Leibniz Universität Hannover
Fakultät für Elektrotechnik und Informatik
Institut für Praktische Informatik
Fachgebiet Software Engineering

Konzept für eine präventive Stimmungsanalyse von Nachrichten in Software-Entwicklungsteams

Concept for a preventive sentiment analysis of messages in software development teams

Masterarbeit

im Studiengang Informatik

von

Lennart Schroth

Prüfer: Prof. Dr. rer. nat. Kurt Schneider
Zweitprüferin und Betreuerin:
Dr. rer. nat. Jil Ann-Christin Klünder

Hannover, 19. April 2022

Zusammenfassung

Die moderne Softwareentwicklung ist mit komplexen Anforderungen in Bezug auf Zeit und Effizient konfrontiert. Dabei spielt die Kommunikation in Software-Entwicklungsteams eine elementare Rolle. Zur Erfassung des Kommunikationsklimas können verschiedene, mit viel Aufwand verbundene Methoden, wie anonyme Umfragen oder persönliche Gespräche genutzt werden. Die automatisierte Stimmungsanalyse schriftlicher Kommunikation kann in diesem Zusammenhang ein wichtiger Baustein sein, Stimmung zielgerichteter zu detektieren. Diese ist allerdings bisher nur retrospektiv möglich und bedarf einer zentralen Sammlung persönlicher Daten. Um Probleme und Konflikte im Vorfeld zu erkennen und dementsprechend präventive Maßnahmen treffen zu können, ist es sinnvoll, eine „Hintergrundstimmung" in Nachrichten zwischen Entwickler:innen „live" zu detektieren. Daher war das Ziel dieser Arbeit, ein Konzept zu entwickeln, um eine „Echtzeit"-Detektion zu ermöglichen und die Stimmung in Entwicklerteams vorbeugend zu analysieren. Dessen Implementierung als Software-Assistent bzw. „Live-Sentiment-analyse" soll Nutzer:innen unterstützen, eventuell auftretende „Stimmungstiefs" zu identifizieren und entsprechend zukünftig Mechanismen zu etablieren, die sich positiv auf die Teamstimmung auswirken können. Die Konzeptentwicklung umfasst die Aspekte Nachrichtenerfassung, Stimmungsanalyse und Datenvisualisierung. Die Stimmungsanalyse basiert auf der Kombination verschiedener bereits existierender Verfahren. Zur praxisnahen Evaluation des Konzeptes wurde die Live-Sentiment-Analyse in qualitativen Studien von Proband:innen aus unterschiedlichen Software-Entwicklungsteams eingesetzt. Die Studienergebnisse zeigen im Kern, dass die vorbeugende Live-Sentiment-Analyse als unterstützendes System erachtet wird, um durch positive schriftliche Nachrichten zwischen Entwickler:innen wiederum auf das Kommunikationsklima positiv zu wirken. Des Weiteren wurde deutlich, dass mittels anonymer Weiterleitungen der Einzelstimmungen zudem ein Teamdurchschnitt ermittelt werden kann, allerdings unter Berücksichtigung, dass alle Mitglieder die Teamdaten transparent einsehen können. Perspektivisch gilt es, Langzeitstudien durchzuführen, um einerseits das Nutzerverhalten in der Praxis zu erfassen und andererseits mehr Daten zu erheben, um die Genauigkeit der Stimmungsanalysen zu optimieren.

Abstract

Modern software development is faced with complex requirements in terms of time and efficiency. Communication in software development teams plays a fundamental role here. Various methods that involve a great deal of effort, such as anonymous surveys or personal interviews, can be used to record the communication climate. In this context, the automated sentiment analysis of written communication can be an important building block for detecting sentiment in a more targeted manner. However, this has so far only been possible retrospectively and requires a central collection of personal data. In order to recognize problems and conflicts in advance and to be able to take preventive measures accordingly, it makes sense to detect a "background sentiment" in messages between developers "live". Therefore, the aim of this work was to develop a concept to enable real-time detection and to analyze the sentiment in development teams as a preventative measure. Its implementation as a software assistant or "live sentiment analysis" is intended to support users in identifying any "low sentiment" that may occur and to establish future mechanisms that can have a positive effect on team spirit. The concept development includes the aspects of news gathering, sentiment analysis and data visualization. The sentiment analysis is based on the combination of various existing methods. For the practical evaluation of the concept, the live sentiment analysis was used in qualitative studies by test persons from different software development teams. The core of the study results shows that the preventive live sentiment analysis is considered a supporting system in order to have a positive effect on the communication climate through positive written messages between developers. Furthermore, it became clear that a team average can also be determined by anonymous forwarding of the individual votes, but taking into account that all members can view the team data transparently. In the future, it is important to carry out long-term studies in order to record user behavior in practice on the one hand and to collect more data on the other hand in order to optimize the accuracy of the sentiment analysis.

Inhaltsverzeichnis

1 Einleitung **1**
1.1 Motivation . . . 1
1.2 Ziel der Arbeit . . . 2
1.3 Struktur der Arbeit . . . 3

2 Grundlagen **5**
2.1 Verfahren zur Stimmungserkennung . . . 5
2.1.1 Stimmunganalyse in Sätzen . . . 5
2.1.2 Statistische Metriken . . . 7
2.2 Stimmungsanalyse . . . 10
2.2.1 SentiStrength . . . 10
2.2.2 SentiStrength-DE . . . 10
2.2.3 SentiStrength-SE . . . 11
2.2.4 Senti4SD . . . 11
2.2.5 SentiCR . . . 11
2.3 Technik . . . 12
2.3.1 Keylogger . . . 12
2.3.2 Plug-in . . . 12
2.3.3 MQTT . . . 13

3 Verwandte Arbeiten **15**

4 Konzeptentwicklung **19**
4.1 Vision . . . 20
4.1.1 Szenarien . . . 21
4.2 Kontext . . . 22
4.3 Datenerfassung . . . 24
4.4 Datenanalyse . . . 24
4.4.1 Vorselektion von Verfahren zur Stimmungsanalyse . . . 25
4.4.2 Entscheidungen . . . 25
4.4.3 Standardisierung der Analysen . . . 27
4.4.4 Grenzen der vorhandenen Analysen . . . 27
4.5 Datensammlung und -visualisierung . . . 28

5 Konzeptimplementierung **29**
5.1 Software-Entwurf . . . 29
5.1.1 Software-Anforderungen . . . 30
5.1.2 Architektur-Entscheidungen . . . 31
5.2 Implementierung des Software-Assistenten . . . 33
5.2.1 Modul zur Datenerfassung . . . 34
5.2.2 Module zur Stimmungsanalyse . . . 35
5.2.3 Modul zur Datenvisualisierung . . . 38

6 Konzeptevaluation und Ergebnisse **43**
6.1 Vorstudie . . . 43
6.1.1 Teilnehmer:innen . . . 43
6.1.2 Durchführung . . . 44
6.1.3 Ergebnisse . . . 44
6.2 Verbesserungen . . . 46
6.3 Hauptstudie . . . 50
6.3.1 Teilnehmer:innen . . . 50
6.3.2 Durchführung . . . 51
6.3.3 Aufbereitung der Ergebnisse . . . 52
6.3.4 Ergebnisse . . . 53

7 Diskussion **61**
7.1 Zusammenfassung der Ergebnisse . . . 61
7.2 Antwort auf die Forschungsfragen . . . 62
7.3 Interpretation der Studienergebnisse . . . 63
7.4 Limitationen . . . 65
7.5 Empfehlungen . . . 66

8 Zusammenfassung und Ausblick **71**
8.1 Zusammenfassung . . . 71
8.2 Ausblick . . . 72

A Unterlagen zur Hauptstudie **73**
A.1 Einladung . . . 74
A.2 Fragebogen mit Ergebnissen . . . 76

Abbildungsverzeichnis **79**

Literaturverzeichnis **79**

Kapitel 1

Einleitung

Die moderne Softwareentwicklung wird mit verschiedenen komplexen Anforderungen konfrontiert. Eine moderne Software muss immer mehr Aufgaben in schnellerer Zeit lösen und dabei muss auch immer die Bandbreite von Anwendern berücksichtigt werden [1]. Zeitdruck bei der Entwicklung, neue Kollegen und schlechte Kommunikation sind nur einige Faktoren, die einen negativen Einfluss auf die Stimmung in einem Entwickler-Team haben können. Dabei reicht es schon, wenn einzelne Kollegen schlechte Stimmung verbreiten und dabei das gesamte Team ausbremsen. Eine steigende negative Stimmung nimmt die Energie aus einem Team und hemmt gleichzeitig die Produktivität [2].

1.1 Motivation

Damit Softwareprojekte erfolgreich abgeschlossen werden können, müssen viele Faktoren für die Entwickler:innen stimmen. Die Stimmung in einem Team ist dabei ein wichtiger Faktor, gleichzeitig einer, der nur schwer gemessen oder in Zahlen festgehalten werden kann. Als Geschäftsführung oder Projektleitung die Stimmung in einem Entwickler-Team einzuschätzen geht nur in persönlichen Gesprächen oder in Meetings, mit anonymen Umfragen oder über Informationen, die auf dem „Flur" ausgetauscht werden [3]. Gerade in dezentralen Teams, aber auch in Zeiten der Corona-Pandemie findet Kommunikation online in virtuellen Konferenzen, Chats und E-Mails statt [4]. In diesen Bereichen wird die Stimmung aber aber nur retrospektiv mit einer gewissen Verzögerung erfasst, da zuvor private Daten zentral erfasst und ausgewertet werden müssen. Um aber eine schwankende oder gar schlechte Stimmung in einem Team zu identifizieren, kann es sinnvoll sein, präventiv diese „Hintergrundstimmung" zu detektieren. Wie elementar Stimmung für die Produktivität sein kann, haben Kaur et al. [2] gezeigt, die Kommentare von Entwicklern über mehrere Monate untersucht haben. In diesem Rahmen ist aufgefallen, dass schlechte bzw. negative Stimmung die Motivation im Team drückt und die Produktivität sinkt. Die Identifizierung von Stimmung wird mit verschiedenen Sentimentanalysen durchgeführt. Dabei können einzelne Sätze oder ganze Dokumente klassifiziert werden. Je nach Klassifizierung wird in verschiedenen Skalen zwischen positiv, neutral und negativ unterschieden.

Eine positive Stimmung entspricht dann beispielsweise dem Wert 1, eine neutral ist äquivalent zur 0 und eine negative Stimmung wird mit -1 dargestellt. Die meisten bekannten Forschungsergebnisse in diesem Bereich, wie die von Pang et al. [5] oder Liu [6], beziehen sich auf den verbreiteten Anteil von englischsprachigen Daten. Sie bilden die Grundlage für eine computergestütze Analyse von Stimmung, meist basierend auf öffentlich zugänglichen Daten aus dem Internet von Beiträgen, Kommentaren oder Foren. Wie die Stimmung „jetzt" gerade in Nachrichten im eigenen Team ist, wird allerdings mit dieser Technik noch nicht erfasst.

1.2 Ziel der Arbeit

Das übergeordnete Ziel dieser Arbeit ist das „Proof-of-Concept", um die Durchführbarkeit der Echtzeit-Analyse von Textnachrichten in Software-Entwicklungsteams zu belegen. In diesem Zusammenhang wird ein Konzept entwickelt, implementiert und evaluiert, um präventiv Stimmung zu detektieren. Dabei handelt es sich bei dem theoretischen Konzept um einen Bauplan, der ein System skizziert, das die Stimmungsanalyse durchführt. Dieses System soll Entwickler:innen assistieren und die möglicherweise hervorgerufene Stimmung gerade verfasster Nachrichten erkennen und zurückmelden. Dieser Software-Assistent unterstützt somit die Reflexion der Stimmung in einem Team und dient infolgedessen der Optimierung der Kommunikation, ohne dabei auf einer zentralen, retrospektiven Analyse aufzubauen. Aufbauend auf einer praxisnahen Evaluation werden im Rahmen dieser Arbeit die folgenden Forschungsfragen (Research Question) beantwortet:

RQ1

Inwiefern kann Stimmung in Textnachrichten präventiv detektiert werden?

RQ2

Wie und unter welchen Bedingungen kann der Teamstimmung-Durchschnitt automatisiert berechnet werden?

1.3 Struktur der Arbeit

In dem sich dieser Einleitung anschließenden Kapitel 2 werdenbereits existierende Verfahren zur Stimmungserkennung vorgestellt und Techniken erläutert, die für den im Rahmen dieser Masterarbeit entwickelten Software-Assistenten von Relevanz sind. In Kapitel 3 werden verwandte, wissenschaftliche Arbeiten der angrenzenden Forschungsgebiete vorgestellt und in einen Kontext zu dieser Arbeit gesetzt. Der Fokus liegt dabei auf Erkenntnissen, die bei Entscheidungen den Software-Assistenten betreffend einen Einfluss hatten. In Kapitel 4 wird ein Konzept für die Erstellung eines Assistenten zur Erfassung der Stimmung in Textnachrichten vorgestellt. Dabei werden die Themen Datenerfassung, Datenanalyse und Visualisierung skizziert. Aufbauend auf diesem Konzept wird in Kapitel 4 der Entwurf des Software-Assistenten dargestellt sowie dessen Implementierung erläutert. Gegenstand von Kapitel 6 ist die Vorstellung der Studien zur Evaluierung des Konzepts sowie deren Ergebnisse. Das Kapitel 7 beinhaltet die Diskussion und Interpretation der Studienergebnisse, die zunächst anhand der Forschungsfragen erläutert werden. Abgeschlossen wird dieses Kapitel durch eine Empfehlung für den Nutzen und zukünftige Weiterentwicklungen des Assistenten zur Erkennung von Stimmungen in Textnachrichten. Im letzten Kapitel werden die Ergebnisse dieser Arbeit zusammengefasst und es werden im Ausblick Anregungen für mögliche weitere Forschungsprojekte gegeben.

Kapitel 2

Grundlagen

In dem folgenden Kapitel werden die relevanten Grundlagen ausgeführt, auf denen diese Masterarbeit aufbaut. Zunächst wird auf die verschiedenen Methoden und Metriken zur Stimmungsanalysen eingegangen, die für die Analyse von Textnachrichten in Frage kommen. Anschließend werden relevante Techniken erläutert, die für die durchgeführte Implementation des Konzeptes in einen Assistenten genutzt wurden.

2.1 Verfahren zur Stimmungserkennung

Die Stimmungsanalyse oder Sentimentanalyse ist ein Oberbegriff für verschiedene Analyseverfahren aus dem Bereich Text-Mining. Dabei werden einzelne Sätze oder ganze Dokumente in Bezug auf ihre Stimmung analysiert. Die Klassifikation bei der Stimmungsanalyse erfolgt dabei meist zwischen den drei Polaritäten positiv, neutral und negativ [7]. Eine positive Stimmung umfasst verschiedene Emotionen wie Euphorie (euphoria), Begeisterung (enthusiasm) oder Vergnügen (pleasure). Panik (panic) oder Wut (fury) sind beispielsweise Emotionen, die einer negativen Stimmung zugeordnet werden [8]. Es ist üblich, nicht die genaue Emotion zu detektieren, sondern nur eine Aussage darüber zu treffen, wie stark ein Satz positiv oder negativ ist [9]. Als neutral werden alle Wörter und Sätze interpretiert, die weder positiv noch negativ sind. Erwähnenswert ist, dass nicht alle Analyseverfahren eine Aussage treffen, wie positiv oder negativ eine Aussage ist. Einige[10, 11] teilen auch nur ein, ob ein Satz positiv oder negativ ist, aber nicht wie stark. Nachfolgend werden verschiedene Verfahren skizziert, die Wörter sowie andere Merkmale analysieren, die einen Rückschluss auf die Stimmung geben.

2.1.1 Stimmunganalyse in Sätzen

Im Folgenden werden drei Verfahren zur Stimmungsanalyse vorgestellt, die die verwendeten Wörter in einem Satz bezüglich ihrer Polarität untersuchen und damit aufzeigt, wie der Satz bei der oder dem Rezipierenden ankommt. Vor dem Hintergrund, dass Wörter von verschiedenen Personen oder Gruppen unterschiedlich

bewertet werden, sind diese Wortpolaritäten subjektiv, sodass hier von subjektiven Verfahren gesprochen wird.

Lexikonbasiertes Verfahren
Anwendungen, die nach dem lexikonbasierten Verfahren arbeiten, nutzen ein zuvor erstelltes Lexikon. In diesem befinden sich Wörter oder Sätze mit einer zuvor klassifizierten Polarität. Üblich ist es, dass jedes Wort im Satz einzeln interpretiert wird, damit im Anschluss der gesamte Satz evaluiert werden kann. Da sowohl positive als auch negative Wörter in einem Satz stehen können, werden diese meist zusammen gerechnet, um ein daraus resultierendes Ergebnis über die Stimmung im Satz zu ermöglichen. Positive Wörter können so mit (+1), negative mit (-1) und neutrale mit (+0) gewertet werden.

Zum Beispiel wird durch den Satz „Ich freue(+1) mich, dass du eine neue schöne(+1) Stelle gefunden hast. Schade(-1), dass du nicht mehr unser Kollege bleibst." folgende Gesamtstimmung ausgedrückt: = +1 +1 -1 = 1. Damit ist der Satz insgesamt als positiv zu bewerten. Abhängig von der Implementierung eines solchen Verfahrens können verschiedene Abstufungen vorgenommen werden. So können Wörter beispielsweise auch in größeren Intervallen gestaffelt sein, wie durch folgenden Beispielsatz deutlich wird: „Das war super(+3)! Gute(+2) Arbeit. Dumme(-1) Sache. Scheiß(-3) Fehler(-1)". Eine exemplarische Anwendung dieses Verfahrens wird im Abschnitt 2.2.1 „SentiStrength " vorgestellt.

In einem lexikonbasierten Verfahren sind gewisse Kombinationen von Wörtern schwieriger zu analysieren. Ein „dummer" Computer kann einzelne Wörter interpretieren, aber Wortgruppen sind deutlich schwieriger. Ein Beispiel sind Verneinungen, wie „nicht gut". Bei einer einfachen Analyse nur von Wörtern würde so eine neutrale Aussage „nicht gut(+1)" = 0 entstehen, obwohl eine analysierende Person hier eine klare negative Polarität als Ergebnis erwarten würde. Daher werden solche Verneinungswörter extra herausgesucht, um bei dem nachfolgenden Wort die Polarität umzudrehen. Ein „nicht' dreht die Polarität des nachfolgenden Wortes, um ein „nicht gut(-1)" zu erzeugen. Ein Ausdruck wie „nicht so gut(+1)" wird damit aber noch nicht richtig von dem Programm interpretiert, da das „nicht" nur das neutrale „so" umdreht, was neutral bleibt. Hier gäbe es eine falsche Interpretation der Stimmung. Ähnlich wie Verneinungen sind Ausdrücke mit verstärkenden Wörtern wie „viel", „sehr" oder „mehr" schwieriger zu analysieren. Beispielsweise sollte der Satz „Das war **sehr** gut(+1)" positiver bewertet durch das Wort „sehr" als „Das war gut". Genau so wirkt „Das war sehr schlecht(-1)" durch das „sehr" noch negativer. Solche Zusammenhänge müssen in Stimmungsanalysen immer zusätzlich zu den Wortpolaritäten betrachtet werden. Weil sie aber nicht immer alle Kombinationen abdecken, können Abweichungen in der Praxis vorkommen. Mit dem im Folgenden vorgestellten keywordbasierten Verfahren sind diese Aussagen häufig einfacher zu detektieren.

Keywordbasiertes Verfahren
Neben dem lexikonbasierten Verfahren gibt es in dem keywordbasierten Verfahren die Möglichkeit, in Sätzen nach Schlagwörtern bzw. Keywords zu suchen. Dabei handelt es sind um spezielle Wörter, Ausdrücke oder Phrasen, die eine positive oder negative Stimmung zum Ausdruck bringen. Das kann ein Ausdruck wie „Super!" oder eine Kombination wie „weiter so" sein.
Die Grundlage sowohl des keyword- als auch des lexikonbasierten Verfahrens ist die linguistische Analyse, nach der Wörter in einem Satz interpretiert werden, um eine bestimmte Stimmung des Satzes abzuleiten. Da beide Verfahren aber nach wie vor nicht alle Stimmungen adäquat aufzeigen können, werden Methoden aus dem Bereich des „Maschinellen Lernens" untersucht, trainiert und eingesetzt, mittels derer sprachliche Muster einfacher zu detektieren sind.

Verfahren mit maschinellem Lernen
Beim maschinellen Lernen handelt es sich um ein künstliches System, das mit vielen Trainingsdaten angelernt wird. Darauf aufbauend wird ein statistisches Modell erstellt, das Muster oder Gesetzmäßigkeiten nutzt, um Daten zu interpretieren. Umso mehr Trainingsdaten ein System zum Lernen erhält, umso genauer sind die Analysen. Eine der bekanntesten Anwendungen in dem Bereich bietet Google mit der „Cloud Natural Language".

2.1.2 Statistische Metriken

Neben den vorgestellten Verfahren Polaritäten gewissen Wörtern zuzuordnen, gibt es verschiedene Möglichkeiten Rückschlüsse auf die Stimmung zu gewinnen. Neben den offensichtlichen Ausdrücken von Stimmung über Smileys oder Emoticons gibt es subtilere Varianten [10]. Beispielsweise kann negative Stimmung über Großbuchstaben oder Satzzeichen ausgedrückt werden. In dem Beispiel „Das muss JETZT FERTIG werden!!!" sind die Verwendungen von Großbuchstaben und Satzzeichen ein Merkmal für schlechte Stimmung. Der Begriff der **Metrik** ist hierbei der quantitative Messwert eines Merkmales oder Verfahrens. Die Metrik dahinter wäre die tatsächliche Anzahl der Großbuchstaben. In dem gegebenen Beispiel sind 12 Großbuchstaben und 12 Kleinbuchstaben ist ein sehr „unnatürliches" Verhältnis.

Solche äußeren Formen einer Textnachricht lassen sich durch einfaches Zählen oder die Berechnung von Durchschnitten ermitteln. Im Gegensatz zu den vorgestellten subjektiven Polaritäten von Wörtern sind die Anzahl von verwendeten Großbuchstaben feste Werte. Die Ermittlung dieser **statistischen Metriken** werden als „**objektive Verfahren**" geführt. Insbesondere Häufigkeiten einzelner Wörter oder die Verhältnisse von Wörtern können interessante Metriken sein, mit denen Rückschlüsse auf Stimmung gezogen werden kann [10].

Großbuchstaben - „Das muss JETZT FERTIG WERDEN"
Großbuchstaben zu interpretieren kann ein Mittel sein Stimmung zu detektieren. Dieses muss aber im Kontext gesehen werden, da sonst Eigennamen „GOOGLE" dann negativ interpretiert werden.

Satzzeichen - „Jetzt!!!"
Auch bei Satzzeichen oder Kombinationen kann nicht automatisch ein negativer Zusammenhang hergestellt werden. „Super!!!" oder „Alles klar bei dir???" hängt immer von Kontext ab. „Super!!! noch eine Woche, wo du nichts fertig bekommen hast" oder „Super!!! das freut mich" sind nur ein Verstärker in einem Kontext. Dabei können sowohl bestimmte Satzzeichen betrachtet werden, wie auch die Anzahl der Satzzeichen.

Buchstaben-Satzzeichen-Verhältnis
Das Verhältnis von Buchstaben zu Satzzeichen kann ebenfalls interpretiert werden. Das Beispiel „Macht das! Jetzt! ich warte!" hat ungewöhnlich viele Satzzeichen und lässt auch eine schlechte Stimmung beim Autor schließen. Das sind aber eher konstruierte Beispiele, die in der Realität wohl eher selten anzutreffen sind.

Anzahl von Wörtern und durchschnittliche Wortlänge
Das vermehrte Vorkommen von längeren Wörtern kann für eine komplexe oder fachliche Wortwahl stehen. „Ausschweigende Beschreibungen mit sich sehr langziehenden Begrifflichkeiten und reichlich detaillieren Informationen" lassen sich nur mit viel Zeit und Ruhe schreiben. Kurze, kompakte Nachrichten vermitteln einen gewissen Zeitdruck und werden anders wahrgenommen. „Denk an den Termin heute" vs. „Bitte vergiss nicht, dass nach dem Meeting heute Mittag, der Kunde nochmal angerufen werden muss".

Emoticons
Gerade in privaten Chats häufig anzutreffen sind Smileys und Emoticons. Sie sollen Stimmungs- oder Gefühlszustände ausdrücken und werden gerade von der jüngeren Generation gern genutzt :-). In vielen Chat Programmen werden diese Zeichenkombinationen nach dem Eintippen in kleinere Grafiken umgesetzt, den sogenannten Smileys. Im Jahr 2015 haben Wang und Castanon [12] den Zusammenhang zwischen genutzten Emoticons und den Intentionen in der Kommunikation untersucht. Dabei wurde Emoticons den Klassen positiv, neutral und negativ zugeordnet und bestimmt, mit welcher Wahrscheinlichkeit diese vom Menschen interpretiert werden. Emoticons wie „:D" werden beispielsweise zu 90% positiv gewertet, wo hingegen „:(" zu 95% negativ klassifiziert wurde.

Von einigen Strichen, bis zu hunderten komplexen Grafiken, teilweise mit Animationen, haben sich diese Darsteller von Stimmung in den letzten Jahren immer weiter entwickelt. Wo zu beginn noch ein paar dutzend Emiticons zur Verfügung standen, gibt es mittlerweile über 3000 Grafiken für jede Stimmungslage. [1]

[1] https://de.statista.com/infografik/17371/anzahl-der-verfuegbaren-emoticons/

Anmerkung
Alle hier vorgestellen Verfahren zur Stimmungsanalyse von Wörtern sowie der weiteren Metriken haben Vor- und Nachteile. Daher haben verschiedene Anwendungen in den vergangenen Jahren verschiedene Metriken kombiniert, um die Stimmungsanalyse zu verbessern. Bevor einige der existierenden Anwendungen näher vorgestellt werden sei erwähnt, dass alle nur für einen gewissen Bereich entwickelt worden sind. Dabei deckt eine Anwendung nie alle Metriken mit ab. Sprache im Allgemeinen ist eine sehr komplexe Thematik, die sich zudem ständig weiter entwickelt. Gewisse Programme haben somit auch immer Vor- und Nachteile und können nicht ohne weiteres in verschiedenen Branchen eingesetzt werden.

2.2 Stimmungsanalyse

2.2.1 SentiStrength

Wie eingangs schon erwähnt handelt es sich bei der SentiStrength Anwendung um einen lexikonbasierten Klassifizierer. Dieser Interpretiert Sätze und ganze Texte in Hinblick auf ihre Stimmung. Dabei werden positive Polaritäten mit (+1) bis (+5) und negative von (-1) bis (-5) dargestellt. Die vorher definierten Bewertungen von Wörtern sind in einem Lexikon abgelegt. Diese Lexikon besteht dann aus mehreren Nachschlag-Tabllen (engl. Lookup-Table oder kurz LUT). Jede Lookup-Table befindet sich in einer Datei.

SentiStrength ist ursprünglich in Großbritannien für den englischsprachigen Raum entwickelt worden. Aufbauend auf der ursprünglichen Arbeit von Thelwall et al. aus dem Jahr 2010 sind viele weiteren Erkenntnisse in das Tool und das Lexikon eingeflossen[9]. Für Forschungszwecke darf diese Anwendung sowohl im Internet, aber auf Anfrage auch als Java-Applikation genutzt werden[2].
Mittlerweile haben sind in verschiedenen Ländern Organisationen und Gruppen gefunden, die dieses Tool nutzten, um mit angepassten Lexikas, auch die Stimmungsanalyse von Texten mit anderen Sprachen unterstützt. Für den deutschsprachigen Raum kann so das Lexikon „SentiStrength-DE" in die Java-Anwendung eingebunden werden.

2.2.2 SentiStrength-DE

Ein deutschsprachiges Lexikon für die SentiStrength-Anwendung zur Erkennung von Stimmung in Texten gibt es zum Zeitpunkt dieser Arbeit nur von dem Austrian Research Institute for Artificial Intelligence (OFAI)[3]. Dieses Lexikon kann über deren Internetseite bezogen werden [4].

Dafür bekommt SentiStrength einen Verweis auf das Lexikon mit den deutschsprachigen Lookup-Tables (LUT). Um dies zu verdeutlichen wird ein Auszug aus dem deutschsprachigem Lexikon vorgestellt.

Das Lexikon umfasst ein Arbeitsverzeichnis mit verschiedenen Nachschlage-Dateien, in denen sich jeweils eine Lookup-Table befindet. Um nun die Stimmung von Wörter zu bestimmen, sind diese mit Polaritäten in den einzelnen Dokumenten gespeichert. Diese werden als txt-Datei verwaltet und können auch manuell ergänzt werden. SentimentLookupTable.txt umfasst dabei die Wörter mit den zugewiesenen Polaritäten. NegatingWordList.txt enthält eine Lookup-Table mit Worten, die eine verneinende oder invertierende Funktion haben. BoosterWordList.txt umfasst verschiedene Verstärker wie „extrem", „sehr" aber auch negative wie „irrsinnig",

[2] http://sentistrength.wlv.ac.uk/
[3] https://www.ofai.at/research-areas
[4] https://www.ofai.at/resources/sentistrength_de

„eventuell” oder „wenig”.

Sentiment-LUT.txt

himmlische	4
schön*	3
aktiv*	2
bonus	1
allein	-1
streit	-2
mist	-3
angst	-4

NegatingWordList.txt

anti
kein
keine
keineswegs
nicht
nie
nix
ohne

BoosterWordList.txt

absolut	1
definitiv	1
extra	1
hoffentlich	-1
kaum	-1
sehr	1
voll	1
zweifellos	1

Tabelle 2.1: Auszüge aus dem deutschsprachigem SentiStrength Lexikon

Hinweis zu dem tabellarischen Auszug: Auf Wörter mit * können weitere Buchstaben folgen „schön**es**(+3) Beispiel”. Bei allen Tabellen handelt es sich nur um einen Auszug, der ein Verständnis für die Funktionalität zeigen soll und nicht um das gesamte Lexikon.

2.2.3 SentiStrength-SE

Eine Sentimentanalyse gibt es allerdings nicht nur für eine gewisse Sprache, sondern auch zugeschnitten an Bereiche oder Domänen. Für die Domäne des Software Engineering (SE) gibt es das Lexikon SentiStrength-SE. Dieses gibt es zum Zeitpunkt dieser Arbeit nur in englischer Ausführung. Im Englischen sind Wörter wie „Bug” oder „Error” negativ konnotiert, aber in dieser Branche fast alltägliche Begriffe. Daher wurde die Klassifikationen und Polaritäten dahingehend angepasst, um eine fehlerhafte Vorhersage über Stimmung zu vermeiden [13].

2.2.4 Senti4SD

Bei Senti4SD handelt es sich um eine Erweiterung von SentiStrength. Neben dem lexikonbasierendem Vorgehen, wird auch eine keywordbasierende und eine semantische Analyse genutzt. Ähnlich dem SentiStrengthSE soll die Kommunikation der SE Domäne betrachtet werden. Senti4SD stellt somit eine „Sentiment polarity detection for software development” [14]. Die open source Anwendung läuft unter Java 8 oder R [5].

2.2.5 SentiCR

Bei SentiCR handelt es sich um ein Framework, welches mit Hilfe von Verfahren aus dem maschinellen Lernen trainiert wurde, um Stimmungen in Sätzen zu detekietren.

[5] https://opensourcelibs.com/lib/senti4sd

Das „CR” steht dabei für „Code Reviews” das hier als Schwerpunkt für das Framework steht [15].

2.3 Technik

Neben Techniken zur Analyse von Stimmung in Sätzen, bedarf es auch Methoden und Protokolle diese Textnachrichten zuerst zu erfassen und zu übertragen.

2.3.1 Keylogger

Ein Keylogger ist ein Programm, das Eingaben in der Tastatur parallel zu anderen Programmen mithört und protokolliert. Der Begriff wird häufig mit Hackern oder Ermittlungsbehörden in Verbindung gerade, um an persönliche oder vertrauliche Daten zu kommen und ist daher eher negativ konnotiert. Allgemein ist es aber nur eine Software, um die getätigten Tastatureingaben zu verfassen. Dafür können verschiedene Programmiersprachen genutzt werden, die dann einen asynchronen Eingang mit Zeichen sogenannten Chars erhält. Solche Zeichen werden dann zu Sätzen, den Strings, zusammengefügt.
Vorteile eines solchen Programmes:

- Einfach und schnell zu implementieren
- In verschiedenen Programmiersprachen umsetzbar
- Bedarfen wenig Rechenleistung und Speicher

Da aber Eingaben und Veränderungen mit der Maus oder anderen Eingabegeräten nicht mit erfassen werden, können Wörter an der falschen Stelle landen.

2.3.2 Plug-in

Ein Plug-in ist eine Software-Erweiterung und hat den Namen vom englischem „plug in”, was für „einstecken” oder „anschließen” steht. Ein ähnlicher Begriff der häufig als Synonym zu finden ist, ist „Add-on”. Als Erweiterung dienen sie der Hauptanwendung als zusätzliche Möglichkeiten, können aber ohne die Anwendung nicht eigenständig existieren.

2.3.3 MQTT

Message Queuing Telemetry Transport (kurz MQTT) ist ein Protokoll für den Austausch von kleinen Nachrichten in und ein neuer Standard für IoT-Nachrichten (IoT = Internet of things). Dabei ist MQTT auf Umgebungen mit geringen Bandbreiten und hohen Latenzen spezialisiert und ein ideales Protokoll für die Machine-to-Machine-Kommunikation (M2M). Es ist netzwerkbasiert, kann somit sowohl auf einem Rechner oder verschiedenen verteilten Systemen in einem Netzwerk genutzt werden.
Eine zentrale Schnittstelle, der „MQTT-Broker" organisiert den Datenverkehr. Ein MQTT-Client verbindet sich mit dem Broker und meldet sich mit einer ID an. Dann kann ein Client dem MQTT-Broker mitteilen, welche Kanäle, hier „Topics', er abonniert. Wenn eine Nachricht auf ein Topic veröffentlicht bzw. „gepublished" wird, bekommen alle Abonnenten, diese Information vom Broker weitergeleitet. Dieses Abonnieren eines Topics wird mit einem „subscribe"-Befehl nach dem connect gemacht. Kommt es nach einem „Publish" einer Nachricht zu einer Störung, kann der Broker verschiedene Methoden nutzen, um die Qualität einer Verbindung aufrecht zu halten. Daten können so mit verschiedenen Service-Stufen übertragen werden. Der Quality-of-service (kurz QoS) sorgt dafür, dass die Nachricht bei Fehler garantiert empfangen oder verworfen werden.
QoS = 0 (at most once) sorgt dafür, dass eine Nachricht einmal ankommt, wenn sie verloren geht, ist dies nicht schlimm.
QoS = 1 (at least once) sorgt dafür, dass eine Nachricht mindestens einmal bei jedem Empfänger ankommt.
QoS = 2 (exactly once) sorgt in jeder einzelnen Verbindung mit jedem einzelnen Subscriber für das Erreichen der Nachricht beim Abonnenten.

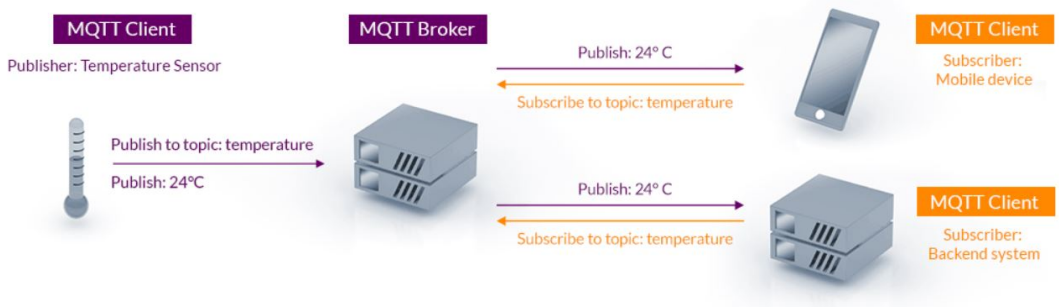

Abbildung 2.1: MQTT Architektur mit Beispiel

Für verschiedenste Endgeräten und Programmiersprachen gibt es Bibliotheken, die genutzt werden können, um Daten effizient zu nutzten [6].

[6] https://mqtt.org/

Kapitel 3

Verwandte Arbeiten

In diesem Kapitel wird ein Überblick über verwandte Forschungsarbeiten in dem Bereich Stimmungsanalyse in Software-Entwicklerteams gegeben. Die Darstellung konzentriert sich dabei auf ausgewählte Publikationen, die einen besonderen inhaltlichen Bezug zu den im Rahmen der vorliegenden Arbeit fokussierten Fragestellungen aufweisen.

Stimmungsanalysen im Bereich Software-Engineering (SE)

Obaidi und Klünder [16] liefern mit ihrer Publikation aus dem Jahr 2021 einen Überblick von Arbeiten in dem Themenfeld der Stimmungsanalyse in der Software-Entwicklung [16]. Auf der Basis verschiedener wissenschaftlicher Online- Datenbanken (ACM, IEEE, GoogleScholar, ScienceDirect und Springer) haben sie einen Pool von 6763 Arbeiten zusammengestellt, die sich dem Thema „Development and Application of Sentiment Analysis Tools in Software Engineering" widmen. Von den 80 Arbeiten, die genauer betrachtet wurden, waren es in 59 englischsprachige Open-Source-Softwareprojekte, die als Grundlage einer Stimmungsanalyse dienten. In 29 Papern wurde SentiStrength und in 10 SentiStrenghth-SE genutzt, um Stimmung zu detektieren. Ebenfalls häufiger wurde NLTK mit 13 mal und Senti4SD mit 9 mal genutzt.

Die Nutzung von Daten aus GitHub oder Stack Overflow aus Open-Source-Projekten zeigt die Herausforderung an Daten zu kommen, um Verfahren zur Stimmungsanalyse zu testen und zu verbessern. Ebenfalls schwierig sind aktuelle Forschungsthemen, wie das Identifizieren von Ironie und Sarkasmus.

Grundlegende Erkenntnisse zu Verfahren mit Sentimentanalyse stammen aus dem Jahr 2008 von Pang und Lee [5]. Ihre Analyse sollte aufzeigen, wie Menschen Gedanken und Gefühle schriftlich festhalten. Dazu wurden Online-Blogs und Review-Seiten zur Analyse herangezogen. Die extrahierten Texte sollten hinsichtlich ihrer positiven oder negativen Stimmung hin klassifiziert werden. Entsprechende Wörter

mit Polarität wurden dafür gezielt extrahiert.

Für eine automatisierte Analyse mit einer Software, stellten Thelwall et al. [9] im Jahr 2010 das Tool SentiStrength vor. Mit SentiStrength wurden die ersten Studien erarbeiten, in denen mithilfe von Lexika die Polarität von Wörtern in Sätzen erkannt wurde um Rückschlüsse auf deren Stimmung zu erhalten [9, 17].
Aufbauend auf Thelwall et al. wurde das Tool für weitere Sprachen angepasst. SentiStrength und die Lexika für unterschiedliche Sprachen, können über die Internetseite [1] frei getestet werden.

Für die Domaine Software-Engineering (SE) wurde das lexikonbasierte SentiStrength im Jahr 2017 zu SentiStrengthSE weiter entwickelt [13]. Diese Spezialisierung soll eine höhere Genauigkeit liefern, da in SentiStrength der allgemeine Kontext nur betrachtet wird. So wurden weitere Anpassungen an den Lexikas von SentiStrength vorgenommen, um diese auf die spezielle Domäne Software-Engineering anzupassen. In einer Studie haben Islam et al. [13] anschließend gezeigt, dass SentiStrengthSE Texte aus dieser Domain genauer interpretiert.

In einer Untersuchung im Bereich Software-Engineering (SE) haben Novielli et al. [18] im Jahr 2020 vier gängige Sentiment-Analysen mit verschiedenen Datensätzen gegeneinander verglichen. Der „Off the shelf"-Test sollte zeigen, wie genau Senti4SD, SentiStrength-SE, SentiCR und DEVA in der Praxis abschneiden [18]. Dafür wurden verschiedene Daten von GitHub, Jira und Stack Overflow herangezogen. Eine der gewonnen Erkenntnisse besteht darin, dass ohne weitere Trainingszyklen für die Verbesserung von Verfahren mit überwachtem Lernen (supervised learning), die lexikonbasierten Verfahren eine höhere Genauigkeit aufgewiesen haben. SentiStrength-SE hat dabei eine konstante Genauigkeit zwischen 75 - 80 % gezeigt.

Eine praxisnähere Anwendung von kombinierten Verfahren zur Stimmungsanalyse ist Gegenstand der Masterarbeit von Specht [19] aus dem Jahr 2021. Um Stimmungsanalysen zu verbessern, wurden verschiedene Metriken und Verfahren mit englischen Datensätzen von Stack Overflow untersucht. Diese Nachrichten aus Software-Entwicklerteams sind aus öffentlichen Projekten entnommen und anschließend evaluiert worden. Die subjektive Analyse mit SentiStrength lieferte dabei die zuverlässigsten Ergebnisse, trotz einem Genauigkeitsmaß von ungefähr 80% auf. Mithilfe von objektiven Metriken, wurde das Genauigkeitsmaß darauf aufbauend verbessert. Dazu wurden die objektiveren Metriken Informationsgehalt, durchschnittliche Anzahl von Wörtern und die Analyse von Emoticons herangezogen, um durch eine Kombination genauere Ergebnisse zu erzielen. Die Genauigkeit der Vorhersage konnte damit auf ein Maß von etwa 86% erhöht werden [19]. Dessen ungeachtet sind auch bei diesen Untersuchungen primär englischsprachige Daten untersucht worden, da Testdaten in größeren Stückzahlen aus öffentlichen Projekten

[1] http://sentistrength.wlv.ac.uk/

bezogen werden können. Ebenfalls ist die Forschung für englischsprachige Analysen am weitesten verbreitet.

Deutschsprachige Stimmungsanalysen im SE

In vielen Forschungsarbeiten, wie auch bei den eben vorgestellten, handelt es sich primär um englischsprachige Sentiment-Analysen. Stimmungsanalysen, die auch die deutsch Sprache interpretieren können, sind in der Forschung deutlich seltener.

Die Erkennung von Stimmung in einem deutschsprachigem Entwicklungsteam war Thema der Masterarbeit von Horstmann [10]. Mittels natural language processing (NLP) wurde die schriftliche Kommunikation aus Meetings von Software-Entwicklerteams analysiert. Dafür hat er ein Verfahren entwickelt, das im ersten Schritt mittels NLP verschiedene Metriken aus allen Textnachrichten eines Chatsverlaufs extrahiert. Horstmann kombinierte dazu verschiedene Metriken. Neben inhaltlichen Metriken, die die Polarität einzelner Wörter im Satz interpretieren, wurden auch statistische Metriken genutzt. Hierzu wurden die Längen und die Häufigkeiten bestimmter Wörter betrachtet. Die inhaltlichen Metriken basierten auf Datensäten von Waltinger [20] und Remus et al. [21]. Mittels maschinellen Lernen wurde ein Modell mit den Trainingslabel positiv, neutral und negativ trainiert. Ein entsprechender evolutionärer Algorithmus nutzt die verschiedenen Metriken, um das beste Modell zu trainieren. Zur Klassifikation wurden drei Verfahren herangezogen: Random Forests, Support Vector Machine und Naive Bayes. Ein VotingClassifier gewichtet abschließend die mehrheitliche Klasse. Mit einem Datensatz von Textnachrichten aus einem privaten Unternehmen hat Horstmann diese Masterarbeit abschließend validiert [10]. Hierbei wurde eine Genauigkeit von 63% erreicht[10].

Diese Erkenntnisse zur Stimmungsdetektion wurde von Klünder und Horstmann in einer Fallstudie untersucht und in einem Paper aufbereitet [22]. Hierbei wurden 1947 gesendete Nachrichten aus einem Gruppenchat aus dem Open-Source-Chat Zulip über 5,5 Monate untersucht. Ziel war die Analyse der Gruppenstimmung in einer industriellen Fallstudie. Mit einer durchschnittlichen Genauigkeit von rund 63% wurde die Textnachrichten in positiv, neutral und negativ klassifiziert. Diese grobkörnige Einteilung ließ zusammen mit dieser Genauigkeit Tendenzen in der Gruppenstimmung ausreichend genau detektieren [22].

Aufbauend auf der Analyse von Stimmung in Textnachrichten hat Herrmann [11] sich in seiner Bachelorarbeit mit der Analyse sprachlicher Kommunikation in Meetings beschäftigt. Dazu wurde während eines Meetings der Ton aufgezeichnet und transkribiert. Das verschriftliche Meeting wurde im Anschluss auf Stimmung hin untersucht, um dem Projektleiter am Ende ein direktes Feedback über den Verlauf des Meetings zu gewähren [11].

Abgrenzung der Arbeit

Bei allen hier vorgestellten Arbeiten wurden die zu analysierenden Daten zentral gesammelt und retrospektiv ausgewertet. Textnachrichten und Meeting-Skripte aus Software-Entwicklerteams werden bereits analysiert. Eine dezentrale Echtzeit-Stimmungsanalyse, die Gegenstand dieser Arbeit ist, wurde aber noch nicht durchgeführt. Die erwähnten Arbeiten zeigen, dass die vorhandenen Verfahren an gewisse Branchen oder Sprachen angepasst wurden. Das lexikonbasierte SentiStrength, das Lexika für verschiedene Sprachen unterstützt und darüber hinaus auch anpassbar für Branchen wie die Software-Entwicklung ist, stellte sich bei den Recherchen für diese Masterarbeit als eines der zuverlässigsten und meist verbreitetsten Verfahren heraus. Aufbauend auf diesen Vorarbeiten wird die Echtzeit-Stimmungsanaylse für deutschsprachige Textnachrichten in den nun folgenden Kapiteln näher betrachtet.

Kapitel 4

Konzeptentwicklung

Das nachfolgend vorgestellte Konzept skizziert ein System, das als Software implementiert wird und den Nutzer:innen assistiert, indem gerade verfasste Nachrichten in Hinblick auf deren Stimmung in Echtzeit analysiert werden. Daher wird dieser Assistent in den folgenden Kapiteln als „Live-Sentiment-Analyse" bezeichnet. Negative oder positive Nachrichten werden automatisch erkannt und geben Rückschlüsse darauf, wie diese beim Empfänger ankommen. Das Bewusstsein für Stimmung soll so beim Verfasser:in geschärft werden. Der Software-Assistent wird aufbauend auf dem in diesem Kapitel ausgeführten Konzept prototypisch realisiert (Kapitel 5), um anschließend zur Evaluation des Konzeptes herangezogen zu werden (Kapitel 6).

Vor diesem Hintergrund wird das Konzept zunächst in einen visionären Zusammenhang gestellt. Im Anschluss daran wird ein Systemüberblick für das zu entwerfende Konzept erläutert. Das Konzept umfasst folgende drei wesentliche Bausteine bzw. zu bearbeitende Teilsysteme: die Erfassung der geschriebenen Nachrichten, die Stimmungsanalyse mit verschiedenen Verfahren und Metriken sowie die Sammlung, Darstellung und Auswertung der Daten. Analog dazu werden die Datenerfassung, die Datenanalyse sowie die Datensammlung und -visualisierung in diesem Kapitel beschrieben.

4.1 Vision

Eine Vision ist ein Mittel, das das Leitmotiv einer Software oder eines Produktes erläutert und aufzeigt, wie ein Problem gelöst oder wie Bedürfnissen von Kund:innen optimal entsprochen werden kann[1]. Genutzt wird eine Vision beispielsweise in der agilen Entwicklung, um ein zukünftiges Produkt eindeutig zu beschreiben[2]. In Anlehnung an diesen Ansatz wird im Folgenden eine Vision skizziert, wie der zu entwickelnde Software-Assistent zur Erkennung von Stimmung in Textnachrichten im Jahr 2030 genutzt werden könnte. Daraus werden zwei Szenarien abgeleitet, die den Bezug zu den in Abschnitt 1.2 vorgestellten Forschungsfragen herstellen und eine exemplarische Interaktion eines Benutzers mit der Live-Sentiment-Analyse skizzieren.

Vision: Live-Sentiment-Analyse im Jahr 2030

Analog zu den Rechtschreib-Assistenten von Chat-Programmen, ist es im Jahr 2030 genau so üblich, dass getätigte Texteingaben nicht nur auf Rechtschreibfehler, sondern auch auf ihre Stimmung hin untersucht werden. Eine negative Stimmung kann viel größere Auswirkungen als ein Rechtschreibfehler haben und bereits einzelne unzufriedene Mitglieder können die Stimmung in einem Team dämpfen. Daher ist es bis zum Jahr 2030 ein Standard geworden, Sentiment-Analysen in verschiedenen Chat- und Kommunikationsprogrammen zu verwenden. Bis zum Jahr 2025 haben verschiedene Personen in Entwickler-Teams diese Technik genutzt und verbessert, sodass sich diese Schritt für Schritt auch in anderen Bereichen und Branchen durchgesetzt haben. Sprachliche Vorschläge, die von den Stimmungsassistenten unterbreitet werden, ermöglichen in verschiedenen Sprachen eine positivere Kommunikation. Aber auch die Erkennung von Stimmung in Teams ermöglicht es den Projektleiter:innen und der Geschäftsführung, Maßnahmen abzuleiten, um die Stimmung in einem Team zu verbessern. Dieser Durchbruch von Stimmungsassistenten hat dazu geführt, dass sich die Diskussionskultur weiterentwickelt hat. Eine positive Ausdrucksweise ist der neue Standard in E-Mails und Chat-Verläufen von Microsoft Teams, Telegram und ähnlicher Dienste. Die Unzufriedenheit einzelner Personen und ganzer Teams ist aufgrund dessen zurückgegangen.

[1] https://www.it-agile.de/agiles-wissen/agiles-produktmanagement/was-ist-eine-produktvision/ 04.04.2022

[2] https://www.agile-academy.com/de/product-owner/ein-produkt-entwickeln-das-die-nutzer-wollen-mit-dem-vision-board-von-der-idee-bis-zum-backlog/ 04.04.2022

4.1.1 Szenarien

Szenarien werden genutzt, um Interaktionsfolgen zu dokumentieren, die zu einem Mehrwert für die Benutzer:innen (oder einen anderen Stakeholder) führen [23]. Diese dienen dem Verbildlichen einer Anwendung und der Beschreibung des erwarteten Verhaltens eines Systems. Das nachfolgend skizzierte, erste Szenario bezieht sich auf die im Rahmen dieser Arbeit formulierten Forschungsfrage RQ1 und analog das zweite Szenario auf RQ2.

Szenario 1

Der Qualitätsbeauftragte eines relativ jungen Entwicklerteams sieht, dass der zu testende Quellcode auf dem Kundensystem überhaupt nicht läuft und ständig Fehler wirft. Nach einer Stunde erfolglosen Testens schreibt er per Microsoft-Teams die Entwickler:innen an. Bei der Eingabe „Das Ding läuft überhaupt nicht! Das muss sofort überarbeitet werden!" gibt ihm der Assistent für die Stimmungsdetektion den Hinweis aus, dass die beiden Sätze sehr negativ bei den Empfänger:innen ankommen werden. Die Anzeige der Textstimmung geht in den roten, negativen Bereich und stimmt den Verfasser nachdenklich. Nach kurzem Überlegen beschließt er, die Sätze zu überarbeiten. Er schreibt daraufhin „Das hat heute leider nicht wie gewollt funktioniert. Das müssen wir nächste Woche nochmal besser machen.". Die Anzeige geht nun in den positiven Bereich und der Qualitätsbeauftragte sendet die Nachricht an seine Kolleg:innen.

Szenario 2

Die Projektleiterin eines Teams mit acht Entwicklern:innen setzt ein Online-Meeting an. Das Meeting verläuft leider unter Zeitdruck und nicht alle notwendigen Informationen und Arbeitsschritte werden thematisiert. Da sie auch keine Zeit für Rückfragen hat, verlässt sie das Meeting und die Stimmung im Team ist etwas angespannt. Um ihrem Unmut etwas Luft zu verschaffen, tauschen sich die Entwickler:innen anschließend mit direkten Nachrichten aus. Im Hintergrund läuft nun die Live-Sentiment-Analyse der geschriebenen Nachrichten. Aus verschiedenen Analysen, die die Entwickler:innen live sehen können, wird ein Durchschnitt berechnet. Zwischen 0 und 1 wäre er positiv, wie die Stimmung der verfassten Nachrichten. Aber nach dem Meeting schwankt der Durchschnitt immer zwischen -0,1 und -0,3 und verdeutlicht die negative Stimmung. Ohne persönliche Details werden nur die Durchschnitte der Entwickler:innen anonym ausgewertet. Die Projektleiterin sieht die Stimmung in einem kleinen Fenster auf ihrem Rechner und bemerkt, dass eine Stunden nach dem Meeting die Stimmung ziemlich gekippt ist. Bevor es nun eskaliert, setzt die Projektleiterin ein neues Online-Meeting an, in dem sie sich die Zeit nimmt, Fragen zu beantworten und Unklarheiten zu beseitigen.

4.2 Kontext

Das zu entwerfende Konzept beschreibt ein theoretisches System, das als Bauplan für die Implementierung der „Live-Sentiment-Analyse" in Form einer Software dient. Damit das zu erstellende System die Funktionen, die in den oben ausgeführten Szenarien beschrieben sind, erfüllen kann, müssen verschiedene Rahmenbedingungen gegeben sein. Da die Entwicklung des Software-Assistenten bzw. der „Live-Semtiment-Analyse" im Rahmen dieser Arbeit für eine Proband:innen-Gruppe erfolgt, die in der IT-Branche in unterschiedlichen Funktionen tätig sind, basieren in der ersten Entwicklungsphase verschiedene Aspekte und Anforderungen auf Annahmen. In der sich anschließenden Vorstudie werden diese Annahmen auf Grundlage des Feedbacks von Expert:innen überprüft und ggf. angepasst.

Ziel des Konzeptes

Das zu erstellende Konzept dient als Bauplan für die Entwicklung eines Software-Assistenten, der die Stimmung in Textnachrichten detektieren und visualisieren kann. Dabei soll nicht nur gezeigt werden, ob ein Satz positiv oder negativ ist, sondern auch, bis zu welchem Grad und was ihn positiv oder negativ macht. Der Benutzer muss dabei vollen Zugriff auf seine Daten haben, das System jederzeit starten und stoppen können sowie die Stimmungsanalysen grafisch aufgearbeitet einsehen können.

Einsatzumgebung

Die erwähnte Datensicherheit ist bei diesem Konzept ein zentrales Thema, da ein übliches Vorgehen bei den bestehenden Sentiment-Analysen darauf basiert, persönliche Daten über einen längeren Zeitraum zu sammeln und diese im Anschluss auf einem zentralen System auszuwerten. Um die privaten Daten nicht zentral sammeln zu müssen, sollten die Stimmungsanalysen auch lokal auf jedem Endgerät laufen können, sodass jeder Nutzer seine Daten und Analysen selbst verwalten kann. Dementsprechend muss das System kompakten Speicherplatz nutzen und wenig Rechenleistung vom Endgerät in Anspruch nehmen.

Bausteine und Prozessablauf

Das Konzept zur „Live-Sentiment-Analyse" muss dabei die folgenden drei Schritte beinhalten:

1. **Datenerfassung**: Die Erfassung der geschriebenen Nachrichten
2. **Datenanalyse**: Die Stimmungsanalyse mit verschiedenen Verfahren
3. **Datenaufbereitung**: Die Sammlung, Darstellung und Auswertung der Daten

Für den Austausch der Daten ist es sinnvoll, einen einheitlichen Dienst zu nutzen. Um auch zukünftig eine hohe Erweiterbarkeit des Systems zu gewährleisten, wird auf den in den Grundlagen vorgestellte MQTT-Kurznachrichten-Austausch zurückgegriffen.

Benutzerakzeptanz

Eines der zentralen Aspekte bei dem Entwurf eines Software-Assistenten ist die spätere Akzeptanz der Benutzer:innen. Die Zielgruppe des Systems sind in erster Linie Software-Entwickler:innen, aber auch Kolleg:innen aus der technischen Redaktion oder dem Qualitätsmanagement, die Teil eines Entwickler-Teams sind. Darüber hinaus haben auch bzw. insbesondere Projektleiter:innen, die ein solches System nutzen, Interesse an einem Überblick über die Stimmung von einzelnen Personen oder dem ganzen Team, und die es infolgedessen zu betrachten gilt. Die Privatsphäre und die Datensicherheit sind hier elementar, damit einzelne Personen nicht bloßgestellt oder ausgegrenzt werden. Die Nutzung des Systems ist freiwillig und darf folglich weder aufgezwungen noch aufgedrängt werden. Zudem muss berücksichtigt werden, dass der Assistent nicht mehr benutzt wird, sobald das Systeme mehrmals Stimmungen falsch detektiert hat.

Genauigkeit der Stimmungsanalysen und Fehldetektionen

Eine Software verliert massiv an Glaubwürdigkeit, wenn sie ihre Aufgabe nicht erfüllt. In Bezug auf einen Assistenten zur Stimmungsanalyse bedeutet dies, dass er nicht mehr verwendet wird, wenn er mehrfach die Stimmung von den Benutzer:innen falsch interpretiert hat. In diesem Fall wäre eine mögliche Lösung, verschiedene Verfahren zu kombinieren, damit mehrere Aspekte berücksichtigt werden. Ebenfalls sollte die Modularität und die Erweiterbarkeit berücksichtigt werden, um Analysen eventuell zu tauschen (beispielsweise bei verschiedenen Sprachen) und um Analysen auch zukünftig zu verbessern.

Systemkomponenten

Das System soll als kompakter Stimmungsassistent auf einem lokalen Rechner laufen. Dabei sollen keine externen Dienste genutzt werden. Die Stimmungsanalyse eines Entwicklers läuft auch nur auf seinem System.

Machbarkeitsanalyse

Durch das Herunterbrechen des Gesamtsystems auf Teilsysteme, die jeweils analysiert werden, lässt sich die Machbarkeit des Gesamtsystems einschätzen. Bei der Erfassung der Daten kann hierbei auf vorhandene Techniken zurückgegriffen werden, wie der Keylogger, der die Tastatureingaben der Benutzer protokollieren kann. Mit Hilfe der in den Grundlagen vorgestellten Verfahren zur Stimmungsdetektion und ihrer bereits erforschten Genauigkeiten und Einsatzgebiete lässt sich auch das Teilsystem „Datenanalyse" betrachten. Hier können durch eine hohe Modularität des Systems verschiedene, etablierte Verfahren implementiert werden. Auch bei der Visualisierung der Daten kann auf vorhandene Oberflächen zurückgegriffen werden. Zwar gibt es wenige, die eine Stimmungsanalyse ermöglichen, dafür aber funktional ähnliche Elemente zur Verfügung stellen. Dazu zählen Dashboards, die verschiedene Sensorwerte

aufbereiten und grafisch darstellen. Auch wenn es für die im Rahmen dieser Arbeit zu entwickelnde präventive Stimmungsanalyse keine Software-Bausteine gibt, die direkt verwendet werden können, gibt es zumindest ähnliche Systeme, die abgewandelt werden können. Auf dieser Grundlage lässt sich abschließend festhalten, dass das zu entwerfende Konzept durchaus in Software umsetzbar ist.

4.3 Datenerfassung

Um Textnachrichten analysieren zu können, muss der Text oder der Entwurf einer Nachricht in Echtzeit erfasst werden. Hierbei kann entweder für einen bestimmten Messenger-Dienst ein Plug-in geschrieben werden oder ein Keylogger als Hintergrund-Prozess zeichnet die Tastatureingaben parallel zum Chat mit auf. Ein **Plug-in** ist eine Software-Erweiterung eines bestehenden Hauptsystems. Für die Erfassung von Nachrichten kann das beispielsweise das weit verbreitete Microsoft-Teams sein. Ein entwickeltes Plug-in kann dann an einem Rechner lokal installiert und an den Messager-Dienst „eingesteckt" werden.

Um unabhängig von einem bestimmen Programm Textnachrichten zu erfassen, kann ein **Keylogger** genutzt werden. Dieser protokolliert die über die Tastatur eingegebenen Buchstaben und Zeichen, kann aber Interaktionen mit der Maus und damit einhergehende Manipulationen nicht erfassen. Auch wenn die korrekte Reihenfolge von Wörtern bei bestimmten Stimmungsanalysen nicht immer eine Rolle spielen, können Abweichungen entstehen, da Teile vom Satz anders oder nicht richtig interpretiert werden können.

4.4 Datenanalyse

In der aktuellen Sentiment- bzw. Stimmungsanalyse werden Daten über Wochen und Monate erhoben, um diese im Nachhinein auf positive oder negative Stimmung hin zu untersuchen. Diese retrospektive Analyse an einer zentralen Stelle ermöglicht einen guten Überblick über die Kommunikation und die Art der Kommunizierenden. Diese Analyse ist aber immer nur nachgeschaltet, sodass Ergebnisse die Stimmung betreffend nicht für die in Echtzeit stattfindende Kommunikation vorliegen und genutzt werden können. Zudem müssen Analysten dann private Chats und Daten zentral sammeln. Für eine Echtzeit-Stimmungsanalyse sollen bereits bekannte, aber schnelle Verfahren genutzt werden, um die erfassten Daten lokal auf einem Endgerät analysieren zu können.

Specht [19] hat in seiner Arbeit gezeigt, dass die Kombination mehrerer Analysen eine höhere Genauigkeit mit sich bringt. Ein zentrales Ergebnis seiner Arbeit ist, dass eine subjektive Analyse mit SentiStrength ein Genauigkeitsmaß von ungefähr

80% generiert. Mithilfe von objektiven Metriken wurde das Genauigkeitsmaß weiter verbessert. Dazu wurden der Informationsgehalt, die durchschnittliche Anzahl von Wörtern und die Analyse von Emoticons ergänzt. Die Vorhersage konnte somit auf ein Genauigkeitsmaß von etwa 86% erhöht werden. Obwohl Emoticons in den Datensätzen von Stack Overflow in nur sehr geringem Maße vorkamen, ist davon auszugehen, dass sie in den Textnachrichten zwischen befreundeten Entwicklern deutlich häufiger ein kommunikatives Ausdrucksmittel sind.
In Anlehnung an die Erfahrungen von Specht wird auch in dieser Arbeit eine Kombination von Verfahren zur Stimmungsanalyse genutzt. Hierzu wurde zunächst eine Vorselektion vorgenommen, um darauf aufbauend Verfahren für die Nutzung in der Software festzulegen.

4.4.1 Vorselektion von Verfahren zur Stimmungsanalyse

Um abschätzen zu können, welche Verfahren und Metriken für eine Live-Sentiment-Analyse in Frage kommen, wurden deren Vor- und Nachteile gegenüber gestellt (s. Tabelle 4.1. Außerdem wurde die Expertise von einem Experten im Bereich der Sentiment-Analyse mittels Interview in den Selektionsprozess eingebunden. Die Vorselektion erfolgte unter Berücksichtigung nachfolgend aufgeführter Aspekte.

- Datensicherheit
  (Sind unverschlüsselte, private Chats an einen öffentlichen Server zu senden?)
- Umsetzbarkeit und Wiederverwendung von Baugruppen
  (Was kann wiederverwendet oder schnell implementiert werden?)
- Zeit für Analyseergebnisse
  (Die Interpretation eines Satzes sollte in weniger als einer Sekunde erfolgen.)
- Größe des Analyseverfahrens
  (Eine Analyse sollte nicht mehr als 1 GB Speicherplatz beanspruchen.)
- Mehrwert für die Gesamtanalyse
  (Verschiedene Verfahren können sich in Bezug auf die Funktionen „überlappen".)

Die Priorisierung in der Tabelle 4.1 wurde für den Entscheidungsprozess hinzugefügt, um deutlich zu machen, welche Verfahren eher oder sogar gar nicht in Frage kommen.

4.4.2 Entscheidungen

Auf Basis der Vorselektion wurden die verschiedenen Verfahren zur Stimmungsanalyse abschließend bewertet. Auf Grundlage der von Experten in dem oben erwähnten Interview geäußerten Erfahrungen wurde die Datenanalyse auf dem SentiStrength-Verfahren aufgebaut und mit weiteren, objektiveren Analysen verfeinert. Neben der Möglichkeit, ein deutsches Lexikon zu nutzen, haben auch Novielli et al. gezeigt,

Prio	Verfahren/Metrik	Vorteile	Nachteile	Sprache
1	Emoticons	Einfach in verschiedenen Programmiersprachen zu implementieren		De, En
2	Anzahl Großbuchstaben	Einfach zu implementieren	Eigennamen	De
3	bestimme Satzzeichen	Auch Verhältnis Satzzeichen-Buchstaben, oder Satzzeichenfolgen		De, En
4	SentiStrenghtDE	JAR-Anwendung vorhanden		De
5	SentiStrenghtSE	Speziell für Software-Entwicklung	Nur Englisch	En
6	Senti4SD	Wenn SentiStrenghtSE nur auf englisch verfügbar ist	Nur Englisch	En
7	Interjektionen	Einfach (Empfindungswörter wie Oh, Ihhh, pfui, hä?)	Muss selber erstellt werden	De
8	Anzahl Wörter	Auf einen sehr sehr langen Text, wird eine ausführliche Antwort erwartet		De
9	Anzahl Zeichen	Extrem umschwängliche und ausgiebige Formulierungen		De
10	Durchschn. Wortlänge	Kurz und Knapp oder möglichst viele Umformulierungen		De
11	Informationsgehalt	Sehr interessant	Analyse dauert 5 - 10 sek	De
12	NLP	Neuro-Linguistisches Programmieren (Denken, Fühlen mittels Sprache)	Technisch sehr aufwändig	De
13	NLTK	Natural Language Toolkit	Lange Analysezeit	En

Tabelle 4.1: Verfahren und Metriken zur Stimmungsanalyse im Vergleich

dass ohne weiteres Training bei einem „Off-the-shelf"- Test SentiStrength eines der zuverlässigsten Verfahren ist [18]. Obaidi und Klünder haben darüber hinaus ermittelt, dass SentiStrength weit verbreitet ist [16]. Da in Software-Entwicklungsteams auch englische Begriffe genutzt werden, sollte neben dem deutschen SentiStrength-DE auch das englischsprachige unterstützt werden.
Eine weitere Entscheidung viel auf die Analyse von Emoticons als objektive Metrik. Die Zeichenfolgen von z.B. fröhlichen, nachdenklichen oder irritierten Gesichtern wird heutzutage gerade von jüngeren Personen gerne eingesetzt und lässt die Stimmung beim Verfassen von Textnachrichten gut interpretieren.
Für eine erste Implementierung des Konzeptes in Form eines Software-Assistenten wurden die folgenden drei Verfahren in Betracht gezogen:

- SentiStrength-DE
- SentiStrength-EN
- Emoticons-Analyse

4.4.3 Standardisierung der Analysen

Da für die Echtzeit-Analyse von Stimmungen in Textnachrichten verschiedene Verfahren genutzt werden, müssen diese standardisiert werden. Neben der Visualisierung der verschiedenen Stimmungsanalysen in einer Oberfläche, muss auch die Berechnung eines gemeinsamen Durchschnittes bei dieser Standardisierung berücksichtigt werden. Um die Stimmung in Werten auszudrücken bietet es sich an, diese in einem Intervall zwischen **-1 bis 1** zu begrenzen. Dabei spiegeln negative Werte eine negative Stimmung wieder. Die Null ist auch in der Stimmung neutral und positive Werte bis eins stehen für eine positive Stimmung. Diese Form ist somit als Bild ähnlicher als ein Intervall von 1 bis 5.
Für mehr Transparenz der Analysen gegenüber dem Benutzer ist es zudem möglich, auch eine Begründung darzustellen. Diese soll verdeutlichen, welcher Teil eines Satzes der Grund für die entsprechende positive oder negative Interpretation war. Eine solche Begründung kann dann beispielsweise direkt hinter einem positiv oder negativ polarisiertem Wort stehen: „Das war super(+3)! Gute(+2) Arbeit. Dumme(-1) Sache. Scheiß(-3) Fehler(-1)". Aber auch ein „:-)" kann als Begründung für einen positiv interpretierten Satz angeführt werden.

4.4.4 Grenzen der vorhandenen Analysen

Bei der Recherche zu den Verfahren für die Stimmungsanalyse im Rahmen dieser Arbeit wurde deutlich, dass einige Grenzen schnell erreicht sein werden. Neben primär englischsprachigen gibt es weniger deutschsprachige Verfahren. Darüber hinaus werden komplexe Zusammenhänge ebenso wenig erkannt wie Ironie oder Sarkasmus. Dessen ungeachtet können Genauigkeiten zwischen 70% bis 85% als Grundlage dienen, um in den meisten Fällen eine positive oder negative Stimmung richtig zu erkennen.

4.5 Datensammlung und -visualisierung

Für das Sammeln und Visualisieren der Ergebnisse von Stimmungsanalysen stehen zwei Optionen zur Verfügung. Die erste Option ist die Darstellung in einem vorhandenen Programm mit Hilfe einer Plug-in-Lösung oder in einem externen Programm, um unabhängig von einem vorhandenen Dienst zu sein. Wie bereits bei der Datenerfassung und in den Grundlagen ausgeführt, kann ein Plug-in eine sinnvolle Erweiterung eines vorhandenen Systems sein. Für gängige Messenger-Dienste wie Microsoft Teams oder Telegramm gibt es Möglichkeiten, eine eigene Anwendung als Plug-in zu integrieren. Hierbei können nicht nur Daten, wie die Eingabe einer Nachricht, mit genutzt werden, sondern es kann auch die Benutzeroberfläche erweitert werden.
Die zweite Möglichkeit besteht in der Neuentwicklung einer Benutzeroberfläche unabhängig von einem spezifischen Nachrichtendienst. Der wesentliche Vorteil besteht dabei darin, dass in mehreren Chats oder E-Mail-Programmen eine Live-Analyse durchgeführt werden kann und diese programmübergreifend visualisiert wird. Es muss nicht für jeden Messenger-Dienst ein eigenes Plug-in geschrieben werden, denn es unterstützt so verschiedene, bereits vorhandene Programme. Eine zu entwickelnde Oberfläche kann mit verschiedenen Programmiersprachen, wie Java oder C# realisiert werden. Elementar ist nur die Anbindung eines MQTT-Clients, um über den Datenaustausch zu verwalten.

Das Diagramm in Abbildung 4.1 gibt einen Überblick über die in diesem Kapitel vorgestellten Bausteine.

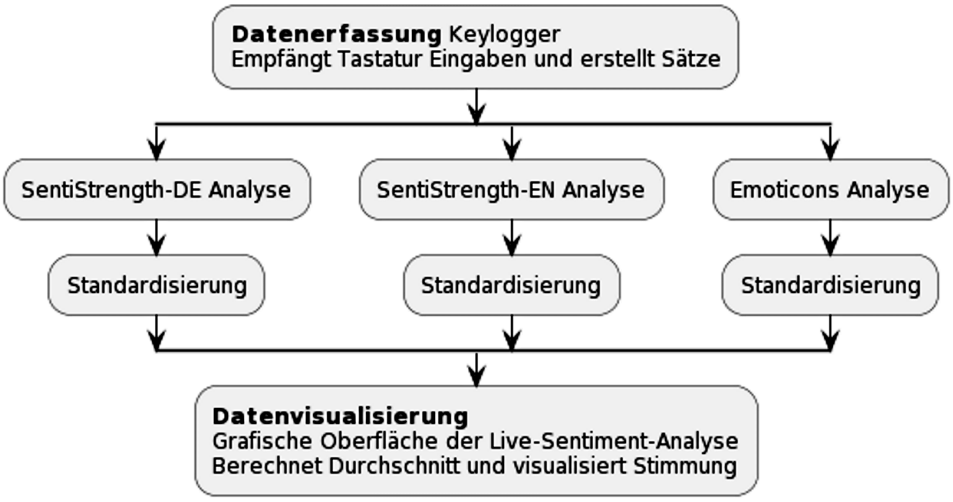

Abbildung 4.1: Bausteinansicht eines „Live-Sentiment-Analyse”-Systems (Aktivitätsdiagramm erstellt mit www.plantuml.com)

Kapitel 5

Konzeptimplementierung

In diesem Kapitel wird eine mögliche Umsetzung des vorgestellten Konzeptes in einer Software beschrieben. Die Software, die dem Benutzer assistieren soll und die Echtzeit-Analyse von Stimmung in schriftlichen Nachrichten durchführt, wird dabei als „Live-Sentiment-Analyse" bezeichnet. Dafür wird zunächst der theoretische Entwurf des Software-Entwicklung ausführlich vorgestellt. Dabei werden Rahmenbedingungen, funktionale Anforderungen, wie der reibungslose Datenaustausch, und nicht-funktionale Anforderungen, wie die Privatsphäre, erfasst und dokumentiert (Absatz 5.1). Auf Grundlage des Entwurfs wird anschließend die Realisierung des Software-Prototypen ausgeführt (Absatz 5.2). Dabei werden verschiedene technische, für die Implementierung der Software relevante Merkmale erläutert.

5.1 Software-Entwurf

Die Erfassung, Dokumentation und Verwaltung von Anforderungen ist ein wichtiger Baustein im Systems-Engineering-Prozess [24]. Gut erfasste funktionale und nichtfunktionale Anforderungen sind von elementarer Bedeutung. Da ein realer Kunde diese Entwicklung nicht begleitet, wurden im ersten Schritt die wichtigsten Rahmenbedingungen des Konzeptes festgehalten. Im zweiten Schritt werden diese Annahmen in Software-Anforderungen dokumentiert.

Obwohl die Prozesse in einer „Live-Sentiment-Analyse" in „Echtzeit" arbeiten sollen, gibt es Echtzeit von technischer Seite nicht. Auch wenn Daten parallel in verschiedenen Verfahren analysiert werden, gibt es bei rechnergestützen Systemen keine Echtzeit, wie es der Begriff erwarten lässt. Von technischer Seite gibt es einen definierten Zeitrahmen, in der ein System eine Antwort geben muss. Für die Live-Sentiment-Analyse gilt hier eine Antwortzeit von unter einer Sekunde.

5.1.1 Software-Anforderungen

Die hier vorgestellten Anforderungen wurden aus dem Abschnitt 4.2 abgeleitet.Die Generierung von Anforderungen geschieht mittels Schablonen. Diese dient als Bauplan für die Struktur der zu erstellenden Anforderungssätze. Rupp und Joppich [25] bieten unterschiedliche Anforderungsschablonen für die natürlichsprachlichen Anforderungen an. Neben Anforderungsschablonen für die funktionale oder nichtfunktionale Anforderungen (Eigenschaften des Systems oder der Umgebung) gibt es auch Vorlagen für Bedingungssätze [26, 27]. Unter dem Begriff MASTeR (Mustergültige Anforderungen- die SOPHIST Templates für Requirerments) werden diese gesamten Anforderungsschablonen bezeichnet [28].

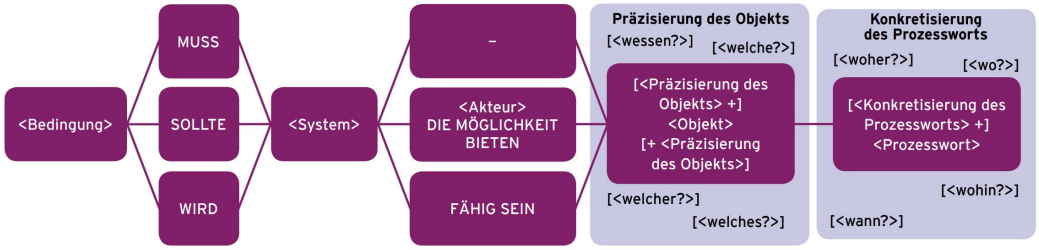

Abbildung 5.1: Detaillierter FunktionsMASTER mit Bedingung - Anforderungsschablone [25]

Die folgenden, an das System gestellten Anforderungen wurden mit Hilfe der Anforderungsschablone nach Rupp erstellt. Die <Bedingung> ist dabei durch die Pfeile als optionaler Baustein in der Schablone gekennzeichnet.

1. Um die privaten Daten nicht zentral sammeln zu müssen, muss die Stimmungsanalyse lokal auf einem Endgerät laufen

2. Nach dem Start des Systems durch den Benutzer muss der Software-Assistent die Stimmungsanalyse durchführen.

3. Der Assistent muss private Textnachrichten lokal nur auf dem Gerät des Benutzers auf Stimmung untersuchen.

4. Während der Darstellung der Ergebnisse wird der Assistent kurzfristig persönliche Daten zwischenspeichern bzw. buffern.

5. Während der Stimmungsanalyse von Texten sollte der Assistent dabei eine bestmögliche Genauigkeit anstreben.

6. Beim Beenden muss der Assistent alle persönlichen Daten löschen.

7. Nach einer Texteingabe muss der Assistent ein Ergebnis innerhalb einer Sekunde dem Benutzer anzeigen.

8. Unabhängig vom Betrieb sollte der Assistent von Entwicklern verbessert werden können.

9. Unabhängig vom Betrieb sollte der Assistent die Möglichkeit bieten einzelne (Analyse) Module auszutauschen.

5.1.2 Architektur-Entscheidungen

Die Software Architektur beschreibt die grundlegenden Module und Komponenten und deren Interaktion innerhalb eines Systems. Robert C. Martin nutzt ihn als Metapher, ähnlich dem Bau eines neuen Gebäudes, obwohl der Vergleich mit der heutigen Software-Entwicklung mit schnellen Release-Zeiten und agilem Projektmanagment weniger passt [29]. Eine entsprechende Vorbereitung vor dem eigentlichen Programmieren ist aber für den den Entwicklungsprozess von Vorteil. Die Nutzung von Design-Pattern, um Modularität und Erweiterbarkeit zukünftig zu gewährleisten, sollen diesen Prozess genauso begleiten wie das Erstellen von Diagrammen und Skizzen, um Abläufe zu visualisieren und zu optimieren. Abgeleitet von dem vorgestellten Konzept, wurde eine Architektur entworfen, die wiederum als Bauplan für die eigentliche Implementierung verstanden wird. Dabei müssen einige Entscheidungen begründet getroffen werden, um eine bestmögliche Realisierung zu gewährleisten.

Eine grundlegende Entscheidung ist die Wahl der Kommunikationsschnittstellen. Diese verbindet die einzelnen Baugruppen bzw. Software-Module und soll zugleich für Veränderbarkeit und Erweiterbarkeit stehen. Das Festlegen auf MQTT (Message Queuing Telemetry Transport) zum Datenaustausch ermöglicht mit der Verwendung bestehender MQTT-Clients die Entwicklung auf Basis verschiedener Programmiersprachen. Der Austausch der Daten ist somit über ein festes, aber einfaches Mittel festgelegt, sodass jedes Modul individuell betrachtet werden kann. So ist es möglich für gewisse Programme ein Skriptsprache zu nutzen, während eine Oberfläche mit einer Programmiersprache geschrieben werden kann, die für Graphical User Interface(GUI)-Entwicklungen angepasst wurde. Ein neues Modul kann so unabhängig vom restlichen System entworfen werden, da Ein- und Ausgaben über einen einheitlichen Dienst organisiert werden.

Für die zu entwerfende Software-Architektur wurde, die in dem Konzept vorgestellten Teilsysteme von Datenerfassung, -analyse und -visualisierung in entsprechende Module übernommen. Die daraus abgeleiteten Module können mit Hilfe der Standardisieren MQTT Schnittstelle individuell betrachtet werden und bei Bedarf ausgetauscht oder einzeln erweitert werden. Zudem können einzelne Analysen im Prozess unabhängig verbessert werden, um zukünftig die Gesamtanalyse weiter zu optimieren.

Ein exemplarischer Ablauf für ein mögliches „Live-Sentiment-Analyse”-System ist im folgenden Diagramm (s. Abb. 5.2) skizziert.

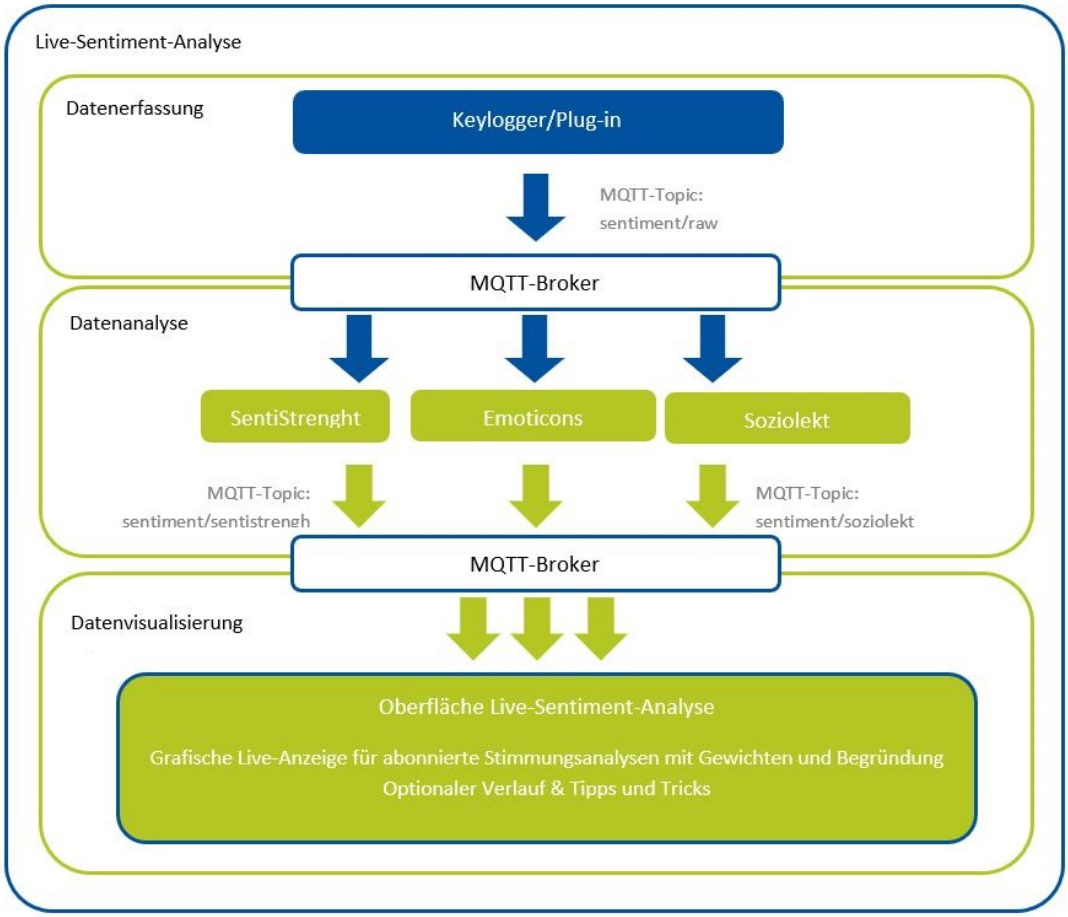

Abbildung 5.2: Überblick über die Bausteine und den Prozessablauf eines „Live-Sentiment-Analyse”-Systems

Der zeitliche Verlauf von Daten wird in Abbildung 5.3 in einem UML-Sequenzdiagramm visualisiert. Obwohl die Daten über den MQTT-Broker an alle Analysen gleichzeitig verteilt werden, brauchen unterschiedliche Analyseverfahren bei verschiedenen Eingaben immer unterschiedlich lang. Daher ist das Sequenzdiagramm nicht zu 100 % genau, sondern soll eine Idee über den zeitlichen Verlauf geben.

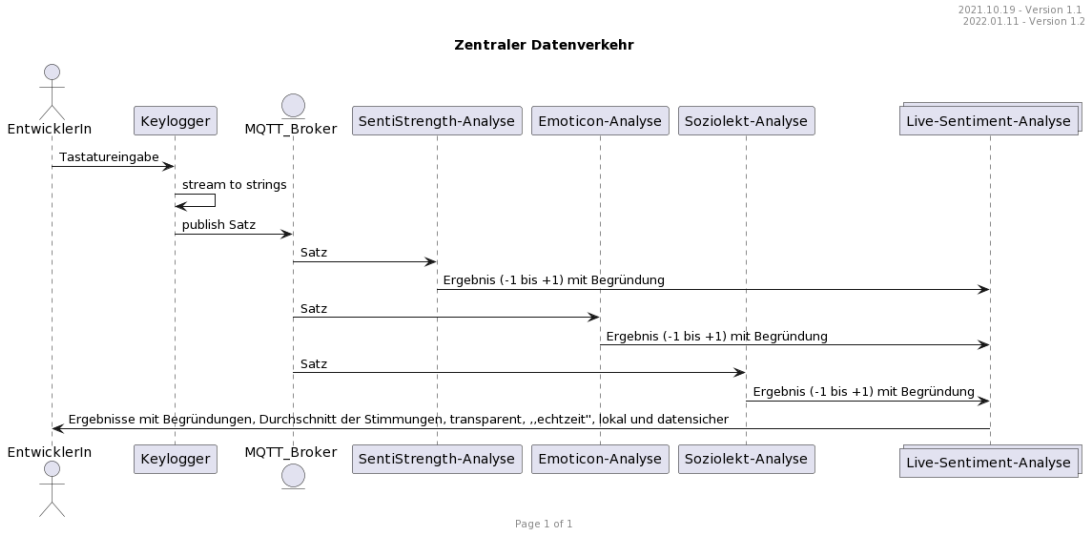

Abbildung 5.3: UML-Sequenzdiagramm Live-Sentiment-Analyse (plantuml.com)

Ein gemeinsames Klassendiagramm wird an dieser Stelle nicht aufgeführt, da jedes Modul aus einer Klasse oder nur wenigen Klassen besteht und diese nicht von einander abhängig sind. Die Schnittstellen werden über MQTT beschrieben und standardisiert. Durch die Einbindung bestehender MQTT-Clients für unterschiedliche Programmiersprachen ist die Komplexität auf ein Minimum reduziert worden, um auf dieser Grundlage das Projekt auch zukünftig weiter entwickeln zu können.

5.2 Implementierung des Software-Assistenten

Wie bereits in dem Kapitel zu dem Entwurf der Architektur erwähnt, basiert die Konzeptentwicklung auf den drei Teilsystemen Datenerfassung, -analyse und -visualisierung, die in drei verschiedenen Baugruppen bzw. Modulen realisiert und nachfolgend vorgestellt werden. Alle Baugruppen enthalten dabei einen MQTT-Client und nutzen den MQTT-Broker zur Kommunikationsverwaltung. Ein vereinfachtes Beispiel ist der Abbildung 5.4 zu entnehmen.

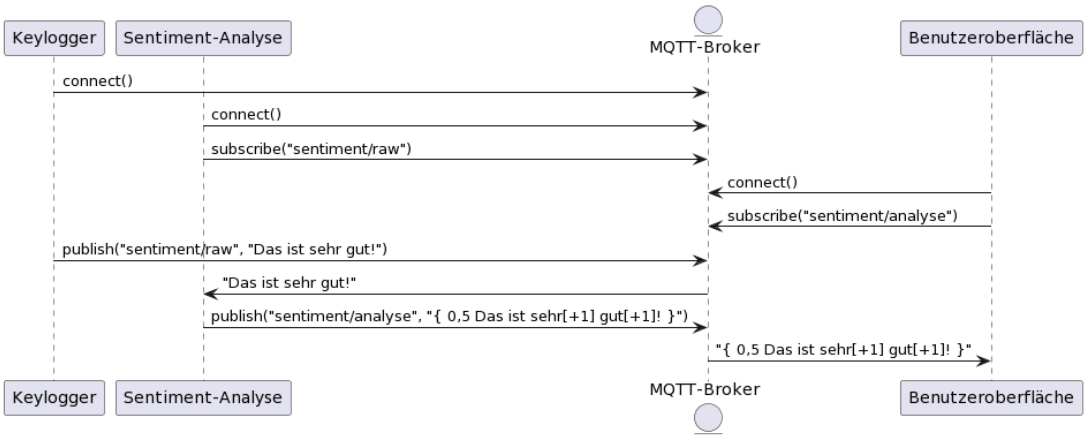

Abbildung 5.4: UML-Sequenzdiagramm zum MQTT-Datenaustausch

5.2.1 Modul zur Datenerfassung

Um die Daten für das Proof-of-Concept in einer Studie aus verschiedenen Eingabequellen zu erfassen, wird für den Software-Assistenten ein Keylogger implementiert. Dieser protokolliert die eingegebenen Textnachrichten über die Tastatur parallel zum Chat- oder E-Mail- Programm. So können die Nachrichten sowohl bei Microsoft-Teams als auch bei Outlook oder einem ähnlichen Dienst erfasst werden. Darüber hinaus können auch die Eingaben bei einem Schreibprogramm wie Microsoft Word, OpenOffice Writer oder einem simplen Texteditor erfasst werden. Über solche Texteditoren können Eingaben im Rahmen einer Studie wieder gelöscht werden und verbleiben so nicht in einem Verlauf eines externen Dienstes. Zudem ist die Nutzung des Keylogger offline möglich, wenn ein Nachrichten-Dienst keine Internetverbindung hat.
Für die Implementierung des Keyloggers wurde ein Skript für die Arbeit im Hintergrund mit der Programmiersprache Python entwickelt. Python hast sich in den vergangenen Jahren zu einem der meist genutzten Programmiersprachen entwickelt [1]. Neben der Eignung für Programmier-Einsteiger wird es verstärkt in den Forschungsbereichen Data-Analytics und Machine-Learning eingesetzt. Auch wenn ein Keylogger in den Programmiersprachen Java oder C ähnlich gut zu realisieren ist, wurde für die Programmierung des Software-Assistenten Python gewählt, damit auch zukünftig an diesem System weiter gearbeitet werden kann.
Von dem Python-Skript werden die einzelnen Zeichen der Tastatureingabe parallel empfangen und hintereinander zu einem Satz aufgebaut. Auch Sonderzeichen werden in diesen Satz eingebaut. Mit der Betätigung der Enter-Taste wird der so entstehende Satz als MQTT-Paket verpackt und an die einzelnen Analysen weiter gesendet. Der MQTT-Client im Skript baut dafür eine Verbindung zum MQTT-Broker auf, der den

[1] https://www.it-markt.ch/news/2021-01-07/python-ist-die-programmiersprache-des-jahres-2020

Austausch auf verschiedenen Kanälen (MQTT-Topics) verwaltet.

Auszug aus dem Python-Quellcode:

```
1 client.connect(''localhost'', 1883, 60)
2 client.publish(''sentiment/raw'', string_to_analyse , qos=2)
```

Um die Daten auf einem Rechner lokal zu verwalten, ist im Quellcode keine IP-Adresse eines Netzwerkgerätes angegeben, sondern ein Verweis auf den "localhost", der den MQTT-Broker auf dem eigenen Rechner verwaltet. Der Port „1883" ist der für MQTT spezifische Port und die „60" beschreibt die „Keep alive time" in Sekunden. Der Publish-Befehl führt dann das veröffentlichen des Satzes an die Analysen aus. Um nun Stimmungsanalysen hinzuzufügen oder auszutauschen, müssen die Analysen das Topic „sentiment/raw" abonnieren, um die Daten zu erhalten. Das Veröffentlichen mit Quality-of-Service (QoS) mit dem Wert „2" dient dazu, dass der MQTT-Broker als Nachrichtenvermittler dafür sorgt, dass jede Analyse diesen Satz genau einmal erhält. Auch wenn es Fehler bei der Übertragung gibt, garantiert der Service, dass die Nachricht zugestellt wird.

5.2.2 Module zur Stimmungsanalyse

Wie in der Konzeptentwicklung bereits gezeigt, ist die Realisierung des Software-Assistenten mit verschiedenen Stimmungsanalyse-Verfahren eine sinnvoller Schritt zu einer genaueren Analyse. Diese Kombination stellt allerdings die Software-Entwicklung vor einige Anforderungen. Dazu gehört die Verteilung der Daten und das anschließende Sammeln der Ergebnisse und die damit einhergehende Synchronisierung. Weiterhin ist es notwendig, die Struktur der Software so modular zu gestalten, dass einzelne Analysen einfach verbessert, ausgetauscht oder gar deaktiviert werden können.
In Anlehnung an den Ansatz bei der Datenerfassung wurden auch die Datenanalysen als Python-Skript implementiert. Der verbaute MQTT-Client verbindet sich ebenfalls zum MQTT-Broker auf dem „localhost". Nach dem Verbindungsaufbau wird das Topic „sentiment/raw" abonniert. Jedes Analyseverfahren hat dabei ein eigenes Skript und einen eigenen MQTT-Client, damit die verschiedenen Verfahren nicht sequenziell sondern parallel berechnet werden können.
Nach dem Eingehen einer Nachricht werden dann mit verschiedenen Verfahren die Sätze analysiert. Je nach Verfahren gibt es im Anschluss eine individuelle Standardisierung, um die Ergebnisse einheitlich weiterleiten zu können. Jede der Datenanalysen sendet ihre Ergebnisse auf ein bestimmtes MQTT-Topic. Dieses ermöglicht es, dass eine empfange Benutzeroberfläche wiederum Topics abonnieren oder deabonnieren kann. Wenn eine Datenanalyse für gewisse Aufgaben falsche Ergebnisse liefert (z.B. bei einer anderen Sprache) oder für einen bestimmten Benutzer nicht benötigt wird, kann ein Topic deabonniert werden, um diese Analyse nicht in eine Gesamtbewertung einzuberechnen.

Genutzte Analyseverfahren

Auf Grundlage der in der Konzeptentwicklung 4.4.1 untersuchten Verfahren, wurden für die erste Version des Software-Assistenten die folgenden drei Verfahren implementiert:

Datenanalysen mit SentiStrength

Um SentiStrength als vorhandenes Verfahren zur Stimmungsanalyse zu nutzen, wurde ein Python-Skript entwickelt, das die Daten vom Keylogger-Modul empfängt und aufbereitet. Im Anschluss werden die Daten an SentiStrength übergeben. Nach der Analyse von SentiStrength werden deren Ausgaben vom Skript interpretiert und als standardisiertes Paket verpackt. Per MQTT werden die Daten mittels MQTT-Broker an das Modul zur Auswertung und grafischen Anzeige gesendet.
Da die im Rahmen dieser Arbeit durchgeführte Sentiment-Analysis auf Daten bzw. Texten aus dem Bereich der Software-Entwicklung stammen, wird davon ausgegangen, dass die Nachrichten bzw. deutschen Texte zahlreiche englische Fachbegriffe aufweisen. Demzufolge wird neben dem deutschsprachigen Analyse-Tool SentiStrength-DE auch das englischsprachige SentiStrength-EN implementiert. Für die Nutzung von SensiStrength als Programm mit verschiedenen Lexikas gibt es neben der Analyse auf der Internetseite [2] auch eine JAR-Datei zum Ausführen auf dem lokalen Endgerät. Zum Testen kann diese JAR beispielsweise in der Windows-Eingabeaufforderung bzw. Kommandozeile wie folgt aufgerufen werden.

```
C:\...\SentiStrengthDE>java -jar SentiStrength.jar
    sentidata C:/.../SentiStrengthDE/SentiStrength_Data_DE/
    text "SentiStrength ist toll!" explain\\
```

Um SentiStrength mit Python aufzurufen bietet sich ein Jar-Wrapper an. Dieser Startet das Java-basierte Programm und ermöglicht den Zugriff aus folgendem Python-Skript:

```
    args = ['SentiStrength.jar', 'sentidata',
            'C:/.../SentiStrengthDE/SentiStrength_Data_DE/',
            'text', message, 'explain']
    result = jarWrapper(*args)
```

In beiden Fällen wird folgendes Ergebnis geliefert:

```
3 -1 SentiStrength ist toll[2] ![+1 punctuation emphasis] [sentence: 3,-1]
```

Der Parameter „sentidata" gibt das Verzeichnis des Lexikons an. Das Stichwort „text" steht vor der zu analysierenden Nachricht. Der optionale Parameter „explain" weißt SentiStrength an, eine Begründung mit auszugeben.

[2] http://sentistrength.wlv.ac.uk/

Das verwendete deutschsprachige Lexikon SentiStrengthDE [3] wird über die offizielle Internetseite von SentiStrength empfohlen und basiert auf den Forschungen des Austrian Research Institute for Artificial Intelligence.
Die Ergebnisse der deutschsprachigen Sentiment-Analyse werden von der JAR-Anwendung als Zeichenfolge bzw. „String" an das Python-Skript zurückgegeben. Dieser String wird zerlegt, um den positiven (+1 bis +4) und negativen Stimmungswert (-1 bis -4) daraus zu generieren. Um diesen Wert zu standardisieren, wird der negative Wert auf den positiven addiert. Durch die Negation des negativen entsteht eine Subtraktion. Durch eine Division wird der Stimmungswert in den Bereich zwischen -1 und +1 genormt. Der positive Wert 3 und der Negative -1 ergeben so den Wert 2. Dieser wird durch 4 dividiert und ergibt einen positiven Wert von 0,5.
Die mit dem Parameter „explain" erwartete Begründung der Stimmungsanalyse wird zwischengespeichert und dem Paket zur Übertragung mit MQTT hinzugefügt.
Das so entstehende Paket, das an die Datensammlung und Visualisierung per MQTT weiter geleitet wird, ist wie folgt aufgebaut:

```
1 message = ''{0,5 SentiStrength ist toll[2] !
2           [+1 punctuation emphasis] [sentence: 3,-1]}''
3 TOPIC = ''sentiment/sentistrenghtde''
4 client.publish(TOPIC, str(message), qos=2)
```

Um andere Sprachen mit SentiStrength anzubieten, kann das entwickelte Skript kopiert und der Verweis auf ein anderes Lexikon angepasst werden. Abhängig von der nachfolgenden Auswertung muss das Topic, das vorgibt, welches Modul die Daten anschließend empfängt, angepasst werden. Dementsprechend wurde analog zum deutschen das englischsprachige SentiStrength implementiert. Neben dem Verweis auf das entsprechende Lexikon wurde das Ziel-Topic auf „sentiment/sentistrenghten" geändert. Eine Weiterleitung an das gleiche Topic ist technisch zwar möglich, wird aber nicht empfohlen. Verschiedene Topics ermöglichen der nachgeschalteten Gesamtanalyse das (de)abonnieren von Analysen, um nicht benötigte auszuschließen oder neue einfach einzubinden.

Datenanalyse für Emoticons

Emoticons oder die grafisch weitergeführten Smileys werden in Chats erfahrungsgemäß von jüngeren Personen bevorzugt verwendet. Dabei unterstützen diese nicht nur grafisch die emotionale Aussage des Textes, sondern werden oft auch ganz ohne Text als Antwort auf eine Nachricht verschickt. Im deutschspachigen Lexikon[4] von SentiStrength wurden eine Look-up-Table (LUT) mit Emoticons und einem Stimmungswert hinzugefügt. Diese werden mit der zur Verfügung stehenden JAR-Anwendung von SentiStrength nicht unterstützt. Um die gängigsten 115 Emoticons trotzdem mit abzudecken, wurden diese in eine separate LUT im JSON-Format kopiert.

[3] https://www.ofai.at/resources/sentistrength_de
[4] https://www.ofai.at/resources/sentistrength_de

Jede Zeichenfolge hat dabei die Werte zwischen -1 und 1 erhalten. Der folgende, kleine Auszug zeigt fünf der 115 Emoticons und stellt die verwendete Struktur da.

```
1 {
2   "lut": [
3       { "emo": ":-)",      "senti": 1       },
4       { "emo": "(^ ^)",    "senti": 1       },
5       { "emo": "=(",       "senti": -1      },
6       { "emo": "o.O",      "senti": 0       },
7       { "emo": "@}-->--",  "senti": 1       }
8   ]
9 }
```

Die von der Datenerfassung gesendeten Nachrichten werden hinsichtlich verwendeter Emoticons untersucht. Wird ein „ :-) ” gefunden, wird die Stimmung „1” als positiver Wert mit der Begründung „ :-)[1] ” weitergeleitet. Werden mehrere Emoticon verwendet, wird aus diesen der Durchschnitt berechnet. Bei dem Beispiel „Ich finde das gut (^ ^), aber das doof =(da müssen wir nochmal ran :-)” würden zwei positive Ergebnisse und ein negatives Ergebnis zu dem Wert +1 +1 -1 = 2 zusammengerechnet und durch die Anzahl der gefundenen Emoticons (hier 3) dividiert. Im Ergebnis würde das Paket übertragen werden als { 0,66 (^ ^) [1] :([-1] :-) [1] }.

Daten-Standardisierung

Durch die Kombination von vorhandenen und neuen Analysen musste ein standardisiertes Stimmungsformat gewählt werden. Die Anwendung von SentiStrength gibt zwei Werte für positiv und negativ zurück, die mit 1 bis 5 und -1 bis -5 dargestellt werden. Um nun aber auch einen Durchschnitt von den verschiedenen Analysen berechnen zu können, werden alle Stimmungswerte umgerechnet. Als gemeinsamer Standard wurde der Bereich von -1 bis 1 gewählt, wobei 0 neutral ist. Damit nicht nur zwischen positiv, neutral und negativ unterteilt werden muss, können Stimmungen mit beliebigen Kommastellen in diesem Bereich liegen. Eine Stimmung von 0,2 ist somit positiv und besser als -0.1.
Um diesen Zusammenhang einheitlich in einer Software zu nutzen, wird der Stimmungswert und die Begründung mit geschweiften Klammern in einem Paket verwendet. Wenn bei SentiStrength der Satz „Das war gut.” als Eingabe kommt, gibt die Analyse nach der Standardisierung ein „{ 0,5 Das war gut [+1] }” aus. 0,5 spiegelt die positive Stimmung wieder und die Wörter mit Bewertung werden dem Benutzer angezeigt, um zu zeigen, welcher Teil des Satzes wie interpretiert wurde.

5.2.3 Modul zur Datenvisualisierung

Um im Rahmen einer ersten Studie ein Proof-of-Concept durchzuführen, fiel die Entscheidung gegen die Darstellung der interpretierten Stimmung in einem Plug-in für Microsoft (MS) Teams. Dieses lag primär an der Tatsache, dass alle Eingaben bei

MS Teams langfristig gespeichert werden. Auch wenn zukünftig eine Live-Sentiment-Analyse die Datenerfassung, -analyse und -visualisierung auf einer Plattform wie MS Teams realisiert werden muss, ist es für diese Arbeit und die im folgenden Kapitel beschriebene Studie der sinnvollere Weg, hier eine eigene Benutzeroberfläche zu erstellen.

Wie bei grafischen Oberflächen üblich, wurde auch in diesem GUI Modul das Design Pattern „Model-View-Controller" (kurz MVC) genutzt, um Daten, grafische Darstellung und Steuerung zu trennen. Die Datenverwaltung wird asynchron über einen MQTT-Client in einen Zwischenspeicher bzw. Buffer geschrieben. Dabei werden zwar einige Datenpakete zwischengespeichert bzw. gebuffert, aber keinerlei Daten werden dokumentiert oder fest abgespeichert. Die Zahl der gebufferten Pakete ist auf zehn pro Analyseverfahren begrenzt, kann aber beliebig angepasst werden.

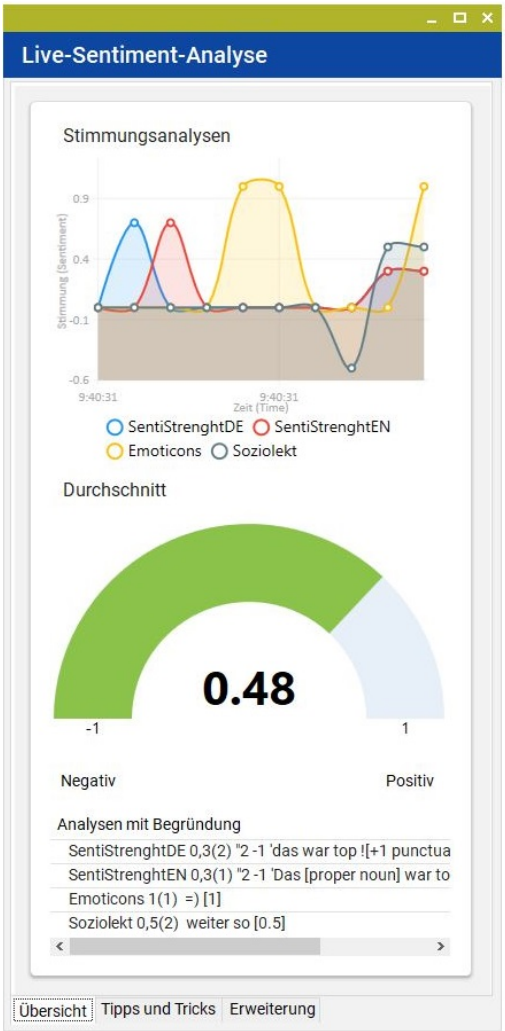

Abbildung 5.5: Screenshot des Software-Assistenten

Für die Studie wurde eine Windows Forms-Anwendung mit C# entworfen. Der links abgebildete Screenshot (s. Abb. 5.5) zeigt die Start- bzw. Hauptansicht. Im oberen Drittel ist ein Linien-Diagramm abgebildet, das die verschiedenen aktivierten Stimmungsanalysen anzeigt (Y-Achse für die Stimmung zwischen -1 und 1 und X-Achse für die Zeit). In der Mitte befindet sich der Gesamtdurchschnitt, der über alle Stimmungsanalysen berechnet worden ist (vgl. Abschnitt Durchschnittsberechnung). Das untere Drittel enthält eine Tabelle, in der die aktuellsten Analyseergebnisse aufgelistet sind. Das Fenster lässt sich flexibel skalieren. Die schmale Darstellung soll dabei unterstützen die Oberfläche links oder rechts von einem Chat-Programm zu positionieren. Für horizontale Ausrichtung des Programmes lassen sich die einzelnen Diagramme vergrößern oder verkleinern, um beispielsweise die Durchschnittsanzeige als Hauptansicht festzulegen. Die letzte Eingabe zum Zeitpunkt des Screnshots war „Das war top! weiter so =)". SentiStrengthDE hat in diesem Beispielsatz eine Stimmung von 0,3 detektiert. Die „(2)" ist dabei eine Gewichtung. Dieses ermöglicht den verschiedenen genutzten Analysen noch einen Faktor zuzuordnen und höher gewichtete Analysen zu priorisieren. SentiStrengthDE fällt somit doppelt so stark ins Gewicht gegenüber SentiStrengthEN. Analog wurde das Emoticon „=)" positiv erkannt. Im Anschluss folgt hinter jedem Stimmungswert dessen Begründung.

Durchschnittberechnung

Für die Berechnung des Gesamtdurchschnittes werden die letzten eingehenden Ergebnisse der einzelnen Stimmungsanalysen zugrunde gelegt. Aufgrund dessen, dass die einzelnen Analysen unterschiedliche Berechnungs- und Übertragungszeiten in Anspruch nehmen, kommen die Ergebnisse asynchron in die Live-Sentiment-Analyse-

Anzeige hinein. Bei jeder eingehenden Nachricht wird der Durchschnitt aus den letzten Analysen berechnet. Das System filtert dabei den Worst-Case, das einige oder alle Analysen falsch oder gar nicht senden, heraus.
Die eigentliche Berechnung des Durchschnittes erfolgt mit einer Iteration über alle Analysen. Wenn die Analyse zu einem Satz einen Stimmungswert geliefert hat, wird dieser mit seinem Faktor multipliziert und einer Gesamtsumme hinzugefügt. Diese Summe wird durch die summierten Faktoren dividiert, um einen Durchschnitt zu erhalten. Bei dem oben abgebildeten Screenshot berechnet sich die Durchschnittsstimmung wie folgt:
0,48 = (0,3*2 + 0,3*1 + 1*1 + 0,5*2) / (2+1+1+2) = 2,9 / 6 = 0,4833
Dieser Wert wird anschließend auf zwei Stellen nach dem Komma gerundet, um ihn grafisch anschaulicher darstellen zu können.

Weitere Funktionen

Eine kleine Erweiterung lässt sich über die Funktion „Manuelle Analyse” öffnen. Um die Live-Sentiment-Analyse ohne ein Chatprogramm und den Keylogger zu nutzen, können manuelle Eingaben über ein entsprechendes Textfenster eingegeben und manuell zur Analyse geschickt werden. Das Fenster ist somit eine grafische Alternative für die Datenerfassung und kann manuelle Eingaben verwalten, wenn kein Keylogger genutzt werden kann oder soll.

Das weiße Fenster zur Haupteingabe kann dabei mehrere Sätze mit Umbrüchen enthalten, die an die Stimmungsanalysen weiter geleitet werden sollen. Das ausgegraute und nicht editierbare Textfeld unten links zeigt das Topic „sentiment/raw” an, welches angibt, auf welchem MQTT-Topic oder vereinfacht gesagt, Datenkanal die Daten weitergeleitet werden. Über den Button „ANALYSIERE” wird die oben getätigte Eingabe so an alle aktivierten Stimmungsanalysen gesendet. Aktiv heißt in diesem Zusammenhang: Alle Analyse-Skripte, die laufen und das Topic „sentiment/raw” abonniert haben.

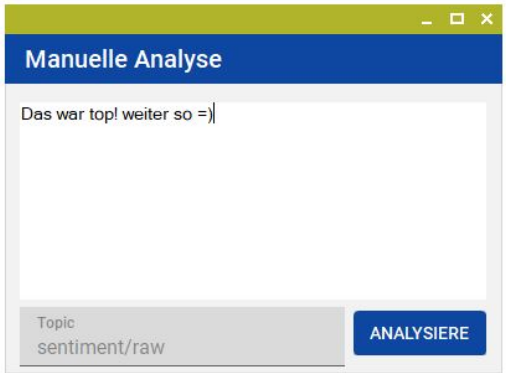

Abbildung 5.6: Abbildung des Software Prototypen

Im Rahmen der Studie war es so möglich, vorhandene Nachrichten aus älteren Chats oder E-Mails zu kopieren und dort einzufügen. So konnte ohne zusätzliche Eingabe über den Keylogger auch längere Texte auf deren Stimmung hin analysiert werden.

Kapitel 6

Konzeptevaluation und Ergebnisse

Zur möglichst praxisnahen Evaluierung des im vorangegangen Kapitels vorgestellten und implementierten Software-Assistenten zur Echtzeit-Stimmungsanalyse wird dieser im Rahmen einer qualitativen Studie eingesetzt. Nachfolgend werden zunächst die Vorstudie und deren Ergebnisse vorgestellt. Im Anschluss daran werden die auf den Erkenntnissen dieser Vorstudie vorgenommenen Verbesserungen an dem Software-Assistenten skizziert. Abschließend wird die Durchführung der auf dem optimierten Software-Assistenten basierenden Hauptstudie vorgestellt und es werden die Ergebnisse ausgeführt. Die durchgeführten Untersuchungen dienen der Beantwortung der bereits in Abschnitt 1.2 formulierten Forschungsfragen.

6.1 Vorstudie

Ziel der Vorstudie war es, auf die erste Version des Software-Assistenten, die primär für die Vorstudie ausgelegt war, das Feedback einer Expertengruppe einzuholen, um auf dieser Grundlage Verbesserungen an dem Assistenten für dessen Einsatz in der qualitativen Hauptstudie vorzunehmen. Die Durchführung der Vorstudie erfolgte im Rahmen dieser Arbeit in dem Zeitraum von November 2021 bis Januar 2022.

6.1.1 Teilnehmer:innen

Die vier Teilnehmer:innen der Vorstudie bildeten die Expertengruppe. Dabei handelte es sich ausschließlich um Personen aus der Projektleitung, dem Qualitätsmanagement und der Forschung mit langjähriger Berufserfahrung, davon ausgehend, dass sie dementsprechend in der Lage sind, ein fundiertes Feedback zu dem Prototypen sowie dem Gesamtkonzept geben zu können.

6.1.2 Durchführung

Im Rahmen der Vorstudie wurden die vier Teilnehmer:innen direkt angeschrieben. Aufgrund der Coronavirus-Pandemie fanden die Treffen in Präsenz, jeweils einzeln und unter den gesetzlich vorgeschriebenen Schutzmaßnahmen statt. Dabei wurden ihnen zuerst die Vision des Konzeptes sowie die Szenarien vorgestellt. Im Anschluss daran gab es eine kurze Demonstration des Assistenten,indem ein Beispieltext auf dessen Stimmung hin untersucht wurde, und eine Erklärung, wie die Analyseergebnisse auf der Benutzeroberfläche zu interpretieren sind. Daraufhin haben die Proband:innen selber mit dem Prototypen gearbeitet. Diese Phase basierte auf dem Think-aloud-Prinzip, d.h., dass die Teilnehmenden ihre Gedanken laut äußerten und diese protokolliert wurden. Das Format der offenen Fragen sollte den Prozess unterstützen, möglichst früh das Feedback in den Entwicklungs- und Verbesserungsprozess des Software-Assistenten einzubringen.

6.1.3 Ergebnisse

Generell wurden das Konzept und der Prototyp sehr positiv aufgenommen. Alle Proband:innen waren der Auffassung, das die Live-Analyse von Stimmung in Textnachrichten ein wichtiger Baustein sein kann, um die Stimmung in einem Team zielgerichteter zu erfassen und bei deren Verschlechterung präventive Maßnahmen zu treffen. Die nachfolgend ausgeführten Ergebnisse basieren auf den Diskussionen mit den Proband:innen der Expertengruppe und sind nach dem Grad der Relevanz vorgestellt.

Zwei der vier Expert:innen äußerten Bedenken hinsichtlich des Gesamtkonzeptes zu dem Aspekt der Akzeptanz.
Wenn der Verfasser einer Nachricht häufiger und bewusst negativ schreibt, muss ihm eine Software nicht anzeigen, dass diese beim Empfänger negativ ankommen wird. Die Live-Sentiment-Analyse biete nur einen geringen Mehrwert, wenn diese dem Verfasser nur anzeigt, was dieser eigentlich selber weiß. Die zwei Proband:innen merkten weiter an, dass es zwar der Selbstreflexion diene, wenn über die verfassten Nachrichten nachgedacht werde, aber der „Benefit", wie er (vermutlich) von den zwei Proband:innen und zukünftigen Nutzern erwartet wird, nicht immer gegeben sei.

Dessen ungeachtet wurden auch einige Merkmale des vorgestellten Prototypen mit den Proband:innen diskutiert. Neben einer Vereinfachung der Benutzeroberfläche stand die Verbesserung der Analysen im Vordergrund, da gewisse Aussagen von SentiStrength nicht interpretiert wurden. Spezifische Aussagen wie „Weiter so!" oder „Wenn es sein muss." werden von der Polarität der Wörter und ihrer Zusammenstellung von dem Assistenten her nur neutral interpretiert, obwohl diese Sätze positiv bzw. negativ von den Proband:innen aufgefasst wurden. Die Erwartungen der Vorstudien-Teilnehmer in Bezug auf die Genauigkeit wurden also nicht voll erfüllt. Vor diesem Hintergrund rückte die Analyse des Soziolekts in den Fokus des Interesses, auf die in

dem Abschnitt „Verbesserungen” eingegangen wird.

Zum Zeitpunkt der Vorstudie hatte der Software-Prototyp eine Funktion, die noch einen größeren Verlauf über die letzten zehn Analysen grafisch darstellte. Neben den Sätzen waren auch alle Begründungen aufgeführt. Ein Proband merkte an, dass diese Funktion in der Zukunft ausgeblendet oder „erstmal versteckt” werden sollte, da die zukünftigen Benutzer sonst gerade am Anfang die Analysen vergleichen würden. Hierbei könne es passieren, dass Benutzer den Verlauf der Stimmungsanalyse überinterpretieren und vielleicht die Akzeptanz des Systems mit der Zeit verlieren.

Neben diesen Aspekten gab es weitere Punkte, die mit der Expertengruppe diskutiert wurden, aber nicht weiter in die Verbesserungen dieser Arbeit einfließen konnten und nur in den Empfehlungen erläutert werden. In diesem Zusammenhang ist die Verbesserung der Analyse der Stimmung und ihrer Interpretation durch die Betrachtung des Kontextes zu nennen, d.h. die Analyse von Themen und Zusammenhängen im gesamten Chat und nicht nur die einzelner Sätze.

Mit der Festlegung auf Textnachrichten in Software-Entwicklerteams hat sich diese Arbeit primär mit Chats zwischen Entwickler:innen beschäftigt. Gerade der Kontakt mit Kunden über E-Mail wäre aber, so drei der Teilnehmer der Expertengruppe, ein sehr interessantes Anwendungsgebiet, denn der Umgang mit Kunden sei einer der risikobehafteten Kommunikationswege. Hier kam von den Proband:innen aber auch der Einwand, dass die Stimmungsanalyse dieses Bereichs zwar von hohem Interesse ist, aber schwieriger sein wird, da zum einen die Menge an Daten sehr umfangreich ist und zum anderen der Grad der Geheimhaltung sehr hoch sein kann. Mit Hilfe des Keyloggers oder der Eingabe über die „manuelle Analyse” können zwar auch Texte von E-Mails erfasst werden, aber aufgrund der vorliegenden geringen Datenmenge dieser Textsorte in dieser Arbeit wird darauf nicht weiter eingegangen.

Eine weitere Anmerkung gab es bezüglich der Corona-Pandemie-Situation und das damit zusammenhängende vornehmliche Arbeiten im Homeoffice. Die Verlagerung von persönlicher Kommunikation hin zu Plattformen wie Microsoft Teams habe das Arbeitsumfeld verändert. Explizit wurde angemerkt, dass einige Personen aus dem beruflichem Umfeld der Proband:innen mit der Umstellung besser klargekommen würden als andere. Auch nach Monaten gebe es Personen, die mit der vermehrt schriftlichen Kommunikation weniger gut klarkommen und diese auch nicht bevorzugen würden. Zwei der Proband:innen merkten an, dass zudem die Art der schriftlichen Kommunikation sehr wichtig sei, da einige Personen E-Mails als Medium favorisieren, wo hingegen andere Mitarbeiter mehr auf direktere Chats setzten.

6.2 Verbesserungen

Aufbauend auf den Erkenntnissen der Vorstudie wurde eine weitere Entwicklungsphase Ende Dezember 2021 bis Anfang Januar 2022 angelegt, in der die mit den Expert:innen diskutierten Verbesserungen in den Software-Assistenten mit eingeflossen sind. Der Hauptaspekt lag auf der Frage, was der Mehrwert einer Stimmungsanalyse mittels eines Software-Assistenten für Entwicklerteams sei. Auch wenn die Stimmungsanalyse zur Selbstreflexion beim Verfassen von Nachrichten diene, biete sie laut zwei Proband:innen keinen großen Mehrwert, da der Assistent einem nur sage, was einem als Verfasser:in eh bewusst sein sollte. Natürlich helfe es dem Team, wenn einige Teilnehmer sich reflektieren und Nachrichten verbessern, aber davon kann nicht immer ausgegangen werden. In Anlehnung an diese Kritik der Proband:innen der Expertengruppe wurde das Konzept dahingehend erweitert, dass nicht nur die Stimmung eines einzelnen Entwicklers mittels des Software-Assistenten analysiert werden kann, sondern die in dem gesamten Team herrschende Stimmung. Hierzu wurde eine Funktion implementiert, mittels derer jeder Nutzer der Live-Sentiment-Analyse die Interpretation der Team-Stimmung in Form des Stimmungsdurchschnittes anonym weiterleiten kann. Wenn (fast) alle aus dem Team dies freiwillig machen würden, könnte jeder sein Team besser überblicken und sich auch selber darin einordnen.
Wenn alle Entwickl:innen dem Projektleiter neben den Einzelanalysen auch ihre Stimmungsdurchschnitte zur Verfügung stellen würden, könnte dadurch indirekt ein Zeichen von Vertrauen übermittelt werden. Wesentlich ist dabei, dass kein Gruppenzwang entsteht. Auf dieser Basis könnte die Stimmung in einem Team nicht nur live analysiert werden, sondern es könnten auch entsprechend effiziente Maßnahmen getroffen werden, um die Stimmung - insofern sie schlecht ist - zu verbessern.

Vor dem Hintergrund, dass Software-Entwicklerteams näher betrachtet werden, sind gruppenspezifische Ausdrücke zu erwarten. Neben regionalen Unterschieden und verschiedenen Interjektionen werden auch sogenannte „Insider" erwartet, also kurze Sätze, die nicht jeder versteht, weil entsprechende Hintergrundinformationen nur bestimmten, eingeweihten Personen (vom engl. inside = innen) zur Verfügung steht. Die so entstehende „**Soziolekt**"-Analyse umfasst somit den Sprachgebrauch einer sozialen Gruppe und im Zusammenhang mit dieser Arbeit insbesondere die Berufssprache und deren soziokulturelle Färbung.
Hier entstehen häufig eigene Ausdrücke, die auch Stimmung widerspiegeln, aber von der Zusammenstellung der Wörter von einer SentiStrength-Analyse nicht erkannt werden. „Wenn's sein muss" enthält keine negativ polarisierten Wörter, ist aber in Gruppen oft ein negativer Ausdruck. Die Implementierung einer Soziolekt-Analyse resultierte aus der Vorstudie und soll Ausdrücke wie „Weiter so!", „Wenn es sein muss." oder ähnliche Aussagen klassifizieren, die von SentiStrength nicht interpretiert werden. Das keywordbasierte Verfahren nutzt als Datengrundlage ebenfalls eine Look-Up-Table (LUT), um Keywords oder Key-Phrases zu erkennen. Die LUT umfasst z.B. folgende

Aussagen und Wertungen:

```
{
  "lut": [
     { "emo": "wenn es sein muss", "senti": -0,5},
     { "emo": "gar nichts",      "senti": -0,5   },
     { "emo": "jaja",            "senti": -0,5   },
     { "emo": "machs besser",    "senti": -0,5   },
     { "emo": "weiter so",       "senti": 0,5    }
  ]
}
```

Jeder Mensch und auch jede Gruppe hat gewisse Begriffe, die regelmäßig genutzt werden, aber nicht durch die in der Studie genutzten Metriken analysiert werden. Ein „na gut”, das mittels der Sentistrengh-Analyse durch das Wort „gut” positiv interpretiert wird, obwohl in Kombination mit „na” eher negativ konnotiert ist, kann so durch eine Soziolekt-Analyse in eben diesen negativen Bereich gezogen werden.

Da sich aber auch Berufssprachen, wie die der Informatik und Gruppendynamiken ändern und diese entsprechend abgebildet werden sollen, kann diese LUT vom Benutzer über eine grafische Eingabe manuell ergänzt werden. Der Benutzer ein Wort oder eine Aussage mit einer entsprechenden Polarität zwischen -1 und +1 dem Lexikon hinzufügen.

Damit Nutzer bessere, d.h. genauere Ergebnisse von Stimmungsanalysen erhalten können, ist es sinnvoll, diese zu individualisieren. Auch während der Laufzeit der Stimmungsanalyse der Soziolekte ist es möglich, neue Wörter oder ganze Aussagen hinzuzufügen. Dazu gibt es ein zusätzliches Fenster in der Benutzeroberfläche. Der in Abb. 6.1 gezeigte Screenshot wurde von dieser Oberfläche gemacht. Unter der Lookup-table im Screenshot ist es möglich, einen neuen Satz hinzufügen und diesen mit einem Stimmungswert zu versehen.
Im Rahmen der Studie konnten Teilnehmer eigene Aussagen oder Wörter hinzufügen, wenn diese durch die Analyse nicht abgedeckt wurden. Die abgebildeten Einträge wurden durch die Proband:innen manuell ergänzt. Diese Personalisierung wurde parallel zum laufenden Betrieb durchgeführt und wurde nach der Eingabe auch direkt bei der Stimmungsanalyse „Soziolekt" berücksichtigt. Den entsprechenden Stimmungswert konnten die Proband:innen in 0,1-er Schritten selbst festlegen. Auch Interjektionen (haha, ah, würg, oh usw.) können hier individuell eingebaut werden.

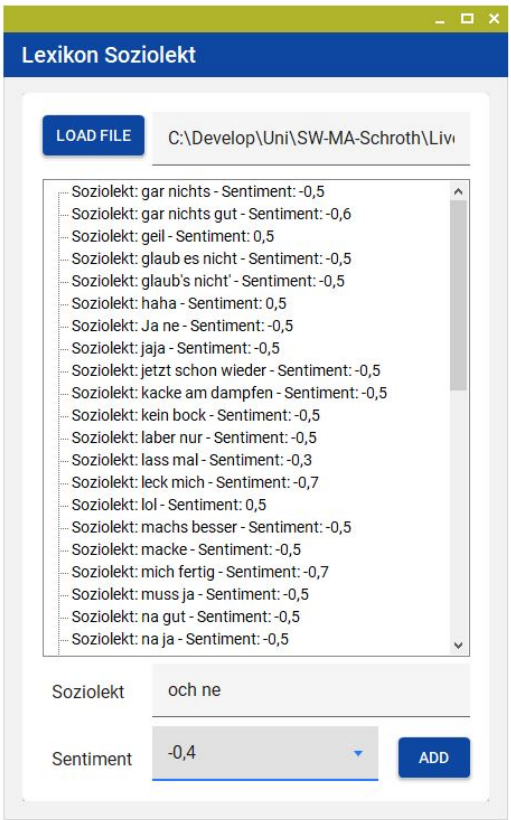

Abbildung 6.1: Personalisierung der Soziolekt-Analyse

Ob ein „haha" positiv im Sinne von „Das war lustig, ich lache." oder negativ gemeint war, kann der Assistent nur über den Kontext erkennen. Da dies noch nicht möglich ist, wird hier auf die Interpretation des Benutzers gesetzt, um die Gesamtanalyse zu verbessern. Wenn dieser „haha" als „das ist lustig" auffasst, kann er dieses Wort mit einer entsprechenden Bewerbung, beispielsweise +0,5 dem Soziolekt-Lexikon hinzufügen. Eine zu entwickelte Kontextanalyse wird in den Empfehlungen erläutert.

Neben der Implementierung der neu entwickelten Soziolekt-Analyse wurden auch kleine Änderungen am deutschen Lexikon von SentiStrength vorgenommen. In dem Satz „Ich krieg noch was von dir!" wird standardmäßig das Wort „krieg" durch die negative Konnotation auch negativ mit -0,5 (SentiStrength 1 -3, nach Standardisierung 1+ (-3) / 4 = -0,5) interpretiert. Analog wurden folgende Werte im Lexikon angepasst:
- krieg 0 statt -3
- kriege 0 statt -3
- wie 0 statt 2

- stark 0 statt -2

Durch die parallele Verwendung von StentiStrength-EN gab es zudem deutsche Wörter, die im Englischen falsch interpretiert wurden. Dazu wurde auch hier eine Anpassung an das englische Lexikon vorgenommen. Das Wort bzw. der Artikel „die", der im deutschen Sprachgebrauch sehr häufig vorkommt, bedeutet im Englischen „sterben" oder „stirb". Daher wurde dieses spezielle Wort aus dem englischen Lexikon entfernt. Gleiches gilt für das (relativ) häufig vorkommende, deutsche Wort „war", das im Englischen „Krieg" bedeutet. Es ist davon auszugehen, dass die Liste der Wörter mit gleicher Schreibweise aber (zum Teil eklatanter) unterschiedlicher Bedeutung in der deutschen und englischen Sprache und die aufgrund dessen zu Fehlinterpretationen führen, erweitert werden müsste.
Auch die Oberfläche des Software-Assistenten wurde auf Basis der Vorstudien-Ergebnisse überarbeitet. Das Linien-Diagramm im oberen Drittel hat eine andere Zeitanzeige bekommen, damit Nutzer ihre Nachrichtenanalyse im zeitlichen Verlauf besser verfolgen können.

Der mittlere Stimmungsdurchschnitt hat eine Anpassung der Farbe erhalten. Hier wurden ein helles und ein dunkleres Grün für den Bereich zwischen 0,1 und 0,5 und 0,5 bis 1 eingefügt.

Der Verlauf mit den Begründungen im unteren Drittel wurde auf die zuletzt durchgeführte Analyse reduziert. Für den späteren Betrieb in der Hauptstudie riet ein Projektleiter als Proband in der Vorstudie dazu, dieses Feature auszublenden. Seiner Meinung nach sei es somit nicht mehr möglich, vergangene Analysen mit der aktuellsten zu vergleichen. Dies würde das subjektive Vertrauen in den Datenschutz verbessern. Zum subjektiven Vertrauen sei erwähnt, dass während der Laufzeit vom Programm die letzten Datenpakete gebuffert werden. Ältere Pakete werden automatisch gelöscht. Persönliche Daten werden also nur während der Laufzeit des Programmes und nur für einen gewissen Zeitraum angezeigt. Umso weniger Daten angezeigt werden, umso höher sollte das Vertrauen in die Software sein, so der Experte.

6.3 Hauptstudie

Ziel der Hauptstudie war es, die Anwendbarkeit des Konzeptes in der Praxis zu prüfen und die eingangs formulierten Forschungsfragen zu beantworten. Die qualitative Hauptstudie wurde in der zweiten Januarhälfte 2022 mit dem verbesserten Software-Assistenten durchgeführt. Dazu wurde zwischen dem 18. und 25. Januar 2022 unter Einhaltung aller zu der Zeit geltenden Corona-Maßnahmen ein Aufbau realisiert, der sowohl in Präsenz als auch in einem Online-Meeting von den Proband:innen genutzt werden konnte. Die Face-to-Face-Umfrage erfolgte mittels eines Fragebogens, der im Anhang (A.2) dieser Arbeit zu finden ist.
Die durchgeführte Studie, betrachtet primär die Stimmungsanalyse einzelner Entwickler:innen. Die Funktionalität, mehrerer Stimmungen verschiedener Entwickler:innen zu erfassen, um eine Teamstimmung daraus abzuleiten, war ein Ergebnis der Vorstudie. Dementsprechend wurden im Anschluss an die Vorstudie verschiedene, oben beschriebene Verbesserungen vorgenommen, damit das System diese Analyse der Teamstimmung technisch durchführen kann. Dennoch sind die Datensätze aus der Hauptstudie zu gering, um eine Evaluation dieser Teamstimmung-Funktion durchzuführen. Dafür müsste eine Studie über einen längeren Zeitraum erfolgen.

6.3.1 Teilnehmer:innen

An der Hauptstudie nahmen zwölf Proband:innen aus drei mittelständischen Unternehmen der Informations- und Telekommunikationsbranche teil. Die drei Unternehmen haben mindestens einen ihrer Sitze in der Stadt oder der Region Hannover und beschäftigen zwischen 50 und 250 Mitarbeiter:innen. Von den zwölf in Entwicklung-Teams von Software-Lösungen tätigen Proband:innen waren zum Zeitpunkt der Durchführung der Studie sieben als Programmierer:innen, zwei als Projektleiter:innen, zwei als technische Redakteur:innen und einer als IT-Consultant tätig. Die Proband:innen-Gruppe setzte sich aus vier Frauen und acht Männern zusammen. Im Durchschnitt arbeiten sie ungefähr zehn Jahre lang in der IT- Branche, wobei sich die Bandbreite an Erfahrungen von einem Jahr bis 24 Jahre erstreckt (s. Anhang A.2).
Abbildung 6.2 zeigt die Branchenerfahrung bzw. Arbeitsjahre der Proband:innen. Neben jüngeren Entwicklern und Quereinsteigern konnten auch fünf Projektleiter:innen und Entwickler:innen mit zehn bis 24 Jahren Berufserfahrung für die Studie gewonnen werden. Im Durchschnitt waren die Proband:innen ca. 9,6 Jahre in der Branche tätig.

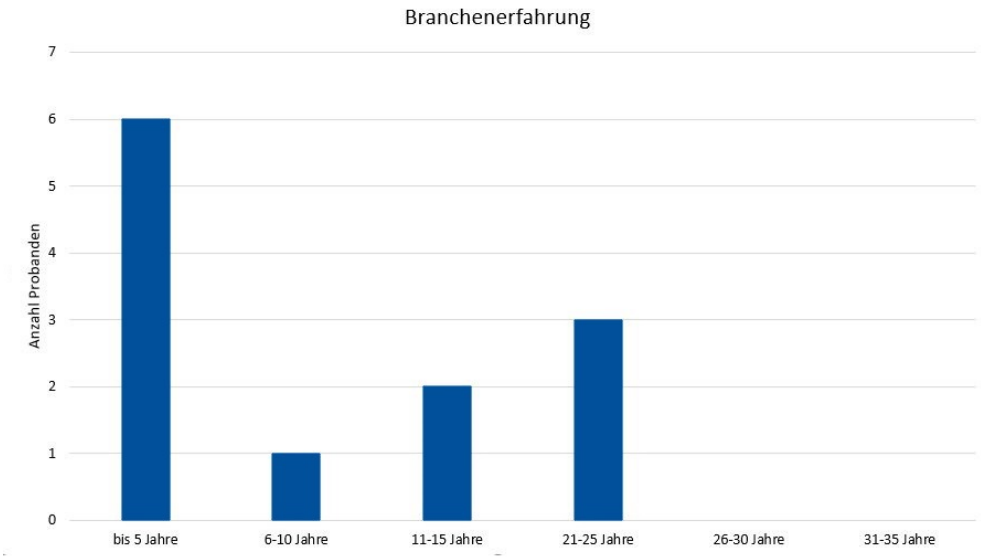

Abbildung 6.2: Erfahrungen der Proband:innen in der Branche

6.3.2 Durchführung

Alle Proband:innen erhielten per E-Mail eine Einladung zu der Studie (s. Anhang A.1). Die Terminvergabe für den Januar 2022 erfolgte mittels einer Doodle-Umfrage. Dabei konnten sie angeben, ob sie an der Durchführung der Studie in Präsenz oder online teilnehmen wollten. Pro Proband waren für die Durchführung 45 bis max. 60 Minuten eingeplant.
Der Zeitplan sah folgendes Vorgehen vor:

1. **Einführung in das Thema** (5 Minuten)
   Den Proband:innen wurde die Idee des Konzeptes erläutert, d.h., dass ihre Texteingaben auf Stimmung hin untersucht werden, um zu erkennen, wie ihre Nachrichten beim Empfänger ankommen werden. Es wurde ebenfalls erläutert, wie die Nachrichten erfasst und verarbeitet werden und dass keine Eingaben über die Studie hinaus gespeichert werden.

2. **Fragebogen Teil 1** (5 Minuten)
   Neben der schriftlichen Zusage an dieser Studie teilzunehmen, wurden primär persönliche Informationen (Tätigkeit, Erfahrung in der Branche, Geschlecht) der Proband:innen und Angaben bezüglich des aktuell bevorzugten Kommunikationsweges (E-Mail, Telefon etc.) in einem Fragebogen (s. Anhang A.2) erfasst.

3. **Prototyp erkunden** (5 Minuten)

In den ersten fünf Minuten konnten die Proband:innen selbstständig Eingaben tätigen und ihre Echtzeit-Analyse betrachteten. Hier wurden Eingaben wie „Guten Morgen :-)" kurz vorgeführt oder schon selbstständig getestet.

4. **Prototyp testen** (10-20 Minuten)
   Nach der kurzen Eingewöhnungsphase wurden den Proband:innen fiktive, aber möglichst praxisnahe Szenarien von mir mündlich beschrieben. Wie das Szenario, dass ein Kolleg:in über das Wochenende mit viel Engagement einen wichtigen Quellcode fertig gestellt hat. Die Aufgabe bestand nun darin ihr oder ihm eine positive Nachricht zu senden, in denen der Dank dafür ausgedrückt wird. Analog wurden positive, negative oder sachliche Szenarien gestellt, zu denen Inhalte verfasst und von dem Software-Assistenten analysiert wurden. Bei fast allen Proband:innen gab es nach diesen offenen Aufgaben die Rückfrage, ob sie das System auch „an die Grenze bringen" könnten. Mit sehr unnatürlich positiven oder negativen Eingaben sollten die Stimmungsgrenzen von +1 oder -1 erreichen werden, was in Extremfällen auch gelang. Dieses „Spielen" würde in den meisten Fällen als Möglichkeit genutzt, um vom Prototyp wieder auf den Fragebogen zu wechseln.

5. **Fragebogen Teil 2** (5 Minuten)
   Um die Erfahrungen der Proband:innen festzuhalten, sollten diese auf dem zweiten Teil des Fragebogens Einschätzungen auf einer Skala zur Live-Sentiment-Analyse festhalten. Zudem konnte nach dem „I wish, i like, i wonder"-Schema, Feedback zum Assistenten abgegeben werden. Feedback sollte von den Proband:innen anhand der drei Kategorien „Ich wünsche mir...", „Ich mag..." und „Ich wundere mich, dass..." gegeben werden. Dieses sollte das spätere sortieren und kategorisieren vereinfachen und eine kleine Hilfestellung sein, um Feedback kompakt zu formulieren [30].

Obwohl der Fragebogen als Grundlage zum Festhalten der Eindrücke und Wünsche diente, wurden einige Aspekte ähnlich einer Think-aloud-Studie auch persönlich mit den einzelnen Proband:innen besprochen und protokolliert. Der Fragebogen mit den Ergebnissen der Teilnehmer ist dem Anhang zu entnehmen A.2. Die Durchführung der Studie pro Proband lag wie geplant zwischen 45 und 55 Minuten. Keiner der Teilnehmenden war schneller oder brauchte länger als 55 Minuten.

6.3.3 Aufbereitung der Ergebnisse

Für die Darstellung der Ergebnisse sowie der folgenden Interpretation im Kapitel 7, wurden die Daten aus der Hauptstudie zuerst aufbereitet. Neben der Visualisierung der Daten und der Berechnung von Mittelwerten, Minimas, Maximas sowie Standardabweichungen mithilfe eines Excel-Dokuments wurden die Freitext-Antworten nach inhaltlichen Übereinstimmungen kategorisiert und sortiert.
Die Analyse der offen gestellten Fragen folgen dabei einem offenen Coding.

Da verschiedene Aspekte ähnlich einer Think-aloud-Studie auch mündlich diskutiert worden sind, wurde diese separat protokolliert. Diese Ergebnisse wurden ebenfalls kategorisiert und ergänzen das schriftliche Feedback 2.2 auf dem zweiten Teil des Fragebogen (s. Anhang A.2).

6.3.4 Ergebnisse

Im Folgenden werden die Ergebnisse der Hauptstudie und des Fragebogens in Bezug auf die Forschungsfragen vorgestellt. Grundsätzlich kamen alle befragten Software-Entwickler:innen, Projektleiter:innen und technische Redakteur:innen sehr schnell mit dem Assistenten zurecht. Die Bedienung war für alle intuitiv und die grafischen Anzeigen wurden richtig interpretiert. Der erste Teil des Fragebogens beschäftigt sich primär mit der Erfassung der Daten der Proband:innen sowie ihrem Arbeitsumfeld, während beim zweiten Teil direkte Bezüge zu den Forschungsfragen zu finden sind.

Einblick in die Probandengruppe

Die Antworten auf die ersten drei Fragen sind dabei dem Absatz 6.3.1 „Teilnehmer:innen” zu entnehmen. Dort wurden die zwölf Proband:innen und die drei Unternehmen der Hauptstudie vorgestellt.

Kommunikationsverhalten der Proband:innen

In der vierten Aufgabe sollten die Proband:innen sieben Aussagen bewerten. Mittels der Aussagen 4.1 bis 4.7 sollten weitere Einblicke in das Arbeitsumfeld der Proband:innen und ihrem Kommunikationsverhalten abgeleitet werden.
Den Bewertungen der Aussage 4.1 ist zu entnehmen, das eine ausreichende Kommunikation der Projektführung zu den Proband:innen vorherrscht. Hier haben die Proband:innen 4 von 5 Punkten im Durchschnitt (ohne größere Ausreißer) für den Informationsfluss bezüglich ihren jeweiligen Tätigkeiten gegeben (Fragebogen Teil 1 - Bewertung der Aussage 4.1).
Ob auch die Proband:innen genug Informationen mit ihren Kollegen und der Projektführung austauschen wurde mit der Aussage 4.2 überprüft. Rückfragen zu ihren Tätigkeiten gibt es, aber nicht ungewöhnlich viel.
Regelmäßige Meetings werden in allen Unternehmen ebenfalls durchgeführt.
Die Aussage 4.4 soll Rückschlüsse ermöglichen, ob die Kommunikation von der Geschäftsführung zu den Teams und Entwicklern funktioniert. Da nach Angaben der Teilnehmer eher selten Informationen „auf dem Flur” geteilt werden, lässt sich darauf schließen, dass in den Unternehmen die Informationen durchaus gut bei den Teams ankommt.

Die Abbildung 6.3 zeigt die Bewertungen der Aussagen 4.5 bis 4.7 der Proband:innen.

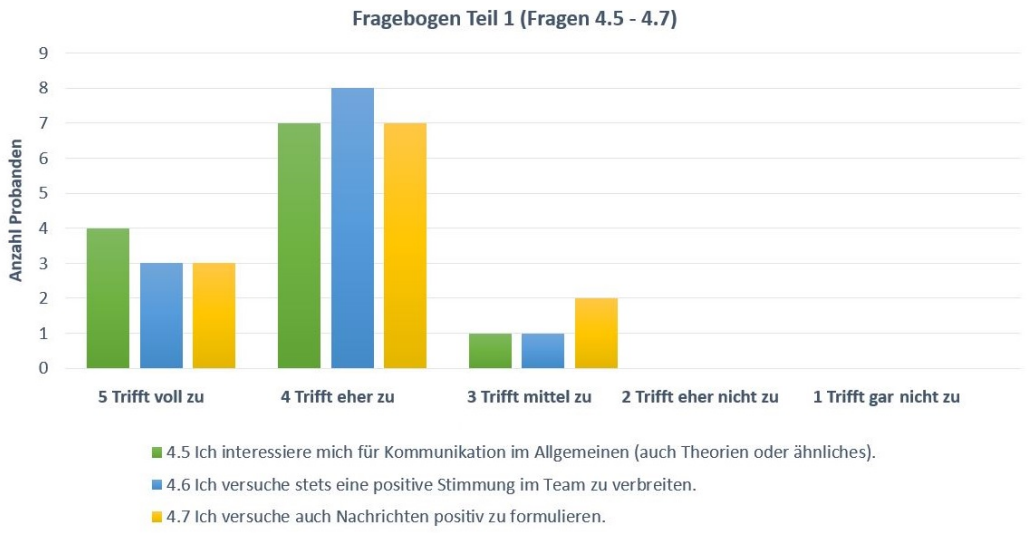

Abbildung 6.3: Fragebogen Teil 1 - Bewertung der Aussagen 4.5, 4.6 und 4.7

Das allgemeine Interesse an Kommunikation ist bei allen Proband:innen hoch (Aussage 4.5). Der regelmäßige Austausch von Informationen wird von allen als zentraler Aspekt für den Erfolg eines Teams erachtet. Ebenso ist für die Teilnehmer die Bedeutung der Stimmung elementar, da fast alle Proband:innen stets versuchen, die Stimmung im Team zu halten oder zu verbessern. Schlechte Erfahrungen im Zusammenhang mit negativer Stimmung haben fast alle Proband:innen mündlich bekundet. Der Wunsch nach einer Möglichkeit zur Erfassung der Teamstimmung mittels eines Software-Tools mit dem Ziel der Optimierung der Stimmung wurde bereits am Anfang der einzelnen Hauptstudien, d.h. bevor die eigentliche Software von den Proband:innen genutzt wurde, von vielen Proband:innen geäußert.

Häufigkeit der Kommunikationswege (Fragebogen Teil 1 - Frage 5)

Da mit der Live-Sentiment-Analyse nur die schriftliche Kommunikation auf Stimmung hin untersucht wird, gilt es herauszufinden, wie häufig diese Kommunikationswege genutzt werden.

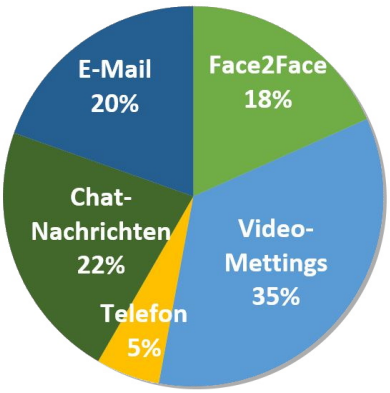

Abbildung 6.4: Genutzte Kommunikationswege der Proband:innen in der Hauptstudie

Dazu konnten die Proband:innen die Häufigkeiten ihrer Kommunikation angeben. Schriftliche Kommunikation, bestehend aus Chat-Nachrichten 22% und E-Mails 20% decken in den betrachteten Unternehmen im Durchschnitt etwas weniger als die Hälfte der Gesamtkommunikation ab. Hinzu kommt, dass die (Standard-) Abweichungen hier mit 8,5% und 11,5% recht groß sind, da im Rahmen der Corona-Situation in einigen Teams die Face2Face Kommunikation (fast) komplett weg gefallen ist und durch Video-Meetings und Chats ersetzt werden. Bei einigen Proband:innen wird so die schriftliche Kommunikation häufiger genutzt als alle anderen Kommunikationswege (bis zu 60%). Abschließend muss festgehalten werden, dass diese Verteilung in Abbildung 6.4 eine Momentaufnahme für den Januar 2022 ist.

Wünsche der Proband:innen für die interne Kommunikation

Neben der Abschätzung der Kommunikation und Kanäle der Proband:innen, konnten die Proband:innen im letzten Absatz des ersten Teil des Fragebogens Wünsche zum Thema Kommunikation geben. Dabei konnten allgemeine Ideen oder konkrete Verbesserungen im Unternehmen festgehalten werden. Diese Wünsche sollten dazu dienen einen abschließenden Einblick in die Kommunikation der Proband:innen zu erhalten und vorhandene Probleme in der Kommunikation aufdecken. Aspekte, die in keinem Bezug zu den Forschungsfragen stehen, werden hier allerdings nicht aufgeführt.

Mehr persönlicher Austausch (Face2Face) war einer der meist genannten Wünsche der Teilnehmer. Gerade durch den Wechsel zu Online-Meetings und dem vermehrten schriftlichen Austausch, ist die Reduzierung persönlicher Kontakte für viele Proband:innen in den letzten Monaten schwer gefallen. Dieser Umstieg bewirkte zudem, dass trotz strukturierter Online-Meetings viele Informationen doppelt besprochen, verschickt oder dokumentiert worden sind.

Diese redundanten Informationen sind auch in Gruppenchats von Teams ein relevantes Thema. Teamspezifische Kommunikation ist gerade mit neuen oder wechselnden Teammitgliedern ein Aspekt, der zu Konflikten führen kann. Der Aufbau von Gemeinsamen Wissen gepaart mit dem Austausch von zu vielen technischen Details kann die Übersichtlichkeit solcher Chats verschlechtern. Gerade in solchen Teamchats wird sich von einigen Teilnehmern noch eine zielorientierter

Kommunikation gewünscht. Dazu wurde sich von zwei Proband:innen mehr Struktur (Von Grob nach Fein) gewünscht, wie auch die vermehrte Nutzung einheitlicher Vokabeln (FDS = Functional Design Specification, FR = Functional Requirements).
Aber auch Chats einzelner Entwickler können mit Projektfragen oder mit zu vielen Details unübersichtlich werden. Zwei Teilnehmer merkten an, dass zu viele technische oder Projektdetails in persönlichen Chats geteilt werden, anstatt diese Informationen zentral zu verwalten.
Ebenfalls wünschten sich zwei Proband:innen gerade im Homeoffice weniger Telefonate und E-Mails. E-Mails benötigen für die Bearbeitung mehr Zeit und erlauben den Austausch weniger Informationen.
Gerade von Teilnehmern, die regelmäßig im Homeoffice arbeiten, wurde auch ein ausgeglichenes Verhältnis zwischen Meetings und Arbeitsphasen schriftlich festgehalten.
Von einer Probandin wurde ein zuverlässigere Beantwortung von schriftlichen Nachrichten gewünscht. Neben schnelleren Antworten wurde auch die Ausführlichkeit der Antworten bemängelt.
Von einem jüngeren Teilnehmer, der seit 1,5 Jahren in dem Unternehmen tätig ist, wurde angemerkt, dass Kritik professioneller und freundlichen gehalten werden sollte.

Feedback zum Software-Assistenten (Fragebogen Teil 2)

Der zweite Teil des Fragebogen beschäftigt sich mit dem Feedback zum Konzept und dem Software-Assistenten. Hierbei stehen die Fragen 1.1, 1.2, 1.3 im Bezug zur ersten Forschungsfrage:
RQ1: Inwiefern kann Stimmung in Textnachrichten präventiv detektiert werden?
Ob die Live-Sentiment-Analyse die Stimmung der Nachrichten der Proband:innen erkannt hat, sollten diese wieder mit 1 bis 5 Punkten bewerten. Die Hälfte der Proband:innen gab an, dass die Analyseergebnisse der verfassten Nachrichten die von den Proband:innen gesendete Stimmung (Fragebogen Teil 2 - Aussage 1.1) voll zu trafen, drei Teilnehmer:innen gaben an, dass sie eher zu treffen. Auch einzelnen Analysen haben eine ähnliche gute Bewertung erhalten.
Die Aussage 1.3 „Eine solche Software könnte ich Zukunft hilfreich sein" wurde sogar noch häufiger mit „trifft eher zu" und „trifft voll zu" bewertet. Dies unterstreicht die von vielen Proband:innen am Anfang der jeweiligen Studien getätigten Wünsche nach einem solchen System.

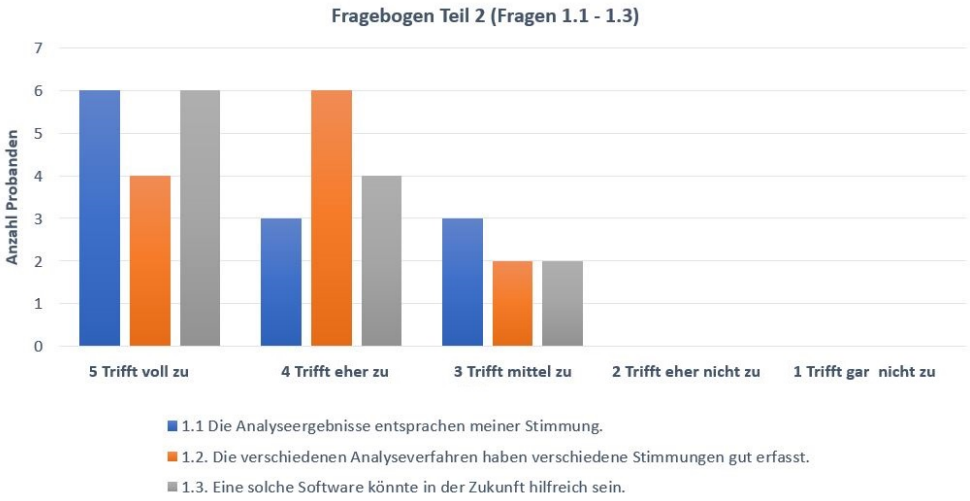

Abbildung 6.5: Ergebnisse Fragebogen Teil 2 - 1.1 bis 1.3 Aussagen bewerten

Die Bewertung der Aussage 1.4 spiegelt das subjektive Vertrauen der Proband:innen wieder, der für den Nutzen eines Systems relevant ist. Allen Teilnehmern der Studie war bewusst, dass die Daten ausschließlich lokal auf dem Endgerät analysiert werden und keinerlei Daten auf externen Geräten überprüft werden. Obwohl der technische Datenschutz somit gewährleistet ist, ist die subjektive Wahrnehmung beim ersten Kontakt mit einem solchem System gemischt.

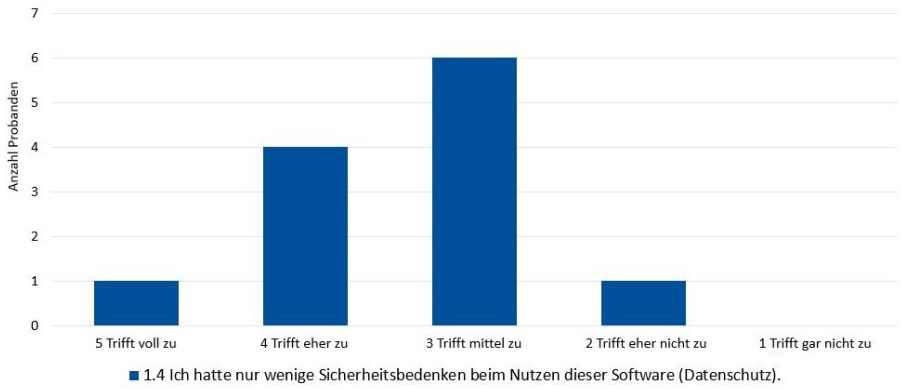

Abbildung 6.6: Ergebnisse Fragebogen Teil 2 - 1.4

Bei den meisten Proband:innen traf die Aussage 1.4 nur mittel zu. Die technische Sicherheit spiegelt sich nicht in der subjektiven wieder.

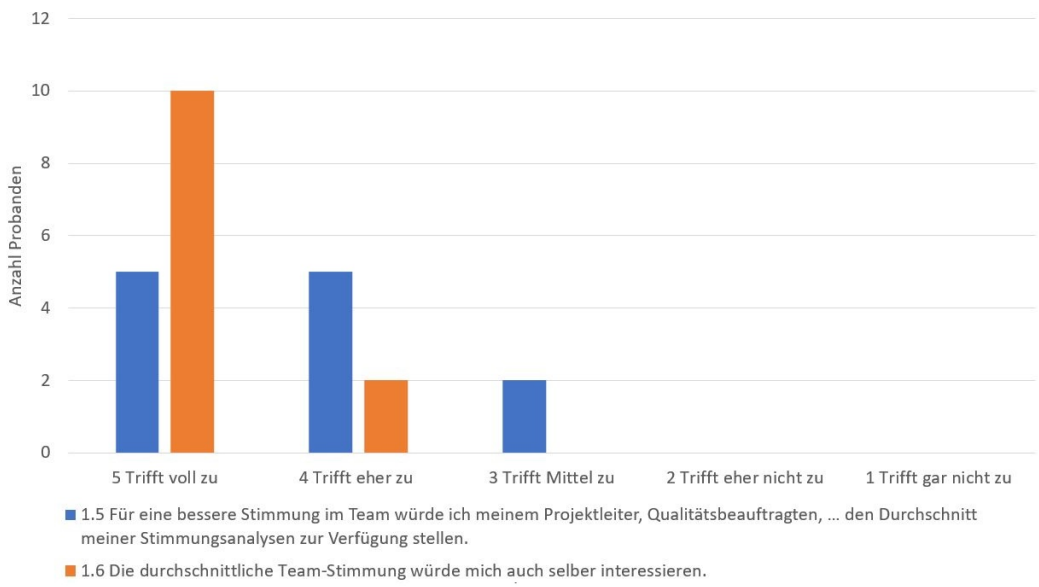

Abbildung 6.7: Ergebnisse Fragebogen Teil 2 - 1.5 und 1.6 Aussagen bewerten

Der Aussage 1.5 „Für eine bessere Stimmung im Team würde ich meinem Projektleiter, Qualitätsbeauftragten, ... den Durchschnitt meiner Stimmungsanalysen zur Verfügung stellen" stimmen die Proband:innen zu. Viele Proband:innen stellten aber die Bedingung, dass die Projektleitung die Teamstimmung nur einsehen kann, wenn die Entwickler:innen dies auch können. Das Team soll gemeinsam seine Teamstimmung einsehen können.
Der Aussage 1.6 „Die durchschnittliche Team-Stimmung würde mich auch selber interessieren." haben fast alle Proband:innen, bis auf zwei, die volle Punktzahl 5 gegeben was wiederum bedeutet, dass die Funktion der Live-Teamstimmung ein sehr hohes Interesse findet.

I wish, i like, i wonder - Feedback

Nach dem Schema „I wish, I like, i wonder" wurden die Proband:innen im letzten Absatz des Fragebogens um schriftliche Bemerkungen gebeten. Diese wurden nach dem vorgegebenen Schema kategorisiert und entsprechend der Anzahl an Proband:innen, die sich in ähnlicher Form geäußert haben, priorisiert. Eine stichpunktartige Auflistung ist dem Fragebogen im Anhang A.2 hinterlegt

I like - „Ich mag..."
Gelobt wurde die einfache Bedienung und die simple grafische Darstellung der

Daten. Ohne große Erklärungen war allen Proband:innen schnell ersichtlich, dass die Stimmung mittels verschiedener Verfahren erkannt wird und in einen Stimmungsdurchschnitt einfloss.
Ebenfalls wurde von der Hälfte der Proband:innen die Personalisierung der Stimmungsanalysen durch die Lexikonverwaltung der Soziolekt-Analyse positiv bemerkt.

I wish - „Ich wünsche mir...”
Für den zukünftigen Einsatz einer solchen Live-Sentiment-Analyse, haben sich die Hälfte der Proband:innen für eine direkte Integration des Assistenten für den Messager-Dienst „Microsoft-Teams” ausgesprochen. Eine Live-Anzeige der Stimmung neben dem Textfeld war dabei ebenso von den Proband:innen angesprochen worden, wie die grafische Einarbeitung im Texteingabe-Feld. Positive oder negative Aussagen könnten direkt grün oder rot markiert werden. Dabei könnte eine automatische Analyse beispielsweise gewissen Wörter, wie bei einer Rechtschreibkorrektur unterstreichen. Von zwei Proband:innen kam zudem die Idee nicht alle Nachrichten auf Stimmung hin zu untersuchen, sondern mit einem Prüf-Button oder einem Tastenkürzel eine manuelle Analyse auszulösen.
Im Rahmen der Studie wurden nur die ausgehenden Nachrichten der Proband:innen auf Stimmung hin von Assistenten untersucht. Von vier Proband:innen wurde die Idee geäußert, dass eingehende Nachrichten von dem Empfängern manuell auf Stimmung bewertet werden. Bei einem Vergleich der Stimmungsanalyse des Assistenten von den gesendeten Nachricht und der einer menschlichen Analyse auf Empfängerseite, könnte dazu genutzt werden, die Sentiment-Analyse in Zukunft in Richtung Genauigkeit zu optimieren.
Zwei Teilnehmer gaben an, dass sie ein solches System gerne über einen längeren Zeitraum im Praxistest untersuchen möchte.
Ein Entwickler hielt einen Wunsch schriftlich fest, dass nur bestimmte (Gruppen-) Chats von Assistenten betrachtet werden sollten. Hierzu könnten gewünschte Chats ausgewählt werden, um eine Stimmungsanalyse nur auf diesen auszuführen.

I wonder - „Ich wundere mich, über ...”
Eine Entwicklerin hat sich insbesondere zum Datenschutz geäußert. Obwohl bekannt ist, dass die Analyse nur lokal auf dem eigenen Gerät geschehen, kann es mögliches Gefühl von Beobachtung oder gar Überwachung entstehen. Damit einhergehend können Meinungsäußerungen weniger kritisch geäußert werden. Die Funktion freiwillig die Stimmung weiter zu leiten, um einen Teamdurchschnitt anonym berechnen zu können, setzt voraus, dass (fast) alle Teammitglieder dieses tragen. Personen können sich ausgegrenzt fühlen, wenn sie ihre anonymen Werte nicht weiterleiten wollen. Im Worst-Case kann es damit zu einer Abspaltung von Personen in einem Team kommen, so die Probandin. Aber selbst diese Probandin hielt schriftlich fest, dass „solche Analysetools sehr hilfreich sein könnten. [...] es ist immer gut zu wissen, wie die Stimmung in einem Team ist.”.

Kapitel 7

Diskussion

Nachfolgend werden auf Grundlage der Ergebnisse der Hauptstudie aus Kapitel 6 Rückschlüsse auf den zukünftigen Nutzen des in dieser Arbeit entwickelten Software-Assistenten zur Stimmungsanalyse von Textnachrichten gezogen. Dazu werden zu Beginn die relevanten Ergebnisse noch einmal kurz zusammengefasst, um diese im nächsten Schritt in Beziehung zu den zwei Forschungsfragen zu setzen. Auf dieser Basis werden anschließend die Ergebnisse der Studie interpretiert. Auf die Aussagefähigkeit der Studienergebnisse bzw. auf die dieser Arbeit gesetzten Grenzen wird im nächsten Abschnitt eingegangen. Abgeschlossen wird das Kapitel mit Empfehlungen für unabhängige, weiterführende Untersuchen, die auf dem Konzept und den Erkenntnissen dieser Arbeit aufbauen können.

7.1 Zusammenfassung der Ergebnisse

Mit dem Ziel, die Stimmung in einem Team von Software-Entwickler:innen in Echtzeit über Textnachrichten zu erfassen - statt zentral und retrospektiv - wurde in dieser Arbeit das Konzept der „Live-Sentiment-Analyse" entwickelt, vorgestellt und evaluiert. Der hierfür realisierte Software-Assistent dient den einzelnen Entwickler:innen dazu, dass die Stimmung, die mittels der von ihnen verfassten Textnachricht an das Team transportiert wird, unmittelbar angezeigt wird. Im Wesentlichen werden dabei negative oder positive Nachrichten mittels verschiedener Analyseverfahren erkannt und visualisiert.

Aufbauend auf dem verbreiteten, subjektiven Verfahren SentiStrenth [9], wurden zwei weitere Verfahren zur Optimierung der Genauigkeit der Live-Sentiment-Analyse eingesetzt, und zwar die Verwendung von Emoticons und von Soziolekten betreffend. Des Weiteren untersucht ein objektives Verfahren die Eingaben hinsichtlich der Häufigkeit und der Auswahl von Emoticons. Die Emoticon-Analyse nutzt diese statistische Metrik und erfasst die durch die Zeichenfolgen und Smileys transportierten Stimmungen. Aufbauend auf Ergebnissen der mit Experten in der Vorstudie geführten Diskussionen wurde in einer weiteren Entwicklungsphase eine Soziolekt-Analyse

integriert. Durch eine entsprechende grafische Schnittstelle, können Entwickler:innen die Analyse mit eigenen Aussagen personalisieren.

Im Rahmen der Hauptstudie haben die zwölf Proband:innen den Software-Assistenten genutzt. Grundsätzlich standen alle Proband:innen der Software zur Echtzeit-Analyse von Stimmung in Nachrichten sehr positiv gegenüber. Sowohl das schriftliche Feedback als auch die mündlichen Aussagen, die protokolliert wurden, bringen zum Ausdruck, dass alle zwölf Proband:innen einen Software-Assistenten zur Stimmungsanalyse in der Zukunft durchaus nutzen würden. Obwohl nach Ansicht der Proband:innen die eigentlichen Analyseergebnisse gewisse Abweichungen der „realen" Stimmung aufwiesen, seien diese dennoch ausreichend genau, um ihre vermittelten Stimmungen wiederzuspiegeln.

Zusammenfassend ist festzuhalten, dass der im Rahmen dieser Arbeit realisierte Software-Assistent sowie dessen Evaluierung die wichtigsten Anforderungen an ein Konzept für eine präventive Stimmungsanalyse von Nachrichten in einem Software-Entwicklungsteam abdeckt und somit als „Proof of Concept" dient.

7.2 Antwort auf die Forschungsfragen

In Anbetracht des Ziels dieser Masterarbeit, ein Konzept zu entwickeln, zu implementieren und zu evaluieren, um in Echtzeit die Stimmung in schriftlichen Nachrichten zwischen Software-Entwicklern:innen zu detektieren, wurden folgende zwei Forschungsfragen (s. Abschnitt 1.2) abgeleitet:

RQ1

Inwiefern kann Stimmung in Textnachrichten präventiv detektiert werden?

RQ2

Wie und unter welchen Bedingungen kann der Teamstimmung-Durchschnitt automatisiert berechnet werden?

Die Beantwortung dieser zwei Forschungsfragen erfolgt primär auf Grundlage der Ergebnisse der Hauptstudie.
Das entwickelte Konzept und der daraus resultierende Software-Assistent zeigen eine praxistaugliche Möglichkeit, die Stimmung in Textnachrichten zu erkennen. Die Kombination von verschiedenen Stimmungsanalysen ermöglicht es, eine ausreichend genaue Stimmungsanalyse von Sätzen zu gewährleisten. Das hohe Interesse an einem solchen System für den zukünftigen Nutzen in der Praxis haben alle Teilnehmer:innen der Studie schriftlich und mündlich bekundet.

Beantwortung der Forschungsfrage RQ1

Durch die Realisierung des Konzeptes in einer Software können Rückschlüsse auf die Stimmung in Textnachrichten in Echtzeit gewonnen werden. Auf Grundlage dieser Stimmungsanalyse können weitere präventive Maßnahmen zur Aufrechterhaltung oder Verbesserung von Stimmung in Texten folgen. Welche Maßnahmen zukünftig getroffen werden, müssen aber umfangreichere Untersuchungen zeigen.

Alle Proband:innen der Studie standen der Erfassung der Teamstimmung durchweg positiv gegenüber, allerdings unter der Bedingung, dass alle diese Teamstimmung einsehen können und diese aus anonymen und freiwilligen Einzelstimmungen berechnet wird. Die Berechnung eines Mittelwertes von anonymen Einzelstimmungen kann ein sinnvolles Mittel sein, die Stimmung in einem Team zu erfassen.

Beantwortung der Forschungsfrage RQ2

Mittels anonymer Weiterleitungen der Einzelstimmungen aus den Live-Sentiment-Analyse-Tools der Entwickler:innen, kann ein Teamdurchschnitt ermittelt werden. Dieser kann aufzeigen, wie sich die Hintergrundstimmung in einem Team entwickelt. Eine solche Erfassung der Teamstimmung kann aber nur gelingen, wenn alle Personen in einem Team alle Daten transparent einsehen können.

In Bezug auf das zweite Szenario (s. Abschnitt 4.1) kann die Teamstimmung bereits mit wenig Details genutzt werden, um präventive Maßnahmen zu treffen. Wenn der/die Projektleiter:in sieht, dass die Stimmung in der vergangenen Woche von einer beispielhaften positiven Stimmung von 0,2 auf 0,05 abgerutscht ist, spiegelt dieses Unzufriedenheiten im Team schon wieder. Aber auch Abweichungen können in den Augen der Proband:innen relevant sein. Wenn in der vergangenen Woche der Durchschnitt bei +0,2 lag und die maximalen Abweichungen davon +- 0,1 waren, wären Schwankungen mit +-0,3 schon ein Indiz für eine Stimmungsänderung. Für die Evaluation des tatsächlichen Nutzens in der Praxis müssen längere Studien durchgeführt werden.

7.3 Interpretation der Studienergebnisse

Aus der Interpretation der Ergebnisse der Hauptstudie (s. Abschnitt 6.3.4) ergibt sich generell, dass der zur Stimmungsanalyse dienende Software- Assistent viel Interesse bei den Proband:innen hervorgerufen hat. Die erfassten Ergebnisse lassen darauf schließen, dass die Echtzeit-Erkennung von Stimmung in Nachrichten technisch nicht nur machbar, sondern vor allem auch von der Praxis nachgefragt wird. Die Optimierung von Stimmung in Entwicklerteams ist für viele ein wichtiger Aspekt für zukünftige Entwicklungsprozesse. Die steigende Komplexität von Projekten mit immer

leistungsstärkeren Geräten stellt für Entwickler:innen und deren Zusammenarbeit neue Herausforderungen dar.

Nutzung des Software-Assistenten
Die Ergebnisse lassen aber auch erkennen, dass zwei Annahmen die einzelnen Funktionen des Software-Assistenten betreffend, die in dieser Arbeit getroffen wurden, weniger genutzt wurden. So kamen die englischsprachige Analyse „SentiStrength-EN" kaum zum Einsatz. Dies mag darauf zurückzuführen sein, dass die verwendeten englischen Begriffe in Bezug auf die Polarität meist neutral waren („file", „change") und dass bestimmte Wörter wie „cool" bereits eingedeutscht sind. Auch die Darstellung der einzelnen Analyseverfahren wurde nur selten betrachtet. Das Liniendiagramm im oberen Drittel der Software (s. Abb 5.5) wurde von den Proband:innen in der Hauptstudie deutlich seltener besprochen, als der zentral dargestellte Stimmungsdurchschnitt. Den Proband:innen war es meist wichtig, wie positiv oder negativ ihre Nachricht war. Auf welcher Grundlage diese Berechnung der Stimmung einer Aussage basierte, wurde meist nur am Anfang kurz betrachtet. In den späteren Verläufen der Studien wurde nur der vereinfachte Stimmungsdurchschnitt betrachtet. Zukünftig können solche Anzeigen für die einzelnen Stimmungsverfahren wohl ausgeblendet werden, sodass nur die Gesamtstimmung der zuletzt getätigten Eingabe angezeigt wird.

Erfahrungsabhängige Kommunikationsmuster
Darüber hinaus zeigen die in den Studien getätigten Eingaben, dass die Kommunikation sowohl von Kolleg:innen mit weniger Berufserfahrung untereinander als auch von erfahrenen mit weniger erfahrenen Kolleg:innen oft umfangreicher ist und Sachverhalte vermehrt umschrieben werden. Außerdem wurde Wert darauf gelegt, dass neutrale Aussagen getroffen wurden. Einige Kolleg:innen mit noch nicht so viel Berufserfahrung haben bei den Eingaben gesagt, dass sie professionell klingen wollen. Hierbei wurde eine gewisse Professionalität scheinbar immer mit Freundlichkeit gleichgesetzt. In der Studie sollten auch negative Sätze und Kritik geäußert werden, in diesem Zusammenhang kam dann von einigen Teilnehmer:innen oft der mündliche Einwand „So böse schreibe ich eigentlich nicht.". Eine Textnachricht, um „über Jemanden zu lästern" wurde in der Studie nicht verfasst. Einige Proband:innen gaben an, dass hierfür immer auf Face2Face, ein Telefonat oder einen privaten Messanger-Dienst wie WhatsApp ausgewichen wird.
Kolleg:innen, die schon lange zusammenarbeiten, schreiben deutlich kürzere und direktere Nachrichten. Hier werden auch negative Sachverhalte nicht umschrieben. Folglich schreiben erfahrene Entwickler:innen ihren ebenso erfahrenen Kolleg:innen „Das geht so gar nicht!", während jüngere Entwickler:innen eher dazu neigen, die Kritik zu umschreiben durch Formulierungen wie „Ich habe gesehen, dass du das Problem noch nicht gelöst hast. Das musst du nochmal verbessern.". Vor diesem Hintergrund wurde in der gesamten Studie negative Stimmung seltener detektiert als

positive.

Allgemein hat sich, nach Ansicht der Proband:innen, die Art der Kommunikation in den betrachteten Unternehmen durch die Verlagerung auf vermehrte Online-Meetings und Chats geändert. Fast alle Proband:innen haben in dieser Zeit sowohl in den Unternehmen, als auch im Home-Office gearbeitet. Dieser Wandel zu der vermehrten digitalen Kommunikationskanälen ermöglicht den vermehrten Einsatz solcher automatisierten Analysemöglichkeiten, wie er in dieser Arbeit vorgestellt wurde. Fast alle Entwickler:innen wollen aber zukünftig wieder mehr auf den persönlichen Austausch Face2Face setzten. Auch wenn dieser Umstand vor allem Online- und Präsenz-Meetings betrifft, kann es in verschiedenen Unternehmen auch vorkommen, dass die Kommunikation per Chats und E-Mail abnimmt.

7.4 Limitationen

Die Ergebnisse dieser Arbeit sind sowohl durch äußere als auch durch interne Voraussetzungen in ihrer Aussagekraft begrenzt. Es sind drei den äußeren Rahmenbedingungen geschuldete Aspekte, die die Aussagekraft der Ergebnisse dieser Arbeit begrenzen. Zum einen können die vier Expert:innen bzw. die zwölf Proband:innen, die an der im Rahmen dieser Masterarbeit durchgeführten Vor- bzw. Hauptstudie teilgenommen haben, mit ihrer Expertise den Bereich Software-Entwicklerteams nicht in vollem Umfang abbilden. Zum anderen war das Zeitfenster, in dem die Studien durchgeführt werden konnten, um die Umsetzbarkeit des in dieser Arbeit erstellten Konzeptes zu prüfen, limitiert, sodass damit einhergehend auch auch eine größere Datenerhebung nicht möglich war. Schließlich ist zu berücksichtigen, dass sowohl die Vorstudie als auch die Hauptstudie während der Coronavirus-Pandemie und somit unter entsprechenden Sicherheitsmaßnahmen durchgeführt werden mussten. Die Durchführung einer länger angelegten Studie, wird bei einer „normalisierten" Lage zudem einfacher durchzuführen.
Neben diesen drei äußeren Faktoren waren es zwei die Studie direkt betreffenden Aspekte, die Grenzen der Aussagekraft der Ergebnisse verdeutlichen, die aber in engem Zusammenhang mit den oben ausgeführten Rahmenbedingungen stehen. Dabei handelt es sich zum einen um die Genauigkeit der Stimmungsanalyse und um die Ermittlung des Durchschnitts der Teamstimmung. Im Rahmen dieser Arbeit wurde kein Verfahren vorgesehen, das die Genauigkeit der Stimmungsanalyse in einem wissenschaftlichem Wert abbildet. Da auch die Teilnehmerzahl für eine quantitative Studie zu gering war, wurde nur die subjektiv wahrgenommene Stimmung erfasst. In diesem Zusammenhang wurden längere Sätze in der Regel genauer wahrgenommen als kurze Aussagen. Etwa die Hälfte der Teilnehmer:innen hat auch ironische Aussagen getestet. Diese wurden vom Software-Assistenten meist als neutral interpretiert, was die „gefühlte Genauigkeit" bei den Proband:innen verschlechterte. Die Ergebnisse des Fragebogens und die mündlichen Aussagen der Proband:innen lassen insgesamt auf eine geschätzte Genauigkeit von 75% +- 10 % schließen. In dieser Arbeit wurde

ebenfalls nicht die Funktion zur Ermittlung der Teamstimmung weiter untersucht. Der zeitliche Rahmen dieser Arbeit reichte nicht aus, um eine Untersuchung dieser Funktion in der Studie mit abzudecken. Eine solche Betrachtung sollte ein einem oder mehreren fest definierten Teams über mehrere Monate untersucht werden. Es wurde gezeigt, dass die Ermittlung der Teamstimmung machbar ist und von Entwickler:innen akzeptiert werden kann, wenn alle Teammitglieder Einsicht in die Teamstimmung haben. Diese Arbeit behandelt aber nicht, wie Daten zur Teamstimmung ermittelt werden, um weitere Rückschlüsse auf diese Funktion zu erhalten.

7.5 Empfehlungen

Das in dieser Masterarbeit entwickelte Konzept stieß bei den Proband:innen der drei Unternehmen auf großes Interesse. In diesem Zusammenhang werden nachfolgend Vorschläge für weiterführende Untersuchungen gegeben.

Empfehlungen für die Praxis

Auch wenn für die im Rahmen dieser Arbeit durchgeführten Studie ein Keylogger verwendet wurde, um Textnachrichten von Entwickler:innen zu erfassen, ist die Entwicklung von verschiedenen Plug-ins auf unterschiedlichen Plattformen für länger angelegte Studien sinnvoller. Verschiedene Plattformen wie Microsoft Teams oder Telegramm binden zudem weitere Funktionen ein. Somit können Funktionen wie die Auswahl von Emoticons oder das Senden einer Nachricht mit der Enter-Taste besser berücksichtigt werden. Zusätzlich können die Ergebnisse auch direkt im Chat-Fenster angezeigt werden. Dazu können beispielsweise Texte wie bei einer Rechtschreibkontrolle farblich unterstrichen oder „gehightlighed” werden. Aber auch Verbesserungsvorschläge, Tipps und Tricks oder Erinnerungen für Lob und Dank können direkt integriert werden. So können in einem Plug-in alle Bausteine kombiniert werden. Für eine erste Studie wäre hier der Messanger-Dienst Microsoft Teams in Betracht zu ziehen. Zukünftig sollte zudem eine Cross-Plattform-Nutzung in Erwägung gezogen werden, um verschiedene Betriebssysteme zu unterstützen. Abbildung 7.1 zeigt einen mit Photoshop bearbeiteten Microsoft Teams-Screenshot, um eine Integration der Live-Sentiment-Analyse als Plug-in zu skizzieren.

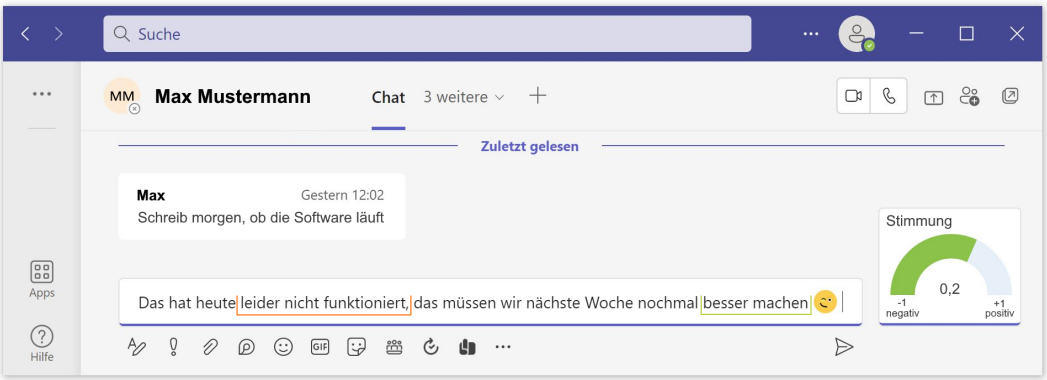

Abbildung 7.1: Plug-in Option (Mit Photoshop bearbeiteter Microsoft Teams-Screenshot)

Darauf aufbauend kann somit der Wunsch einiger Teilnehmer:innen erfüllt werden, um die Anwendung eines solchen Systems über einen längeren Zeitraum zu untersuchen. Zentral würde die Entwicklung eines Plug-in für MS Teams notwendig sein. Neben der Dauer einer Eingewöhnungsphase wird vor allem das längere Nutzungsverhalten interessant sein. Dabei haben die Proband:innen über folgende Forschungsfragen spekuliert:

- Werden Nachrichten (anfangs) anders formuliert und gezielt die Stimmung in Nachrichten zu verbessern?
- Inwiefern werden sich Routinen einstellen, sodass sich das Nutzungsverhalten nach den ersten Wochen bzw. Monaten normalisiert?
- Ist die Betrachtung aller Chats sinnvoll? Oder reichen zentrale Team- bzw. Gruppenchats?
- Welche Maßnahmen werden getroffen, wenn der Einblick in die Teamstimmung eine Verschlechterung der Stimmung signalisiert?

Um zukünftig diese Einzelstimmungen in einer gemeinsamen Teamstimmung nutzen zu können, kann ein Server im internen Netzwerk eines Unternehmens integriert werden. Auf diesem könnte ein MQTT-Client die anonymen Einzelnstimmungen zu einer Teamstimmung zusammenrechnen.

Empfehlungen für weiterführende Forschung

Weiterer Forschungsbedarf ergibt sich hinsichtlich der Genauigkeit der Stimmungsanalysen. Als zentrale Forschungsfrage kann untersucht werden, inwieweit die Analyse des Kontextes, beispielsweise des Chat-Verlaufes, die Genauigkeit der Stimmungsanalyse verbessert. Ironische Beiträge oder kurze Aussagen sind beim Lesen häufig nur über den Kontext zu verstehen. Ob ein „Ich glaub es nicht." positiv oder negativ gemeint ist, kann von einer anderen Aussage abhängen. Ein „Das sieht

super aus!" oder „Das geht überhaupt nicht!" in einer davor oder danach getätigten Aussage gibt dem „Ich glaub es nicht." erst die Bedeutung. Ironische Aussagen wie „Danke! Für nichts!" können auch über solche Kontextanalysen erfasst werden. Dass die Erkennung von Ironie und Sarkasmus weiterführender Untersuchen bedarf, zeigen bereits Obaidi und Klünder [16].

Inwieweit Kommunikationshäufigkeit, Antwortraten und Antwortverhalten einen Einfluss auf die Stimmung haben, konnte in dieser Arbeit ebenfalls nicht mit in die Analyse von Stimmung einfließen. Eine Probandin berichtete von einer langen verfassten Nachricht und erwarte als Antwort mehr als nur „okay". Personen, die schnell antworten, wünschen sich ein ähnliches Verhalten. Ähnlich werden bei langen Nachrichten, auch längere Antworten gewünscht.

Inwieweit Tipps und Hinweise positiveres Schreiben zu folgen haben, könnte in einer größer angelegten Studie ebenfalls mit untersucht werden. Für professionelles, positives und höfliches Schreiben könnten Formulierungsvorschläge auch direkt im Chat-Fenster mit angeboten werden. Ein Vorschlag für ein negatives „Entschuldigung für die Verspätung" könnte vom Assistenten durch ein positives „Danke fürs Warten" gegeben werden. Ein „Kein Problem" kann durch ein „Freut mich, dass ich helfen konnte" umformuliert werden. Analog wären:
- Ich denke wir sollten vielleicht -> Besser wäre es
- Ich musste viel schreiben -> Wir sollten das persönlich besprechen
- Ich hoffe das macht Sinn -> Lass mich wissen, wenn du Fragen hast
- Entschuldigung, das habe ich ganz vergessen -> Danke fürs Bescheid geben

Welche Erkenntnisse über die schriftlichen Kommunikation, auch in der mündliche Kommunikation (Telefon, Online-Meeting vllt. Face2Face) anzuwenden sind, könnte ebenfalls näher untersucht werden In diesem Zusammenhang hat Herrmann [11] in seiner Masterarbeit erste Erkenntnisse zu der Analyse von Meetings beigesteuert. Hier könnten neben der Umsetzung von Sprache in Text zur Analyse mit bekannten Verfahren auch mit anderen Metriken (Stimmlage, Lautstärke, Schwankungen in der Stimme etc.) gearbeitet werden. Das die Erkennung von Stimmung in Gesprächen,über die schriftliche Kommunikation hinaus, wichtigst ist, wurde bereits in der Vorstudie von den Experten geäußert.

Darüber hinaus wurde der Kontakt mit Kunden und Personen außerhalb des Teams von den Experten der Vorstudie als wichtiger Forschungsaspekt zukünftiger Arbeiten angesehen. Kommunikation ist in vielen Bereichen elementar, sodass zukünftige Stimmungsanalysen nicht nur verschieden Medien oder Kanäle (schriftlich, mündlich, hybrid) weiter analysieren sollten, sondern auch eine umfassendere Bandbreite an Nutzer betrachtet werden muss. Software-Entwicklerteams sind stark von Kommunikation abhängig und sind als Probandengruppe für diese Studie eine gute Wahl gewesen. Wie die Vision aber auch angedeutet hat, haben Stimmungsanalysen

durchaus das Potential, um in vielen Bereichen die Kommunikation zu optimieren. Der höfliche Umgang mit Kunden und Projektpartnern ist hierbei ein elementarer Baustein für den Projekterfolg, sowie darüber hinausgehend für die Zufriedenheit und die zukünftige Zusammenarbeit.

In dieser Arbeit wurden Alle Teammitglieder auf einer „Ebene" untersucht. Die Untersuchung von Hierachien auf die Art des Schreibens ist ebenfalls ein Aspekt, der zukünftig beleuchtet werden könnte. Nachrichten zum Chef:in oder Projektleiter:in werden anders formuliert als zum Kolleg:in. Ob oder inwieweit dieses Wechseln von Schreibarten Einfluss auf die Stimmung haben, kann allerdings noch nicht abgeschätzt werden.

Obwohl der entwickelte Assistent sowohl die Stimmungsanalysen von deutscher und englischer Sprache unterstützt hat, wurde ein Wechsel von Sprachen nicht betrachtet. Fraglich ist, ob dieser einen Einfluss auf die Stimmung hat. Wer selten in anderen Sprachen Nachrichten verfasst, wird nicht zwangsläufig auf positive Stimmung achten. Hier könnte ein Assistent mit Tipps und Formulierungsvorschlägen, für positive verfasste Nachrichten den Nutzer unterstützen.

Die folgenden Aspekte wurden im Rahmen der Studie mit einigen Proband:innen oder Expert:innen diskutiert. Daher werden die folgenden Punkte nur zur Vollständigkeit aufgeführt:

- Einfluss von Witzen auf die schriftliche Stimmung
- Erkennung von Rassismus
- Energie in Sätzen und die Verbindung zur Stimmung
- Verbesserung der Stimmung durch mehr gemeinsames Verständnis
  Weniger Missverständnisse und weniger technische Details mittels Planung von Kommunikation und Wissen
- Weitere Einbindung von Expert:innen der Bezugsdisziplinen wie Sprach- und Kommunikationswissenschaften
- Hermeneutik
- Barrierefrei (Blinde; Gehörlosigkeit)
- Symmetrische und asymmetrische Kommunikation
- Männer Frauen

Die angesprochenen Aspekte zeigen die Dimension bzw. die Interdisziplinarität des Themenkomplexes Stimmungsanalyse, ohne diese im Rahmen der Arbeiter weiter zu vertiefen.

Kapitel 8

Zusammenfassung und Ausblick

In diesem letzten Kapitel werden noch einmal alle wichtigen Ergebnisse dieser Arbeit kurz zusammengefasst. Abschließend wird ein Ausblick gegeben auf mögliche weiterführende Forschungsvorhaben.

8.1 Zusammenfassung

Die retrospektive Untersuchung der Stimmung von Teams im Kontext der Software-Entwicklung ist bereits Gegenstand einer Vielzahl von Arbeiten. Der Fokus dieser Masterarbeit lag auf der Beantwortung der Forschungsfragen, **inwiefern die Stimmung in einem Team präventiv auf der Grundlage von Textnachrichten detektiert werden kann** und **wie und unter welchen Bedingungen der Teamstimmung-Durchschnitt automatisiert berechnet werden kann**. Mit der prototypischen Implementation eines Software-Assistenten, der auf Basis eine „Live-Sentiment-Analyse" realisiert wurde, konnte die Praxistauglichkeit des Konzepts im Rahmen einer Studie unter Beweis gestellt werden. Es wurde nachgewiesen, dass das Konzept mittels Keylogger nicht nur in der Lage ist, beliebige Texteingaben zu erfassen, sondern auch mittels kombinierter Stimmungsanalysen die Textstimmung der Proband:innen zu interpretieren. Nachdem zu Beginn dieser Arbeit die Rahmenbedingungen für die Analyse von Nachrichten einzelner Entwickler:innen gelegt wurden, stelle sich im Rahmen der Vorstudie heraus, dass der Mehrwert nicht zwangsläufig gegeben sei. Dieses Defizit und führte zu einigen Verbesserungen an dem Software-Assistenten. Die Implementierung einer auf Basis der Einzelstimmungen berechneten Teamstimmung wurde in diesem Kapitel als weitere Entwicklungsphase weiter ausgeführt. Die Analyse des Soziolektes wurde in diesem zweiten Entwicklungsschritt ebenfalls implementiert, damit gruppenspezifische Aussagen erkannt und in die Analyseergebnisse mit einfließen können.

8.2 Ausblick

Die in dieser Arbeit vorgestellten Ergebnisse liefern eine Grundlage für weiterführende Forschungsvorhaben im Bereich der automatisierten Erfassung der Teamstimmung. Hierfür wurden in Kapitel 7.5 bereits sehr konkrete Vorschläge für Folgeforschungsprojekte zusammengestellt. Potenzial zur Verbesserung liefer in jedem Fall der vorgestellte Software-Assistent, der in seiner Genauigkeit noch begrenzt ist. Zum einen hat der Kontext unmittelbaren Einfluss auf die Analyseverfahren, sodass hier weitere Untersuchungen im Chatverlauf erforderlich sein könnten. Zum anderen sind für solche Forschungsvorhaben die Erfassung von Daten über einen längeren Zeitraum sinnvoll. Eine Implementation des Konzeptes als Plug-in für Microsoft Teams wird hier als nächster Entwicklungsschritt empfohlen. Die freiwillige und anonyme Weiterleitung von Einzelstimmungen zur Bildung eines Teamdurchschnittes sollte in diesem Rahmen ebenfalls weiter untersucht werden. Die Betrachtung eines oder mehrerer Teams sollte hier über einen längeren Zeitraum erfolgen. Diese Masterarbeit leistet durch das vorgestellte Konzept und der Live-Sentiment-Analyse einen Beitrag zur Echtzeit-Erkennung von Stimmung in Textnachrichten und stellt somit eine Alternative zur retrospektiv und zentral durchgeführten Stimmungsanalyse in Entwicklungsteams dar.

Anhang A

Unterlagen zur Hauptstudie

Im Rahmen der Hauptstudie zur Evaluation des Konzeptes bekamen die Proband:innen verschiedene Dokumente. Die Einladung zur Studie erfolgte per E-Mail (s. Anhang A.1). Den Fragebogen (s. Anhang A.2) erhielten die Proband:innen in zu Beginn der Studie.

A.1 Einladung

Guten Morgen in die Runde,

Im Rahmen meiner Masterarbeit beschäftige ich mich mit der Echtzeitanalyse von Stimmung in Nachrichten zwischen Software Entwicklern.
(Der Titel der Arbeit: „Konzept für eine präventive Stimmungsanalyse von Nachrichten in Software-Entwicklungsteams")
Für diesen Zweck suche ich freiwillige Probanden aus der Software-Entwicklung, die sich 30-45 min Zeit nehmen können, um neben einem kurzen Fragebogen einen Software Prototypen zu testen. Diese soll in Text-Nachrichten (Chats, E-Mails, etc.) detektieren, wie die Stimmung beim Gegenüber ankommen kann.

Da dies sehr subjektiv ist, soll diese erste qualitative Studie die Machbarkeit einer Live-Sentiment-Analyse prüfen.
Wichtig zu erwähnen ist, dass hier zwar Nachrichten von euch analysiert werden, diese aber nicht gespeichert oder weitergegeben werden.
Hier geht es nicht darum eine perfekte Stimmungsanalyse zu entwickeln oder zu verbessern, sondern nur um die Machbarkeit zu zeigen.
In den Prototypen werden verschiedene Analyseverfahren genutzt, um einen Durchschnitt der Stimmung eurer Nachrichten lokal zu berechnen.
Es wird in dieser Arbeit ebenfalls überprüft ob es sinnvoll wäre diesen Durschnitt anonymisiert an den Projektleiter oder Qualitätsbeauftragten weiter zu senden, damit ggf. präventive Maßnahmen gegen schlechte Stimmung getroffen werden können.

Da diese Arbeit ein Proof-of-concept ist, könnte mit der erfolgreich durchgeführten Studie ein neuer Bereich in der Forschung eröffnet werden.
Zurzeit ist nur eine nachgeschaltete Analyse von Stimmung möglich, die auf persönliche Daten zugreifen muss. Mit dem neuen Konzept soll es zukünftig möglich sein, präventive Stimmungsanalysen durchzuführen und frühzeitig zu deeskalieren oder Verbesserungen im Team vor zu nehmen. Ihr habt somit die Möglichkeit Ideen mit rein zu bringen, welche vielleicht auch Auswirkungen die zukünftige Entwicklung in Software-Teams hat.

Ablauf:
5 min Einführung ins Thema
5 min Fragebogen Teil 1
5 min Prototyp erkunden
10 – 20 min Prototypen testen
5 min Fragebogen Teil 2

Die Anmeldung ist unter
https://doodle.com/poll/9bkizsd47cmevex5?utm_source=poll&utm_medium=link
möglich.

Sollte kein Termin zeitlich passen, können auch Termine in der nachfolgenden Woche gemacht werden.

Für Rückfragen stehe ich gerne unter Lennart.Schroth@atsonline.de zur Verfügung.

Mit freundlichen Grüßen,
Lennart Schroth

Anhang:

Szenario:

Eine Projektleiterin eines Teams mit relativ jungen Entwicklern setzt ein Online-Meeting an.
Das Meeting verläuft unter Zeitdruck und nicht alle notwendigen Informationen und Arbeitsschritte werden thematisiert.
Da sie auch keine Zeit für Rückfragen hat, ist die Stimmung bei dem jungen Team etwas angespannt.
Um ihrem Unmut etwas Luft zu verschaffen, tauschen sich die Entwickler mit direkt Nachrichten aus.
Im Hintergrund läuft nun die Live-Analyse der geschriebenen Nachrichten.
Aus verschiedenen Analysen, die die Entwickler live sehen können, wird ein Durchschnitt berechnet.
Zwischen 0 und 1 wäre er positiv, wie die Stimmung der verfassten Nachrichten.
Aber nach dem Meeting schwankt der Durchschnitt immer zwischen 0 und -1 und verdeutlicht die negative Stimmung.
Ohne persönliche Details werden nur die Durchschnitte der Entwickler anonym ausgewertet.
Der Projektleiter kann die Stimmung in einem kleinen Fenster auf seinem Rechner nachverfolgen und bemerkt, dass nach dem Meeting die Stimmung ziemlich gekippt ist.
Bevor es nun eskaliert, nimmt sich die Projektleiterin die Zeit, um Fragen oder Unklarheiten zu klären.

A.2 Fragebogen mit Ergebnissen

Leibniz Universität Hannover

Konzeptevaluation eines Assistenten zur präventiven Stimmungsanalyse von Nachrichten in Software-Entwicklungsteams

Fragebogen zur Studie (Teil 1)

Teilnehmer-Nr.	Freiwillige Kontaktdaten	Datum
		18.01 – 25.01.2022

1. Was ist ihr Beruf (Entwickler, Projektleiter, Student, …)?

7x Entwickler, 2x Technische Redaktion, 2x Projektleiter, 1x IT-Consultant
4 Frauen, 8 Männer

2. In welcher Branche sind Sie tätig bzw. wollen Sie zukünftig arbeiten? (Informations-/ Telekommunikation, Automobilbau, Elektroindustrie, Logistik, …)

Drei Unternehmen aus der Informations-/ Telekommunikation (in und um Hannover)

3. Wie lange sind Sie schon ungefähr in der Branche tätig?

1 **9,6** **24 Jahre** **Durchschnittlich 9,66 Jahre**

Bewerten Sie folgende Aussagen	Trifft voll zu 5	Trifft eher zu 4	Trifft mittel zu 3	Trifft eher nicht zu 2	Trifft gar nicht zu 1
4.1 Ich erhalte genügend Informationen zur Ausübung meiner Aufgaben.				Durchschnitt 4,08	
4.2. Es gibt nur selten Rückfragen zu meinen Tätigkeiten und erledigen Arbeiten.					3,5
4.3. Es werden im Team regelmäßig Team-Meetings zum Austausch der aktuellen Projekte durchgeführt.			4,67		
4.4. Es werden Informationen auf dem „Flur“ geteilt, die besser durch die Geschäftsführung geteilt werden sollten, um eine Verzerrung der Informationen zu verhindern.					2,45
4.5 Ich interessiere mich für Kommunikation im Allgemeinen (auch Theorien oder ähnliches).				4,25	
4.6 Ich versuche stets eine positive Stimmung im Team zu verbreiten.				4,17	
4.7 Ich versuche auch Nachrichten positiv zu formulieren.				4,08	

1

5. Schätzen Sie die Häufigkeit Ihrer aktuellen Kommunikationswege (verteilen Sie 100% an:)	
Direkt (face 2 face)	**18,3 % (Standardabweichung 17,1)**
Video-Online-Meetings	**34,6 % (Standardabweichung 11,6)**
Telefon-Anruf	**5,4 % (Standardabweichung 8,28)**
Chat-Nachrichten (z.B. MS Teams)	**22,1 % (Standardabweichung 8,53)**
E-Mail	**19,6 % (Standardabweichung 11,4)**
Sonstige (Handschriftlich, …):	

6. Was wünschen Sie sich für eine optimale interne Kommunikation?
4x Regelmäßige Abstimmungsmeetings ohne redundante Inhalte (Mit Kommunikation über MS Teams als Standard sind alle zufrieden) 4x Mehr persönlicher Austausch 2x Zentrale Chats für Informationen von Projekten, Keine Projektdetails in persönlichen Chats 2x Weniger Telefonate und E-Mails (gerade im Homeoffice) 1x Ausgeglichenes Verhältnis zwischen Meetings und Arbeitsphasen 1x Zuverlässige Beantwortung schriftlicher Kommunikation 1x Videos Meetings mit Kamera (Keine Pflicht, nicht zu früh am Morgen) 1x Kritik bei jüngeren professionell und freundlich halten 1x Gemeinsames Verständnis - Einheitliche Vokabeln (OR, FDS, Frameworks) 1x Teamspezifische Kommunikation (Wenn habe ich in der Kommunikation – welches Wissen hat der Teilnehmer) 1x Zielorientierte Kommunikation (Von Grob nach Fein) ggf. nicht alles in einem Meeting klären wollen.

Vielen Dank für das Beantworten des ersten Teils des Fragebogens.

Sie bekommen nun den Software-Prototypen, um Erfahrungen mit dem System zu machen.

Im Anschluss wird der zweite Teil des Fragebogens ausgefüllt.

Leibniz Universität Hannover

Fragebogen zur Studie (Teil 2)

Bewerten Sie folgende Aussagen	Trifft voll zu 5	Trifft eher zu 4	Trifft mittel zu 3	Trifft eher nicht zu 2	Trifft gar nicht zu 1
1.1 Die Analyseergebnisse entsprachen meiner Stimmung.				Durchschnitt 4,25	
1.2. Die verschiedenen Analyseverfahren haben verschiedene Stimmungen gut erfasst.				4,18	
1.3. Eine solche Software könnte in der Zukunft hilfreich sein.				4,33	
1.4 Ich hatte nur wenige Sicherheitsbedenken beim Nutzen dieser Software (Datenschutz).					3,42
1.5 Für eine bessere Stimmung im Team würde ich meinem Projektleiter, Qualitätsbeauftragten, … den Durchschnitt meiner Stimmungsanalysen zur Verfügung stellen.				4,25	
1.6 Die durchschnittliche Team-Stimmung würde mich auch selber interessieren.			4,83		

2. I wish – I like – i wonder Feedback

I like
- anpassbare Soziolekt-Lexikonverwaltung (6 positive Probanden)
- Grafische Darstellung gut und verständlich (3)
- eine neue Möglichkeit sich auszudrücken (1)

I wish
- Praxistest über längeren Zeitraum (2)
- Live Anzeige der Stimmung im Chatfeld (6)
- Eingehende Nachrichten bewerten, um Analyse zu verbessern (4)
- Projektbezogene Team-Chats (1)
- Team-Stimmung für bestimmte Chats und Gruppen und Teams (man sieht nur sein Team) (1)
- ähnliche Anzeige wie eine Rechtschreibkorrektur (Grün/rot unterstrichen) oder Prüf-Button (mit Tastenkürzel)(2)

I wonder
- Mögliches Gefühl von Beobachtung/ Überwachung (1)
- Weniger kritisch und keine direkte Meinungsäußerung (1)
- Spaltung von Team, wenn man Durchschnitt nicht teilen möchte (1)

Bedingung von 10 Personen: 1.5 nur wenn 1.6 gilt
- Alle müssen die Teamstimmung sehen können und nicht nur der Projektleiter (der auch mitmachen muss/soll)

Abbildungsverzeichnis

2.1 MQTT Architektur mit Beispiel 13

4.1 Bausteinansicht eines „Live-Sentiment-Analyse”-Systems (Aktivitätsdiagramm erstellt mit www.plantuml.com) 28

5.1 Detaillierter FunktionsMASTER mit Bedingung - Anforderungsschablone [25] 30
5.2 Überblick über die Bausteine und den Prozessablauf eines „Live-Sentiment-Analyse”-Systems 32
5.3 UML-Sequenzdiagramm Live-Sentiment-Analyse (plantuml.com) 33
5.4 UML-Sequenzdiagramm zum MQTT-Datenaustausch 34
5.5 Screenshot des Software-Assistenten 40
5.6 Abbildung des Software Prototypen 41

6.1 Personalisierung der Soziolekt-Analyse 48
6.2 Erfahrungen der Proband:innen in der Branche 51
6.3 Fragebogen Teil 1 - Bewertung der Aussagen 4.5, 4.6 und 4.7 54
6.4 Genutzte Kommunikationswege der Proband:innen in der Hauptstudie . 55
6.5 Ergebnisse Fragebogen Teil 2 - 1.1 bis 1.3 Aussagen bewerten 57
6.6 Ergebnisse Fragebogen Teil 2 - 1.4 57
6.7 Ergebnisse Fragebogen Teil 2 - 1.5 und 1.6 Aussagen bewerten 58

7.1 Plug-in Option (Mit Photoshop bearbeiteter Microsoft Teams-Screenshot) 67

Literaturverzeichnis

[1] G. Stepanek, *Software Programming Secrets: Why Projects Fail.* APress, 2012.

[2] R. Kaur and K. K. Chahal, "Analysis of developers' sentiments in commit comments," in *International Conference on Advanced Informatics for Computing Research*, pp. 3–12, Springer, 2020.

[3] E. Bonnie, "Complete collection of project management statistics 2015," *Retrieved February*, vol. 1, p. 2016, 2015.

[4] L. Waizenegger, B. McKenna, W. Cai, and T. Bendz, "An affordance perspective of team collaboration and enforced working from home during covid-19," *European Journal of Information Systems*, vol. 29, no. 4, pp. 429–442, 2020.

[5] B. Pang, L. Lee, *et al.*, "Opinion mining and sentiment analysis," *Foundations and Trends® in information retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.

[6] B. Liu, *Sentiment analysis: Mining opinions, sentiments, and emotions.* Cambridge university press, 2020.

[7] R. S. Lazarus, *Emotion and adaptation.* Oxford University Press, 1991.

[8] S. J. McAllister, A. Vincent, A. L. Hassett, M. O. Whipple, T. H. Oh, R. P. Benzo, and L. L. Toussaint, "Psychological resilience, affective mechanisms and symptom burden in a tertiary-care sample of patients with fibromyalgia," *Stress and Health*, vol. 31, no. 4, pp. 299–305, 2015.

[9] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "Sentiment strength detection in short informal text," *Journal of the American society for information science and technology*, vol. 61, no. 12, pp. 2544–2558, 2010.

[10] J. Horstmann, "Computer-gestützte Analyse des Kommunikationsverhaltens in Entwicklerteams unter Berücksichtigung digitaler Medien," *Master's thesis, Gottfried Wilhelm Leibniz Universität Hannover*, 2019.

[11] M. Herrmann, "Automatische Klassifikation von Aussagen in Meetings von Entwicklungsteams," *Bachelor's thesis, Gottfried Wilhelm Leibniz Universität Hannover*, 2021.

[12] H. Wang and J. A. Castanon, "Sentiment expression via emoticons on social media," in *2015 ieee international conference on big data (big data)*, pp. 2404–2408, IEEE, 2015.

[13] M. R. Islam and M. F. Zibran, "Leveraging automated sentiment analysis in software engineering," in *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*, pp. 203–214, IEEE, 2017.

[14] F. Calefato, F. Lanubile, F. Maiorano, and N. Novielli, "Sentiment polarity detection for software development," *Empirical Software Engineering*, vol. 23, no. 3, pp. 1352–1382, 2018.

[15] T. Ahmed, A. Bosu, A. Iqbal, and S. Rahimi, "Senticr: a customized sentiment analysis tool for code review interactions," in *2017 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 106–111, IEEE, 2017.

[16] M. Obaidi and J. Klünder, "Development and application of sentiment analysis tools in software engineering: A systematic literature review," *Evaluation and Assessment in Software Engineering*, pp. 80–89, 2021.

[17] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment strength detection for the social web," *Journal of the American Society for Information Science and Technology*, vol. 63, no. 1, pp. 163–173, 2012.

[18] N. Novielli, F. Calefato, D. Dongiovanni, D. Girardi, and F. Lanubile, "Can we use se-specific sentiment analysis tools in a cross-platform setting?," in *Proceedings of the 17th International Conference on Mining Software Repositories*, pp. 158–168, 2020.

[19] A. Specht, "Identifikation von relevanten Metriken zur Analyse von Kommunikation in Entwicklerteams," in *Masterarbeit*, 2021.

[20] U. Waltinger, "Sentiment analysis reloaded-a comparative study on sentiment polarity identification combining machine learning and subjectivity features.," in *WEBIST (1)*, pp. 203–210, 2010.

[21] R. Remus, U. Quasthoff, and G. Heyer, "Sentiws-a publicly available german-language resource for sentiment analysis," in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, 2010.

[22] J. Klünder, J. Horstmann, and O. Karras, "Identifying the mood of a software development team by analyzing text-based communication in chats with machine learning," in *International Conference on Human-Centred Software Engineering*, pp. 133–151, Springer, 2020.

[23] K. Lauenroth, E. Sikora, and K. Pohl, "Positive Effekte von Szenarien und Features in einem Softwarepraktikum.," in *SEUH*, pp. 27–40, 2007.

[24] M. Broy and K. Webers, "Anforderungsmanagement in der Softwareentwicklung," *ATZelektronik*, vol. 8, no. 1, pp. 64–69, 2013.

[25] C. Rupp and R. Joppich, "Anforderungsschablonen - der MASTER-plan für gute Anforderungen," p. 215–246, 2014.

[26] T. Y. LE, "Anwendung von natural language processing zur Evaluierung der Anforderungsbeschreibungen," 2021.

[27] C. Rupp, M. Simon, and F. Hocker, "Requirements engineering und management," *HMD Praxis der Wirtschaftsinformatik*, vol. 46, no. 3, pp. 94–103, 2009.

[28] K. Pohl, *Requirements engineering fundamentals: a study guide for the certified professional for requirements engineering exam-foundation level-IREB compliant.* Rocky Nook, Inc., 2016.

[29] R. C. Martin, *Clean Code-Refactoring, Patterns, Testen und Techniken für sauberen Code: Deutsche Ausgabe.* MITP-Verlags GmbH & Co. KG, 2013.

[30] M. Lewrick, P. Link, and L. Leifer, *The design thinking toolbox: A guide to mastering the most popular and valuable innovation methods.* John Wiley & Sons, 2020.

Erklärung der Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbständig und ohne fremde Hilfe verfasst und keine anderen als die in der Arbeit angegebenen Quellen und Hilfsmittel verwendet habe. Die Arbeit hat in gleicher oder ähnlicher Form noch keinem anderen Prüfungsamt vorgelegen.

Hannover, den 19. April 2022

Lennart Schroth