | 1.032 | 0.161 | 0.323 | H0 |
| 3 | 3.15 | 3.31 | 0.15 | 0.90 | 0.85 | 1.52 | 0.365 | 0.361 | 0.721 | H0 |
| 4 | 3.85 | 3.92 | 0.08 | 0.55 | 0.28 | 0.64 | 0.433 | 0.336 | 0.673 | H0 |
| 5 | 2.23 | 2.08 | -0.15 | 1.17 | 1.19 | 1.52 | -0.365 | 0.361 | 0.721 | H0 |
| 6 | 3.08 | 2.69 | -0.38 | 0.86 | 1.25 | 1.50 | -0.923 | 0.187 | 0.374 | H0 |
| 7 | 3.54 | 3.46 | -0.08 | 0.52 | 0.52 | 0.76 | -0.365 | 0.361 | 0.721 | H0 |
| 8 | 2.69 | 3.15 | 0.46 | 1.03 | 0.80 | 1.13 | 1.477 | 0.083 | 0.165 | H0 |
| 9 | 2.92 | 2.69 | -0.23 | 0.95 | 1.11 | 1.36 | -0.610 | 0.277 | 0.553 | H0 |
| 10 | 3.46 | 3.77 | 0.31 | 1.13 | 0.60 | 1.38 | 0.805 | 0.218 | 0.436 | H0 |
| Tot. | 75.19 | 77.31 | 2.12 | 10.08 | 10.78 | 16.16 | 0.472 | 0.323 | 0.646 | H0 |

Table 4: Results of the evaluation of the SUS questionnaire, comparing C1 with C2 for each question and the overall SUS-score. The first ten rows represent the 10 questions of the SUS, the last the total SUS-score. The first three columns represent the means and their differences for the results of all participants. The second group of three, present the standard-deviations of the means and differences of the means. The last group of three show the results of the t-test while t is the t-value and p1 & p2 are the p-values of the test-statistic. Hereby p1 represents the p-value calculated from a one-tailed test and p2 from a two-tailed test. The H column presents the accepted statistical hypotheses.

the majority of users.

**NASA TLX** The NASA TLX was evaluated with almost the exact same procedure as the SUS with the only difference being the calculation of "NASA TLX-score". In this case the total value was simply an aggregation of the results of each item of the survey. The results of this evaluation, indicate no significant difference between C1 and C2 overall and for each item. Consequently, users of Chattitude will perceive no higher workload in all respects

when using the Signature Feature of Chattitude. As such, the perceived workload will have no confounding influence on the results of the study.

| Q | $\mu_{C1}$ | $\mu_{C2}$ | Diff | $s_{C1}$ | $s_{C2}$ | $s_{C2}$ -$\mu_{C2}$ | t | p1 | p2 | H |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 5.77 | 6.62 | 0.85 | 3.94 | 5.94 | 8.05 | 0.38 | 0.36 | 0.71 | H0 |
| 2 | 2.31 | 2.00 | -0.31 | 2.32 | 2.16 | 3.04 | -0.37 | 0.36 | 0.72 | H0 |
| 3 | 4.69 | 5.62 | 0.92 | 5.02 | 5.41 | 7.95 | 0.42 | 0.34 | 0.68 | H0 |
| 4 | 8.00 | 7.62 | -0.38 | 4.58 | 4.19 | 4.29 | -0.32 | 0.38 | 0.75 | H0 |
| 5 | 4.92 | 5.46 | 0.54 | 4.35 | 4.96 | 7.25 | 0.27 | 0.40 | 0.79 | H0 |
| 6 | 3.62 | 3.00 | -0.62 | 3.40 | 2.68 | 4.59 | -0.48 | 0.32 | 0.64 | H0 |
| Tot. | 29.31 | 30.31 | 1.00 | 20.40 | 20.99 | 30.36 | 0.12 | 0.45 | 0.91 | H0 |

Table 5: Results of the evaluation of the NASA TLX questionnaire, comparing C1 with C2 for each question and the total aggregated value. The first six rows represent the six items of the NASA TLX, the last the total aggregated perceived workload value. The first three columns represent the means and their differences for the results of all participants. The second group of three, present the standard-deviations of the means and differences of the means. The last group of three show the results of the t-test, while t is the t-value and p1 & p2 are the p-values of the test-statistic. Hereby p1 represents the p-value calculated from a one-tailed test and p2 from a two-tailed test. The H column presents the accepted statistical hypotheses.

**Evaluation of the Logs** During the conversations between the participants Chattitude logged various kinds of data. In Table 6 the top-level readings of the logged data is presented. In total the conditions have very balanced number of messages and message lengths. In total 476 messages were sent by the participants. Note that the logs of the study leader were excluded, as these are not evaluated due to the bias they would have on the results. The study leader wrote a total of 768 messages(Warm-up included), with an average length of 33.35 tokens, 9 of them being profane. This results in a total of 1244 being sent during the experiment.

| | N | $\mu_N$ | $\mu_{\|msg\|}$ | min.$\mu_{\|msg\|}$ | max.$\mu_{\|msg\|}$ | $N_{pro}$ | $\mu_{Npro}$ |
|---|---|---|---|---|---|---|---|
| C1 | 235 | 28.42 | 28.73 | 2.08 | 78.58 | 4 | 0.33 |
| C2 | 241 | 26.00 | 27.47 | 2.50 | 89.33 | 2 | 0.17 |

Table 6: Overall results from the evaluation of the logging information gathered from the conversations by participants in the user study. N = Total number of messages sent in each condition, $\mu_N$ = average number of messages sent in each conversation, $\mu_{|msg|}$ = average length of each message, min/max $\mu_{|msg|}$ = minimal/maximal average length of each message, $N_{pro}$ = Number of profane messages, $\mu_{Npro.}$ = average number of profane messages in a conversation

One specifically important kind of logged information are the results the emotion-classifier reported for each whole message. These log-entries contained data which identified message, the top-level result and the probability of each emotion being the correct label for the message. To enable a comparison of this data for both conditions, the messages of each participant were grouped. In particular, the total number of messages with a specific label was counted as well as their averages. For most of the emotion-labels the numbers were fairly similar except for "joy" and "sadness". In each conversation in C2 1.50 more messages were labeled with "joy" on average and 1.08 less were labeled with "sadness".

| Label | $N_{C1}$ | $N_{N,C2}$ | Diff | $\mu_{N,C1}$ | $\mu_{N,C2}$ | Diff | $\mu_{P,C1}$ | $\mu_{P,C2}$ | Diff |
|---|---|---|---|---|---|---|---|---|---|
| anger | 30 | 27 | 3 | 2.50 | 2.25 | 0.25 | 0.40 | 0.48 | -0.084 |
| fear | 79 | 85 | -6 | 6.58 | 7.08 | -0.50 | 0.32 | 0.34 | -0.025 |
| surprise | 16 | 11 | 5 | 1.33 | 0.92 | 0.42 | 0.38 | 0.33 | 0.059 |
| joy | 12 | 30 | -18 | 1.00 | 2.50 | -1.50 | 0.41 | 0.48 | -0.070 |
| sadness | 35 | 22 | 13 | 2.92 | 1.83 | 1.08 | 0.37 | 0.31 | 0.058 |
| disgust | 3 | 2 | 1 | 0.25 | 0.17 | 0.08 | 0.33 | 0.23 | 0.098 |
| none | 60 | 64 | -4 | 5.00 | 5.33 | -0.33 | 0.51 | 0.52 | -0.008 |

Table 7: Overview over the results of the emotion-classifier for each participant aggregated of each label. The first group of three show the absolute number of messages that where labeled with particular emotion. The second group of three show the average number of messages that where labeled with a specific emotion for each conversation grouped by condition. The last group of three show the average probability for a particular label the classifier reported. The last column shows the difference of the reported probabilities of C1 and C2.

With a total of 15 profane of all 1244 sent messages, the participants were too friendly and respectful for the profanity warning to have any effect. Future studies, therefore, need to choose more polarizing topics and select more aggressive participants.

### 4.4.2 Evaluation Of The Final Survey

**Star Ratings** In the final survey the user was asked to rate Chattitude and its Signature Features based on the commonly used five star rating system, which is a simple Likert Scale from 1 to 5 (1 = 1 Star $\widehat{=}$ bad, 5 = 5 Stars $\widehat{=}$ good). In Figure 9 the distribution of these user ratings is presented.

The average of the star rating for Chattitude totals to 3.5 stars with a standard deviation of 0.67. The highest-rated feature on average is the

Emotion-Chart with 3.91 stars and a standard deviation of 1.24, followed by the profanity warning with 3.83 stars and a standard deviation of 1.11. The least popular feature is message-representation, with 3 stars on average and a standard deviation of 1.28. The explanation for the lower scores can be found in the qualitative feedback, where it was mentioned that the highlighting tends to distract the user.

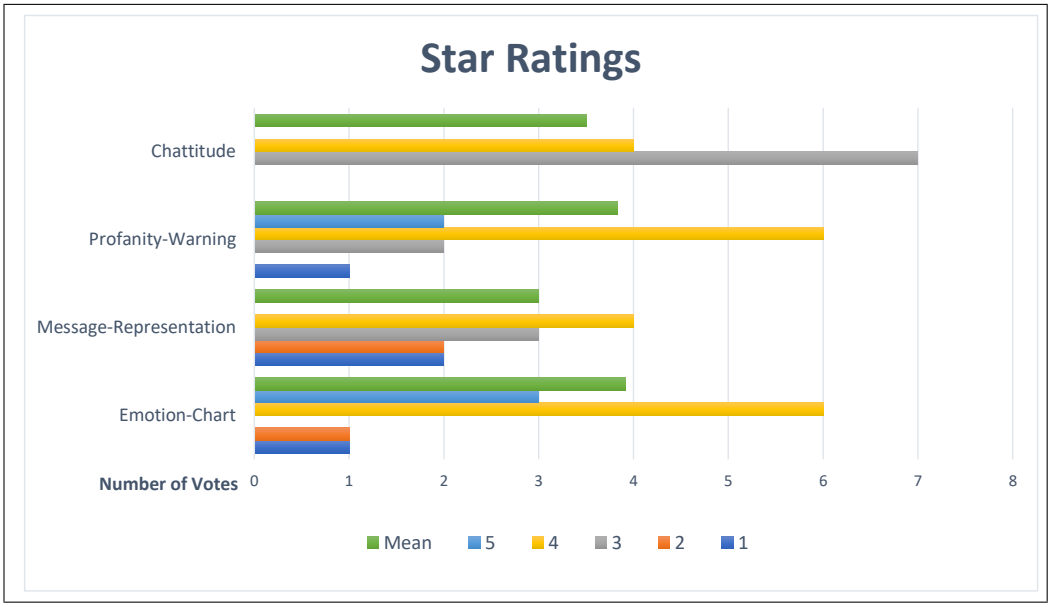



Figure 9: Visualization of the user ratings of Chattitude and it Signature Features. In particular the profanity warning, Emotion-Chart and combination of message, Emotion-Chart and highlighting

**Qualitative Feedback** The secondary purpose of the final survey was to give participants the possibility to give qualitative feedback. Three questions were asked in this context concerning the good and improvable attributes of Chattitude and a final question for free feedback. The results for the question of the users liked about Chattitude are shown in Table 8. For the question of what could be improved, many users mentioned that the highlighting was quite distracting. Furthermore they wished for a typing indicator and more control over the visualizations.

The question for general feedback did not yield any information of importance. In summary users liked the idea and concept of Chattitude, and liked to see improvements for the highlighting, accuracy of the classifier and some comfort features, like a typing indicator for the chat-partner.

| | What do you think is good about Chattitude? |
|---|---|
| 1 | I like the idea of gathering people in a healthy discussion environment. |
| 2 | That it makes emotions visible and helps evaluate statements. Cool! |
| 3 | the feedback for the written messages on both sides |
| 4 | The model performed consistently. Given a different data set and biases it should be able to more accurately determine sentiments. |
| 5 | Seeing the emotions visually is pretty great |
| 6 | It's nice to see how emotional my message is. If one gets too emotional during a debate/conversation I believe it is a good thing to be able to take a step back and visually point out the different emotions the message expresses. To keep control of what one says and always stay respectful. |
| 7 | Analysing the emotions and give directly feedback. |
| 8 | The Emotion Chart is very nice I want it in other chat apps |
| 9 | The smily representation is quick to see and understand. Also the pop up window using bad language. |
| 10 | the idea and design |
| 11 | Like the feature :) |
| 12 | Tells you your emotions with a fairly good accuracy makes you think how your words are portrayed and recieved |

Table 8: Shows the qualitative answers to the question "What do you think is good about Chattitude?" in the final survey of the user study.

## 4.5 Discussion

Overall the results of the SUS and the final survey show, that the participants of this study are fairly satisfied with the quality of Chattitude. Furthermore there is no significant difference in usability and perceived workload, when turning the Signature Features on or off. Consequently, the hypotheses 2 and 3 stated in section 4.1 can be accepted. The results of the evaluation of the conversations through the custom survey and the logs, however, show that the detection systems need to be fine-tuned so that they have a positive effect on the conversations. After the evaluation of the logging data, one cannot confidently accept 1, but the objective is still reachable with after adjustments in the selection of the participant.

While the high SUS-scores, the star-ratings, and the positive qualitative feedback indicate that the users were fairly satisfied with the quality of the application there is still room for improvement. Nevertheless, the goal of the UX not being a con-founding influence in the user study, seems to be achieved, because the scores of the SUS do not significantly differ. However, in the future the sensitivity of the classifier and the amount of highlighting need to be reduced, so they are supporting rather than distracting the user. This could be achieved by improving the classifier, by pre- or post-filtering the tokens that should be analyzed and highlighted. Another approach could be to reduce the color intensity, for highlighted words that are either less likely to actually be labeled with the particular emotion or have a less emotional intensity.

The distraction caused by the high sensitivity of the classifier might also be the reason for the lower mutual understanding that participants reported in the custom survey for conversations. In other words, the mutual understanding is probably similar, but the participant is occupied with understanding the highlighting, distracting the participant from a fast paced discussion.

The high sensitivity seems to also frame the messages of the chat-partner as less intellectual, as the Emotion-Chart and the highlighting might indi-

cate more emotionality as there might be. The other view could be that the users become more aware of the emotions hidden in messages, but in turn think that the chat-partner is less intellectual or factual, because of a unconscious association of an anti-proportional correlation between intellectual and emotional.

This might raise the question how to fine-tune the highlighting so that it doesn't distract the user but is not ceasing to exist. Chattitude seemed to have been too sensitive, but if the user needs to edit his message to achieve a specific emotion, a higher sensitivity might be needed. Consequently there might be different use-cases that demand a specific sensitivity, which in turn poses a challenge for the fine-tuning of the analysis and the visualization. A question that might also have consequences on the visualizations of different linguistic analysis types, such as Named-Entity-Recognition (NER) or Co-Referencing. These examples could ease the comprehensibility of a message by helping defining certain words of importance. Increasing the comprehensibility could help improving the mutual understanding in a conversation, which is crucial in discussions.

Future studies also need to focus on the selection of participants and generation of conflicts, as the average participant in this study was a 28 year old student. This group of people apparently is quite friendly and respectful. So to test Chattitude with more aggressive content, participants from an aggressive environment need to be selected. The range of topics in this study were quite intellectual, so the level of the topics needs to be lowered. In future studies, more provocative topics based on specific tweets for example could be chosen. Furthermore the personal involvement in a specific topic was quite low because of ethical concerns. This could cause a lower emotional involvement and in turn lowers the amount of profanity, as it is difficult to provoke the participants.

# 5 Conclusion

In this thesis a prototype web-application named Chattitude was developed, which mimics common chat-applications like WhatsApp or Telegram. The goal of this application was to deescalate a chat conversation, so for that purpose two linguistic analyses were used to detect profanity and emotions in each message. In the GUI the results of these detection systems, were visualized, so the user is encouraged to reflect more on the emotional and profane contents of his or her message.

Through a user study the quality of Chattitude and the effect of its Signature Features on conversations was investigated. Results of this study show that the participants perceived Chattitude to be usable with an average SUS-score of 76.26. With an average rating of 3.5 stars, Chattitude competes with the upper-range of the middle-field of published applications.

The use of the Signature Features had no effect on the quality and perceived workload of the application, however the sensitivity of the emotion-classifier needs to be fine-tuned so that it does not distract the users. The distraction lowers the mutual understanding and frames the messages of the chat-partner as less intellectual or factual.

# 6 Future Work

There are multiple areas in the context of this project, that can be improved, extended or continued. In general there is the technical and experimental field in which the most potential work can be done.

### 6.0.1 Technical

The technical sector is focused on improving or adding of features for Chattitude, especially those that are needed to make it full fledged application,

that users can use in daily live. Future feature might be an encryption system, user management, a sarcasm detection system, dedicated structuring of conversations and moderating roles.

**Encryption** Since WhatsApp introduced the end-to-end encryption, this feature became an important argument for the trustworthiness of applications with private communication. For Chattitude this feature poses a huge challenge as it obstructs an automatic analysis of the content of a message. It might even be impossible to implement a classic end-to-end-encryption with an automatic analysis, if it needs too many resources. In the case of Chattitude the analysis pipeline needed to be implemented on a server to shift workload from the client. To be able to extend a classic spell checker, the analysis processes would need to become a lot faster. If this is achieved an end-to-end-encryption might be possible.

Till then, a possible approach could be to give the user the control over every application besides the message-exchange service and make the message decipherable only for permitted applications.

**Improving The Emotion Classification** As mentioned in the discussion of the user study, the emotion visualization needs more fine-tuning as the high sensitivity caused the user to be distracted. Furthermore, the accuracy of the classifier could be increased by implementing a correction system for the user. This way the correction could serve as annotated data and could be used to train the classifier. To improve the highlighting either a dictionary approach could be tested or a new classifier needs to be created which features an attention layer to trace back the results to inputs. Furthermore, if multiple classifiers are to be utilized, an AI-management system could be tried out, that also features a domain specific detection. This way multiple languages could be supported, as the management system could choose the optimal classifier for the correct domain. A domain in this context could be the areas of expertise, languages or a conversation specific dictionary.

**Sarcasm Detection & Other Detection Techniques** The use of sarcasm can intensify a conflict. There are already systems that can detect sarcasm as Joshi et al. (2017) discussed various approaches. However, this task does not only require the pure linguistic detection of sarcasm, but in this context it also requires a separate visualization and the analysis pipeline needs to be extended.

**Automatic Moderation** The fundamental concept of Chattitude is to mimic an established chat-application. A simple chat-application attempts to simulate a normal spoken conversation on textual basis, creating a new kind of conversation. Structuring conflicts or generally discussions would be a set of features that is attempting to give a conversation a specific structure so that it helps running through the phases of a conflict or discussion. Possible phases could be the following:

1. Identifying and defining the conflict or important aspects of a conflict.
2. Identifying possible solutions
3. Research
4. Evaluating and rating of the possible solutions
5. Negotiating for competing solutions
6. Defining the final solution

Using the knowledge of the current phase, Chattitude could attempt to guide or moderate the discussion by asking questions typical for the specific phase (e.g.: @userX: "What could be a solution for you?" in the second phase). Chattitude could then ask each conflict party if it accepts a specific answer to find common ground step-by-step. Furthermore, supporting features a like a voting or a calendar-synchronization system could added as well. The voting system could be used to finalize certain decisions, as a calendar-system

could automatically identify and attempt to solve scheduling conflicts. Other features dedicated to the gamification of such conversations, could be achievements for clean and respectful language and a reputation system, where users can reward each other for positive behavior. These features would follow the approach of rewards rather than punishments. Additionally, other approaches of CSCW mentioned by Easterbrook et al. (1993) should be investigated and integrated into Chattitude if possible.

**Manual Moderation** In certain conflicts the difficulty to resolve is quite high, which causes conflict parties to struggle a lot to solve them. In such a case a moderator or mediator is could help to go through the discussion without with less problems or violence. In the context of Chattitude this would involve the implementation of user roles, in this case specifically the role of a moderator. A role should then come with a set of features that support that particular role. In this case the moderator will need to be highlighted in a certain way and he needs to have an interface where he can easily communicate with both parties privately. In certain contexts the moderator could also make use of rights like correcting or deleting messages, that would insult the other party or obstructs the flow of the conflict.

### 6.0.2 Experimental

In this section other approaches are discussed that either investigate new factors in conversations of chat-applications or approaches that could improve the user study conducted in this thesis.

**Improvements** The evaluation of the conversation by a custom survey (Table 1) could be improved, by evaluating the survey itself and rephrasing of the items if necessary. Furthermore, it needs to by psychometrically tested, to exclude bias that could be caused by the survey itself.

Another improvement of the user study is improving the conflict generation approach. The approach in this study was to pre-select a group of topics and investigate the trends among the participants. The problem however is the selection of the topics, because the topics in this study had a low personal involvement. This caused participants to be less obliged to enforce their own position and consequently less likely to use more verbally violent measures to enforce their interests. In the future it would be advisable to either use topics that are truly known for their balance and high polarization, and recruit involved people as participants. This could be done by finding a controversy on platforms like Twitter or YouTube, and invite people that are actively involved in this controversy. Another approach could be to recruit participant with a dedicated provocation role, to artificially increase the conflict potential. Speaking of participants, it might also be beneficial for the study to select participants that are younger, however, the ethical factor always needs to be respected.

**New Factors** The user study conducted here, is based on a single online encounter between two people that did not know each other. This made the use of a survey that evaluated the quality of a relationship impossible. In the future a long term study that investigates the influence of Chattitude on longer encounters or relationships could be conducted.

Another dimension not investigated is the number of simultaneous participants, which was two in this study. In the future conversation with a one-to-many or many-to-many characteristic could be investigated, for account for group dynamics.

# A Appendix

| | Statement |
|---|---|
| 1 | I think that I would like to use this website frequently. |
| 2 | I found this website unnecessarily complex. |
| 3 | I thought this website was easy to use. |
| 4 | I think that I would need assistance to be able to use this website. |
| 5 | I found the various functions in this website were well integrated. |
| 6 | I thought there was too much inconsistency in this website. |
| 7 | I would imagine that most people would learn to use this website very quickly. |
| 8 | I found this website very cumbersome/awkward to use. |
| 9 | I felt very confident using this website. |
| 10 | I needed to learn a lot of things before I could get going with this website. |

Table 9: Items of SUS used in the User Study of this the Thesis. Each Statement is scored by participants on a Likert Scale of 1-5 (5 = "Strongly Agree").

| Scale | Description |
|---|---|
| Mental Demand: | How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc)? Was the task easy or demanding, simple or complex, exacting or forgiving? |
| Physical Demand: | How much physical activity was required (e.g. pushing, pulling, turning, controlling, activating, etc)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious? |
| Temporal Demand: | How much time pressure did you feel due to the rate of pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic? |
| Performance: | How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals? |
| Effort: | How hard did you have to work (mentally and physically) to accomplish your level of performance? |
| Frustration: | How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task? |

Table 10: Items of the NASA TLX Survey used in the User Study of this Thesis

| | Keyword | Statement |
|---|---|---|
| T1 | DemoPupils | Pupils should be forbidden to attend demonstrations (e.g. Fridays for Future) |
| T2 | DivingBan | More bans on driving in major cities should be implemented to reduce fine dust issues. |
| T3 | VoteParent | Parents should be allowed to vote on behalf of their underage children. |
| T4 | GeneEditing | We need gene splicing and other gene manipulation methods to combat inherited diseases. |
| T5 | EgoShooter | Ego shooters and other violence encouraging games should fall under the same laws as physical small arms, requiring registration via id. |
| T6 | Embryos | Regulations of research on embryos should be lightened. |
| T7 | Smoking | Smoking and vaping should be banned. |
| T8 | E-Mobility | Any kind of e-mobility (e-car, e-scooter, e-bike) is superior to fuel based mobility and should be encouraged. |
| T9 | GenderSTEM | Politics should invest a lot more resources and put more regulations in place to over come the gender gap in STEM (Science, Technology, Engineering, Maths) subjects. |
| T10 | RentLimit | The solution to high renting prices is limiting them by law. |
| T11 | Military | Germany should invest more resources into military and police. |
| T12 | Dresscode | Schools should enforce a dress code (not uniform). |
| T13 | AfricaAid | More resources should be invested as development aid into Africa. |

| | | |
|---|---|---|
| T14 | ClimatTech | The only solution for the stabilization of the climate is technology. |
| T15 | China | China will be a good future economic partner of Germany. |
| T16 | PoliceDrones | Drones should be used more by the police |
| T17 | Soccer | Soccer is an overrated sport, and other sportive activities should be promoted. |
| T18 | Brexit | The UK should have exited the EU on the 31. of October. |

Table 11: List of topics participants needed to state their opinion on a Likert Scale from 1 to 7 (7 = "Strongly Agree") and an option to not state an opinion. In Figure 8 the results of the answers to this topic list are presented.

# Glossary

**JSON Schema** This is used to give JSON a structure with which a given JSON object can be examined and verified whether it is structured as defined in the schema. 18, 58

**Angular** A framework similar to Vue.js [17] or React.js [18], for creating web applications. It is based on TypeScript . 19, 26, 59

**API** Application Programming Interface. 16, 25, 26, 60

**asynchronous** "In computer programs, asynchronous operation means that a process operates independently of other processes, whereas synchronous operation means that the process runs only as a result of some other process being completed or handing off operation. A typical activity

[17] `https://vuejs.org/`(last accessed: 27.11.2019)
[18] `https://reactjs.org/`(last accessed: 27.11.2019)

that might use a synchronous protocol would be a transmission of files from one point to another. As each transmission is received, a response is returned indicating success or the need to resend. Each successive transmission of data requires a response to the previous transmission before a new one can be initiated." [19]. 20

**C1** Condition One. 27, 29, 32, 33, 35, 36, 38–43, 59

**C2** Condition Two. 28, 29, 33, 35, 36, 38–43

**CAPTCHA** completely automated public Turing test to tell computers and humans apart. 10

**Chat-Service** A JavaScript package that extends Socket IO to "handle common public network messaging problems like reliable delivery, multiple connections from a single user, real-time permissions and presence" [20]. 16, 26, 60

**Chattitude** The Name of the web application used as a Prototype in this thesis.. 7–13, 15–19, 21–29, 32, 33, 35, 38–41, 43–52, 57–60

**CMC** Computer Mediated Communication. 7

**CSCW** Computer Supported Cooperative Work. 7, 51

**emotion** In the context of Chattitude, the range of emotions is limited to the generalized terms of anger, sadness, surprise, joy, disgust, fear and "none". Each message of Chattitude is analyzed and is labeled with this set of emotions. This applies to each word in a message as well. 6, 11, 14, 17–20, 29, 42, 43, 46, 48, 58, 59

[19] `https://searchnetworking.techtarget.com/definition/asynchronous`(last access: 05.11.2019)

[20] https://github.com/an-sh/chat-service (last access: 05.11.2019)

**Emotion-Chart** A Signature Feature of Chattitude, which appeares next to a message, and displays the emotions in the message. 13–16, 24, 27, 44, 46, 59, 60

**GUI** Graphical User Interface. 11, 13, 26, 48

**HCI** Human-Computer-Interaction. 7

**highlighting** In Chattitude highlighting refers to the visualization of the emotion of a given word in a message by using a color. 14, 24, 27, 44–47, 49, 59, 60

**HTML** Hypertext Markup Language. 23

**IRC** Internet Relay Chat. 26

**JavaScript** A programming language that is mainly used in webdevelopment to implement the behavior of the front-end of a website or to build server-application with e.g. Node JS. 57, 59, 60

**JSON** Javascript Object Notation. 18, 56, 58

**LRU** Least-Recently-Used. 22

**message** In the context of Chattitude a message is first of all a piece of text created by a user of Chattitude but it is modeled using JSON Schema. By applying multiple NLP techniques this model is augmented by information about the Emotion and profanity, as well as a tokenized representation of the message. The model can easily be extended with additional information generated by present or future NLP techniques, if needed. 13, 15, 16, 18–22, 24, 29, 32, 41–44, 46–48, 57–59

**NASA TLX** NASA Task Load Index. 32, 35, 36, 40, 41, 54, 59

**NER** Named-Entity-Recognition. 47

**NLP** Natural Language Processing. 6, 8, 9, 13, 18, 19, 58, 60

**Node JS** A JavaScript based framework for mostly web applications[21]. 16, 19, 58

**OS** Operating System. 32

**perceived workload** In the context of the NASA TLX, perceived workload refers to . 27, 28, 33, 41, 46

**POS** Part of Speech. 20

**Preview-Message** The preview of the message that the user is currently editing. In C1 it features the Emotion-Chart and the highlighting. 15, 16, 24

**Prime NG** A user interface suite, similar to matarial.io [22], featuring multiple components for Angular like menus, inputfields, panels and drag & drop areas . 19

**profanity** In the context of Chattitude, profanity is a label of a message, indicating whether the message contains any words of foul language. It is very similarly modeled as emotions, with only minor differences described in the thesis. 11, 14, 15, 17–20, 48, 58

**profanity warning** In Chattitude the profanity warning is an overlay warning the user has to confirm in order to send a message containing profane language. 13–15, 28, 29, 43, 44, 60

**Python** A programming language, widely used by developers for artificial intelligence and NASA TLX. 19

**REST** REpresentational State Transfer. 16

[21] `https://nodejs.org/en/about/`(last access: 05.11.2019)

[22] `https://material.io/`(last accessed: 28.11.2019)

**Signature Feature** The features of Chattitude that set it apart from other chat applications. Namely these are the Emotion-Chart, profanity warning and the highlighting. 11, 13, 14, 22, 27–30, 32, 33, 38, 41, 43, 44, 46, 48, 58

**Socket IO** A JavaScript library enabling real-time bi-directional communication between a server and their clients. In Chattitude it serves as the underlying communication protocol for the transmission of messages, analysis requests and other kinds of events. In the back-end of Chattitude, an extension of this library, called Chat-Service is used. 16, 57

**Stanford CoreNLP** A toolkit for NLP functions which also features a server with an API developed by Manning et al. (2014). 17, 18, 20

**SUS** Sytem Usability Scale. 32, 35, 36, 38–40, 46, 48, 53

**TypeScript** A programming language that represents a snytactical superset of JavaScript. 26, 56

**URL** Uniform Resource Locator. 12, 35

**UX** User Experience. 24, 29, 32, 46

# References

Jennifer Aaker and Andy Smith. *The Dragonfly Effect: Quick, Effective, and Powerful Ways To Use Social Media to Drive Social Change*. John Wiley & Sons, September 2010. ISBN 978-0-470-61415-0.

Aaron Bangor, Philip Kortum, and James Miller. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *J. Usability Studies*, 4(3):114–123, May 2009. ISSN 1931-3357. URL `http://dl.acm.org/citation.cfm?id=2835587.2835589`.

Anat Ben-David and Ariadna Matamoros-Fernandez. Hate speech and covert discrimination on social media: Monitoring the Facebook pages of extreme-right political parties in Spain. *International Journal of Communication*, 10:1167–1193, 2016. ISSN 1932-8036. URL `http://ijoc.org/index.php/ijoc/article/view/3697`.

Laura Ana Maria Bostan and Roman Klinger. An analysis of annotated corpora for emotion classification in text. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2104–2119. Association for Computational Linguistics, 2018. URL `http://aclweb.org/anthology/C18-1179`.

John Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

Herbert H Clark, Susan E Brennan, et al. Grounding in communication. *Perspectives on socially shared cognition*, 13(1991):127–149, 1991.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ICWSM '17, pages 512–515, 2017.

Steve M Easterbrook, Eevi E Beck, James S Goodlet, Lydia Plowman, Mike Sharples, and Charles C Wood. A survey of empirical studies of conflict. In *CSCW: Cooperation or conflict?*, pages 1–68. Springer, 1993.

Johan Galtung and Hedda Wagner. *Strukturelle Gewalt: Beiträge zur Friedens-und Konfliktforschung*. Rowohlt Reinbek, 1975.

Sandra G Hart and Lowell E Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In *Advances in psychology*, volume 52, pages 139–183. Elsevier, 1988.

Aditya Joshi, Pushpak Bhattacharyya, and Mark J. Carman. Automatic Sarcasm Detection: A Survey. *ACM Comput. Surv.*, 50(5):73:1–73:22,

September 2017. ISSN 0360-0300. doi: 10.1145/3124420. URL http://doi.acm.org/10.1145/3124420.

Aycan Kaya, Reha Ozturk, and Cigdem Altin Gumussoy. Usability Measurement of Mobile Applications with System Usability Scale (SUS). In Fethi Calisir, Emre Cevikcan, and Hatice Camgoz Akdag, editors, *Industrial Engineering in the Big Data Era*, Lecture Notes in Management and Industrial Engineering, pages 389–400, Cham, 2019. Springer International Publishing. ISBN 978-3-030-03317-0. doi: 10.1007/978-3-030-03317-0_32.

Josh Kjar. The Trouble with Twitter. *Student Publications*, August 2019. URL https://scholarsarchive.byu.edu/studentpub/279.

Roman Klinger, Orphée De Clercq, Saif M. Mohammad, and Alexandra Balahur. IEST: WASSA-2018 Implicit Emotions Shared Task. *arXiv:1809.01083 [cs]*, September 2018. URL http://arxiv.org/abs/1809.01083. arXiv: 1809.01083.

Elizabeth D Liddy. Natural Language Processing. page 15, 2001.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit, 2014. URL http://www.aclweb.org/anthology/P/P14/P14-5010.

O. Hobart Mowrer. *Learning theory and behavior.* Learning theory and behavior. John Wiley & Sons Inc, Hoboken, NJ, US, 1960. doi: 10.1037/10802-000. URL https://archive.org/details/learningtheorybe00mowr/page/n15.

Atte Oksanen, James Hawdon, Emma Holkeri, Matti Näsi, and Pekka Räsänen. Exposure to online hate among young social media users. *Sociological studies of children & youth*, 18(1):253–273, 2014.

Ronald D. Rogge, Frank D. Fincham, Dev Crasta, and Michael R. Maniaci. Positive and negative evaluation of relationships: Development and validation of the Positive-Negative Relationship Quality (PN-RQ) scale. *Psychological Assessment*, 29(8):1028–1043, August 2017. ISSN 1939-134X. doi: 10.1037/pas0000392.

Jitendra Kumar Rout, Kim-Kwang Raymond Choo, Amiya Kumar Dash, Sambit Bakshi, Sanjay Kumar Jena, and Karen L. Williams. A model for sentiment and emotion analysis of unstructured social media text. *Electronic Commerce Research*, 18(1):181–199, March 2018. ISSN 1572-9362. doi: 10.1007/s10660-017-9257-8. URL `https://doi.org/10.1007/s10660-017-9257-8`.

Jeremy Waldron. *The Harm in Hate Speech*. Harvard University Press, June 2012. ISBN 978-0-674-06508-6. Google-Books-ID: mJ6mEAbQ9koC.

Ali Yadollahi, Ameneh Gholipour Shahraki, and Osmar R. Zaiane. Current State of Text Sentiment Analysis from Opinion to Emotion Mining. *ACM Comput. Surv.*, 50(2):25:1–25:33, May 2017. ISSN 0360-0300. doi: 10.1145/3057270. URL `http://doi.acm.org/10.1145/3057270`.

# Websites

Cambridge. `https://dictionary.cambridge.org/us/dictionary/english/troll`, 2019. Accessed: 27.11.2019.

Ben Friedland. profanity-filter. `https://github.com/ben174/profanity`, 2013. Accessed: 22.11.2019.

Roman Inflianskas. profanity-filter. `https://github.com/rominf/profanity-filter`, 2019. Accessed: 22.11.2019.

Jigsaw. Toxic Comment Classification challenge. https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/overview, 2017. Accessed: 14.11.2019.

Linus Neumann. Die trolldrossel (erkenntnisse der empirischen trollforschung). https://linus-neumann.de/2013/05/die-trolldrossel-erkenntnisse-der-empirischen-trollforschung/, 2013. Accessed: 27.11.2019.

Priori-Data. Daily active users (dau) of leading messaging and communication apps from the google play store in germany during august 2019 (in 1,000s) [graph]. https://www.statista.com/statistics/858971/leading-android-messaging-daily-active-users-dau-germany/, 2019.

Victor Zhou. profanity-check. https://github.com/vzhou842/profanity-check, 2019. Accessed: 22.11.2019.