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

Anwendung von sozialer Netzwerkanalyse auf Meetings in Softwareprojekten - Applying Social Network Analysis to Meetings in Software Projects

Masterarbeit

im Studiengang Informatik

von

Pia Lehmann

Prüfer: Prof. Dr. rer. nat. Kurt Schneider
Zweitprüfer: Dr. rer. nat. Jil Ann-Christin Klünder
Betreuer: Dr. rer. nat. Jil Ann-Christin Klünder

Hannover, 12. September 2021

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 12. September 2021

Pia Lehmann

Zusammenfassung

Softwareprojekte brauchen Teamarbeit. Hierbei spielen viele Faktoren eine Rolle, wie soziale Konflikte und Kommunikation, die sich auf die Stimmung, Motivation und damit auch auf die Produktivität des Teams auswirken. Besonders viel wird dabei in Meetings kommuniziert. Ein unangemessener Umgang miteinander sowie Teilnehmer, die sich nicht am Meeting beteiligen, können hier oft für Probleme sorgen.

In dieser Arbeit wird soziale Netzwerkanalyse eingesetzt, um Muster in der Kommunikation und den Beziehungen innerhalb von Meetings zu erkennen. Hierfür werden aus den Meetings Netzwerke konstruiert, die die Kommunikation und Beziehungen der Entwickler repräsentieren. Zudem werden verschiedene Maße definiert, mit denen Schwachstellen und Probleme gefunden werden können.

Das entwickelte Konzept wurde anschließend auf 38 Meetings studentischer Software-Projekte angewandt und evaluiert. Dabei konnte kein eindeutiger Zusammenhang zwischen den Kennzahlen der Netzwerke und der Leistung und Stimmung des Teams nachgewiesen werden. Es sind lediglich Tendenzen sichtbar. Es zeigen sich jedoch Auffälligkeiten in einigen Netzwerken, die auf mögliche Problemstellen in der Beziehungsstruktur der Meetingteilnehmer hinweisen.

Abstract

Software projects need teamwork. For this many aspects are relevant like social conflicts and communication which can influence the mood, motivation and therefore the productivity of the team. Especially in meetings there is a lot of communication. Treating each other inapropriatly as well as attendees who don't participate in the meeting can often cause a lot of problems here.

In this work social network analysis is used to identify patterns of communication and relationships during meetings. For this, networks are constructed from meetings representing the communication and relationships of the developers. Additionally a couple of different measurements are defined to help identify possible problem spots.

The developed concept was then applied to 38 meetings of student software projects and evaluated. It wasn't possible to prove a clear connection between the key figures of the networks and the performance and mood of the teams but there where tendencies visible. There were however abnormalities visible in some networks that point out problem areas in the structure of relationships of the developers.

Inhaltsverzeichnis

Kapitel 1

Einleitung

Software ist aus der modernen Welt nicht mehr wegzudenken. Egal ob in Autos, Fahrkartenautommaten oder dem privaten Smartphone, überall kommt Software zum Einsatz. Der Bereich der Software-Entwicklung ist daher schon länger immer weiter in den Fokus geraten.

Die meisten Software-Projekte sind dabei heutzutage viel zu groß und umfangreich, um sie als eine einzelne Person bewältigen zu können. In der Regel wird als Team zusammen gearbeitet. Bei besonders großen Projekten, wie sie heutzutage in beinahe allen Bereichen der Software-Entwicklung vorkommen, können sogar Entwickler aus verschiedenen Nationen gemeinsam arbeiten. Eine gute Zusammenarbeit ist hier unabdingbar für ein erfolgreiches Projekt.

Ein großer Aspekt innerhalb der Software-Entwicklung sind Meetings [37]. Egal ob mit Kunden oder im Entwicklerteam, Meetings sind fester Bestandteil des Entwicklungsprozesses. Innerhalb eines Meetings kann viel kommuniziert werden und es sind mehrere, teils sogar alle, Entwickler aus dem Team beteiligt. Aus diesem Grund wird das Verhalten der Entwickler in Meetings bereits in einigen Forschungsarbeiten als Grundlage für Analysen der Teamkommunikation und der Zusammenarbeit genutzt [21].

Auch die Analyse der Beziehungen zwischen den Entwicklern gerät hierbei langsam mehr in den Fokus. Soziale Konflikte können negativen Einfluss auf die Stimmung haben. Die Stimmung innerhalb des Teams wiederum hat einen nicht zu vernachlässigenden Einfluss auf den Erfolg eines Projekts [10] [8].

Für die Analyse von Beziehungen und Meetings ist die soziale Netzwerkanalyse verbreitet. Sie konzentriert sich unter anderem auf die Darstellung und Analyse von sozialen Strukturen, Informationsflüssen und Beziehungen [44].

In dieser Arbeit soll die Kommunikation von Entwicklern in Meetings im Rahmen von Software-Projekten analysiert werden. Ziel ist es, mit Hilfe der sozialen Netzwerkanalyse, die Beziehungen zwischen den Entwicklern zu erkennen, einzuschätzen und einen Zusammenhang zwischen diesen Beziehungen und der Team-Stimmung und -Leistung nachzuweisen.

1.1 Problemstellung

Soziale Konflikte innerhalb eines Teams können den Erfolg des Projekts gefährden [10]. Schnittstellen innerhalb des Programmcodes spiegeln immer auch ein Stück weit die Kommunikation zwischen den Entwicklern wieder [5]. Liegen Spannungen zwischen zwei Mitgliedern des Teams vor, so ist die Stimmung angespannt, das Arbeitsumfeld geladen und die Zusammenarbeit geht eher schleppend voran.

Aus diesem Grund ist ein frühzeitiges Erkennen von sozialen Konflikten und negativen Beziehungen sehr wichtig. Nur auf diese Weise kann rechtzeitig gegen gesteuert werden. Eventuell sind klärende Gespräche nach einer Auseinandersetzung oder jahrelang verstecktem Groll notwendig. Vielleicht ist die Situation auch noch nicht so kritisch, sollte aber im Auge behalten werden. Alle diese Schritte lassen sich nur dann Einleiten, wenn das Problem rechtzeitig bekannt ist.

Leider sind Menschen bei Fragen nach der persönlichen Meinung über Kollegen und die eigene Beziehung mit ihnen nicht immer ehrlich. Viele Menschen scheuen sich davor, soziale Konflikte direkt anzusprechen. Gründe hierfür können Angst vor Konfrontation verschiedener Auffassungen oder den Konsequenzen eines Gesprächs sein [30]. Zudem können diese Nachfragen von den Entwicklern als unangenehm empfunden werden und sie aufgrund des Zeitaufwandes von der Arbeit abhalten.

Das Problem, das sich daher nun stellt, ist eine andere Möglichkeit zu finden, die Beziehung und soziale Konflikte zwischen Entwicklern erkennen und analysieren zu können, ohne dabei nur auf die Aussagen der betroffenen Personen angewiesen zu sein. In dieser Arbeit wird das Problem behandelt, wie aus dem Verhalten und der Kommunikation der Entwickler innerhalb von Meetings auf die Beziehungen der Entwickler untereinander geschlossen werden kann und ob die so ermittelten Werte einen Einfluss auf die Team-Stimmung und -Leistung haben.

Eine Möglichkeit aus Meetings auf Beziehungen zu schließen wäre hilfreich als Unterstützung bei der Verbesserung der Team-Stimmung. Da Meetings in der Software-Entwicklung für gewöhnlich regelmäßig mit mehreren oder allen Entwicklern des Teams stattfinden, bieten sie eine

gute Datengrundlage. Eine Analyse dieser könnte auf Konflikte innerhalb eines Teams hindeuten, die so weiter beobachtet oder angesprochen werden können. Allerdings muss hier, beispielsweise mittels einer Anonymisierung der Daten, auf die Rechte der Entwickler und den Datenschutz Rücksicht genommen werden.

1.2 Lösungsansatz

In dieser Arbeit werden soziale Netzwerke verwendet, um das Verhalten und die Kommunikation der Entwickler innerhalb von Meetings darzustellen und Beziehungen zu analysieren.

Hierfür werden aus Transkripten von Meetings soziale Netzwerke konstruiert. In der Gesamtauswertung soll so ein Netzwerk entstehen, dessen Kantengewichte eine Tendenz über die Positivität der Beziehung zwischen den entsprechenden Entwicklern angibt. Ein hohes, positives Kantengewicht steht hierbei für eine gute Beziehung und negative Kantengewichte deuten eher auf ein schlechtes Verhältnis hin.

Dieser Ansatz basiert darauf, dass Kommunikation als ein Ausdruck von Beziehung gesehen wird und durch Kommunikation zweier Menschen unweigerlich auch eine Beziehung zwischen ihnen entsteht [26]. Die Beziehung zweier Menschen ist dabei stark davon abhängig, wie viel sie miteinander reden [27].

Die Häufigkeit, mit der wir mit einem anderen Menschen sprechen, ist allerdings nicht allein ein zu berücksichtigender Faktor für die Beziehung. Menschen mit unterschiedlichen Persönlichkeiten sprechen unterschiedlich viel [16]. So kann eine eher ruhige Person trotz weniger Kommunikation ein besseres Verhältnis zu seinem Kollegen haben, als jemand, der gern und viel redet. Aus diesem Grund wird in dem hier verfolgten Lösungsansatz zusätzlich zur Kommunikationshäufigkeit auch die Art und Weise, mit der kommuniziert wird, einbezogen. Dabei wird auf Dinge wie Lob, Ermutigung, aber auch Beleidigungen und Ähnliches geachtet.

Der hier verfolgte Lösungsansatz hat den Vorteil, dass eine Analyse der Kommunikation in Meetings weniger durch die Bereitschaft der Entwickler, über soziale Konflikte zu sprechen, abhängig ist. Auf diese Weise können Tendenzen schlechter Beziehungen, die sich in der Kommunikation zeigen oder aus viel negativer Kommunikation entstehen können, erkannt werden. Diese würden von den Beteiligten sonst erst zu spät oder gar nicht angesprochen werden. So kann das Problem unausgesprochenen Grolls

behoben werden, bevor die Situation unter Umständen eskaliert und den Erfolg des Projekts gefährdet.

Um den hier vorgestellten Lösungsansatz später auch evaluieren zu können, wird das Konzept in Kapitel 6 auf einen Datensatz angewendet. Hier wird dann nach einem Zusammenhang zwischen den ermittelten Beziehungen und der Team-Stimmung und -Leistung gesucht. Allerdings können die Netzwerke auch ohne einen solchen Zusammenhang Tendenzen über die Art und Weise der Kommunikation der Entwickler untereinander liefern. So kann auf eine positivere Kommunikation hingearbeitet werden.

1.3 Struktur der Arbeit

Diese Arbeit ist wie folgt strukturiert:

In Kapitel 2 werden alle für die Arbeit notwendigen Grundlagen beschrieben. Die hier behandelten Aspekte sind für das Verständnis der Arbeit unabdingbar.

Kapitel 3 liefert eine Übersicht über bereits existierende Arbeiten im Bereich der Meetinganalyse, sozialen Netzwerkanalyse und Kommunikationsanalyse, sowie eine Abgrenzung dieser Arbeit von den Beschriebenen.

In Kapitel 4 wird ein Konzept zur Lösung des beschriebenen Problems erarbeitet und erläutert.

Kapitel 5 befasst sich mit der Implementierung des erarbeiteten Konzepts als Prototyp. Hier werden alle Besonderheiten, wichtige Informationen und verwendete Librarys für die Implementierung beschrieben.

In Kapitel 6 findet eine Evaluation des erarbeiteten Konzepts statt. Dafür wird der implementierte Prototyp auf einen Datensatz angewendet und ein Bezug zu Team-Stimmung und -Leistung hergestellt.

Kapitel 7 befasst sich mit der Interpretation der Ergebnisse der Evaluation. Außerdem werden hier Einschränkungen der Ergebnisse durch Methodik und Datenlage behandelt.

In Kapitel 8 wird ein Überblick über die gesamte Arbeit gegeben sowie ein Ausblick auf die Möglichkeiten zukünftiger Arbeiten.

Kapitel 2

Grundlagen

Im folgenden Kapitel werden die für das Verständnis der Arbeit notwendigen Grundlagen vermittelt.

2.1 Soziale Netzwerkanalyse

Netzwerke bestehen aus einer Menge von Elementen, zum Beispiel ein Mitarbeiter, und Beziehungen zwischen diesen Elementen [15]. Formal kann ein Netzwerk durch einen Graph repräsentiert werden. Ein Graph (V, E) besteht aus einer Knotenmenge V und einer Kantenmenge $E \subseteq (V \times V)$. Den Kanten kann über eine Gewichtsfunktion $d: E \rightarrow \mathbb{R}$ ein Kantengewicht zugeordnet sein. Die Visualisierung eines ungerichteten Graphen mit Kantengewichten ist in Abbildung 2.1 dargestellt.

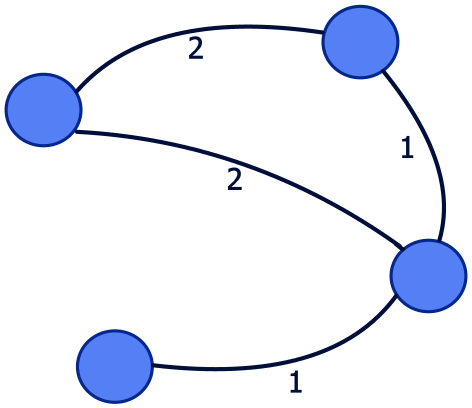

Abbildung 2.1: Ungerichteter Graph

Ein Netzwerk kann viele verschiedene Beziehungen und Strukturen repräsentieren. Durch eine Netzwerkanalyse können Besonderheiten oder

Unstimmigkeiten in einer Struktur aufgezeigt werden und diese dementsprechend optimiert werden. Da diese Arbeit sich mit der Analyse von zwischenmenschlichen Beziehungen im Kontext von Meetings der Software Entwicklung beschäftigt, werden hier soziale Netzwerke eingesetzt.

Ein soziales Netzwerk besteht aus einer endlichen Menge von Akteuren und den zwischen ihnen definierten Beziehungen [24]. Hierbei werden in der Regel die sozialen Strukturen oder Informationsflüsse betrachtet. Die soziale Netzwerkanalyse zielt darauf ab, unterschiedliche Strukturen und Charakteristika zu untersuchen, um bestehende Probleme oder Risiken zu erkennen und Verbesserungen an der vorhandenen Struktur oder Lage einleiten zu können. Sie können Aufschluss darüber geben, wie Informationen in einem Netzwerk von Personen verteilt werden [3]. Ein soziales Netzwerk kann auch als Graph G definiert werden [20]:

$$
\begin{gathered}
G = (V, E), \qquad (2.1) \\
mit \\
\text{Indexmenge } I \\
V = \{v_i : i \in I\} : \text{ die Menge der Knoten} \\
E = \{(v_j, v_k) : v_j, v_k \in V \times V\} : \text{ die Menge der Kanten}
\end{gathered}
$$

In der sozialen Netzwerkanalyse treten unterschiedliche Typen von Netzwerken auf. Im Bereich der Software-Entwicklung sind vor allem drei Arten verbreitet. Im Rahmen dieser Arbeit werden verschiedene Netzwerke konstruiert, um die Beziehung von Entwicklern anhand von Meetings zu analysieren.

Ein **Kollaborations-Graph oder -Netzwerk** stellt die Zusammenarbeit der Akteure dar. Eine Kante (v_i, v_j) bedeutet hierbei, dass die Akteure v_i und v_j bereits zusammengearbeitet haben. Die genauere Bedeutung von Zusammenarbeit ist hierbei anhängig von dem jeweiligen Kontext [41]. Ein Kollaborationsnetzwerk könnte beispielsweise darstellen, welche Mitarbeiter in einem großen Unternehmen gemeinsam an einem Projekt arbeiten. Eine Kante von einem Entwickler A zu einem Entwickler B würde in diesem Fall bedeuten, dass A und B im selben Projekt arbeiten. Dadurch können komplexere Strukturen dargestellt werden, vor allem, wenn Mitarbeiter parallel in mehreren Projekten sind.

In einem **Kommunikationsnetzwerk** wird die Kommunikation zwischen den Akteuren visualisiert. Eine Kante zwischen zwei Knoten steht hierbei für eine direkte Kommunikation zwischen den jeweiligen Akteuren, im Falle gerichteter Kanten in der entsprechenden Richtung. Die Kanten können hier

um Kantengewichte ergänzt werden, die dann beispielsweise die Häufigkeit oder Intensität der Kommunikation repräsentieren können [4] [38].

Ein solches Netzwerk kann zum Beispiel für die Teamkommunikation innerhalb einer Projektwoche erstellt werden. Dabei wird jedes Teammitglied als ein Knoten dargestellt. Spricht nun ein Teammitglied A mit einem Anderen B während dieser Woche, so wird dem Netzwerk eine Kante zwischen A und B hinzugefügt. So kann am Ende der Woche erkannt werden, wer mit wem kommuniziert hat.

Dieses Netzwerk ist für diese Arbeit besonders relevant, da vor allem die Kommunikation der Mitarbeiter in Meetings analysiert wird.

Ein **Affiliation Netzwerk** hat anders als die vorangegangenen Netzwerktypen zwei verschiedene Arten von Knoten. Die Akteure bleiben als eine Gruppe erhalten, die andere Gruppe bilden sogenannte *events*, also Ereignisse. Bei diesen Ereignissen kann es sich je nach Kontext um Aktivitäten, Meetings oder auch Gesprächsabschnitte handeln. Kanten bestehen ausschließlich zwischen Akteuren und Ereignissen, Verbindungen zwischen zwei Akteuren oder zwei Ereignissen sind nicht vorgesehen [33] [9].

In Abbildung 2.2 ist eine Visualisierung eines Affiliation Netwerks dargestellt. Dabei sind die Akteure in blau und die Ereignisse in rot gekennzeichnet.

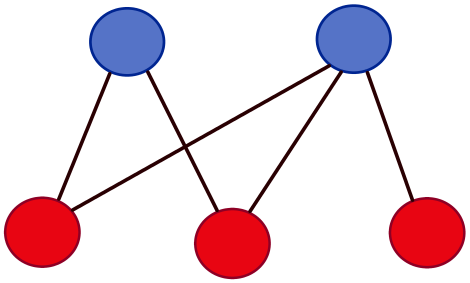

Abbildung 2.2: Affiliation Netzwerk

Es kann mit Hilfe eines Affiliation Netzwerks beispielsweise dargestellt werden, wer in einem Meeting sich zu welchen Themen äußert. So können in einem Meeting mit Tagesordnung die einzelnen Punkte auf dieser als events und die einzelnen Teammitglieder als Akteure dargestellt werden. Trägt nun ein Entwickler E etwas zum Tagesordnungspunkt T bei, so wird eine Kante zwischen E und T dem Netzwerk hinzugefügt. Dadurch kann einfach erkannt werden, wer sich in welchen Bereichen beteiligt.

In dieser Arbeit werden Affiliation Netzwerke genutzt, um die Beteiligung der Meeting Teilnehmer in unterschiedlichen Abschnitten des Meetings zu analysieren.

Um die im Rahmen dieser Arbeit konstruierten Netzwerke analysieren und interpretieren zu können, kommen im Rahmen der Evaluation verschiedene, aus den Netzwerken berechnete Größen zum Einsatz. Die für die Arbeit wichtigen Netzwerkgrößen der sozialen Netzwerkanalyse gehören zu den Zentralitätsmaßen. Zentralitätsmaße sind eine Menge von berechenbaren Werten in einem sozialen Netzwerk, die Informationen über die „Wichtigkeit“ von Akteuren liefern sollen. Im folgenden werden zwei solcher Zentralitätsmaße definiert.

2.1.1 Maße und Metriken

Die **Grad-Zentralität** eines Knoten v spiegelt den Anteil der anderen Knoten im Netzwerk wider, mit denen v verbunden ist [20] [24]. Ein Knoten mit einer Grad-Zentralität von 1 ist beispielsweise mit allen anderen Knoten verbunden, ein Knoten mit einer Grad-Zentralität von 0.5 hingegen nur mit der Hälfte aller Knoten. Es wird also angegeben, mit wie viel Prozent der anderen Knoten v verbunden ist.

Für die Berechnung der Grad-Zentralität, wird der sogenannte Grad eines Knoten benötigt. Der Grad oder Degree eines Knoten v ist die Anzahl an Kanten von v [24]. Hierbei kann bei gerichteten Graphen zwischen einem In-Degree, der nur reinkommende Kanten berücksichtigt, und einem Out-Degree, der die ausgehenden Kanten zählt, unterschieden werden [23]. Der gesamte Grad eines Knoten berechnet sich in diesem Fall aus der Summe von In- und Out-Degree, da dann alle Kanten eines Knoten berücksichtigt werden.

Die Grad-Zentralität eines Knoten v wird berechnet, indem der Grad des Knotens durch die Anzahl anderer Knoten, also die Anzahl aller Knoten minus eins, geteilt wird [20] [24]. In der folgenden Formel wird die gesamte Grad-Zentralität berechnet, da diese auch für diese Arbeit verwendet wird. Es kann aber auch hier zwischen In- und Out-Degree unterschieden werden. Hierfür muss lediglich statt des Gestamtgrads des Knoten der In- oder Out-Degree in die Formel eingesetzt werden.

$$C_D(v) := \frac{deg(v)}{n-1}, \tag{2.2}$$

$$mit$$

$$deg(v) = \#\{(v_i, v_j) \in E : v_i = v \text{ oder } v_j = v\}$$

$G = (V, E)$: soziales Netzwerk als Graph mit Knotenmenge V und Kantenmenge E

n = Anzahl der Knoten

Die **Closeness-Zentralität** gehört ebenfalls zu den Zentralitätsmaßen. Sie basiert allerdings nicht wie die Grad-Zentralität auf der Anzahl an verbundener Knoten, sondern auf der Distanz zu ihnen. Die Closeness-Zentralität gibt also an, wie „schnell" die anderen Knoten von einem Knoten v erreicht werden können [20] [24] [32].

Die geodätische Distanz zwischen zwei Knoten, auch Pfaddistanz genannt, ist die Länge des kürzesten Pfades zwischen diesen beiden Knoten. Müssen mindestens zwei Kanten abgelaufen werden, um von Knoten A zu Knoten B zu kommen, so ist die Distanz zwischen diesen Knoten also 2. Sind den Kanten eines Netzwerkes Gewichte zugeordnet, so können diese für die Berechnung der Distanz verwendet werden. In diesem Fall werden nicht die Anzahl Kanten auf einem Pfad gezählt, sondern die Gewichte der jeweiligen Kanten addiert.

Die Gesamtdistanz eines Knoten v ist die Summe der Distanzen von v zu jedem anderen Knoten in dem Netzwerk. Die Closeness-Zentralität ist der Kehrwert dieser Gesamtdistanz.

$$C_C(u) = d(u)^{-1}, \tag{2.3}$$

$$mit$$

$$d(u) := \sum_{v \in V \setminus \{u\}} d(u, v)$$

$$d(u, v) = \text{geodätische Distanz zwischen u und v}$$

2.2 Kommunikationsmaße

Neben den klassischen Netzwerkgrößen der sozialen Netzwerkanalyse werden in dieser Arbeit auch einige Größen verwendet, die etwas über die Kommunikationshäufigkeit und -intensität aussagen. Diese kommen, wie die Netzwerkgrößen auch, in der Evaluation und Interpretation der konstruierten Netzwerke zum Einsatz.

2.2.1 Communciation Score und Communication Intensity

Sowohl der Communication Score als auch die Communication Intensity sind Größen, die die Kommunikation von Teammitgliedern untereinander beschreiben und die mithilfe eines Kommunikationsnetzwerkes berechnet werden können. Sie werden bei Schneider et al. [38] beschrieben und definiert.

Die **Communication Intensity** I_{ij} beschreibt die Intensität mit der Akteur i mit Akteur j kommuniziert hat. Diese wird in der Regel von den Akteuren selbst eingeschätzt und kann für verschiedene Kommunikationskanäle separat oder für alle gemeinsam bewertet werden.

Der **Communication Score zwischen zwei Akteuren** bezeichnet den Durchschnitt der Communication Intensities der Beiden mit dem jeweils Anderen. Dieser Communication Score wird teilweise auch als Communication Intensity der beiden Akteure bezeichnet. Da verschiedene Medien, über die kommuniziert wird, unterschiedliche Einflüsse auf die Intensität der Kommunikation haben, ist jedem Medium ein eigener Score zugeordnet. Dieser wird bei der Berechnung mit beachtet.

$$C_{ij} = \varnothing\{I_{ij}, I_{ji}\} * \sum_{x \in X} (max\{M_{ij}^x, M_{ji}^x\}), \tag{2.4}$$

$$mit$$

$$i, j \in T = \text{Menge aller Teammitglieder}$$

$$X = \text{Menge aller Medien}$$

$$M_{ij}^x = \begin{cases} S^x, \text{wenn i mit j über Medium x kommuniziert} \\ 0, \text{sonst} \end{cases}$$

$$S^x = \text{Score von Medium x}$$

Der **Communication Score eines Teammitglieds i** ist die Summe aller paarweisen Communication Scores, an denen i beteiligt ist. Er sagt also aus, wie viel i insgesamt kommuniziert.

$$C_i = \sum_{\substack{j \in T \\ j \neq i}} C_{ij} \tag{2.5}$$

2.2.2 Overall Communication Intesity

Die **Overall communication intensity** ist, wie Communication Intensity und Communication Score, eine Größe die in einem Kommunikationsnetzwerk berechnet werden kann und wird ebenfalls bei Schneider et al. [38] beschrieben. Sie spiegelt die Gesamtkommunikation eines Teams wider und wird über den Durchschnitt aller paarweisen Communication Scores bzw. Communication Intensities berechnet. Zusätzlich wird die Dauer aller Team Meetings in die Berechnung mit eingebracht, dies kann aber bei fehlender Information über die Länge wegfallen.

$$C_{Team} = \varnothing_{\substack{i,j \in T \\ i \neq j}}(C_{ij}) + 0.25 * \sum_{b \in B} duration(b), \tag{2.6}$$

$$mit$$

$$B = \text{Menge aller offiziellen Team Meetings}$$

2.3 Meetinganalyse

Meetings sind aus der modernen Softwareentwicklung nicht mehr wegzudenken. Sie werden zur Planung und Koordination sowie zur Informationsweitergabe genutzt und finden in der agilen Sotwareentwicklung sogar täglich statt [20]. Die Qualität von Meetings kann erheblichen Einfluss auf den Erfolg eines Projektes haben, weshalb das Gebiet der Meetinganalyse von großer Relevanz ist [21]. Ziel ist es, durch Beobachtung und Untersuchung der Struktur und Kommunikation in Meetings, mögliche Schwachstellen oder Risiken für den Projekterfolg zu identifizieren und entsprechende Gegenmaßnahmen zu ergreifen [17]. Eine dafür verwendete Methode wird im folgenden behandelt. Diese wird in dieser Arbeit zur Analyse der Interaktion innerhalb der Meetings verwendet.

2.3.1 act4teams

act4teams ist ein Kodierschema für Aussagen von Teammitgliedern während Meetings, um ihre Zusammenarbeit analysieren zu können. Die Abkürzung steht hier für *advanced interaction analysis for teams* [17]. In diesem Schema werden einzelne Aussagen einer von 44 Kategorien zugeordnet. Diese Kategorien sind aufgeteilt in Kommunikation, die positiv zu einem Meeting beiträgt, und solche, die einen negativen Einfluss hat. Zudem können die Kategorien in vier Arten von Aussagen eingeteilt werden. Die vier Kompetenzbereiche sind [20] [17] [25]:

1. Problemfokussierte Aussagen (professionelle Kompetenz), die sich direkt mit dem Identifizieren und Lösen von Problemen sowie der Evaluation solcher Lösungen beschäftigen

2. Prozeduale Aussagen (Methodenkompetenz), die sich mit dem Strukturieren von Meetings beschäftigen

3. Sozio-emotionale Aussagen (Sozialkompetenz), die andere Teammitglieder mit einbeziehen und somit Beziehungen widerspiegeln und beeinflussen

4. Aktionsorientierte Aussagen (Selbstkompetenz), die die Bereitschaft zur Arbeit und Verbesserung ausdrücken

Eine Übersicht der Kompetenzbereiche und dazugehörigen Kategorien nach [17] sind im Anhang A aufgeführt.

Die Kodierung der Aussagen innerhalb eines Meetings werden in der Regel mithilfe einer Videoaufzeichnung durchgeführt. Hierfür wird das Meeting in „Sinneinheiten“, also einzelne, geschlossene Aussagen unterteilt. Diesen werden dann von geschulten, nicht dem Team angehörigen Mitarbeitern einer Kategorie zugeordnet.

2.4 PANAS

Der *Positive and Negative Affect Schedule (PANAS)* ist ein „weit verbreitete[s] [...] Instrument zur Erfassung der emotionalen Befindlichkeit [...]“ (Breyer et al. [2]) und ist eine Methode der Stimmungsanalyse. Die Stimmung beziehungsweise der Stimmungsverlauf einzelner Teammitglieder oder eines gesamten Entwicklerteams kann Hinweise auf Konflikte innerhalb eines Teams liefern und wird daher in dieser Arbeit für die Evaluation der konstruierten Netzwerke genutzt.

Die Skala besteht aus insgesamt 20 Adjektiven aufgeteilt in zwei Gruppen je 10 Begriffe. Die Adjektive beschreiben verschiedene Empfindungen und Gefühle, die entweder positiv und somit zur Dimension des Positiven Affekts oder negativ und daher dem Negativen Affekt zugeordnet sind. Eine Übersicht aller Adjektive mit Verweis auf die Gruppe ist in Anhang A zu finden.

Bei der Anwendung der PANAS Skala werden die Teilnehmenden aufgefordert, einzuschätzen, wie gut das jeweilige Adjektiv ihre aktuelle Stimmung und Gefühlslage widerspiegelt. Dabei sind feste Antwortmöglichkeiten von eins bis fünf vorgegeben [2]: „gar nicht“ (1), „ein bisschen“ (2), „einigermaßen“ (3), „erheblich“ (4) und „äußerst“ (5).

Für die Berechnung des Positiven und Negativen Affekts werden nun die Mittelwerte aller der jeweiligen Gruppe zugehöriger Einschätzungen gebildet. Höhere Werte beim Positiven Affekt repräsentieren hierbei eine

allgemein positivere Stimmung und ein hoher Negativer Affekt eine allgemein negativere Stimmung [2].

Kapitel 3

Verwandte Arbeiten

Die soziale Netzwerkanalyse und die Analyse von Interaktion in Meetings sind bereits seit einiger Zeit Themen der Forschung.

Albrecht et al. [29] beschreiben drei verschiedene Möglichkeiten, wie Kommunikation als Netzwerk dargestellt werden kann.

Die Erste konzentriert sich dabei mehr auf die Beziehungen zwischen den Akteuren, als auf die Kommunikation. In diesen Kommunikationsnetzwerken wird die Kommunikation zwischen den Akteuren als eine Begleiterscheinung der zwischen ihnen existierenden Beziehung gesehen.

Die zweite beschriebene Möglichkeit bezieht sich hingegen auf die sprachliche Grundlage und analysiert sprachliche Strukturen und Muster.

Der dritte Ansatz „modelliert Kommunikation als eigenlogischen Prozess, der relativ unabhängig von sowohl den Akteuren als auch den sprachlichen Strukturen verläuft." (Albrecht et al. [29], S. 34). Bei diesem „Communication oriented modelling" repräsentieren die Knoten des Netzwerks keine Akteure, sondern kommunikative Ereignisse und die Kanten stellen Referenzen zwischen diesen Ereignissen dar.

Bei Schneider et al. [38] werden FLOW Netzwerke verwendet, um die Kommunikation von Studententeams in Software Projekten zu analysieren. Die Intensität der Kommunikation zwischen den Teammitgliedern wurde hierbei über eine Einschätzung der Teammitglieder, der verwendeten Medien sowie der Länge der offiziellen Meetings bestimmt. Die analysierte Kommunikation basiert also auf subjektiven Einschätzungen der Teammitglieder.

Netzwerkgrößen wie z.B. Overall Communication Intensity, wurden mit der Stimmung bzw dem Stimmungsverlauf sowie dem Projekterfolg in Relation gesetzt, um Einflüsse der Kommunikation auf Stimmung und Erfolg erkennen zu können. Für die Stimmungsauswertung wurde hierbei die PANAS-Skala verwendet 2.4 und für die Überprüfung kamen Quality-Gates zum Einsatz.

Zhang et al. [43] verfolgen einen etwas anderen Ansatz zur Erkennung von Beziehungen. Hier werden die Daten von Telefongesprächen analysiert, um die sozialen Strukturen, Beziehungen und Änderungen zu erkennen. Für die Analyse wird hier ein sehr mathematischer Ansatz verfolgt und anhand der Telefondaten ein sogenannter „reciprocity index“ berechnet.

Klünder et al. [21] beschäftigen sich mit der Interaktionsanalyse in Meetings. Der Betrachtungsschwerpunkt hier liegt auf der Identifikation von positivem und negativem Verhalten in Meetings und deren Einfluss auf deren Qualität. Dabei stehen weniger die Beziehungen der Teammitglieder im Vordergrund, sondern der Einfluss von gewissen Aussagetypen und deren Menge auf das Meeting und Team in ihrer Gesamtheit.

Aufgebaut wird der Analyseansatz auf das Kodierschema act4teams 2.3.1, welches an den gesonderten Fall eines Meetings in der Software-Entwicklung zu act4teams-SHORT angepasst wurde. Dadurch kann es leichter und präziser Ergebnisse liefern und erfordert einen geringeren Schulungsaufwand für die durchführenden Mitarbeiter als die allgemeine Vorlage.

In der Dissertation von Klünder [20] wird soziale Netzwerkanalyse verwendet, um Schwachstellen innerhalb der Weitergabe von Informationen aufzuzeigen und kritische Punkte im Netzwerk zu identifizieren. Ziel hierbei ist es, die Informationsweitergabe innerhalb des Teams zu verbessern und die Gefährdung des Projekterfolgs durch falsche oder verloren gegangene Informationen zu vermeiden. Die Netzwerkanalyse wird hier nicht auf Meetings sondern auf Informationsflüsse angewandt.

Salamin et al. [33] nutzen Netzwerke, um die Rollen von Gesprächsteilnehmern erkennen zu können. Dieser Ansatz soll sowohl für geführte Gespräche, wie Talkshows, als auch für spontan verlaufende, wie beispielsweise Meetings, anwendbar sein. Dabei kommen Affiliation Netzwerke zum Einsatz, um für jeden Teilnehmer einen Vektor von Features zu extrahieren. Anhand dieser Vektoren können sie dann anschließend den am besten passenden Rollen zugeordnet werden.

Als Datengrundlage werden hier Tonaufnahmen verwendet, die in einzelne Abschnitte mit jeweils einem Sprecher aufgeteilt werden. Das Gespräch wird in Abschnitte geteilt, die die Ereignisse des Affiliation Netzwerkes bilden. Spricht ein Teilnehmer während eines Ereignisses, so wird dies in seinem Feature-Vektor vermerkt.

Ein ähnlicher Ansatz wird auch bei Garg et al. [9] verfolgt. Hier findet die Identifikation der Rolle eines Meeting-Teilnehmers innerhalb von Software-Projekten anhand von Interaktionsmustern sowie der Ausdrucksweise auf

sprachlicher Ebene statt.

Hier kommt ebenfalls ein Affiliation Netzwerk zum Einsatz. Zur Festlegung der Ereignisse werden hier gleich große Zeitabschnitte gewählt.

Auch der Versuch Beziehungen zwischen Teilnehmern anhand von Meetings zu identifizieren wurde in der Forschung bereits gemacht. Sauer et al. [35] nutzt ein soziales Netzwerk, bei denen der Fokus auf Beziehungen und Verbindungen zwischen den Teilnehmern gelegt wird. Die Auswertung geschieht hier mittels Audio oder Video-Aufnahmen der Meetings.

Hierbei wird die Kommunikation zwischen zwei Teilnehmern innerhalb eines Meetings als eine Verbindung im Netzwerk gewertet. Wenn ein Teilnehmer A also auf eine Aussage von Teilnehmer B antwortet, dann entsteht eine Verbindung im Netzwerk zwischen A und B.

Auch Stegbauer et al. [40] analysiert mithilfe von Netzwerken die Sprecherreihenfolge und somit auch der Informationsfluss. Allerdings wird hier der Fokus nicht auf die Beziehungen zwischen den Teilnehmern, sondern auf die Identifikation verschiedener Kommunikationsrollen gelegt.

Es werden verschiedene Kommunikationsrollen wie Führer und Folger beschrieben, die immer im Zusammenhang zweier Teammitglieder auftreten können. Außerdem werden eine Reihe von Konstellationen und Rollen in der Beziehung dreier Mitglieder, wie z.B. Verstärker oder Redundanz, vorgestellt, die teils auch Gruppenzugehörigkeiten der jeweiligen Sprecher mit einbeziehen, wie beim Representative.

Sauer et al. [34] nutzt die Videoaufnahmen von Meetings, um die Position vom Teamleiter in einem Meeting und dessen Einfluss auf die Produktivität und Team-Zufriedenheit zu untersuchen.

Dabei wurde in einem Netzwerk dargestellt, wer in einem Meeting auf wen reagiert, also in welcher Reihenfolge die Teilnehmer miteinander gesprochen haben. Dabei wurde eine Reaktion durch eine Kante im Netzwerk visualisiert. Die Häufigkeit der Kommunikation wurde dabei durch das Gewicht der Kanten dargestellt.

Der Fokus dieser Arbeit liegt auf der Analyse der persönlichen Beziehungen zwischen einzelnen Teilnehmern in Meetings. Wie in einigen anderen Arbeiten auch, wird in Dieser die Kommunikation von Teilnehmern in einem Meeting als soziales Netzwerk betrachtet und die Reihenfolge der Sprecher als Anhaltspunkt für die Analyse der Beziehung der Teilnehmer untereinander gewertet. Allerdings wird hier nicht nur die Häufigkeit der Kommunikation verwendet, sondern auch die Art und Weise, mit der kommuniziert wird, mit einbezogen.

Dazu wird, ähnlich wie bei Klünder et al. [21] das positive und negative Verhalten mittels eines auf act4teams beruhenden Schemas betrachtet. Es wird jedoch nicht act4teams-SHORT sondern eine Auswahl von act4teams-Kategorien verwendet. Außerdem werden die Kategorien hier nicht als Indikator für die Qualität des Meetings an sich, sondern als spezifischen Ausdruck von Beziehungen zwischen einzelnen Entwicklern eingesetzt.

In dieser Arbeit wird, anders als beispielsweise bei [38] keine selbst eingeschätzten Werte für die Analyse der Kommunikationshäufigkeit verwendet, sondern lediglich die Meetings. Diese liegen, anders als in anderen Arbeiten, als Transkripte vor. Allerdings wird die PANAS-Skala auch in dieser Arbeit für die Auswertung der Netzwerke, neben anderen Größen, zum Einsatz kommen.

Insgesamt fokussiert sich diese Arbeit weniger auf die Analyse gesamter Meetings oder die Rollen einzelner Teammitglieder, sondern auf die persönlichen Beziehungen der Teilnehmer. Dafür werden eine Reihe verschiedener Aspekte betrachtet.

Kapitel 4

Konzept

Aufbauend auf den beschriebenen verwandten Arbeiten wird im folgenden Kapitel ein Konzept entwickelt, mit dem die Beziehung zwischen Entwicklern anhand von Meetings in Software-Projekten analysiert werden kann.

Hierfür wird zunächst erläutert, wie ein soziales Netzwerk aus einem Meeting konstruiert werden kann. Anschließend werden die einzelnen Teilanalysen mit ihren Besonderheiten entwickelt.

4.1 Meetings als soziales Netzwerk

Um die Beziehungen zwischen einzelnen Entwicklern, die sich innerhalb von Meetings zeigen, analysieren zu können, muss aus einem Meeting ein soziales Netzwerk konstruiert werden. Dieser Vorgang ist auf eine Vielzahl von Arten möglich. Im folgenden wird die in dieser Arbeit verwendete Vorgehensweise erläutert.

Die Meetings liegen für die Konstruktion der Netzwerke als Transkripte vor, in denen unter Anderem die im Meeting getätigten Aussagen sowie deren Sprecher in der gesprochenen Reihenfolge aufgelistet sind. Transkripte bieten gegenüber Ton- oder Videoaufzeichnungen den Vorteil, dass die Verarbeitung und Analyse hier einfacher ist. Zum Einen, weil die Suche in Texten nach bestimmten Schlagwörtern deutlich schneller und einfacher möglich ist als in Audio-Dateien, zum Anderen, da so eine eindeutige Gesprächsreihenfolge vorliegt. In Meetings kann es passieren, dass mehrere Leute gleichzeitig sprechen beziehungsweise dass der nächste Sprecher beginnt, bevor der Vorherige seinen Satz komplett beendet hat. Diese Doppelungen können bei der Verwendung von Audio-Dateien zu einer unklaren Gesprächsabfolge führen. Aus diesen Gründen wird in dieser Arbeit mit Transkripten gearbeitet.

Ein Netzwerk besteht aus einer Knoten- und Kantenmenge. Diese müssen also mit Hilfe des Transkripts gebildet werden.

Da das Netzwerk die Beziehungen der Entwickler untereinander darstellen soll, wird in dem konstruierten Netzwerk jeder der Teilnehmer des Meetings als ein eigener Knoten dargestellt. Es gilt also $V = \{v \mid v \text{ ist Teilnehmer im Meeting }\}$.

Die Kanten des Netzwerks repräsentieren die Beziehungen der Entwickler untereinander und werden über das Transkript konstruiert.

Im folgenden wird der generelle Vorgang bei der Konstruktion der Kanten aus einem Meeting-Transkripts erläutert. Auf Besonderheiten innerhalb der Konstruktion sowie die Wahl der jeweiligen Kantengewichte wird in späteren Kapiteln näher eingegangen.

Für die Konstruktion der Kantenmenge wird das Meeting sequentiell durchgegangen, also nach einander jede der Aussagen einmal betrachtet. Für Jede wird der Sprecher der Aussage ermittelt, sowie der Sprecher der folgenden Aussage und eine Kante von dem Ersten zum zweiten Sprecher der Kantenmenge des Netzwerks hinzugefügt. Antwortet ein Entwickler B also Entwickler A innerhalb des Meetings auf eine gerade gestellte Frage, so wird in der Konstruktion des Netzwerks für diese Interaktion eine Kante A $\rightarrow$ B eingefügt.

Da sich die Reihenfolge, in denen Entwickler sprechen, innerhalb des Meetings wiederholen können, ist es möglich, dass in der finalen Kantenmenge E die selben Kanten mehrfach vor kommen, beispielsweise könnte die Kante A $\rightarrow$ B 25 mal in E enthalten sein. Diese Duplikate können zu einer Kante zusammengefasst werden. Die Anzahl der eigentlich vorhandenen Duplikate wird in diesem Fall über das Kantengewicht repräsentiert. Die 25 Kanten A $\rightarrow$ B werden dann durch eine Kante A $\rightarrow$ B mit einem Kantengewicht von 25 ersetzt.

Auf diese Weise kann ein Netzwerk aus einem Meeting konstruiert werden, welches den Gesprächsablauf repräsentiert. Alle in dieser Arbeit verwendeten Netzwerke werden nach diesem Prinzip konstruiert. Allerdings sind für die unterschiedlichen Teilaspekte verschiedene Änderungen vorzunehmen. Diese betreffen vor allen die Kantengewichte und es wird teils nicht aus jeder Aussage eine Kante konstruiert.

Für diese Arbeit werden gerichtete Graphen als Repräsentation des Beziehungsgeflechts innerhalb des Teams verwendet. Die generelle Beziehung zwischen zwei Entwicklern A und B ist dabei durch die Kanten A $\rightarrow$ B und B $\rightarrow$ A dargestellt. Es wäre auch eine zusammengefasste Darstellung in einer ungerichteten Kante möglich. Die Beziehung zwischen zwei Menschen wird aber nicht immer von beiden Beteiligten gleich empfunden und nach außen gezeigt. Es ist beispielsweise möglich, dass jemand eine Abneigung gegen einen Kollegen hegt, dieser aber eine freundschaftliche Beziehung empfindet. Diese unterschiedlichen Wahrnehmungen können sich in dem Verhalten der

Personen widerspiegeln und würden bei der Betrachtung als ungerichteter Graph verloren gehen. Aus diesem Grund werden in dieser Arbeit gerichtete Graphen für die Analyse eingesetzt.

4.2 Teilanalysen

Da Beziehungen von vielen verschiedenen Aspekten beeinflusst werden [26], ist es sinnvoll, die Analyse nicht nur auf einen Teilaspekt von Kommunikation wie beispielsweise die Häufigkeit, aufzubauen. Aus diesem Grund wird die Analyse in diesem Konzept in verschiedene Teilanalysen aufgeteilt, die am Ende zusammen geführt werden.

Für die Wahl der verschiedenen Teilaspekte dienen die verwandten Arbeiten als Orientierung und Inspiration. Im folgenden werden die gewählten Teilaspekte kurz vorgestellt. Eine nähere Erklärung und die jeweilige Vorgehensweise in der Konstruktion der jeweiligen Netzwerke ist in den entsprechenden Unterkapiteln zu finden.

Da die **Abfolge von Sprechern** bereits in vergangenen Arbeiten [35] [40] [34] als Grundlage für die Identifikation von Rollen und Verbindungen der Teammitglieder verwendet wurde, erscheint es sinnvoll sie auch für die Analyse der Beziehung heranzuziehen.

Da allein aus der Häufigkeit der Kommunikation nur begrenzt Rückschlüsse auf die Art der Beziehung möglich sind, wird auch die **Art und Weise**, mit der kommuniziert wird, berücksichtigt. Hierfür bietet sich act4teams an, da das Schema eine Kategorisierung bietet, die ihre Kategorien bereits in positiven und negativen Einfluss trennt.

Obwohl die **Unterbrechung** als Kategorie in act4teams auftritt, wird sie in der Analyse zusätzlich separat betrachtet. Das Unterbrechen einer anderen Person ist ein Verhalten, dass negativen Einfluss auf die Beziehung zu Dieser haben kann, es beeinflusst aber auch die gesamte Meeting-Dynamik. Daher kann eine getrennte Betrachtung hier zusätzliche Hinweise auf eventuelle Probleme in einem Team bieten.

Auch das Einbeziehen eines Affiliation Netzwerks bietet eine interessante Sichtweise auf die Beziehungen, die durch andere Netzwerkarten nicht gegeben ist. In vergangenen Forschungsarbeiten konnten mithilfe von diesen Netzwerken die Rollen der Meetingteilnehmer ermittelt werden, da Teammitglieder mit ähnlichen Rollen in den selben Abschnitten eines Meetings sprechen [33] [9].

Hier liegt der Rückschluss nahe, dass Entwickler mit unterschiedlichen Rollen oder Expertisen in teils sehr verschiedenen **Meetingabschnitten** sprechen und daher weniger miteinander kommunizieren, obwohl dies nicht mit einer guten oder schlechten Beziehung zwischen ihnen zusammenhängt. Eine Analyse dieses Phänomens hat zwar keine Relevanz für die Analyse der Beziehung an sich, kann aber bei der Einordnung der Analyseergebnisse helfen und sollte daher ebenfalls betrachtet werden.

Es erscheint sinnvoll für die Analyse von Beziehungen mehrere Aspekte zu betrachten, da viele verschiedene Dinge Einfluss auf die Beziehung zwischen zwei Menschen haben können und auch die Kommunikation nur eines davon ist. In dieser Arbeit werden die **Sprecherfolge**, **Unterbrechungen**, **positive sowie negative Interaktion** und die Beteiligung an verschiedenen **Gesprächsabschnitten** fokussiert. Für eine bessere und transparentere Auswertung werden die verschiedenen Aspekte sowohl getrennt voneinander als auch als Gesamtheit analysiert.

Im folgenden wird die Herangehensweise an die einzelnen Teilanalysen sowie die Gesamtanalyse näher erläutert.

4.2.1 Sprecherfolge

Eine Analyse der Sprecherfolge in einem Meeting gibt Aufschluss darüber, wer mit wem direkt kommuniziert. Diese Kommunikation spiegelt die Beziehung zwischen den beiden Kommunizierenden wieder [27]. Durch Kommunikation wird auch eine gewisse Beziehung aufgebaut, da wir mit Menschen, mit denen wir interagieren, automatisch eine Beziehung aufbauen, auch wenn diese nur sehr oberflächlich sein mag [26].

Es ist also davon auszugehen, dass alle Teilnehmer eines Meetings, die direkt kommuniziert haben, in irgendeiner Form eine Beziehung miteinander haben. Eine höhere Kommunikation ist ein Zeichen von einer positiveren Beziehung und bietet ebenfalls mehr Möglichkeiten, durch eine stärkere Interaktion eine Beziehung aufzubauen. Daher wird für die Analyse nicht nur betrachtet, ob zwei Entwickler miteinander kommunizieren, sondern auch wie häufig sie dies tun.

Da die Beziehung zwischen zwei Menschen von den Beteiligten unterschiedlich eingeschätzt und empfunden werden kann [26], ist es außerdem relevant, die Kommunikation hier mehr als einseitige Aktion zu sehen, als es üblicherweise getan wird. Daher wird für die Analyse ein Netzwerk mit gerichteten Kanten verwendet. Auf diese Weise können auch Diskrepanzen zwischen der Kommunikationshäufigkeit zweier Entwickler erkannt werden, die unter Umständen zu zusätzlichem Konfliktpotential führen können.

Als Kommunikation zwischen zwei Entwicklern A und B, wird eine direkte Reaktion von B auf eine Aussage von A gewertet. Es spricht also B nach A. Dies wird in einem Kommunikationsnetzwerk als Kante von A nach B dargestellt. Die Kantengewichte repräsentieren in diesem Netzwerk die Häufigkeit der Kommunikation.

Zur Erstellung des Netzwerkes wird ein Transkript des Meetings verwendet, in dem die jeweiligen Sprecher der Aussagen angegeben sind. Die Knotenmenge des Netzwerkes ist die Menge an Sprechern in diesem Transkript. Da es auch vorkommen kann, dass ein Teilnehmer in einem Meeting nicht spricht und dies auch einen Rückschluss der Beziehungen des stummen Teilnehmers zulässt, werden diese zusätzlich zur Menge der Teammitglieder hinzugefügt. Für jede Reaktion im Transkript wird dann eine entsprechende Kante zum Netzwerk hinzugefügt oder das Kantengewicht erhöht. Dieses zeigt die Anzahl der Kommunikation an. Die Erstellung des Netzwerkes wird nach folgendem Algorithmus durchgeführt:

Algorithm 4.1: Sprecherfolge

```
1  input: Transkript T, Anzahl Teammitglieder a
2  output: Graph G

4  G = Graph mit V = Menge der unterschiedlichen Sprecher in T
5  if a > |V| in T:
6          füge G weitere Elemente hinzu bis a = |V|

8  for i = 1, i < Länge T, i++:

10         A = Sprecher von Wortbeitrag T[i-1]
11         B = Sprecher von Wortbeitrag T[i]

13         if not {A → B} in E:
14                 Füge {A → B} mit Kantengewicht 1 zu E hinzu
15         else:
16                 erhöhe Kantengewicht von {A → B} um 1
17 gebe G zurück
```

Zusätzlich zur Betrachtung der absoluten Anzahl von Reaktionen erscheint es sinnvoll auch weitere Variationen der Werte zu analysieren.

Da Meetings sehr verschiedene Längen haben können, sowohl zeitlich als auch bezüglich der Anzahl an Gesprächsbeiträgen, sind die absoluten Werte der Kommunikationshäufigkeit im Vergleich zu anderen Meetings nicht unbedingt sehr aussagefähig. In einem längeren Meeting kann mehr kommuniziert werden und ein Fehlen davon hat mehr Bedeutung als in einem kurzen Meeting, in dem unter Umständen die Gelegenheit für eine solche Kommunikation fehlte. Aus diesem Grund sollten auch auf die Länge der Meetings normalisierte Werte betrachtet werden. Diese werden wie folgt berechnet:

$$d_{norm}(A \to B) = (d_{abs}(A \to B) : |T|) * 100 \tag{4.1}$$
$$mit:$$
$$d_{abs}(A \to B) = \text{absolutes Kantengewicht der Kante A } \to \text{ B}$$
$$|T| = \text{Länge des Transkripts}$$

Da es keinen großen Unterschied für die Beziehung zweier Teammitglieder macht, ob sie innerhalb eines Meetings 111 oder 112 mal miteinander gesprochen haben, aber der Unterschied zwischen 2 und 4 mal deutlich größer ausfallen kann, ist die zusätzliche Betrachtung von logarithmischen Werten sinnvoll. Auf diese Weise können die Größenordnung der Werte und deren Bedeutung besser in Relation gesetzt werden. Zur Berechnung der neuen Kantengewichte wird der natürliche Logarithmus eingesetzt:

$$d_{log}(A \to B) = ln(d_{abs}(A \to B)) \tag{4.2}$$
$$mit:$$
$$d_{abs}(A \to B) = \text{absolutes Kantengewicht der Kante A } \to \text{ B}$$

Um die Unterschiede in der Kommunikation zweier Teammitglieder untereinander schnell und einfach erkennen zu können, werden zusätzlich die Differenzen der Kantengewichte A $\to$ B und B $\to$ A betrachtet. Um ein übersichtliches Netzwerk zu erhalten, existiert in diesem Netzwerk jeweils nur eine Kante zwischen zwei Knoten mit positivem Kantengewicht. Dabei bedeutet eine Kante von A nach B mit einem Kantengewicht von 5, dass das Kantengewicht von (A $\to$ B) gleich dem Kantengewicht von (B $\to$ A) + 5 ist. Berechnet wird die neue Kantenmenge wie folgt:

$$E_{diff} = \{(A \to B) | A, B \in E_{abs},\ d_{abs}(A \to B) > d_{abs}(B \to A)\} \tag{4.3}$$
$$mit$$
$$d_{abs}(A \to B) = \text{absolutes Kantengewicht der Kante A } \to \text{ B}$$
$$d_{diff}(A \to B) = d_{abs}(A \to B) - d_{abs}(B \to A)$$

4.2.2 Unterbrechungen

Jemanden beim Reden zu unterbrechen gilt in Deutschland im Allgemeinen als unhöflich und zeugt von einem gewissen Maße an mangelndem Respekt dem Sprechenden gegenüber [18]. Aus diesem Grund wird eine Unterbrechung hier als ein negativer Einfluss auf eine Beziehung gewertet [19].

Außerdem haben Unterbrechungen einen negativen Einfluss auf Meetings [17], da sie Struktur und Stimmung negativ beeinflussen können und unter Umständen für Verwirrung unter den Teammitgliedern führen. Wenn die Sprechenden sehr häufig unterbrochen werden, ist es schwerer Gedankengänge zu verfolgen und Lösungen für Probleme zu erarbeiten.

Zur Analyse der Unterbrechungen wird ein soziales Netzwerk ähnlich zu dem der Sprecherabfolge verwendet. Allerdings repräsentiert eine Kante A → B hier, dass A von B im Meeting unterbrochen wurde und das Kantengewicht der Kante gibt die Anzahl an Unterbrechungen an.

Zur Identifikation von Unterbrechungen wird das act4teams-Schema verwendet. Das für die Konstruktion des Netzwerks notwendige Meetingtranskript mit Verweisen auf die jeweiligen Sprecher muss hierbei ebenfalls die jeweiligen act4teams Kategorien enthalten, denen die Aussagen zugeordnet sind. Die in diesem Fall relevante Kategorie ist die **Unterbrechung (Unt)**. Sie ist der *Sozialkompetenz* zugeordnet und fällt dort unter den Oberbegriff *Unkollegiale Interaktion*.

Für die Konstruktion des Netzwerkes wird zunächst die Knotenmenge wie zuvor über die verschiedenen Sprecher im Transkript und stummen Teilnehmer gebildet. Anschließend wird für jede Aussage überprüft, ob diese als Unterbrechung klassifiziert wurde. Ist dies der Fall wird eine entsprechende Kante in das Netzwerk eingefügt oder das dazügehörige Kantengewicht erhöht. Die Konstruktion folgt dem folgenden Algorithmus:

Algorithm 4.2: Unterbrechungen

```
1  input: Transkript T, Anzahl Teammitglieder a
2  output: Graph G

4  G = Graph mit V = Menge der unterschiedlichen Sprecher in T
5  if a > |V| in T:
6          füge G weitere Elemente hinzu bis a = |V|

8  for i = 1, i < Länge T, i++:

10         if T[i] als Unterbrechung eingeordnet:
11                 U = Sprecher von Wortbeitrag T[i]
12                 S = Sprecher von Wortbeitrag T[i-1]

14                 if not {S → U} in E:
15                         Füge {S → U} mit Kantengewicht 1 zu E hinzu
16                 else:
17                         erhöhe Kantengewicht von {S → U} um 1
18 gebe G zurück
```

Wie bei der Sprecherfolge auch, scheint es sinnvoll zu sein, nicht nur die absoluten Werte zu betrachten. Um eine sinnvolle Einordnung der Häufigkeit von Unterbrechungen im Vergleich zur Anzahl getätigter Aussagen vornehmen zu können, werden zusätzlich die mithilfe der Transkriptlänge

normalisierten Werte verwendet. Die Berechnung folgt parallel zur Formel (4.1).

Da die Anzahl an Unterbrechungen in einem Meeting deutlich geringer sind als die Anzahl an Kommunikation insgesamt, ist eine Betrachtung von logarithmischen Werten nicht notwendig. Bereits die absoluten Werte bieten einen ausreichenden Überblick über die Größenordnung der Werte.

Da sich auch bei Unterbrechungen die Anzahl und die damit implizierte Beziehung zweier Entwickler zwischen den beiden Richtungen deutlich unterscheiden kann, ist hier die Betrachtung der entsprechenden Differenzen der Übersicht halber dennoch sinnvoll. Die Konstruktion dieses Netzwerks geschieht analog zu Formel (4.3).

4.2.3 Positive und Negative Interaktion

Nicht nur die Häufigkeit mit der Menschen kommunizieren hat Einfluss auf die Beziehung zwischen ihnen, sondern auch die Art und Weise, wie Interaktionen ablaufen und wie Aussagen formuliert sind [7]. Positive Interaktionen, wie beispielsweise Lob, haben einen positiven Einfluss auf eine Beziehung und zeugen von einem guten oder mindestens neutralen Verhältnis zwischen zwei Entwicklern. Negative Interaktionen, wie Beleidigungen, hingegen beeinflussen eine Beziehung negativ und spiegeln ein eher gespanntes Verhältnis wieder [6]. Daher sind sowohl positive als auch negative Interaktion ein Aspekt, der zur Analyse von Beziehungen herangezogen werden sollte.

Da eine Beziehung nicht ausschließlich auf positiven Interaktionen oder ausschließlich auf negativen Interaktionen aufbaut, ist es sinnvoll beide Interaktionsarten nicht nur getrennt voneinander sondern auch gemeinsam zu betrachten. Dadurch kann ein guter Überblick über die generelle Interaktion zwischen zwei Entwicklern gewonnen werden und eine Tendenz, ob die Kommunikation eher positiver oder negativer Natur ist, wird sichtbar.

Menschen reagieren auf positive Interaktionen anders als auf Negative. Negative Ereignisse bleiben einem Menschen fünf mal länger im Gedächtnis als Positive [1]. Aus diesem Grund erscheint es sinnvoll, die positiven und negativen Interaktionen nicht gleich zu gewichten, sondern hier die menschliche Natur mit zu berücksichtigen. Aus diesem Grund werden in der Analyse negative Aussagen 5 mal stärker negativ gewertet, als die positive Wirkung von entsprechend positiven Aussagen. Um dies in der Konstruktion der Netzwerke berücksichtigen zu können repräsentieren die Kantengewichte hier nicht die Anzahl an positiven oder negativen Interaktionen, sondern einen *Score*, der die positive oder negative Wirkung der getätigten Aussagen wieder spiegelt. Damit die Negativen einen 5 mal stärkeren Einfluss haben als die positiven Aussagen, muss der negative Score

5 mal dem Positiven entsprechen und anschließend, da für den negativen Score negative Kantengewichte verwendet werden, mal -1 genommen werden. Es gilt also:

$$Score_{positiv} * 5 = -1 * Score_{negativ} \tag{4.4}$$

In dieser Arbeit wird $Score_{negativ} = -1$ und $Score_{positiv} = 0.2$ verwendet. Es wären aber auch andere Lösungen der Gleichung möglich, wie zum Beispiel $Score_{negativ} = -5$ und $Score_{positiv} = 1$.

Zur Identifikation von positiver und negativer Interaktion bietet sich, wie bei den Unterbrechungen, die Nutzung des act4teams Schemas an, da dieses bereits eine entsprechende Kategorisierung bietet [20] [17] [25]. Allerdings sind nicht alle Kategorien des Schemas für die Analyse relevant. Nur die Aspekte, die einen direkten Einfluss auf die Beziehung zwischen den Entwicklern haben, werden für die Analyse herangezogen. Dabei handelt es sich um die Aspekte des Bereichs der Sozialkompetenz. Alle anderen Bereiche beziehen sich entweder auf die Struktur des Meetings, dessen inhaltlichen Aspekte oder die persönlichen Mitwirkung am Meeting. Nur bei den Kategorien der Sozialkompetenz ist eine klare Beteiligung mehrerer Entwickler und somit eine Beziehung erkennbar.

Positive Interaktion Für die positive Interaktion gibt es im act4teams Schema neun relevante Kategorien. Diese sind im folgenden aufgelistet. Ihre genauere Bedeutung kann im Anhang A nachgelesen werden.

- Ermunternde Ansprache (EA)
- Unterstützung (ZUST)
- Aktives Zuhören (AZ)
- Ablehnung (ABL)
- Rückmeldung (RM)
- Atmosphärische Auflockerung (ATM)
- Ich-Botschaft: Trennung von Meinung und Tatsache (IB)
- Gefühle (G)
- Lob (Lob)

Alle Kategorien werden gleich stark positiv gewichtet mit einem $Score_{positiv}$ von $+0.2$ pro Aussage. Dabei beziehen sich Aussagen von acht der neun Kategorien immer nur auf die spezielle Beziehung zwischen zwei Personen, dem Sprecher der Aussage und demjenigen, an den diese gerichtet ist. Die Atmosphärische Auflockerung hingegen kann nicht dieser Art von Einfluss zugeordnet werden, da ein Scherz oder ähnliches selten an nur eine Person gesondert gerichtet ist. Da solche Aussagen mehr oder weniger an alle Teammitglieder gleichzeitig gerichtet sind, haben sie auch einen positiven Einfluss auf alle anderen Teammitglieder und werden bei der Konstruktion des Netzwerkes gesondert behandelt.

Für die Analyse der positiven Kommunikation wird ein ähnliches Netzwerk wie bei den Aspekten zuvor verwendet. Allerdings repräsentieren die Kantengewichte in diesem Fall nicht die Anzahl an positvier Interaktionen, sondern eine Art *Score*, der sich aus der Anzahl an positiven Interaktionen und der Wertung Dieser ergibt. Eine Kante von A nach B bedeutet dabei, dass A positive Interaktion gegenüber B gezeigt hat.

Anders als bei den vorangegangenen Aspekten, ist die Identifikation des Adressaten einer Aussage nicht immer eindeutig bestimmbar, da zur Analyse lediglich eine Transkription des Meetings verwendet wird.

Eine Möglichkeit die Person, an die eine Aussage gerichtet ist, zu identifizieren, ist die direkte Ansprache. Wird ein Entwickler in einer Aussage namentlich angesprochen, so ist Diese wahrscheinlich an ihn gerichtet. Dies kann über die Suche der Entwickler-Namen in den entsprechenden Aussagen realisiert werden. Dieses Vorgehen ist allerdings auch fehleranfällig, da die Erwähnung eines Namens nicht von einer direkten Ansprache unterschieden werden kann.

Bei Aussagen, in denen keine direkte Ansprache identifiziert werden kann, wird davon ausgegangen, dass diese an denjenigen gerichtet ist, der direkt vor dem Sprecher geredet hat. Da der selbe Sprecher mehrere Aussagen hintereinander tätigen kann, muss hier nach demjenigen gesucht werden, der vor Diesem gesprochen hat. Es reicht also nicht aus nur den Sprecher der vorherigen Aussage zu betrachten.

Die Konstruktion des Netzwerkes wird nach dem folgenden Algorithmus durchgeführt:

Algorithm 4.3: Positive Interaktion

```
1  input: Transkript T, Anzahl Teammitglieder a
2  output: Graph G

4  G = Graph mit V = Menge der unterschiedlichen Sprecher in T
5  if a > |V| in T:
6          füge G weitere Elemente hinzu bis a = |V|

8  for i = 1, i < Länge T, i++:

10         if T[i] enthält positive Interaktion:
```

```
12                  if T[i-1] enthält direkte Ansprache:
13                          A = Sprecher von Wortbeitrag T[i-1]
14                          B = in Wortbeitrag T[i-1] direkt
                                angesprochene Person
15                  else:
16                          B = Sprecher von Wortbeitrag T[i]
17                          b = 1
18                          A = Sprecher von Wortbeitrag T[i-b]
19                          while A = B and b < i:
20                                  b += 1
21                                  A = Sprecher von Wortbeitrag T[i-b]

23                  if T[i] enthält Atmosphärische Auflockerung:
24                          for a ∈ V, a ≠ B:
25                                  if not {B → a} in E:
26                                          Füge {B → a} mit
                                                Kantengewicht 0.2 zu E
                                                hinzu
27                                  else:
28                                          erhöhe Kantengewicht von {B
                                                → a} um 0.2

30                  else:
31                          if not {B → A} in E:
32                                  Füge {B → A} mit Kantengewicht 0.2
                                        zu E hinzu
33                          else:
34                                  erhöhe Kantengewicht von {B → A} um
                                        0.2

36  gebe G zurück
```

Da es sich bei den Kantengewichten in diesem Fall nur um einen *Score* handelt, werden hier nur die absoluten Werte betrachtet. Zwar unterliegen auch diese Werte den Schwankungen der Meetinglänge, allerdings erscheint die separate Betrachtung der normalisierten Werte, die besonders klein ausfallen, als nicht notwendig.

Negative Interaktion Für die negative Interaktion sind nur drei Kategorien des act4teams Schemas relevant. Diese sind im folgenden aufgelistet:

- Tadel/Abwertung (TD)
- Seitengespräch (Seit)
- Reputation (R)

Da die Unterbrechungen bereits separat betrachtet werden, sind sie hier nicht mit berücksichtigt. Sie gehören aber ebenfalls zum Bereich der negativen Interaktion und werden in der Gesamtauswertung in Kapitel 4.2.5 als solche gewertet. Wie bei der positiven Interaktion werden auch hier alle Kategorien gleich gewichtet.

Das hier verwendete soziale Netzwerk ist das gleiche wie bei der positiven Interaktion, die Bedeutung der Kanten und deren Richtung werden analog übernommen. Allerdings sind die Kantengewichte hier negativ. Eine negative Aussage wird mit einem $Score_{negativ}$ von -1 gewertet.

Wie auch bei der positiven Interaktion gilt es bei der Negativen den Adressaten einer Aussage zu ermitteln. Hier bietet sich abermals die direkte Ansprache als Indikator an. Ist diese Identifikation nicht möglich, so wird abermals davon ausgegangen, dass der vorherige Sprecher, der nicht mit dem aktuellen identisch ist, mit dieser Aussage angesprochen werden sollte.

Die Kategorie des Seitengesprächs bildet in diesem Bereich eine Ausnahme. Da ein Nebengespräch eine Art von Respektlosigkeit ausdrückt, beeinflusst sie die Beziehung desjenigen, der das Seitengespräch führt, und des eigentlichen Sprechers in negativer Weise. Zwar sind an einem Seitengespräch immer mehrere Leute beteiligt, allerdings ist dies über ein Transkript nur schwer bis gar nicht identifizierbar. Aus diesem Grund wird dieser negative Einfluss nur dem Teammitglied zugeordnet, dass im Transkript hierzu aufgelistet ist. Außerdem ist der *Adressat* dieser Art von Aussage immer derjenige, der als letztes zuvor gesprochen hat. Die Überprüfung auf direkte Anrede fällt hier demnach weg.

Die Konstruktion des sozialen Netzwerks folgt diesem Algorithmus:

Algorithm 4.4: Negative Interaktion

```
1  input: Transkript T, Anzahl Teammitglieder a
2  output: Graph G

4  G = Graph mit V = Menge der unterschiedlichen Sprecher in T
5  if a > |V| in T:
6          füge G weitere Elemente hinzu bis a = |V|

8  for i = 1, i < Länge T, i++:
9          if T[i] enthält negative Interaktion:

11                 if T[i] ist kein Seitengespräch und T[i-1] enthält
                       direkte Ansprache:
12                         A = Sprecher von Wortbeitrag T[i-1]
13                         B = in Wortbeitrag T[i-1] direkt
                               angesprochene Person
14                 else:
15                         B = Sprecher von Wortbeitrag T[i]
16                         b = 1
17                         A = Sprecher von Wortbeitrag T[i-b]
18                         while A = B and b < i:
19                                 b += 1
20                                 A = Sprecher von Wortbeitrag T[i-b]

22                 if not {B → A} in E:
23                         Füge {B → A} mit Kantengewicht -1 zu E
                               hinzu
24                 else:
25                         verringere Kantengewicht von {B → A} um 1

27 gebe G zurück
```

Wie bei der positiven Interaktion, werden auch hier nur die absoluten Werte berücksichtigt.

Gemeinsame Betrachtung Für die gemeinsame Betrachtung von positiver und negativer Interaktion werden nun die Kantengewichte addiert:

$$d_{ges}(A \to B) = d_{pos}(A \to B) + d_{neg}(A \to B) \tag{4.5}$$

Da die Gewichtung von positiven und negativen Aussagen bereits in der separaten Analyse berücksichtigt wurde, ist dies hier nicht mehr von Nöten.

Zusätzlich zur Betrachtung der absoluten Werte werden hier nun auch die Differenzen der Interaktionswerte zwischen zwei Entwicklern berücksichtigt. Diese können, wie zuvor, schnelle Auskunft und Übersicht in der unterschiedlichen Wahrnehmung und Ausprägung der Beziehung aus Sicht der beider Beteiligten geben. Die Berechnung der neuen Kantengewichte und -richtungen erfolgt analog zu Gleichung (4.3).

4.2.4 Gesprächsabschnitte

Genauso, wie Gesprächsteilnahmen in den selben Abschnitten eines Meetings auf eine ähnliche oder gleiche Rolle hinweisen können, kann fehlende Kommunikation in bestimmten Teilen ein Hinweis auf verschiedene Rollen und Expertisen hindeuten. Menschen, die unterschiedliche Aufgaben und Fachbereiche haben, reden tendenziell weniger miteinander als solche, die direkt zusammenarbeiten. Dies hat dann aber weniger mit der persönlichen Beziehung zwischen diesen zu tun.

Um diese Einschränkungen in der Analyse der Beziehungen berücksichtigen zu können, wird ein Affiliation Netzwerk verwendet [33].

Da die Analyse anhand von Transkripten vorgenommen wird, ist die Nutzung von gleich großen zeitlichen Abschnitten als Events, so wie es in anderen Arbeiten verwendet wurde [9], nicht möglich. Stattdessen wird das Meeting in Sinnabschnitte unterteilt. Ein Sinnabschnitt ist dabei eine Menge von aufeinander folgenden Aussagen, die zum gleichen Thema oder Themenkomplex gehören, wie z.B. eine Diskussion über die graphische Oberfläche. Da eine automatisierte Einteilung in Gesprächsabschnitte nicht den Fokus dieser Arbeit bildet, werden Diese hier händisch festgelegt. Wenn es möglich ist, sollte die Einteilung bereits während des Meetings geschehen, da die Einteilung mittels eines Transkripts nicht immer eindeutig möglich ist.

Für das Netzwerk werden die Teammitglieder als Akteure und die Gesprächsabschnitte als Events gesehen. Die Gesprächsabschnitte werden vor der Analyse im Transkript vermerkt, indem jede Aussage einem Abschnitt über den Namen beziehungsweise einer Bezeichnung zugeordnet wird. Für jede Aussage wird eine Kante vom Sprecher zum Gesprächsabschnitt, dem die Aussage zugeordnet ist, dem Netzwerk hinzugefügt. Eine Kante von S nach T bedeutet also, dass Teammitglied S im Abschnitt T gesprochen hat. Das Kantengewicht repräsentiert dabei die Anzahl an Wortbeiträgen von S in T.

Die Konstruktion des Netzwerks erfolgt nach dem folgenden Algorithmus:

Algorithm 4.5: Gesprächsabschnitte

```
1  input: Transkript T, Anzahl Teammitglieder a
2  output: Graph G

4  G = Graph mit Events = Themenabschnitte, Akteure = Teammitglieder
5  for i = 0, i < Länge T, i++:
6          T = Themenabschnitt von T[i]
7          S = Sprecher von T[i]

9          if not {S → T} in E:
10                 Füge {S → T} mit Kantengewicht 1 zu E hinzu
11         else:
12                 erhöhe Kantengewicht von {S → T} um 1

14 gebe G zurück
```

Da dieser Teil der Analyse nur als Hilfe für die Einordnung der anderen Ergebnisse dient, wird diese separat betrachtet. Eine direkte Einbindung in die Gesamtanalyse geschieht nicht.

4.2.5 Gesamtauswertung

Da eine Beziehung nicht nur durch einen einzelnen Aspekt beeinflusst und ausgezeichnet wird, ist neben den Teilanalysen auch eine gesamte Betrachtung sinnvoll. Hierfür werden die Ergebnisse der vorangegangenen Analyseabschnitte miteinander verrechnet um einen Gesamtwert oder *Score* zu erhalten, welcher die positive oder negative Beziehung und deren Stärke widerspiegeln soll. Nur die Analyse der Gesprächsabschnitte wird hierfür nicht berücksichtigt.

Da nicht davon ausgegangen werden kann, dass alle Teilaspekte die gleiche Wirkung auf die Beziehung zweier Menschen haben, ist eine simple Addition als Gesamtauswertung nicht zielführend. Außerdem unterscheiden sich die Kantengewichte der verschiedenen Netzwerke in ihrer Werteskala.

Bei positiver und negativer Interaktion wird ein *Score* verwendet, der bereits eine gewisse Gewichtung der Aussagen enthält, während bei Unterbrechung und Sprecherfolge lediglich die Anzahl der Vorkommnisse gezählt werden. Da eine Unterbrechung innerhalb des act4teams Schemas ebenfalls unter „Unkollegiale Interaktion“ fällt, ebenso wie die anderen Kategorien, die zur Identifikation negativer Interaktion verwendet werden, kann eine Unterbrechung einem *Score* von -1 gleichgesetzt werden. Für die Gesamtanalyse muss allerdings die Werteskala der Sprecherfolge in sinnvollen Zusammenhang mit der Skala der anderen Aspekte gebracht werden.

Betrachtet man die analysierten Aspekte so fällt auf, dass sich diese in zwei Gruppen unterteilen lassen. Unterbrechungen, positive und negative Interaktion beziehen sich allesamt auf die Qualität der Kommunikation. Hier wird die Art und Weise betrachtet, in der kommuniziert wird. Bei der Sprecherfolge hingegen wird darauf geschaut, wie häufig die Kommunikation stattfindet, unabhängig davon welcher Art sie ist. Hier liegt der Fokus also auf der Quantität. Diese Unterscheidung hilft bei der Entwicklung eines Konzepts für die Gesamtanalyse.

Emmers-Sommer et al. [7] hat in einer Studie den Effekt von Qualität und Quantität von Kommunikation auf die Zufriedenheit in einer Beziehung untersucht. Diese Arbeit bietet einen Anhaltspunkt, um eine Gewichtung zwischen Qualität und Quantität in der Auswertung festzulegen. In der Studie konnten 36% bzw 26% der Varianz in der Intimität bzw Zufriedenheit der Beziehung über die Qualität der Kommunikation erklärt werden. Über die Quantität der Kommunikation konnten hingegen nur 10% bzw 7% der Variation erklären.

Diese Werte zeigen, dass Qualität der Kommunikation wichtiger für eine Beziehung ist als die Quantität. Setzt man die Zahlen in Relation zueinander, so ergibt sich ein Verhältnis von ungefähr 1 zu 3,5 zugunsten der Qualität. Dieses Verhältnis von Quantität und Qualität wird im folgenden für die Gewichtung der Qualitäts- und Quantitätsaspekte genutzt. Daraus ergibt sich, dass die Qualität der Kommunikation mit 78% und die Quantität mit 22% in die Gesamtbewertung einfließt. Der Einfachheit halber sind diese Prozentwerte gerundet.

Trotz dieser Gewichtung ist der Einfluss der Sprecherfolge allerdings noch zu hoch. Da die absoluten Werte der Anzahl an Kommunikation teils sehr hoch sind, ist es nicht sinnvoll diese in ihrer Gesamtheit zu verrechnen. Außerdem ist dies nach wie vor eine andere Werteskala als die der Qualitätsmerkmale. Daher muss hier die Skala angepasst werden.

Kommunikation hat generell einen positiven Einfluss auf die Beziehung zweier Menschen. Aus diesem Grund erscheint es sinnvoll, erfolgte Kommunikation auch wie eine positive Interaktion zu gewichten. Einmal miteinander sprechen entspricht also einem *Score* von $+0.2$

Mit all diesen Überlegungen kann nun eine Formel für die Gesamtanalyse entwickelt werden. Der Gesamtscore setzt sich aus 78% Qualitätsmerkmalen

und 22% Quantitätsmerkmalen zusammen. Für die Qualitätsmerkmale werden positive und negative Interaktion addiert und die Anzahl Unterbrechungen abgezogen. Für die Quantitäsmerkmale wird die Sprecherfolge mit dem Score von 0.2 multipliziert.

Die Berechnung der Kantengewichte für die Gesamtanalyse ergeben sich aus folgender Formel:

$$d_{ges}(A \to B) = d_{Qualitt}(A \to B) * 0.78 + d_{Quantitt}(A \to B) * 0.22 \quad (4.6)$$

$$mit:$$

$$d_{Qualitt}(A \to B) = d_{pos}(A \to B) + d_{neg}(A \to B) - d_{unt}(A \to B)$$

$$d_{Quantitt}(A \to B) = d_{flg}(A \to B) * 0.2$$

d_{pos} : Kantengewichte der positiven Interaktion

d_{neg} : Kantengewichte der negativen Interaktion

d_{unt} : Kantengewichte der Unterbrechungen

d_{flg} : Kantengewichte der Sprecherfolge

Wie bei den einzelnen Teilaspekten auch, werden sowohl die absoluten Scores als auch die Differenzen zwischen je zwei Entwicklern betrachtet.

Kapitel 5

Implementierung

In diesem Kapitel wird die Implementierung des erarbeiteten Konzepts beschrieben. Dabei wird speziell auf die verwendete Programmiersprache und Librarys sowie den Aufbau des Programms eingegangen. Außerdem werden hier alle Informationen folgen, die für die Nutzung des Prototypen notwendig sind, also die benötigten Eingabedaten, Informationen über die Ausgabedaten und eine Übersicht der graphischen Oberfläche.

Der entwickelte Prototyp dient zur automatischen Auswertung von Meetingtranskripten nach dem im Konzept entwickelten Prinzip. Mithilfe des Programms können die beschriebenen Netzwerke automatisch berechnet werden. Zusätzlich werden diese als Graph dargestellt.

Da das erarbeitete Konzept aus mehreren Teilaspekten besteht, die sowohl getrennt als auch gemeinsam betrachtet werden, spiegelt sich dies auch im Prototyp wider. Er bietet die Auswahlmöglichkeit eines einzelnen Teilaspekts oder der in Kapitel 4.2.5 beschriebenen Gesamtauswertung.

Außerdem ist für jeden Teilaspekt eine Auswahl der verschiedenen Wertetypen absolute (abs), normalisierte (norm), logarithmische (log) Werte und Differenzen (diff) möglich. Welcher Wert für welchen Teilaspekt betrachtet werden kann, wird hier noch einmal aufgelistet. Eine Begründung für oder gegen die Betrachtung ist im jeweiligen Kapitel des Konzeptes für den Teilaspekt zu finden.

Tabelle 5.1: Übersicht der Werte Typen für die Teilaspekte

Teilaspekt	**abs**	**norm**	**log**	**diff**
Sprecherfolge	✓	✓	✓	✓
Unterbrechungen	✓	✓		✓
Positive / Negative Interaktion	✓			
Positive und Negative Interaktion	✓			✓
Gesprächsabschnitte	✓			
Gesamtanalyse	✓			✓

5.1 Programmiersprache und Librarys

Bei der Implementierung des Prototypen kamen einige externe Librarys zum Einsatz. Diese, sowie die genutzte Programmiersprache, werden im folgenden kurz erläutert.

Programmiersprache: Python ist eine universelle Programmiersprache, die unter anderem Objektorientierung unterstützt. Außerdem fällt sie durch dynamische Typisierung und die fehlende Verwendung geschweifter Klammern auf. Anders als bei der Programmiersprache Java, wo Diese zur Kennzeichnung von Code-Blöcken verwendet werden, kommen bei Python Einrückungen zum Einsatz.

Die Wahl von Python als Programmiersprache für den Prototyp basiert auf mehreren Gründen.

Zum einen existieren bereits mehrere Librarys für Python, die Funktionen für die soziale Netzwerkanalyse und Darstellung von Graphen, sowie der Berechnung verschiedener Netzwerkgrößen bieten. Dies bietet eine zeitsparende Möglichkeit der Implementierung, die zusätzlich bereits getestet und erprobt sind. Dieser Vorteil ist jedoch nicht exklusiv für die Sprache Python. In anderen Programmiersprachen, wie zum Beispiel Java und R, existieren solche Librarys ebenfalls.

Allerdings handelt es sich bei R um eine Programmiersprache, die vorwiegend für Statistik und andere mathematischen Berechnung verwendet wird und die daher besonders auf diese Anwendungen zugeschnitten ist. Da für die Umsetzung des Konzepts nicht nur dieser Bereich benötigt wird, sondern auch andere Elemente wie eine graphische Oberfläche, ist R hier nicht ideal. Zwar wäre eine Umsetzung auch in dieser Sprache möglich, jedoch bieten sowohl Java als auch Python mehr Möglichkeiten über mathematische Berechnungen hinaus. Aus diesem Grund wurde sich gegen R als verwendete Programmiersprache entschieden.

Gegenüber Java hat Python den Vorteil, dass sie einen einfachen Umgang mit großen Datensätzen und Tabellen bietet. Diese können in einer Zeile eingelesen und anschließend die Werte ähnlich zu einem Array abgerufen werden. Da für das Programm die Transkripte als Tabellen geladen und nur einzelne Spalten davon analysiert werden müssen, ist dies ein Vorteil, der klar für Python spricht.

Aus diesen Gründen wurde für die Implementierung die Programmiersprache Python gewählt.

Librarys: Für die Umsetzung werden die Funktionen verschiedener externer Librarys verwendet. Welche zum Einsatz kommen und wofür diese

verwendet werden, wird im folgenden kurz erklärt.

PySimpleGui ist eine Library, die verschiedene Funktionen und Klassen zur einfachen Erstellung einer graphischen Oberfläche bietet [31]. Sie wird zur Gestaltung der GUI verwendet und unterstützt bei der Umsetzung deren Funktionalität.

Networkx bietet eine Reihe von Funktionen für die Erstellung, Darstellung und Analyse von Netzwerken und Graphen [11]. In dieser Arbeit kommen die Funktionen zur Erstellung und Speicherung von Netzwerken zum Einsatz. Mit ihrer Hilfe werden die sozialen Netzwerke konstruiert. Die Library bietet auch eine Reihe von Funktionen zur Berechnung verschiedener Netzwerkgrößen, die später noch Verwendung finden.

Die Library **Pyvis** ist auf die Visualisierung von Netzwerkstrukturen spezialisiert [14]. Sie bietet Funktionen, um Netzwerke zu erstellen und so darzustellen, dass die Anordnung von Knoten und Kanten innerhalb der Darstellung angepasst werden können. Diese Funktionalität kommt bei der Visualisierung der mit Networkx erstellten Graphen zum Einsatz. Eine Darstellung ist zwar auch mit dieser Library möglich, jedoch werden dort keine Schleifen, also Kanten von einem Knoten zu sich selbst, dargestellt, was aber für die Analyse der Sprecherfolge wünschenswert ist.

Pandas ist eine Library, die eine Vielzahl an Funktionen zur Datenanalyse bietet [28]. In dieser Arbeit werden vor allem die Funktionen zum Einlesen von csv-Dateien und das DataFrame verwendet. Dieses Objekt zur Speicherung von Tabellendaten bietet einfache und schnelle Möglichkeiten auf einzelne Zellen, Zeilen oder Spalten zuzugreifen, diese zu isolieren, gruppieren oder Berechnungen mit ihnen durchzuführen. Diese Funktionalitäten werden unter anderem für den Prozess der Netzwerk-Konstruktion verwendet.

Numpy bietet Funktionalitäten für den Umgang mit Vektoren, Matrizen und Arrays [12]. Sie wird ergänzend zu den Funktionen der Pandas Library für die Berechnung der relevanten Parameter und Kantengewichte verwendet.

5.2 Ein- und Ausgabedaten

Im folgenden werden die für die Nutzung des Prototypen benötigten Eingabedaten und die bei der Analyse entstehenden Ausgabedaten beschrieben.

Eingabedaten: Für die Nutzung des Prototypen werden zwei csv-Dateien benötigt, ein Transkript des zu analysierenden Meetings sowie eine Datei, die Informationen darüber enthält, welche act4teams Kategorie welchen Einfluss in der Analyse haben soll. Das Transkript kann hier vom Nutzer manuell gewählt werden während das Mapping bereits festgelegt ist. Dies kann aber für spätere Analysen angepasst werden.

Die Datei „Mapping.csv“ ist bereits im Ordner des Codes enthalten und wird automatisch eingelesen. Das Auslagern der entsprechenden Informationen in eine Datei hat gegenüber einer direkten Behandlung im Code den Vorteil, dass die entsprechenden Analyseparameter später einfach und schnell angepasst werden können.

Die Datei enthält eine Liste aller act4teams Kategorien. Diesen ist jeweils ein Code zugeordnet, der angibt welchen Einfluss eine Aussage dieser Kategorie auf Beziehung zweier Entwickler hat. Berücksichtigt sind dabei die folgenden Gruppen:

- u → Unterbrechung
- + → positiver Effekt
- a → positiver Effekt auf alle
- - → negativer Effekt
- s → Seitengespräch
- o → kein Effekt

Die Informationen, welche Kategorie aus act4teams zu welcher Gruppe gehört, kann in Anhang C nachgeschlagen werden. Alle Kategorien, die dort nicht aufgeführt sind, gehören zur Gruppe „kein Effekt“.

Das zu analysierende Meeting muss als Transkript im csv-Format vorliegen. Dieses sollte mit dem act4teams Schema kodiert und in einzelne Gesprächsabschnitte aufgeteilt sein. Nur so ist ein Analyse aller Teilbereiche möglich.

Damit die automatisierte Analyse ohne Probleme ablaufen kann, ist es zwingend notwendig, dass bestimmte Spalten in der Tabelle des Transkripts vorhanden sind und diese korrekt benannt wurden. Es werden die folgenden Spalten benötigt:

1. Teilnehmer
2. Code
3. Themenabschnitt

4. Transcript

Es werden nicht alle Spalten für alle Teilabschnitte der Analyse benötigt. Im folgenden wird kurz erläutert, was die jeweiligen Spalten enthalten müssen und wofür sie notwendig sind.

„Teilnehmer“: In dieser Spalte ist vermerkt, wer die jeweilige Aussage getätigt hat. Die Zuordnung kann über die Namen der Entwickler, IDs oder auch Abkürzungen geschehen. Da für alle Analysen identifiziert werden muss, wer wann gesprochen beziehungsweise wer was gesagt hat, ist diese Spalte essenziell wichtig für alle Analyseabschnitte.

„Code“: Diese Spalte enthält die Zuordnung der Aussage zu einer act4teams-Kategorie. Dabei ist in dieser Spalte nicht der Name der Kategorie, wie zum Beispiel „Ermunternde Ansprache“ vermerkt, sondern die dazugehörige Abkürzung, in diesem Beispiel „EA“. Diese Zuordnung ist für alle Analyseteile relevant, die sich auf das act4teams Schema beziehen, also für Unterbrechungen sowie positive und negative Interaktion.

„Themenabschnitt“: Diese Spalte gibt die Gesprächsabschnitte der jeweiligen Aussagen an. Diese können, wie die Sprecher auch, mit Namen, IDs oder Abkürzungen bezeichnet werden und sind nur für die Analyse der Gesprächsabschnitte von Bedeutung.

„Transcript“: Diese Spalte enthält das tatsächlich transkribierte Meeting, also die Aussagen, die im Meeting gefallen sind. Diese werden lediglich zur Identifikation von direkter Ansprache benötigt, also für die Analyse von positiver und negativer Interaktion. Dabei ist es wichtig, dass Namen in den transkribierten Aussagen auf die selbe Weise dargestellt werden, wie in der Spalte „Teilnehmer“, da eine Zuordnung andernfalls nicht möglich ist. Sollte kein Transkript vorhanden sein, kann diese Spalte leer bleiben, sollte aber in jedem Fall in der Tabelle existieren. In diesem Fall wird die Identifikation von direkter Ansprache nicht mit durchgeführt.

Einen Ausschnitt eines möglichen Transkripts findet sich in Anhang D.

Ausgabedaten Die vom Prototyp berechneten Netzwerke werden in drei verschiedenen Dateien und Formaten gespeichert. Der Pfad, in dem diese abgelegt werden, kann im Programmcode angegeben werden. Diese Variable *result_path* ist in der Datei *Analyser.py* zu finden. In dem dort angegebenen Ordner werden drei Unterordner angelegt, einen für jeden Dateityp, in dem die jeweiligen Ergebnisse der Analyse gespeichert werden. Es entstehen die folgenden Dateien:

1. gexf-Datei
2. html-Datei
3. csv-Datei

In der **gexf-Datei** wird das entstandene Netzwerk gespeichert. Auf diese Weise kann es später wieder in einem Programm geladen und wenn nötig weiter analysiert werden. Es werden dabei sämtliche im Netzwerk enthaltene Angaben und Labels mit gespeichert. Die Kantengewichte entsprechen dabei der Wert-Art, die bei der Analyse im Programm ausgewählt wurde.

Die **html-Datei** beinhaltet eine graphische Darstellung des Netzwerks als Graph. Diese Datei wird nicht nur nach der Analyse automatisch gespeichert, sondern auch direkt geöffnet. Wie bei der gexf-Datei sind die Kantengewichte auch hier von der gewählten Option abhängig. Ein Beispiel der Darstellung des Netzwerks für die Analyse der Sprecherfolge mit normalisierten Werten ist in Abbildung 5.1 gegeben.

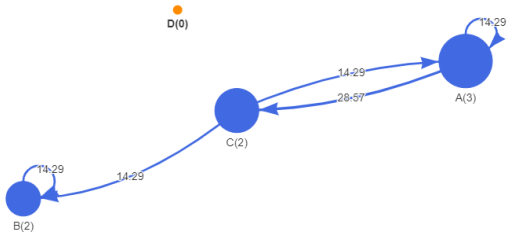

Abbildung 5.1: html-Ausgabe

Die Knoten und Kanten des Graphen können innerhalb der html-Datei verschoben und neu angeordnet werden. Auf diese Weise ist der Graph übersichtlich gestaltbar. Stumme Teammitglieder sind farblich gekennzeichnet und in der Gesamtanalyse werden auch Kanten mit einem negativen Score farblich hervorgehoben.

In einer **csv-Datei** werden sämtliche berechnete Größen gespeichert. Es werden nicht nur die Werte, die als Option ausgewählt wurden, sondern auch alle anderen für diese Teilanalyse möglichen Werte gesichert. Außerdem wird auch die Anzahl der Sprechbeiträge der jeweiligen Entwickler sowie bei der Gesamtanalyse die Ergebnisse aller Teilanalysen angegeben.

In Anhang E sind ein weiteres Beispiel für eine html-Ausgabe sowie Eines für eine csv-Ausgabe gegeben.

5.3 GUI

Im folgenden Kapitel wird die graphische Oberfläche des Prototypen beschrieben. Diese besteht aus drei verschiedenen Fenstern: einem Hauptfenster, das während der gesamten Ausführung des Programms geöffnet ist, einem Fenster für Warnmeldungen und Einem für Fehlermeldungen.

Das **Hauptfenster** bildet die hauptsächliche graphische Oberfläche des Prototypen. Über dieses Fenster werden die Analyse-Parameter definiert und die Analyse gestartet. Es besteht aus sechs Komponenten und ist in Abbildung 5.2 abgebildet. Die einzelnen Interaktionsmöglichkeiten werden im folgenden kurz beschrieben.

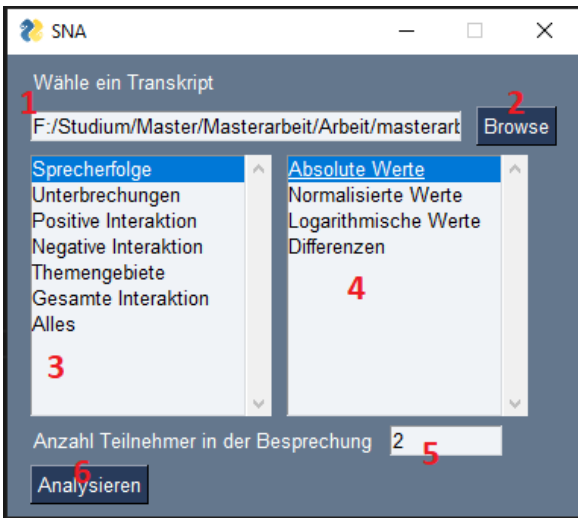

Abbildung 5.2: Hauptfenster

1. **Dateipfad:** Hier kann der Dateipfad für das zu analysierende Meeting-Transkript angegeben werden. Dabei ist es wichtig, dass es sich um einen absoluten Dateipfad handelt. Relative Dateipfade vom Speicherort des Codes oder nur eine Angabe des Dateinamens ist nicht ausreichend.
2. **Browse-Button:** Dieser Button kann zur Auswahl eines Meeting-Transkripts aus dem Speicher des Computers verwendet werden. Er öffnet bei Betätigung einen File-Manager, über den eine Datei im csv-Format von der Festplatte gewählt werden kann. Der Dateipfad der

gewählten Datei wird nach Bestätigung der Auswahl in das Textfeld (1) eingetragen.

3. **Analyse-Aspekte:** Hier wird eine Auswahl der möglichen Analyse-Aspekte angezeigt, inklusive der Möglichkeit einer Gesamtanalyse. Die ausgewählte Option wird farblich gekennzeichnet und für die Analyse verwendet. Eine Auswahl ist für den Start des Prozesses zwingend erforderlich.

4. **Analyse-Optionen:** Diese Liste enthält die zusätzlichen Analyse-Optionen, die bestimmen welche Kantengewichte, zum Beispiel normalisierte Werte, verwendet werden sollen. Da die möglichen Optionen von dem zu analysierenden Teilaspekt abhängen, passt sich die Liste dem Gewählten an. Daher kann die gewünschte Option erst nach Wahl des Analyse-Aspekts ausgesucht werden. Wird keine der Optionen, falls diese vorhanden sind, gewählt, so wird die Analyse zwar durchgeführt, allerdings wird kein Netzwerk angezeigt oder gespeichert. In diesem Fall wird nur die csv-Datei mit sämtlichen Werten erstellt.

5. **Teilnehmerzahl:** Hier kann die Anzahl der Teilnehmer des ausgewählten Meetings angegeben werden. Dies ist vor allem relevant, wenn nicht alle in dem entsprechenden Meeting gesprochen haben. Bleibt das Textfeld leer oder ist die angegebene Zahl kleiner als die Anzahl an Sprechern im Meeting, so wird stattdessen Diese für die Analyse verwendet. Die Angabe der Teilnehmerzahl ist also optional.

6. **Analysieren-Button:** Dieser Button startet die Analyse mit allen ausgewählten Parametern.

Warnung Ein Fenster für Warnmeldungen erscheint immer dann, wenn die Anzahl an Teilnehmern des Meetings, die in der Hauptauswahl angegeben wurde, größer ist als die Anzahl verschiedener Sprecher in dem Transkript. Die Warnung wird ausgegeben, um eventuell falsch angegebene Werte noch einmal korrigieren zu können. Der Analyseprozess ist pausiert solange die Warnmeldung angezeigt wird, kann aber über das Fenster wieder gestartet werden. Schließt man das Warnfenster, so wird auch das Hauptprogramm beendet. Die Oberfläche einer Warnmeldung ist in Abbildung 5.3 dargestellt. Das Fenster besteht aus vier Komponenten, die im folgenden kurz beschrieben werden.

1. **Warnung:** Hier wird die Warnmeldung angegeben. Sie enthält die Anzahl an Teilnehmern, die zwar gemäß der Analyseangaben am Meeting teilgenommen haben, allerdings nicht als Sprecher im Transkript auftreten.

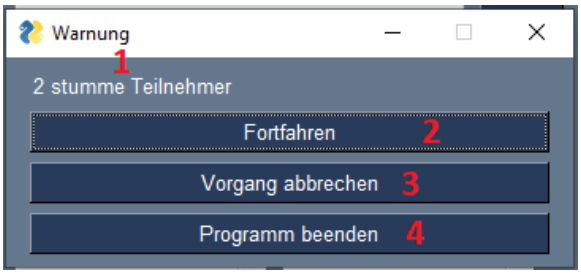

Abbildung 5.3: Warnmeldung

2. **Fortfahren-Button:** Dieser Button bestätigt, dass es sich bei den Angaben nicht um einen Fehler handelt und setzt den Analysevorgang wieder in Gang.

3. **Vorgang abbrechen:** Dieser Button bricht den Analysevorgang ab und bringt den Nutzer wieder zurück zur Hauptauswahl.

4. **Programm beenden:** Dieser Button beendet das gesamte Programm.

Fehlermeldung Eine Fehlermeldung wird angezeigt, wenn während der Ausführung der Analyse ein Problem auftritt, das die Durchführung des Prozesses nicht möglich macht und dieser daher abgebrochen wurde. Dies kann zum Beispiel auftreten, wenn der angegebene Dateipfad zum Transkript nicht existiert. Wie bei dem Warnfenster auch, beendet die Schließung des Fensters mit der Fehlermeldung das gesamte Programm. Das Fenster einer Fehlermeldung ist in Abbildung 5.4 dargestellt und besteht aus 3 Elementen.

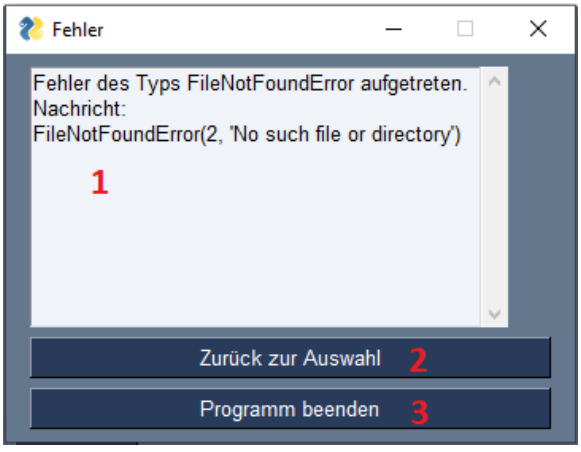

Abbildung 5.4: Fehlermeldung

1. **Fehlermeldung:** Hier wird die vom Programm generierte Fehlermeldung ausgegeben.

2. **Zurück zur Auswahl:** Dieser Button führt den Nutzer zurück zur Hauptauswahl. Dort können eventuell falsche Angaben korrigiert werden.

3. **Programm beenden:** Dieser Button beendet das gesamte Programm.

5.4 Klassen

Der Prototyp besteht aus vier verschiedenen Klassen. Diese werden im folgenden kurz vorgestellt und etwaige Besonderheiten näher beschrieben.

Die Klasse **Main** ist die Hauptklasse des Programms. Über sie wird die GUI gestartet und die Funktionalitäten dieser gesteuert, also die Liste der Optionen aktualisiert und die Analyse gestartet.
Außerdem werden über die Main-Klasse bei Start des Programms die Unterverzeichnisse innerhalb des angegebenen Dateipfads erstellt, in denen die Ergebnisse aller Analysen gespeichert werden. Dabei wird zunächst überprüft, ob das benötigte Verzeichnis bereits existiert und, sollte dies nicht der Fall sein, anschließend ein neues Verzeichnis angelegt.

In der Klasse **PopupWindow** ist das Layout für das Warn- und Fehlerfenster definiert. Da beide Fenster einen ähnlichen Aufbau und ähnliche Funktion haben, wird für beide die selbe Klasse verwendet. Die Entscheidung, welches Fenster angezeigt wird, kann über eine Variable beim Aufruf der Funktion open_window getroffen werden, die das entsprechende Fenster-Objekt zurück gibt. Ein Warnfenster wird ausschließlich aus der Klasse *Analyser* heraus geöffnet, wenn stumme Meeting-Teilnehmer entdeckt wurden. Ein Fehlerfenster kann hingegen sowohl in der Klasse *Main* als auch in der Klasse *Analyser* geöffnet werden. Dies geschieht, wenn ein Fehler mithilfe eines try-except-Konstrukts identifiziert wird.

Die Klasse **SelectionWindow** enthält das Layout für den Hauptteil der graphischen Oberfläche. Ihre Funktionalität wird allerdings von der Main-Klasse übernommen.

In der Klasse **Analyser** werden die Funktionen für die Analyse der Transkripte bereitgestellt. Auch das Laden der Transkripte und die Visualisierung werden hier durchgeführt. Dafür stehen die folgenden Funktionen bereit:

load_ transkript: In dieser Funktion wird das Transkript aus der angegebenen Datei geladen und für die Analyse vorverarbeitet. Es wird eine Liste der Teilnehmer, sowie die Anzahl Sprechbeiträge pro Teilnehmer herausgefiltert. Zusammen mit dem Transkript als DataFrame und einer Liste, die die Reihenfolge der Sprecher beinhaltet, zurückgegeben.

Da die für diese Arbeit analysierten Transkripte einige Unstimmigkeiten enthalten, müssen diese vorher behoben werden. So ist in der Spalte Teilnehmern oft „X“ oder „Y“ als Sprecher eingetragen, die aber keine Teammitglied repräsentieren. Außerdem kommt es vor, dass mehrere Entwickler in einem Sprechbeitrag als Sprecher angegeben sind. Dies ist nicht sinnvoll zu analysieren, da nicht nachvollzogen werden kann ob es sich um einen Formatierungsfehler handelt oder nicht. Daher werden diese Teilnehmer aus sämtlichen Rückgabewerten entfernt. Zudem wird manchmal als Sprecher „alle“ angegeben. Dies sagt aus, dass an diesem Beitrag, in der Regel „lachen“ oder Ähnliches, alle Entwickler beteiligt sind. Dies wird in der Analyse berücksichtigt, muss aber aus der Liste der Teilnehmer entfernt werden, da es sich bei „alle“ um keinen eigenen Teilnehmer handelt.

check_ for_ all: Diese Funktion löst im Transkript auftretende Fälle von „alle“ als Sprecher auf. Diese Funktion wird während der Analyse des Transkripts aufgerufen und ersetzt ein auftretendes „alle“ nacheinander durch jeden Entwickler im Meeting. Allerdings werden Situationen, wo mehrere Wortbeiträge hintereinander von allen kommen nur als ein einzelner Wortbeitrag gewertet.

Für die Visualisierung der berechneten Netzwerke stehen verschiedene Funktionen zur Verfügung, die an die speziellen Anforderungen der verschiedenen Analyseoptionen angepasst sind. In allen Funktionen wird anhand der Teilnehmerliste der Meetings und einer Matrix der in der Analyse berechneten Werte ein Netzwerk konstruiert, angezeigt und gespeichert. Dabei sind die Teilnehmer die Knoten und die Matrix enthält die Kantengewichte. Es gibt die folgenden Funktionen:

- *show_ weightes:* Diese Funktion wird für die absoluten und normalisierten Werte der Teilanalysen Sprecherfolge, positive/negative Interaktion und Unterbrechungen verwendet. Sie zeigt keine Kanten mit Gewicht 0 an

- *show_ log:* Diese Funktion wird für die logarithmischen Werte verwendet, da hier der Wert „-inf“ auftreten kann. Diese Kanten werden nicht angezeigt, während Kanten mit einem Gewicht von 0 farblich hervorgehoben werden.

- *show_ diff:* Diese Funktion stellt die Differenzen dar. Dabei werden

die Kantengewichte alle positiv und die Richtungen der Kanten entsprechend dem Konzept 4 angepasst.

- *show_ affiliation:* Diese Funktion baut spezifisch ein Affiliation Netzwerk für die Gesprächsabschnitte.

- *show_ weightes_ color_ coded:* Diese Funktion wird nur für die Gesamtanalyse verwendet. Sie funktioniert wie die Funktion show_weightes, allerdings sind hier Kanten mit positivem Kantengewicht in einer anderen Farbe gekennzeichnet als Kanten mit einem Negativen.

Für die Berechnung der Werte stellt die Klasse jeweils eine Funktion für die einzelnen Teilanalysen bereit. Die Funktionsnamen beginnen mit *compute_* . Die Berechnung folgt dem jeweiligen Algorithmus der Teilanalyse, wie er in Kapitel 4 beschrieben ist.

Alle bis hier beschriebenen Funktionen sind privat und lediglich zur internen Nutzung vorgesehen. Für das Starten der Analyse über die GUI stehen weitere Funktionen, deren Namen mit *analyse_* beginnen, bereit. Wie bei der Berechnung der Werte gibt es eine Funktion für jeden Teilaspekt. Allerdings stehen zusätzlich noch eine Funktion für die gemeinsame Betrachtung von positiver und negativer Interaktion sowie eine Funktion für die Gesamtanalyse zur Verfügung.

Diese Funktionen erhalten beim Aufruf die Information, welche Werteoption für die Darstellung der Ergebnisse verwendet werden soll. Mithilfe der bereits erläuterten Funktionen werden hier die notwendigen Daten geladen, die Werte berechnet und graphisch dargestellt. In dieser Funktion werden zusätzlich alle berechneten Werte gespeichert. Außerdem werden hier die Berechnungen durchgeführt, die notwendig sind um die gewählte Werteoption darstellen zu können und im Fall der Interaktions- und Gesamtanalyse werden die einzelnen relevanten Teilanalysen entsprechend dem Konzept verrechnet.

Kapitel 6

Evaluation

Im folgenden Kapitel soll das erstellte Konzept und der darauf aufbauend entwickelte Prototyp in Hinblick auf die Problemstellung evaluiert werden.

6.1 Forschungsfrage

In dem entwickelten Konzept werden die Transkripte auf Hinweise für die Beziehung zwischen den Meetingteilnehmern untersucht. Um diese Netzwerke sinnvoll nutzen und interpretieren zu können, sollte nun ein Einfluss dieser Netzwerke und der Stimmung innerhalb des Teams nachweisbar sein. In dieser Arbeit wird sich dabei auf zwei Fragen konzentriert:

Q1: Welcher Zusammenhang besteht zwischen den Größen des Beziehungsnetzwerks und der Stimmung innerhalb des Teams?

Q2: Welcher Zusammenhang besteht zwischen den Größen des Beziehungsnetzwerks und der von den Entwicklern wahrgenommenen Leistung des Teams?

6.2 Daten

Für die Evaluation werden 38 Meeting-Transkripte verwendet, die von qualifiziertem Personal mit dem act4teams Schema kodiert wurden. Die Daten stammen aus studentischen Softwareprojekten im Rahmen einer Lehrveranstaltung der Leibniz Universität Hannover aus den Wintersemestern 12/13, 13/14 und 15/16.

Es liegt jeweils ein Transkript des ersten offiziellen Meetings des Teams vor. Dieses findet direkt im Anschluss an das erste Treffen mit dem

Kunden des Projekts statt. In der Regel hat sich das Team zuvor bereits einmal für erste Besprechungen getroffen, hat aber sonst noch keine großen Verbindungen untereinander.

Da die Transkripte nicht direkt in Gesprächsthemen unterteilt sind, wurden diese für die Arbeit bei einigen Transkripten händisch nachgetragen. Dies ist allerdings unter anderem aufgrund der teils unvollständigen Transkription des Gesprochenen nicht immer eindeutig möglich. Da außerdem für jedes Team nur ein Transkript für die Evaluation vorhanden ist, wird der Bereich der Gesprächsthemen in dieser Arbeit nicht evaluiert. Hierfür erscheint nur eine Beobachtung der Beteiligung über einen längeren Zeitraum als aussagekräftig genug, die mit den vorliegenden Daten nicht möglich ist.

Um einen Zusammenhang zwischen den Werten der konstruierten Netzwerke und somit der vermuteten Beziehungen und der Leistung und Zufriedenheit des Teams herstellen zu können, sind Referenzdaten notwendig, die Auskunft über Teamstimmung und -performance geben. Diese liegen in einer großen csv-Datei für die verwendeten Transkripte vor. Dabei wurden die Teammitglieder sowohl vor als auch nach dem Meeting gebeten, anonymisiert einige Angaben zu ihrer Stimmung und Einschätzung des Teams zu machen. Teilweise wurden auch Kunden und Betreuer der Teams befragt.

Über eine Team-ID können diese Daten den Transkripten zugeordnet werden. Dabei sind die vorliegenden Transkripte aus der ersten Projektwoche. Es werden für die Evaluation also nur diese Daten aus der Datei verwendet. Die Datei gibt zusätzlich für jeden Studenten eine ID an. Diese können allerdings nicht mit den in den Transkripten verwendeten Buchstabenkürzeln für die Studenten gleich gesetzt werden. Eine Zuordnung bestimmter Stimmungsdaten zu den entsprechenden Entwicklern im Netzwerk ist also nicht möglich.

Für die Evaluation werden nicht alle im Datensatz angegebenen Größen verwendet. Es kommen lediglich die folgenden Punkte zum Einsatz: positiver und negativer Affekt (jeweils vor und nach dem Meeting), die Performance, die Menge an sozialen Konflikten (jeweils von den Entwicklern eingeschätzt) und das Vertrauen der Entwickler in ihr Team. Eine vollständige Liste aller im Datensatz vorhandenen Größen ist in Anhang F gegeben.

6.3 Konzept

Um die Forschungsfragen beantworten zu können, wird ein Studiendesign erstellt. Hierfür wird ein Evaluationskonzept entwickelt, dass festlegt, welche Größen betrachtet werden und welche Einflüsse erwartet werden. Wie in Kapitel 6.2 bereits erläutert, wird die Analyse der Gesprächsabschnitte aufgrund der vorliegenden Daten nicht in der Evaluation berücksichtigt.

Netzwerkgrößen: Für die Evaluation wird das Netzwerk durch verschiedene aus dem Netzwerk berechenbare Größen repräsentiert.

Zum einen kommen für die Evaluation die in Kapitel 2.2 erläuterten Kommunikationsmaße zum Einsatz, also die Communication Intensity, der Communication Score sowie die Overall Communication Intensity.

Die Communication Intensity wird in den konstruierten Netzwerken durch die einzelnen Kantengewichte dargestellt und ist hier also, anders als normalerweise, nicht von den Akteuren selbst eingeschätzt, sondern anhand der Transkripte berechnet worden. Da hier nur die Kommunikation innerhalb des Meetings betrachtet wird, ist eine Unterscheidung verschiedener Kommunikationskanäle hier nicht notwendig.

Für die Evaluation wird der Communication Score der Teammitglieder C_i verwendet, nicht der Communication Score zwischen zwei Akteuren C_{ij}. Dieser wird jedoch bei der Berechnung von C_i verwendet.

Da bei der Berechnung der Netzwerke jeweils nur ein Meeting berücksichtigt wird und die gesamte für die Analyse verwendete Kommunikation innerhalb dieses Meetings stattfindet, wird die Dauer des Meetings bei der Berechnung der Overall Communication Intensity nicht mit berücksichtigt. Die Formel ist darauf ausgelegt, die Gesamte Teamkommunikation zu analysieren, also auch die Kommunikation außerhalb des Meetings, und der Einbezug der Meetinglängen soll die generelle Kommunikation in Meetings mit einbeziehen. Da die hier stattfindende Analyse nur auf der Kommunikation in Meetings beruht, ist dieser zusätzliche Einbezug nicht notwendig.

Zusätzlich werden für die Evaluation auch die in Kapitel 2.1 beschriebenen Netzwerkgrößen verwendet. Da die Kanten in den konstruierten Netzwerken allerdings gewichtet sind, müssen die angegebenen Formeln angepasst werden, um auch die Kantengewichte in die Berechnungen mit einzubeziehen.

Für die Grad-Zentralität muss dazu die Art und Weise, mit der der Grad eines Knoten bestimmt wird, angepasst werden. Anstatt für den Grad lediglich die Anzahl der Kanten zu zählen, werden nun die Kantengewichte Aller addiert. Dies entspricht dem normalen Grad des Knotens, wenn alle Kanten ein Gewicht von +1 haben.

Für die Closeness-Zentralität müssen bei der Berechnung der geodätischen Distanz die Kantengewichte einbezogen werden. Eine Kante zwischen zwei Knoten hat dann nicht die Länge 1 sondern die Länge ihres Kantengewichts.

Referenzgrößen: Für die Betrachtung der Team-Stimmung und Performance werden die in 6.2 genannten Größen verwendet.

Der positive Affekt wird als Indikator für die Team-Stimmung genutzt. Um einen Einfluss des Meetings auf die positive Stimmung erkennen zu können, wird die Veränderung des positiven Affekts durch das Meeting

betrachtet. Es liegt der positive Affekt jeweils vor und nach dem Meeting vor, sodass die Differenz der beiden Werte, also die Veränderung durch beziehungsweise während des Meetings betrachtet werden kann. Da der Wert des positiven Affekts zwischen 0 und 5 liegt, befindet sich die Differenz in einem Intervall von -5 und +5.

Der negative Affekt wird, wie der positive Affekt, auch als Indikator für die Team-Stimmung gesehen. Wie beim Positiven wird auch beim negativen Affekt die Differenz der Werte vor und nach dem Meeting und somit die Veränderung verwendet. Der Wertebereich der Differenz ist identisch mit dem des positiven Affekts.

Die Größe Social Conflicts beschreibt die Einschätzung der Teilnehmer, wie viele beziehungsweise wie stark soziale Konflikte im Team bestehen. Da soziale Konflikte negativen Einfluss auf die Stimmung innerhalb des Teams haben, wird auch diese Größe zur Beantwortung von Q1 heran gezogen. Die Angaben der Einschätzungen sind in einem Bereich zwischen 1 (keine / wenig soziale Konflikte) und 6 (viele soziale Konflikte). Dabei wird von jedem Teammitglied ein eigener Wert angegeben.

Für die Einschätzung der Teamleistung wird zum einen die vom Team geschätzte Performance herangezogen. Hierbei handelt es sich um eine Einschätzung eines jeden Entwickler, wie gut die Leistung des Teams ist beziehungsweise war. Die Werte liegen hier zwischen 0 und 5, wobei ein höherer Wert auch eine bessere Performance bedeutet.

Zum anderen wird für die Teamleistung das Vertrauen der Entwickler in das Team betrachtet. Hierfür wurden die Teammitglieder gefragt, wie viel Vertrauen sie darin haben, dass das Team gut zusammen arbeitet und das Projekt erfolgreich abschließen wird. Hier liegen die Werte zwischen 1 und 6.

Erwarteter Einfluss: Um die Ergebnisse der Evaluation besser einordnen und interpretieren zu können, wird im folgenden kurz beschrieben, welcher Zusammenhang zwischen den Größen aus den Netzwerken und den Referenzgrößen vermutet wird.

Im folgenden Absatz wird der Begriff *positiver Effekt* als Synonym für eine positive Veränderung der Teamstimmung und Performance verwendet. Dieser positive Effekt ist dabei immer relativ zum Verlauf, egal ob steigend oder fallend, einer der Größen des Netzwerks zu sehen. Ein positiver Effekt bedeutet für die einzelnen Referenzgrößen das Folgende:

Ein hoher positiver Affekt ist ein Zeichen für eine gute Teamstimmung. Daher wird bei einem positiven Effekt dieser höher.

Ein hoher negativer Affekt hingegen ist ein Zeichen für schlechte Teamstimmung. Daher sinkt der negative Affekt bei einem positiven Effekt.

Da soziale Konflitke einen negativen Einfluss auf die Stimmung im Team haben, wird die Einschätzung der sozialen Konflikte seitens der Meetingteilnehmer bei einem positiven Effekt geringer.

Da sowohl hohe Performance als auch starkes Vertrauen ins Team einen positiven Einfluss auf das gesamte Projekt und dessen Erfolg haben, steigen beide Größen bei einem positiven Effekt an.

Hohe Communication Intensities bedeuten nach dem Konzept gute Beziehungen und viel positive Kommunikation. Aus diesem Grund wäre zu erwarten, dass bei steigenden Communication Intensities ein positiver Effekt eintritt, während sich die Werte bei sinkenden Zahlen entgegengesetzt entwickeln dürften.

Ein gleicher Zusammenhang ist für die Communication Scores und die Overall Communication Intensity zu erwarten, da hohe Communication Scores der Teammitglieder ebenfalls gute Beziehungen und positive Kommunikation repräsentieren.

Da eine hohe Grad-Zentralität eine hohe Verbundenheit des Knotens mit den Anderen bedeutet, ist auch hier von einem positiven Effekt bei steigendem Wert auszugehen.

Da in den konstruierten Netzwerken hohe Kantengewichte und damit hohe godätische Distanzen ein Zeichen von positiver Beziehung sind, ist auch eine hohe Gesamtdistanz eines Knoten besser als eine niedrige. Da die Closeness-Zentralität über den Kehrwert der Gesamtdistanz gebildet wird, ist in diesem Fall also eine kleinere Closeness-Zentralität ein Zeichen für bessere Beziehungen. Daher ist hier ein positiver Effekt bei fallender Closeness-Zentralität zu vermuten.

6.4 Vorbereitungen

Um die Evaluation durchführen zu können, muss der erstellte Prototyp leicht angepasst und erweitert werden.

Da eine Berechnung der Netzwerke aller Transkripte mit allen Teilanalysen und Einstellungen über die graphische Oberfläche sehr lange dauern würde, werden diese für die Evaluation über Programmcode gestartet.

Für die Berechnung vieler Netzwerke in möglichst kurzer Zeit ist es wichtig, dass die Visualisierungen der Netzwerke zwar erstellt und gespeichert, aber nicht automatisch geöffnet wird. Dies ist für die Evaluation nicht notwendig, würde aber dem Programm zusätzliche Arbeit machen und so die Laufzeit verlängern. Um dies zu vermeiden, wurden die Funktionen zum Start der Analyse sowie die zur Visualisierung um eine optionale Variable erweitert. Diese enthält einen Wahrheitswert, der bei fehlender

Angabe auf wahr gesetzt ist. Ist der angegebene Wert wahr, so wird die Visualisierung des Netzwerks, wie in Nutzung des normalen Prototyp üblich, automatisch geöffnet. Andernfalls wird die Visualisierung zwar erstellt und gespeichert, jedoch nicht geöffnet. Während der Evaluation wird immer der Wert *False* übergeben.

Für die Berechnung der Netzwerke, der für die Evaluation benötigten Größen des Netzwerks und das Laden der Referenzgrößen wird eine zusätzliche Klasse „Evaluation“ verwendet. Diese enthält einige Funktionen für Berechnungen und Analyse sowie ein Skript, um alle für die Evaluation genutzten Transkripte zu verwerten. Im folgenden werden die implementierten Funktionen beschrieben. Auf das durchführende Skript wird in Kapitel 6.5 näher eingegangen.

Die Funktion *compute_networks* berechnet alle Netzwerke zu einem übergebenen Transkript. Dafür werden die in der Klasse Analyser bereitgestellten Funktionen für die Teilanalysen und Gesamtanalyse, insgesamt sieben verschiedene Funktionen, jeweils mit allen für sie möglichen Einstellungen aufgerufen. Da die konstruierten Netzwerke bereits während der Berechnung gespeichert werden, muss dies innerhalb der Funktion nicht getan werden.

In der Funktion *compute_network_values* werden die für die Evaluation benötigten Größen aus einem Netzwerk berechnet. Dafür wird das mit Dateipfad übergebene Netzwerk geladen und die Werte entsprechend der in den Grundlagen beschriebenen Formeln berechnet.
Die Berechnung der Closeness-Zentralität ist jedoch in diesem Fall nicht über die von Networkx bereitgestellte Funktion möglich, da diese nicht mit gewichteten Kanten arbeitet. Für die Berechnung der kürzesten Pfade war zudem ein Algorithmus notwendig, der negative Kantengewichte berücksichtigt. Hierfür wurde die Funktion „bellman_ford_predecessor_and_distance“ verwendet. Diese kann aber nicht immer ein Ergebnis liefert und die Gesamtdistanz eines Knoten bei negativen Kantengewichten kann auch 0 sein. Da dies die Berechnung des Kehrwerts nicht immer möglich macht, wird die Closeness-Zentralität in diesen Fällen auf 0 gesetzt.

Die Funktion *load_data* lädt die für die Analyse relevanten Referenzdaten aus dem vorliegenden Datensatz. Dafür wird der Funktion die Team ID des gewünschten Teams übergeben, die in dem Datensatz in der Spalte „Team ID“ angegeben ist. Da die für die Evaluation vorliegenden Transkripte nach dem Schema *Team<ID>.csv* benannt sind, kann die ID für ein Transkript leicht aus dem Dateinamen extrahiert werden. Da die

vorliegenden Transkripte aus der ersten Projektwoche stammen, werden nur die Daten aus dieser Woche geladen.

Für den positiven und negativen Affekt werden nicht nur die Daten geladen, sondern auch gleich die Differenzen zwischen den Werten vor und nach dem Meeting berechnet. Dabei wird der Wert vor dem Meeting von dem Wert nach dem Meeting subtrahiert. Eine positive Differenz bedeutet also einen höheren Wert nach dem Meeting.

In der Funktion *compute_data* werden nun alle für die Evaluation relevanten Daten zusammen geführt. Für ein über den Dateipfad angegebenes Netzwerk werden sowohl die Größen des Netzwerks berechnet als auch die Referenzdaten geladen. Diese werden anschließend in einer gemeinsamen csv-Datei gespeichert. Dabei sind die Werte der Größen jeweils in einer eigenen Spalte angegeben. Die Dateien sind nach dem Schema *Name der Netzwerkdatei + _eval.csv* benannt. Sie werden dann in einem extra angelegten Ordner „EVAL“ innerhalb des Ordners gespeichert, in dem auch die berechneten Netzwerke gespeichert werden.

6.5 Durchführung

Für die Evaluation werden 38 Transkripte analysiert. Über ein Python-Skript werden für jedes Transkript die 7 Teilanalysen mit jeweils 1 - 4 verschiedenen Einstellungen durchgeführt. Insgesamt werden pro Transkript 14 Netzwerke konstruiert, zusammengenommen wären das also 532 Netzwerke. Allerdings sind nur in einem Teil der Transkripte die Gesprächsthemen eingetragen, es werden also insgesamt weniger Netzwerke analysiert. Bei einigen Transkripten war die Einteilung in Gesprächsteile nicht möglich, da keine Transkriptionen des Gesprochenen sondern lediglich die Reihenfolge der Sprecher sowie die Einteilung in act4teams vorhanden sind. Bei den Anderen wurde diese Einteilung teils durch eine große Anzahl von als unverständlich eingetragenen Aussagen und der Aufteilung einer Aussage in mehrere Spalten der Tabelle erschwert. Aus diesem Grund wurden nur sechs Transkripte gewählt, bei denen die Einteilung möglich war und nicht zu stark eingeschränkt wird.

Für die Durchführung der Evaluation werden für alle im angegeben Dateipfad vorhandenen Transkripte sämtliche Netzwerke, die berechnet werden können, konstruiert, anschließend geladen und die entsprechenden Größen berechnet oder aus dem Datensatz extrahiert. Zusätzlich wird für die spätere Darstellung und Interpretation der Daten die Länge der jeweiligen Meetings, in diesem Fall nicht auf die Zeit sondern auf die Anzahl an Sprechbeiträgen bezogen, in einer csv-Datei gespeichert.

Die Ergebnisse der Analysen stehen nun in csv-Dateien zur Verfügung. Diese Darstellung der Daten ist allerdings für eine Interpretation der Ergebnisse nur bedingt geeignet, da lediglich Werte aufgelistet sind, sie aber nicht miteinander in Beziehung gesetzt sind. Um eine übersichtlichere Darstellung der Ergebnisse und der eventuellen Zusammenhänge zwischen den Größen des Netzwerks und den Referenzgrößen zu erreichen, werden die Daten zusätzlich graphisch aufbereitet.

Für die graphische Darstellung wird die Programmiersprache Python mit der Web-Applikation Jupyter Notebook verwendet [22]. Dies ermöglicht eine einfache Generation von Diagrammen und eine übersichtliche Struktur in der Auswertung.

Innerhalb der Auswertung kommen noch zwei zusätzliche Librarys zum Einsatz:

Die Library *Scipy* ist eine Sammlung von numerischen Algorithmen, unter anderem für den Bereich der Optimierung [42]. In der Auswertung kommt sie zum Einsatz, um eine Nährungskurve für die Werte des Diagramms zu berechnen.

Matplotlib bietet eine Sammlung von Funktionen und Klassen, um Diagramme zu erstellen, anzuzeigen und zu speichern [13]. In dieser Arbeit kommen vor allem die Funktionen zur Darstellung und Gestaltung von Liniendiagrammen zum Einsatz.

Die graphische Darstellung der Ergebnisse erfolgt im Rahmen eines Liniendiagramms. Da ein Diagramm nicht alle Daten darstellen kann, werden verschiedene Diagramme erstellt. Für jede der 14 verschiedenen möglichen Netzwerke, also jeder Kombination aus Teilanalyse und Einstellung, wird ein eigenes Diagramm gezeichnet.

Bei den Diagrammen sind die Netzwerkdaten auf der x-Achse und die Referenzdaten auf der y-Achse dargestellt. Für eine übersichtliche Darstellung der Relationen zwischen der jeweiligen Größe des Netzwerks und den fünf Referenzgrößen wird für jede der Referenzgrößen ein eigenes Diagramm erstellt. Da die Darstellung mehrerer verschiedener Werte mit unterschiedlichen Einheiten und Wertebereichen auf der x-Achse sehr unübersichtlich wäre, wird jede der fünf Größen des Netzwerks in einem eigenen Teil-Diagramm dargestellt. Insgesamt werden also für jedes der 14 möglichen Netzwerke fünf Diagramme berechnet.

Ein Diagramm beinhaltet die Daten aller Teams für diese Einstellungen. Jedes Team wird mit einem Punkt innerhalb des Diagramms repräsentiert. Auf diese Weise kann ein eventueller Zusammenhang zwischen Netzwerk- und Referenzgröße einfacher erkannt werden.

Ein Problem bei der Darstellung als Linien-Diagramm für diese Daten ist, dass alle Referenzgrößen mehr als einen Wert für den selben Wert der Netzwerkgröße hat. Dies kommt daher, dass jedes Teammitglied eine separate Referenzgröße hat, sie sich aber die selben Größen des Netzwerks teilen. Eine Darstellung aller einzelnen Punkte würde das Diagramm jedoch zu unübersichtlich machen. Aus diesem Grund werden nur der Durchschnittswert sowie die Spannweite der Angaben dargestellt. Der Durchschnittswert der Angaben für die jeweilige Referenzgröße eines Teams wird als Punkt im Diagramm vermerkt und alle Punkte mit einer Linie verbunden. Zusätzlich wird der Bereich zwischen der kleinsten und größten Angabe der Referenzgröße an jedem Punkt durch eine blaue Fläche gekennzeichnet. Auf diese Weise sind sowohl Mittelwert als auch Streuung für jeden Punkt, also jedes Team, ersichtlich.

Ein weiteres Problem ist, dass es auch bei den einzelnen Größen mehr als einen Wert pro Netzwerk gibt. Mit Außnahme der Overall Communication Intensity, die das gesamte Netzwerk mit einbezieht, sind die berechneten Größen spezifisch auf einen oder zwei Knoten im Netzwerk bezogen. Da eine Zuordnung zu den Referenzdaten allerdings nicht eindeutig möglich ist, kann dieser Konflikt nicht auf diese Weise aufgelöst werden. Stattdessen wird hier der selbe Ansatz wie beim ersten Problem gewählt. Jedes der fünf Teil-Diagramme wird abermals in zwei Teile getrennt. In einem der Beiden wird auf der x-Achse der Durchschnitt aller Werte der jeweiligen Netzwerkgröße für das Team abgebildet, in dem Anderen die Standartabweichung der Werte. Auf diese Weise können auch hier die verschiedenen Werte berücksichtigt werden.

Zusätzlich zur geraden Linie zwischen den jeweiligen Datenpunkten wird in den Diagrammen auch noch eine Nährungsfunktion als gestrichelte blaue Linie angezeigt. Hierfür wird eine automatische Approximation mit der Approximationsfunktion $a * x^3 + b * x^2 + c * x + d$ durchgeführt.

Bei der Betrachtung der auf diese Weise erstellten Diagramme wurde ein zusätzliches Problem sichtbar. Da die Meetings teils sehr unterschiedliche Längen haben und die Kantengewichte der Netzwerke, vor allem bei der Analyse der Sprecherfolge, bei längeren Meetings deutlich größer werden können als bei Kürzeren, ist die Vergleichbarkeit der Team-Meetings untereinander und damit die Aussagekraft der Diagramme recht gering. Um diesen Phänomen entgegenzuwirken, werden die Größen der Netzwerke vor der Erstellung der Diagramme mithilfe ihrer Länge normalisiert. Die Länge der Meetings wurde dabei bereits bei der Berechnung aller Netzwerke und Daten bestimmt und gespeichert. So können die Ergebnisse der einzelnen Teams besser miteinander verglichen werden. Es werden nun allerdings nicht mehr die tatsächlichen Netzwerkgrößen, sondern (Größe / Meeting-Länge) *

100

Im folgenden ist ein Beispiel-Diagramm dargestellt. Dabei handelt es sich lediglich um ein Teil-Diagramm, dass die Durchschnittswerte auf der x-Achse zeigt. Ein Gesamt-Diagramm besteht aus fünf beziehungsweise zehn Teil-Diagrammen, je nach Netzwerkgröße. Eine Sammlung ausgewählter Diagramme, die bei der Evaluation entstanden sind finden sich in Anhang G. Sie werden im folgenden Kapitel 6.6 näher betrachtet.

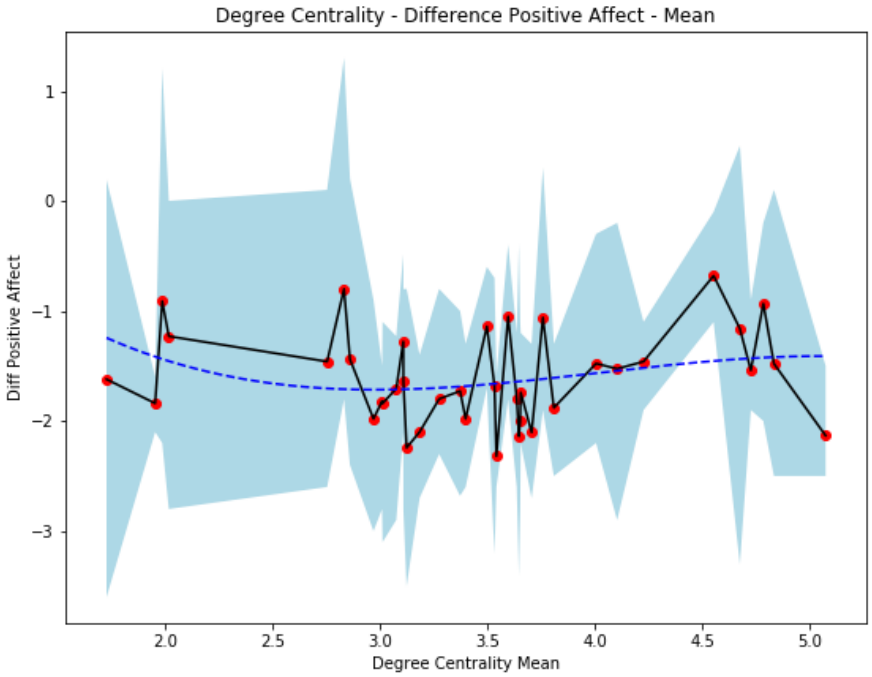

Abbildung 6.1: Beispiel-Diagramm

6.6 Ergebnisse

Nun, da alle Berechnungen abgeschlossen sind und die Daten graphisch aufbereitet wurden, können die Ergebnisse näher betrachtet werden. Dies geschieht in drei Schritten. Zunächst werden einige ausgewählte Netzwerke dargestellt und Auffälligkeiten aufgezeigt. Anschließend werden die Diagramme betrachtet und für den Menschen sichtbare Auffälligkeiten aufgeführt. Als letztes wird eine statistische Untersuchung durchgeführt.

Zunächst werden die **generierten Netzwerke** betrachtet. Hierfür sind in den Abbildungen 6.2, 6.3 und 6.4 drei ausgewählte Netzwerke dargestellt. Es wird jeweils die Gesamtanalyse für die drei Teams abgebildet.

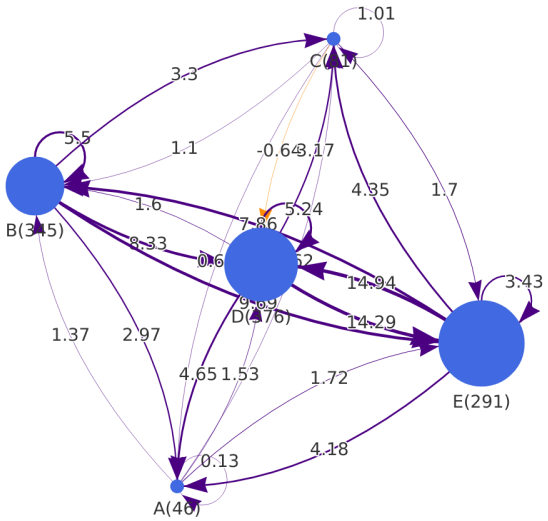

Abbildung 6.2: Graph der Gesamtanalyse von Team 05

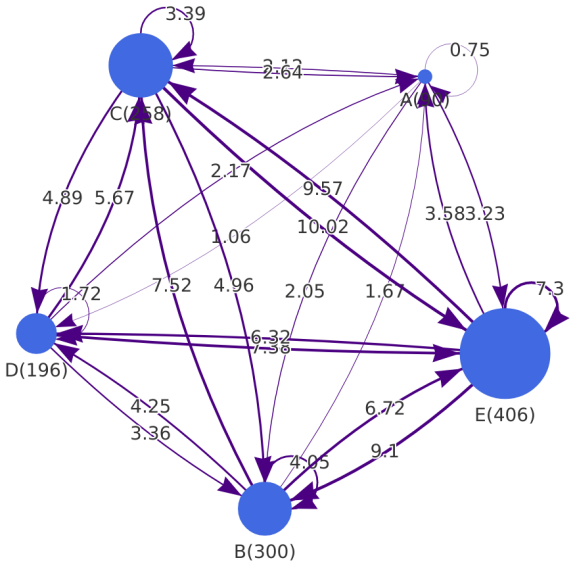

Abbildung 6.3: Graph der Gesamtanalyse von Team 08

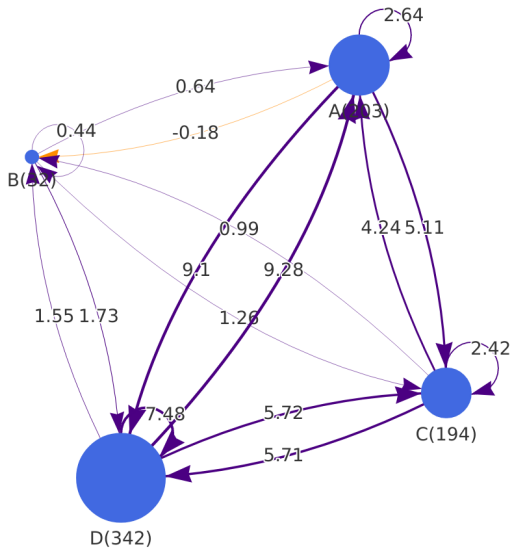

Abbildung 6.4: Graph der Gesamtanalyse von Team 23

Bei der Betrachtung der Netzwerke fällt auf, dass es insgesamt nur sehr wenige Kanten mit negativem Kantengewicht innerhalb der generierten Netzwerke gibt. In einem Großteil der Netzwerke sind gar keine negativen Kanten vorhanden und in den anderen jeweils nur Eine. Dies bedeutet, dass es nur wenig bis gar keine deutlich negativen Beziehungen innerhalb der Teams zu geben scheint. Außerdem zeigen die Netzwerke der Teilanalysen ein sehr geringes Maß an Unterbrechungen und negativen Interaktionen, was das großteilige Fehlen negativer Kanten erklärt.

Eine weitere Auffälligkeit, die ein Großteil der Netzwerke gemeinsam haben, ist die generell höheren Kantengewichte zwischen Entwicklern, die sich viel im Meeting beteiligen. Dies ist logisch, da mehr Kommunikation auch mehr Möglichkeiten bietet eine Beziehung aufzubauen und auszudrücken. Hier ist auch auffällig, dass es häufig einen großen Unterschied in der Beteiligung der einzelnen Entwickler gibt. Diese ist durch die Knotengröße der Netzwerke graphisch dargestellt. In dem Netzwerk in Abbildung 6.2 beispielsweise ist erkennbar, dass sich A und C deutlich weniger beteiligen als die anderen drei Entwickler.

In dem Netzwerk aus Abbildung 6.2 fällt besonders die negative Kante von C nach D auf. Auch wenn hier nur ein negatives Kantengewicht von -0.64 besteht, so deutet diese dennoch auf ein Konfliktpotential hin. Zudem liegt hier eine relativ große Differenz zwischen den Kanten C → D und D→ C sowie den Kanten zwischen B und D vor. Besonders gut scheint die Beziehung zwischen D und E sowie zwischen E und B zu sein, während A und C sich nur wenig im Meeting beteiligen.

In dem Netzwerk aus Abbildung 6.3 fällt auf, dass es keine negativen Kanten, dafür aber viele mit kleinen positiven Kantengewichten gibt. Außerdem ist Entwickler E hier sehr dominant, was auch durch das hohe Kantengewicht der Kante E $\rightarrow$ E deutlich wird, die durch Interaktionen von E mit allen und ein wiederholtes Sprechen von E hintereinander entsteht. Zudem scheinen besonders die Beziehungen zwischen C und E, D und E, E und B, sowie B und C besonders gut zu sein.

In dem Netzwerk aus Abbildung 6.4 fällt zunächst die negative Kante A $\rightarrow$ B auf. Außerdem ist B in diesem Netzwerk extrem wenig beteiligt und scheint lediglich mit D eine ein wenig positivere Beziehung zu haben. Da auch A und C jeweils eine gute Beziehung zu D zeigen, ist D hier eindeutig der zentrale Knoten.

Eine weitergehende Interpretation dieser Auffälligkeiten findet sich in Kapitel 7.1.

Im nächsten Schritt werden zunächst die **Diagramme** analysiert, welche die absoluten Werte der Gesamtanalyse abbilden. Die entsprechenden fünf Diagramme sind in Anhang G zu sehen. Generell sind die Werte in allen Diagrammen sehr ausgeglichen, es lassen sich aber dennoch ein paar leichte Tendenzen erkennen.

Bei der Closeness-Zentralität zeigt sich eine leichte Tendenz, dass bei einem niedrigeren Wert für die Closeness-Zentralität eher eine höhere Performance eingeschätzt wird, sich also ein positiver Effekt für die Performance beobachten lässt. Bei den sozialen Konflikten hingegen sind diese bei einer geringen Closeness-Zentralität eher höher, was einem eher negativen Effekt zuzuordnen wäre. Allerdings sind dies nur Tendenzen. Da ein großer Teil der Teams eine recht kleine Closeness-Zentralität aufweisen, ist eine größere Tendenz nicht erkennbar.

Die Werte der Communication Scores sind gleichmäßiger über die x-Achse verteilt. Dennoch lassen sich auch hier nur leichte Tendenzen mithilfe der Nährungsfunktionen feststellen. So ist bei steigendem Communication Score eine leichte Tendenz zu einem positiven Effekt für die eingeschätzte Performance und die sozialen Konflikte erkennbar. Die von den Entwicklern eingeschätzte Performance zeigt eine leicht steigende und die sozialen Konflikte eine leicht sinkende Tendenz. Allerdings sind hier ebenfalls nur leichte Tendenzen aufgrund der Nährungsfunktion sichtbar.

Einige letzte Tendenzen sind bei der Grad-Zentralität zu beobachten. Diese sind jedoch minimal und vor allem über die Nährungsfunktion erkennbar. Bei einer steigenden Grad-Zentralität sind hier eine leicht steigende Performance und ein leicht sinkender sozialer Konflikt zu beobachten. Diese Tendenzen

sind jedoch nur leicht zu erkennen und werden durch viele recht unterschiedliche Angaben zu sozialen Konflikten und Performance bei ähnlicher Grad-Zentralität deutlich eingeschränkt.

Insgesamt sind bei allen Teams drei Phänomene unabhängig von den Netzwerkgrößen zu beobachten. Bei allen Teams ist der durchschnittliche positive Affekt nach dem Meeting niedriger als davor. Dies lässt normalerweise auf eine Verschlechterung der Team-Stimmung durch das Meeting schließen. Allerdings ist das selbe Phänomen auch beim negativen Affekt zu beobachten. Auch hier sinkt der Wert bei allen Teams nach dem Meeting im Vergleich zu davor, was normalerweise auf eine Verbesserung der Team-Stimmung hindeuten würde. Aufgrund dieser sich entgegenstehenden Werteentwicklungen und der Tatsache, dass sich die Werte bei allen Teams sehr stark ähneln, ist bei positivem und negativem Affekt keinerlei Tendenz der Werteentwicklung erkennbar.

Auch der von den Teammitgliedern durchschnittlich angegebene Wert über ihr Vertrauen in das Team ist in diesen Daten nicht aussagekräftig. Er liegt bei allen Teams sehr hoch, zwischen vier und sechs. Auch hier ist in keinem der Diagramme eine Tendenz erkennbar.

Insgesamt sind in allen Diagrammen keine klaren Zusammenhänge erkennbar. Es zeigen sich zwar an einigen Stellen leichte Tendenzen bezüglich der geschätzten Performance und der sozialen Konflikte, allerdings sind dies nur Tendenzen.

Bei der Betrachtung der Diagramme für die einzelnen Teilanalysen und anderen Werteeinstellungen sind die Ergebnisse und Tendenzen ähnlich zu denen der Gesamtanalyse. Es sind auch dort keine klaren Zusammenhänge erkennbar.

Für die **statistische Untersuchung** werden zwei Korrelationskoeffizienten für alle Diagramme berechnet, die den Grad der Korrelation, also des Zusammenhangs, zwischen den Durchschnittswerten der Netzwerkgrößen und Denen der Referenzgrößen angibt.

Der Pearson Koeffizient liefert einen Wert zwischen -1 und +1 [39]. Werte von 0 oder nahe 0 bedeuten dabei, dass kein Zusammenhang zwischen den Variablen festgestellt werden konnte. Je näher der Wert gegen +1 oder -1 geht, desto stärker ist der festgestellte Zusammenhang. Positive Werte bedeuten hierbei, dass der y-Wert bei steigendem x-Wert größer wird, während negative Werte für einen negativen Zusammenhang, also sinkende y-Werte bei steigendem x-Wert, steht. Der Pearson Koeffizient wird wie folgt berechnet [39]:

$$r_p = \frac{\sum xy}{N\sigma_x\sigma_y} \tag{6.1}$$

$$mit$$

$$x = (X - \overline{X}); y = (Y - \overline{Y})$$

$$\sigma_x = \text{Standardabweichung von X}$$

$$\sigma_y = \text{Standardabweichung von Y}$$

$$N = \text{Anzahl von Paaren in Beobachtung}$$

$$r_p = \text{Korrelationskoeffizient}$$

Der Spearman Koeffizient ist eine Variation des Pearson Koeffizienten und liefert Werte im selben Wertebereich, wird jedoch anders berechnet [39]. Da der Pearson Koeffizient unter der Annahme basiert, dass die untersuchte Population normalverteilt ist, während dies bei Spearman nicht notwendig ist, und dies in diesem Fall nicht garantiert werden kann, werden beide Werte betrachtet. Der Spearman Koeffizient wird wie folgt berechnet [39]:

$$r_s = 1 - \frac{6\sum D^2}{N^3 - N} \tag{6.2}$$

$$mit$$

$$D = \text{Unterschied des Ranges zwischen Paaren,}$$

wobei der Rang durch die numerische Sortierung der Wertereihen gegeben ist

$$N = \text{Anzahl von Paaren in Beobachtung}$$

$$r_s = \text{Korrelationskoeffizient}$$

Eine Auflistung aller berechneten Werte für alle Wertepaare ist in Anhang H gegeben. Ein großer Teil der Koeffizienten liegen im Bereich zwischen -0.1 und +0.1 und ein weiterer Teil zwischen -0.4 und +0.4, also in einem Bereich, wo keine signifikante Korrelation festgestellt werden konnte.

Es gibt jedoch einige Auffälligkeiten. Bei der Analyse der Unterbrechungen ist bei der Beziehung zwischen dem positiven Affekt und Communication Score, Communication Intensity und der Overall Communication Intensity $r_p = 0.43$. Allerdings ist r_s hier deutlich kleiner. Außerdem liegen bei der Beziehung von Social Conflict mit der Grad-Zentralität Werte von $r_p = 0.3$ und $r_s = 0.44$ vor.

Bei der Analyse der Negativen Interaktion sind einige Werte höher als 0.4 beziehungsweise kleiner als -0.4. Dies trifft auf die Werte für den positiven Affekt in Zusammenhang mit den drei Kommunikationsmaßen, die der Sozialen Konflikte für Gradzentralität und Communication Score sowie die für den Punkt Trust und Communication Score, Grad- und Closeness-Zentralität. Hier gehen die Werte teilweise bis zu einem Betragswert von 0.7.

Bei der Betrachtung der Sprecherreihenfolge fällt nur die Beziehung zwischen Sozialen Konflikten und der Grad-Zentralität auf, wo $r_s = 0.42$ ist.

Eine Interpretation der hier beschriebenen Ergebnisse sowie die Beantwortung der gestellten Forschungsfragen folgt in Kapitel 7.

Kapitel 7

Diskussion

In diesem Kapitel werden die Ergebnisse der Evaluation, die in Kapitel 6.6 beschrieben wurden, näher betrachtet. Ziel ist es, aus den Diagrammen Schlüsse über die Beziehung zwischen den Netzwerkdaten und den Referenzdaten zu ziehen, die Ergebnisse zu interpretieren und damit die in Kapitel 6.1 gestellten Fragen zu beantworten.

Zudem wird in diesem Kapitel eine Einordnung der Ergebnisse vorgenommen, indem die Einschränkungen der Arbeit, Analyse und Evaluation aufgezeigt und beschrieben werden.

7.1 Interpretation

Um die in Kapitel 6.1 gestellten Forschungsfragen beantworten zu können, müssen die Ergebnisse aus Kapitel 6.6 nun interpretiert werden.

Insgesamt sind in den Ergebnissen zwar leichte Tendenzen erkennbar, diese sind aber nicht aussagekräftig genug, um auf einen klaren Zusammenhang zwischen Netzwerkgrößen und Referenzgrößen schließen zu können. Die Punkte in den Diagrammen sind trotz ähnlichen x-Werten auf der y-Achse teils sehr stark gestreut, sodass kein eindeutiger Zusammenhang nachgewiesen werden kann.

Ein paar abweichende Punkte könnten als Ausreißer betrachtet und damit aus der Analyse herausgenommen werden, allerdings handelt es sich in den vorliegenden Daten nicht um einige wenige Punkte, die aus der Reihe fallen, sondern um eine größere Menge. Dadurch kann nicht bestimmt werden, ob doch ein Zusammenhang besteht und wenn ja, welche Punkte als Ausreißer zu werten wären.

Die meisten Werte der statistischen Analyse liegen in einem Bereich, in dem kein signifikanter Zusammenhang nachgewiesen werden konnte. Auch

ein großer Teil der Auffälligkeiten liefern zwar höhere Werte als die sonstigen, reichen aber nicht aus, um einen klaren Zusammenhang belegen zu können.

Lediglich die Werte in der Analyse der Negativen Interaktion können Tendenzen aufzeigen. Allerdings zeigen sich beim Betrachten der Diagramme von den Kommunikationsmaßen und dem Positiven Affekt, dass hier die Punkte am besten durch eine Gerade nahezu parallel zur x-Achse repräsentiert werden können. Da die Referenzgröße hier eher als Konstante auftritt, kann kein Zusammenhang zu den drei Größen hergestellt werden.

Bei der Beziehung zwischen der Grad-Zentralität und den Sozialen Konflikten bei der negativen Kommunikation zeigen die Werte eine Tendenz einer negativen Korrelation, also die Tendenz, dass bei steigender Zentralität weniger Soziale Konflikte auftreten. Diese Tendenz wird auch durch das Diagramm bestätigt. Hierbei handelt es sich allerdings lediglich um eine Tendenz. Der Zusammenhang ist nicht groß genug um einen eindeutigen Zusammenhang nachweisen zu können.

Es konnte also kein Zusammenhang zwischen den Netzwerkgrößen und den Referenzgrößen festgestellt werden. Für die gestellten Forschungsfragen bedeutet dies das Folgende:

Q1: In den vorliegenden Daten konnte kein Zusammenhang zwischen den Größen des Beziehungsnetzwerks und der Stimmung innerhalb des Teams nachgewiesen werden.

Q2: In den vorliegenden Daten konnte kein Zusammenhang zwischen den Größen des Beziehungsnetzwerks und der von den Entwicklern wahrgenommenen Leistung des Teams nachgewiesen werden.

Dass in der Evaluation kein Zusammenhang nachgewiesen werden konnte, bedeutet nicht im Umkehrschluss auch, dass keinerlei Zusammenhang zwischen den einzelnen Größen existiert. Die Ergebnisse sagen lediglich aus, dass in den für die Evaluation verwendeten Daten keiner besteht. Um einen Zusammenhang endgültig ausschließen oder andernfalls nachweisen zu können, müsste das Konzept auf weitere und größere Datenmengen angewendet werden. Nur durch große Datenmengen aus verschiedenen Umkreisen ist es möglich, die im folgenden Kapitel 7.2 beschriebenen Einschränkungen dieser Analyse auszugleichen, aufzulösen und eine eindeutige Beantwortung der Forschungsfragen zu ermöglichen.

Bei manueller Betrachtung der Netzwerke sind jedoch einige Auffälligkeiten zu erkennen, die auf mögliche Konflikte hinweisen können oder wo es zumindest möglich ist, dass daraus Probleme entstehen können. Dabei haben sich die folgenden möglichen Konfliktpotentiale ergeben:

Vor allem dadurch, dass in den Netzwerken durch die Wertung genereller Kommunikation als positiven Einfluss auf die Beziehung nur wenige Kanten mit negativen Kantengewichten vorkommen, deuten diese umso mehr auf ein Konfliktpotential hin. Diese Kanten bedeuten, dass entweder sehr viel negativ oder nur wenig überhaupt und dabei eher negativ kommuniziert wird. Dies muss zwar nicht zwangsweise eine schlechte Beziehung, soziale Konflikte und damit ein Problem sein, es ist aber möglich, dass hier ein Problem entstehen könnte. Daher sollte hier im Einzelfall nachgeprüft werden, ob ein Risiko besteht oder sich das Kantengewicht anders erklären lässt.

Auch das Vorhandensein zentraler Knoten, die viel kommunizieren und mit allen anderen Entwickler gute Beziehungen haben während diese unter sich nur leicht positive oder neutrale Verbindungen haben, kann zu einem Problem werden, wenn dieser Knoten wegbricht. Auch hier muss in jedem Fall individuell nachgeforscht werden, ob die durch das Netzwerk entstehenden Bedenken berechtigt sind.

Auch größere Differenzen, im Verhältnis zu den generellen Kantengewichten des Netzwerks einzuschätzen sind, zwischen den beiden Kanten zweier Entwickler untereinander, können ein leichtes Konfliktpotential bieten. Hier ist es unter Umständen möglich, dass unterschiedliche Auffassungen einer Beziehung vorhanden sind und daraus Konflikte entstehen können. Diesen sollte in jedem Fall dann nachgegangen werden, wenn zusätzlich eine der beiden Kanten ein negatives Kantengewicht aufweist.

Auffällig ist, dass die ersten zwei dieser drei Auffälligkeiten in dem Netzwerk in Abbildung 6.4 in Kapitel 6.6 vorliegen. Bei Betrachtung der Angaben der Entwickler bezüglich sozialer Konflikte fällt bei diesem Netzwerk auf, dass diese mit Werten von 2 hoch bis 3,25 höher sind als bei einigen anderen Teams ohne diese Auffälligkeit. Hier ist also ein Zusammenhang möglich.

Insgesamt konnte kein eindeutiger Zusammenhang sondern lediglich leichte Tendenzen zwischen den gewählten Netzwerkgrößen und den Referenzdaten über Stimmung und Leistung des Teams festgestellt werden. Dies bedeutet, dass das erarbeitete Konzept für die vorliegenden Daten nicht geeignet ist, um nur mittels der Netzwerkdaten Prognosen zu Stimmung und Leistung eines Teams machen zu können. Ob dies direkt mit dem Konzept oder mit den für die Evaluation verwendeten Daten zusammenhängt, muss durch weitere Forschung ermittelt werden.

Bei der manuellen Betrachtung hingegen sind in den Netzwerken einige Auffälligkeiten vorhanden und potentielle Probleme für die Teams erkennbar. Obwohl diese durch die vorliegenden Referenzdaten nicht vollständig oder eindeutig belegt werden können und hier weitere Forschungen notwendig sind, erscheint das Konzept doch einen gewissen Aufschluss über potentielle Konfliktstellen zu geben. Diese sollten jedoch von einem Menschen analysiert

und nachgeprüft werden.

7.2 Einschränkungen

Bei der Einordnung der Ergebnisse dieser Arbeit müssen eine Reihe von Einschränkungen und *Threats to validity* berücksichtigt werden. Dabei handelt es sich um Umstände, die sowohl das Konzept oder die verwendeten Daten betreffen können und das Ergebnis der Evaluation beeinflussen.

Grundsätzlich ist die Beziehung zwischen zwei Menschen schwer zu bestimmen und vor allem nicht einfach nur durch die Interaktion zweier Menschen. Bei Linke et al. wird beschrieben, dass die Beziehung zweier Menschen auf zwei grundlegenden Elementen aufbauen [26]. Das eine Element, die Interaktionspraxis, wird in dieser Arbeit betrachtet und analysiert. Das andere Element, die mehr oder weniger bewusste Erfahrung, wird beschrieben als „[...] Auffassung und Deutung der gemeinsamen vergangenen und gegenwärtigen Interaktion durch die an ihr Beteiligten." (Linke et al. Seite 15 [26])

Dieses Element in einer Beziehung ist abhängig von der Wahrnehmung der Beteiligten und wird in dieser Arbeit nicht mit betrachtet, da er nicht eindeutig analysierbar ist. Die Ergebnisse der Beziehungsanalyse spiegeln also nur einen Teil der Beziehung wieder und sind somit mehr als Tendenz in der Beziehung zu verstehen. Diese weitere Variable in der menschlichen Beziehung stellt eine Gefahr für die Internal validity dar.

Außerdem ist eine Einschränkung der *Conclusion validity* in Form von *low reliability of measures* zu vermuten. Die Tatsache, dass für die Analyse der Beziehungen schriftliche Transkripte verwendet werden, schränkt die Ergebnisse ein. Schaub et al. betont die Bedeutung von nonverbalen Komponenten in der Kommunikation: „Welche Worte, welche Gestik, welche Tonlage der Sender gewählt hat, bestimmen in mannigfacher Weise das Gesagte mit." (Schaub et al. Seite 47 [36])

Zwar werden für die Kategorisierung mit dem act4teams Schema Videoaufzeichnungen verwendet, dennoch gehen über diese Zwischenschritte wichtige Nuancen der Kommunikation verloren, die aber einen großen Einfluss auf die Beziehung zweier Menschen haben können.

Außerdem ist die Kategorisierung mit act4teams nicht immer einfach und eindeutig. Die Unterschiede zwischen einzelnen Kategorien sind teilweise sehr gering und manche Aussagen sind nur schwer in eine Kategorie einzuordnen. Daher ist eine Kategorisierung mit act4teams auch immer ein Stück weit eine Interpretation der Aussagen und Situation seitens der Mitarbeiter,

die sie vornehmen. Durch unterschiedliche Hintergründe, Vorwissen und Charaktereigenschaften können daher Diskrepanzen zwischen der act4teams Einschätzung und dem Empfinden der Beteiligten Personen entstehen und somit Ungenauigkeiten in der analysierten Beziehung.

Auch sind eine Reihe von Einschränkungen bezüglich der External validity Aufgrund äußerer Umstände und der Eigenschaften des Datensatzes gegeben:

In dieser Arbeit wurden für die Evaluation des Konzeptes 38 Transkripte verwendet. Dies ist eine relativ kleine Datenmenge. Bei einer größeren Menge an Daten könnten zum einen unter Umständen besser ein eventueller Zusammenhang erkannt werden und zum anderen ist ein Ergebnis mit einer größeren Datenmenge als Analysegrundlage auch aussagekräftiger.

Auch die Tatsache, dass für jedes Team nur ein Transkript als Analysegrundlage vorhanden war, beschränkt die Aussagekraft der Ergebnisse. Ein einzelnes Meeting kann die Beziehung zwischen zwei Menschen nur bedingt repräsentieren, da äußere Einflüsse, wie zum Beispiel die persönliche Stimmung einzelner Teilnehmer, dafür sorgen können, dass eine Beziehung während dieses einen Meetings ganz anders wirkt, als sie ist und normalerweise erkennbar ist. Die Nutzung mehrerer Meeting-Transkripte würde hier für einen besseren Durchschnitt über die Beziehungen sorgen und zudem die Möglichkeit bieten, die Entwicklung von Beziehungen im Laufe der Zeit zu beobachten.

Der Mangel an Transkripten pro Team ist auch der Grund, weshalb der Teilbereich der Gesprächsthemen in dieser Arbeit zwar entwickelt und konzeptioniert, allerdings nicht evaluiert werden konnte.

Auch die Qualität der vorliegenden Transkripte ist teilweise nicht optimal. In manchen Transkripten fehlt die tatsächliche Transkription des Gesagten vollständig oder ist Lückenhaft. Dies macht eine Identifikation von Angesprochenen Teilnehmern in der Analyse teils unmöglich. Auch die Einteilung des Meetings in Gesprächsthemen wird durch die schlechte Qualität der Transkription sehr schwer und bei einem Fehlen nicht möglich gemacht.

Zudem kommen in den Transkripten immer wieder Ausnahmen und Formatierungsfehler im Bereich der Sprecher von Aussagen vor. Teile der Formatierungsfehler konnten manuell behoben werden, die Ausnahmen und übrig gebliebenen Auffälligkeiten mussten für eine funktionierende Analyse jedoch aus dem Transkript herausgefiltert werden. Dies beeinflusst jedoch die Analyse und die Ergebnisse, da das Transkript nun nicht mehr das vollständige Meeting repräsentiert beziehungsweise es weniger akkurat repräsentieren kann als vor den Anpassungen.

Die Tatsache, dass die einzelnen Angaben zu den Referenzdaten nicht mehr eindeutig zu den Teammitgliedern zuzuordnen sind, stellt eine Gefahr für die Construct validity dar und beschränkt die Möglichkeiten der Analyse ebenfalls sehr. Es muss immer das ganze Team betrachtet werden, obwohl die Referenzgrößen und die meisten der Netzwerkgrößen spezifisch für einzelne Teammitglieder beziehungsweise Knoten vorliegen. Durch eine individuelle Zuordnung der jeweiligen Daten für jeden einzelnen Entwickler statt eine Betrachtung der Durchschnittswerte und Abweichungen über das ganze Team könnten unter Umständen Zusammenhänge zum Vorschein kommen, die mit den vorliegenden Daten leider nicht erkennbar sind.

Als letzte Einschränkung der Ergebnisse muss der Rahmen genannt werden, aus dem die verwendeten Daten stammen. Ein studentisches Softwareprojekt im Rahmen einer Lehrveranstaltung ist nur teilweise eine realistische Repräsentation eines Projekts in der Wirtschaft. Dies kann nicht nur als Gefahr für die *Conclusion validity* Aufgrund von *random heterogeneity of respondents*, sondern auch für die *Internal validty* in Form von *Instrumentation* und *External Validity* aufgrund äußerer Umstände und *sample feature* gesehen werden.

Da die Transkripte aus dem ersten Meeting stammen, kennen sich viele der Teilnehmer noch nicht. Es sind also noch kaum Beziehungen vorhanden, die sinnvoll analysiert und interpretiert werden können. Zwar ist es immer mal wieder möglich, dass sich einige der Studenten bereits vorher aus anderen Veranstaltungen oder privat kennen, jedoch ist dies nicht der Regelfall. Ein Großteil der Gruppenmitglieder kennen sich bis zu dem Zeitpunkt des Meetings lediglich vom sehen und haben sich, wenn überhaupt, nur einmal zuvor als Gruppe getroffen. Dies liegt an der großen Anzahl an Studenten, die jährlich die Veranstaltung besuchen, der recht geringen Gruppengrößen und der Tatsache, dass zwar Präferenzen für bestimmte Projekte angegeben, die Gruppen aber nicht selbst gebildet werden können,

Auch die Tatsache, dass das Projekt im Rahmen einer Lehrveranstaltung stattfindet, es sich also um eine Prüfungssituation handelt, beeinträchtigt die Aussagekraft der Ergebnisse. Die angegebenen Referenzdaten können beispielsweise durch unterschiedliche Interpretation der Items in PANAS variieren. Auch eine schlechter werdende Stimmung durch das Meeting kann durch andere Aspekte erklärt werden, die nicht mit Beziehungen der Teammitglieder zusammenhängt. So kann beispielsweise eine schlechtere Stimmung auch dadurch entstehen, dass die Veranstaltungen oder bestimmte Aspekte davon nicht den Erwartungen des Studenten entspricht.

Kapitel 8

Zusammenfassung und Ausblick

8.1 Zusammenfassung

In dieser Arbeit wurde das Problem behandelt, die Beziehungen von Entwicklern anhand ihrer Kommunikation innerhalb von Meetings zu ermitteln. Auf diese Weise sollen soziale Konflikte, die sich unter Umständen unbemerkt entwickelt haben und bislang aus diversen Gründen nicht angesprochen wurden, allerdings die Zusammenarbeit und Stimmung im Team negativ beeinflussen, erkannt werden, damit frühzeitig entschieden werden kann, ob Gegenmaßnahmen eingeleitet werden müssen und wenn ja, welche.

Um dieses Problem zu lösen, wurden soziale Netzwerke aus den Transkripten von Meetings aus Software-Projekten konstruiert. Dabei wurden Netzwerke für verschiedene Aspekte der Kommunikation, nämlich die Häufigkeit, Unterbrechungen, positive und negative Interaktion sowie die Beteiligungen bei verschiedenen Gesprächsthemen, konstruiert und anschließend zu einem Netzwerk vereint, dessen Kantengewichte repräsentieren, wie positiv oder negativ die Beziehung zwischen zwei Entwicklern ist. Aufgrund der fehlenden Datenlage konnte der Teilbereich der Gesprächsthemen leider nicht in die Gesamtauswertung und Evaluation mit einbezogen werden.

Die Evaluation der konstruierten Netzwerke mit dem vorhanden Datensatz, also die Suche nach einem Zusammenhang zwischen den ermittelten Beziehungswerten und der Team-Stimmung und -Performance, haben leider keinen eindeutigen Zusammenhang aufgezeigt. Dies kann jedoch auf die verwendeten Daten zurückzuführen sein.

Da nur ein Meeting für jedes Team vorlag, konnte die Beziehungsanalyse auch nur auf der Grundlage eines einzelnen Treffens, das nicht länger als eine Stunde dauerte, durchgeführt werden. Dadurch, dass es sich bei den verwendeten Meetings um die jeweils ersten Meetings innerhalb des Projekts handelt, und die wenigsten der Teilnehmer zuvor bereits zusammen gearbeitet haben, kommt zusätzlich hinzu, dass nur wenige Beziehungen bereits die Zeit hatten sich zu entwickeln. Daher ist die Aussagekraft der ermittelten Beziehungswerte deutlich eingeschränkt.

Eine manuelle Betrachtung der konstruierten Netzwerke zeigt jedoch einige Auffälligkeiten, die auf mögliche Problemstellen innerhalb der Teams hindeuten können. Diese Auffälligkeiten müssen dann von Fall zu Fall beurteilt und evaluiert werden. Wie signifikant diese Auffälligkeiten in Bezug auf sozialen Konflikte und die Leistungen der Entwickler sind, muss in zukünftigen Arbeiten ermittelt werden.

Insgesamt bietet das erarbeitete Konzept neue Möglichkeiten, mögliche Probleme im Beziehungskonstrukt eines Entwicklerteams zu ermitteln, auch wenn es mit den vorliegenden Daten nicht möglich war einen klaren Zusammenhang zwischen den Netzwerkdaten und der Stimmung und Leistung des Teams herzustellen.

8.2 Ausblick

Diese Arbeit bietet einige Punkte, an denen zukünftige Arbeiten anknüpfen können. Vor allem da die Ergebnisse keinen Zusammenhang aufgezeigt haben und ein Teil der Analyse nicht evaluiert werden konnte, existiert in diesem Bereich viel Potential für weitere Nachforschungen.

Zunächst ist eine Evaluation der Beiteilung von Entwicklern in einzelnen Gesprächsabschnitten ein Aspekt, der in zukünftigen Arbeiten vorgenommen werden sollte. Wenn ein Zusammenhang zwischen der Beteiligung an einzelnen Gesprächsabschnitten, den Aufgabenbereichen und Expertisen der Entwickler und ihrer Kommunikationshäufigkeit innerhalb der Meetings nachgewiesen werden könnte, würde dies bei der Einordnung und Interpretation der Kommunikationshäufigkeit mit berücksichtigt werden und die Ergebnisse so positiv beeinflussen.

Mit Hilfe einer Analyse eines eventuellen Zusammenhangs könnte außerdem das in dieser Arbeit entwickelte Konzept für die Gesamtanalyse verbessert werden, in dem die Gesprächsabschnitte in die Gesamtanalyse

mit einbezogen werden. In welcher Art und Weise dies geschehen könnte, ist dabei stark abhängig von dem ermittelten Zusammenhang.

Für eine solche Analyse sind jedoch eine größere Menge an Daten notwendig. Idealerweise sind hierbei die Aussagen bereits während des Meetings einem Gesprächsabschnitt zugeteilt. Die Informationen der bestimmten Arbeitsbereiche der Mitarbeiter sollten ebenfalls, falls vorhanden, mit in einer Analyse betrachtet werden. Dadurch könnte die Vermutung bestätigt oder widerlegt werden, dass Entwickler in den Gesprächsabschnitten, in denen es um ihren Arbeitsbereich oder ihre Expertise geht, eher mehr sprechen als in anderen.

Auch eine erneute Durchführung der Evaluation mit anderen Daten erscheint für die Zukunft sinnvoll. Auf diese Weise könnten das in dieser Arbeit gefundene Fehlen eines Zusammenhangs bestätigt werden oder aufgezeigt werden, dass dies nur für den verwendeten Datensatz, nicht aber universal gültig ist.

Dafür ist die Verwendung eines größeren Datensatzes mit einer Reihe von Meetings pro Team sinnvoll. Es wäre beispielsweise denkbar, verschiedene Teams in der Wirtschaft für ein Jahr zu begleiten und alle Meetings während dieser Zeit zu analysieren. Auf diese Weise könnte zum Einen gewährleistet werden, dass bestimmte Verhaltensweisen oder Angaben bei Stimmung durch die Tagesform des Entwicklers anders ist als sonst. Zum Anderen könnte hier eine Entwicklung in den Beziehungen beobachtet und auch der Verlauf und die Veränderungen innerhalb der Stimmung und Kommunikation betrachtet werden.

Zusätzlich wäre bei einer erneuten Evaluation die Nutzung anderer Referenzdaten als zusätzliche Faktoren denkbar, wie beispielsweise eine persönliche Einschätzung der Beziehungen zu den Anderen eines jeden Entwicklers. In jedem Fall sollte aber darauf geachtet werden, dass die Angaben der Referenzdaten den Entwicklern in den Meetings, normalerweise anonymisiert, eindeutig zuzuordnen sind. Dies ermöglicht eine direktere und differenziertere Evaluation der Ergebnisse. Auch eine Überprüfung der Auswirkungen der in den Netzwerken gefunden Auffälligkeiten wäre so möglich.

Zu guter Letzt wäre auch eine Betrachtung anderer Konzepte für eine Gesamtauswertung lohnenswert. Die richtige Gewichtung zwischen der Qualität und der Quantität der Kommunikation sowie das Gewichtungsverhältnis zwischen positiver und negativer Interaktion hat einen großen Einfluss auf die Gesamtbewertung der einzelnen Beziehungen. Hier könnten verschiedene Konzepte miteinander verglichen werden.

Anhang A

act4teams Schema

Übersicht des act4teams Schemas unterteilt in die 4 Kompetenzbereiche

Tabelle A.1: Aspekte und Kategorien der „Professionellen Kompetenz“

Professionelle Kompetenz		
Problem-differenzierung	**Lösungs-differenzierung**	**Wissen zur Organisation**
Problem (P) (Teil-)problem benennen	**Sollentwurf(SL)** Visionen, Anforderungen beschreiben	**Organisationales Wissen (WO)** Wissen über Organisation und Abläufe
Problemerläuterung (PE) Problem beschreiben und Beispiele nennen	**Lösungsvorschlag (L)** (Teil-)Lösung benennen	
	Lösungserläuterung (LE) Lösung ausarbeiten und Details beschreiben	
Problemvernetzung	**Lösungsvernetzung**	**Wissensmanagement**
Verknüpfung bei der Problemanalyse (V) z.B. Ursachen und Folgen aufzeigen	**Problem zur Lösung (PL)** Einwände gegen Lösung	**Wissen wer (WW)** Verweis auf Spezialisten
	Verknüpfung mit Lösung (VL) z.B Vorteile einer Lösung benennen	**Frage (F)** Frage nach Meinung, Inhalt, Erfahrung

Tabelle A.2: Aspekte und Kategorien der „Methodenkompetenz“

Methodenkompetenz		
Strukturierung		**Verlieren in Details und Beispielen**
Zielorientierung (Z) auf Thema verweisen bzw. zurückführen	**Zeitmanagement (ZT)** auf Zeit verweisen	**Verlieren in Details und Beispielen (DB)** nicht zielführende Beispiele, Monologe
Klärung/ Konkretisierung (K) Beitrag auf den Punkt bringen, klären	**Aufgabenverteilung (A)** delegieren/übernehmen	
Verfahrensvorschlag (VV) vorschlagen des weiteren Vorgehens	**Visualisierung (VIS)** benutzen von Flipchart und Metaplan o.ä.	
Verfahrensfrage (VF) Frage zum weiteren Vorgehen	**Kosten-Nutzen-Abwägung (KN)** Kosten und Nutzen einer Idee gegenüberstellen	
Priorisieren (PRIO) Schwerpunkte setzen	**Zusammenfassung (ZSF)** Ergebnisse zusammenfassen	

Tabelle A.3: Aspekte und Kategorien der „Selbstkompetenz“

Selbstkompetenz		
Konstruktive Mitwirkung	**Destruktive Mitwirkung**	
Interesse an Veränderungen (IN) Interesse signalisieren	**Kein Interesse an Veränderungen (KI)** z.B. leugnen von Optimierungsmöglichkeiten	**Schuldigensuche (S)** Probleme personalisieren
Eigenverantwortung (EV) Verantwortung übernehmen	**Jammern (J)** Betonung des negativen Ist-Zustandes, Schwarzmalerei, auch Killerphasen	**Betonung autoritärer Elemente (EA)** auf Hierarchien und Zuständigkeiten verweisen
Maßnahmenplanung (MP) Aufgaben zur Umsetzung vereinbaren	**Abbruch (E)** Diskussion vorzeitig beenden (wollen)	**Phrase (AL)** Inhaltsloses Gerede, Worthülse

Tabelle A.4: Aspekte und Kategorien der „Sozialkompetenz“

Sozialkompetenz		
Kollegiale Interaktion		**Unkollegiale Interaktion**
Ermunternde Ansprache (EA) z.B. Stillere ansprechen	**Atmosphärische Auflockerung (ATM)** z.B. Späße	**Tadel/Abwertung (TD)** Abwertung von anderen, „kleine Spitzen“
Unterstützung (ZUST) Vorschlägen, Ideen etc. zustimmen	**Ich-Botschaft: Trennung von Meinung und Tatsache (IB)** eigene Meinung als solche kennzeichnen	**Unterbrechung (Unt)** Wort abschneiden
Aktives Zuhören (AZ) Interesse signaliseren („mmh“, „ja“)	**Gefühle (G)** Gefühle wie Ärger, Freude ansprechen	**Seitengespräch (Seit)** Seitengespräch beginnen oder sich darin verwickeln lassen
Ablehnung (ABL) sachlich widersprechen	**Lob (Lob)** z.B. positive Äußerungen über andere Personen	**Reputation (R)** Verweis auf Diensterfahrung, Betriebszugehörigkeit, etc
Rückmeldung (RM) z.B. signalisieren, ob etwas angekommen, neu, bekannt ist		

Anhang B

PANAS Skala

Items des Positiven Affekts (PA) und Negativen Affekts (NA) nach [2]

Nr.	Item	Dimension
1	aktiv	PA
2	bekümmert	NA
3	interessiert	PA
4	freudig erregt	PA
5	verärgert	NA
6	stark	PA
7	schuldig	NA
8	erschrocken	NA
9	feindselig	NA
10	angeregt	PA
11	stolz	PA
12	gereizt	NA
13	begeistert	PA
14	beschämt	NA
15	wach	PA
16	nervös	NA
17	entschlossen	PA
18	aufmerksam	PA
19	durcheinander	NA
20	ängstlich	NA

Anhang C

Mapping

Alle hier nicht aufgelisteten Kategorien gehören zur Gruppe o (kein Effekt).

Tabelle C.1: Zuteilung der act4teams Kategorien in Gruppen

Kategorie	**Code**	**Mapping**
Ermunternde Ansprache	EA	+
Unterstützung	ZUST	+
Aktives Zuhören	AZ	+
Ablehnung	ABL	+
Rückmeldung	RM	+
Atmoshpärische Auflockerung	ATM	a
Ich-Botschaft	IB	+
Gefühle	G	+
Lob	Lob	+
Tadel/Abwertung	TD	-
Unterbrechung	Unt	u
Seitengespräch	Seit	s
Reputation	R	-

Anhang D

Transkript

Dies ist nur ein Beispiel, wie ein Transkript aufgebaut sein kann. Es handelt sich nicht um ein in der Arbeit tatsächlich verwendetes Dokument.

Tabelle D.1: Aufbau eines möglichen Transkripts

Teil-nehmer	Code	Themen-abschnitt	Transcript
A	EA	GUI	Dann können wir ja mit der GUI starten
B	Lob	GUI	Sehr gute Idee
A	F	GUI	Wie wollen wir sie denn gestalten?
⋮	⋮	⋮	⋮
C	ABL	Datenbank	Also ich find die Datenbank so doof
A	IB	Datenbank	Ich persönlich finde sie nur noch nicht ganz ausgereift.

Anhang E

Ausgaben

Dies ist eine Beispielausgabe im html-Format. Es zeigt den Graph einer Gesamtanalyse.

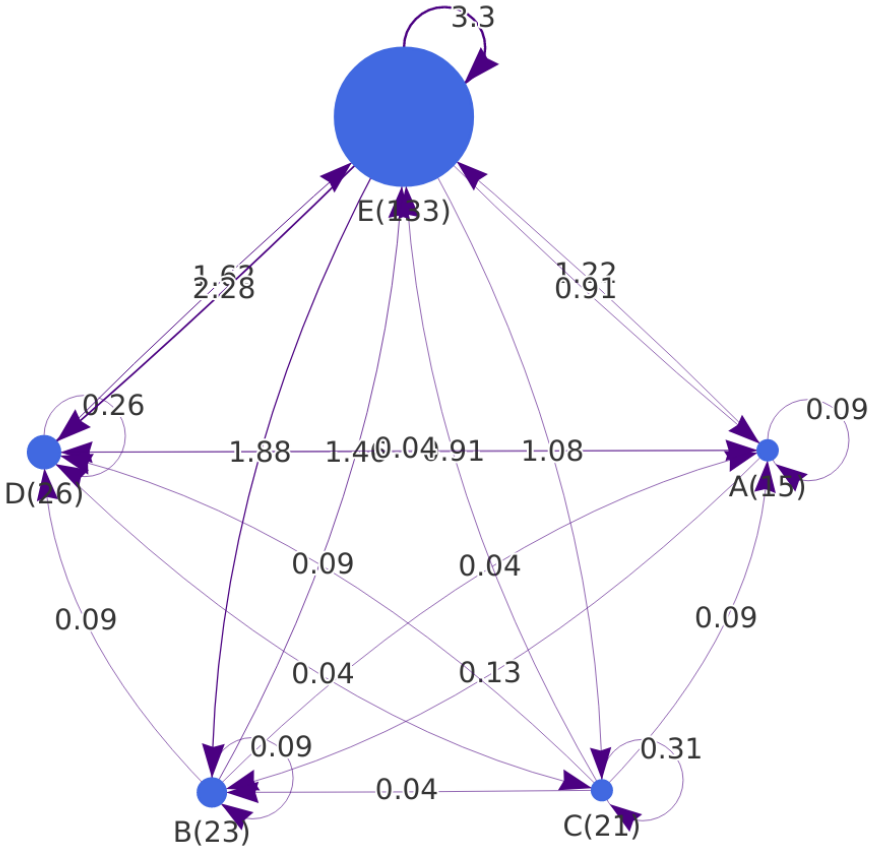

Abbildung E.1: html-Ausgabe

Dies ist eine Beispielausgabe im csv-Format. Es zeigt die Tabelle einer Analyse von Unterbrechungen. Für eine bessere Übersicht ist die Darstellung um 90% rotiert. Die Zeilen in der nachstehenden Tabelle entsprechen also den Spalten in der csv-Datei.

Tabelle E.1: csv-Ausgabe

	A	**B**	**C**	**D**	**E**
A-Count	0	3	0	3	0
B-Count	2	0	0	2	2
C-Count	0	1	0	1	0
D-Count	2	0	1	0	0
E-Count	1	1	0	0	0
A-Norm	0.0	15.79	0.0	15.79	0.0
B-Norm	15.53	0.0	0.0	10.53	10.53
C-Norm	0.0	5.26	0.0	5.26	0.0
D-Norm	10.53	0.0	5.26	0.0	0.0
E-Norm	5.26	5.26	0.0	0.0	0.0
A-Diff	0	0	0	0	0
B-Diff	-1	0	0	0	0
C-Diff	0	1	0	0	0
D-Diff	-1	-2	0	0	0
E-Diff	1	-1	0	0	0
Sum	6	6	2	3	2
Perc	1.84	2.7	1.05	1.62	3.77

Anhang F

Referenzdaten

Im folgenden sind alle Spalten des Referenzdatensatzes aufgelistet. Dabei sind nicht alle Informationen für jedes Team und jedes Teammitglied gegeben.

- Student ID
- Student Code
- Team ID
- Cohort
- Customer_1 ID
- Customer_2 ID
- Coach ID
- Gender
- Age
- Team Size
- Project Week
- Project Leader ID
- Quality Gate
- Holiday Break
- Positive Affekt (BCM)
- Positive Affect (ACM)
- Negative Affect (BCM)
- Negative Affect (ACM)
- Positive Affect (W)
- Negative Affect (W)
- Performance (Team Rated)
- Performance (Customer Rated)
- Performance (Coach Rated)
- Innovative (Team Rated)
- Innovative (Customer Rated)
- Innovative (Coach Rated)
- FLOW-Distance
- FLOW-Centrality
- FLOW-Centralization
- Social Conflicts
- Task Conflicts
- Trust in Team
- Communication Intensity
- Communication F2F

- Communication Video
- Communication Chat
- Communication Telephone
- Communication Email
- Participation Variance
- Meeting (No)
- Meeting (t)
- Meeting Attendance (No)
- Meeting Attendance (t)
- Meeting Completeness
- Code Commitment
- Requirement Compliance
- Project Success

Anhang G

Ergebnisse

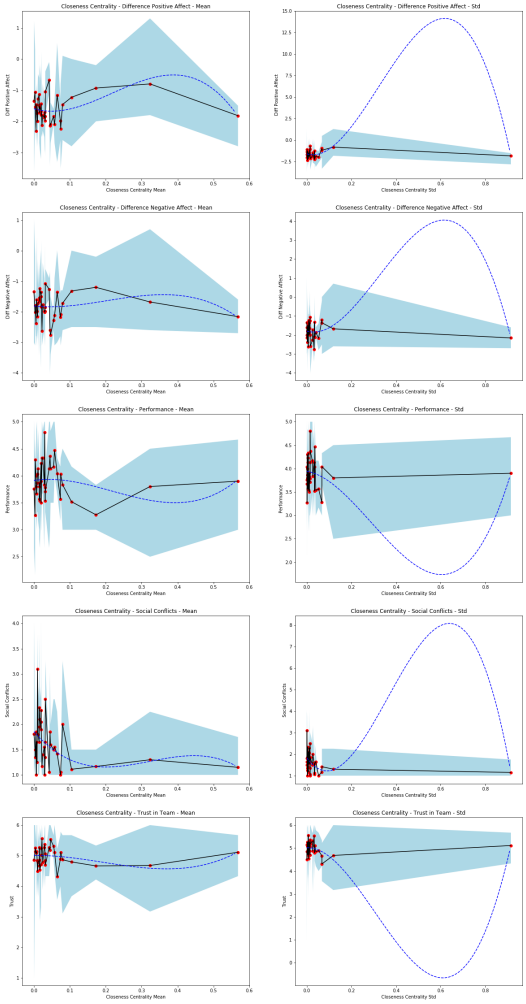

Abbildung G.1: Closeness-Zentralität - Gesamtauswertung

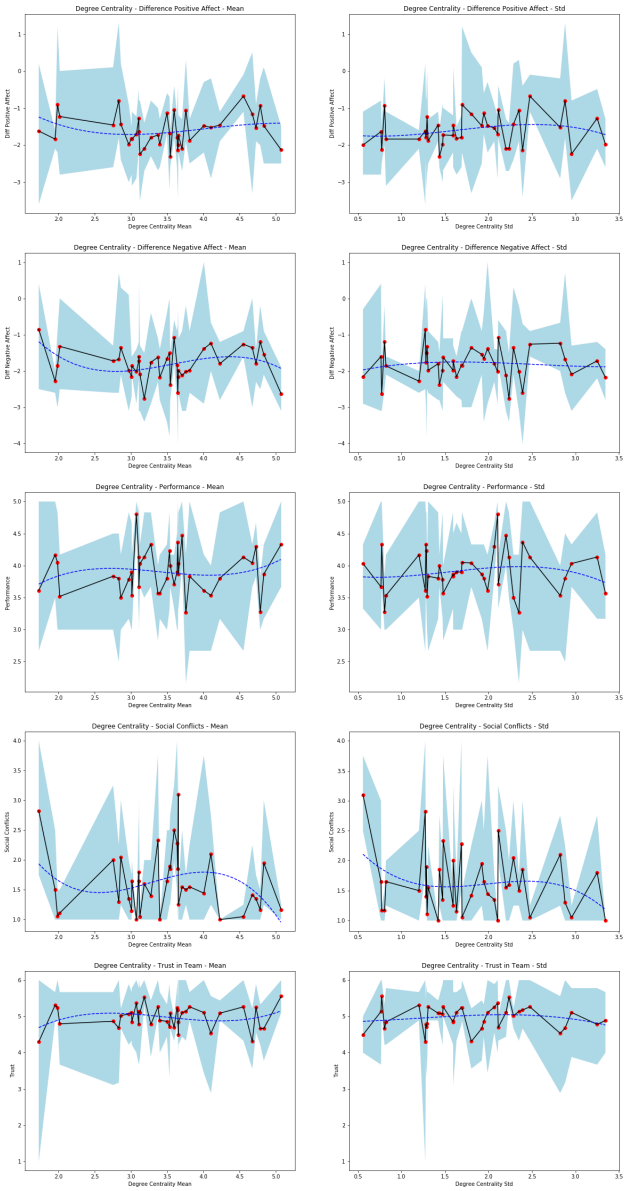

Abbildung G.2: Grad-Zentralität - Gesamtauswertung

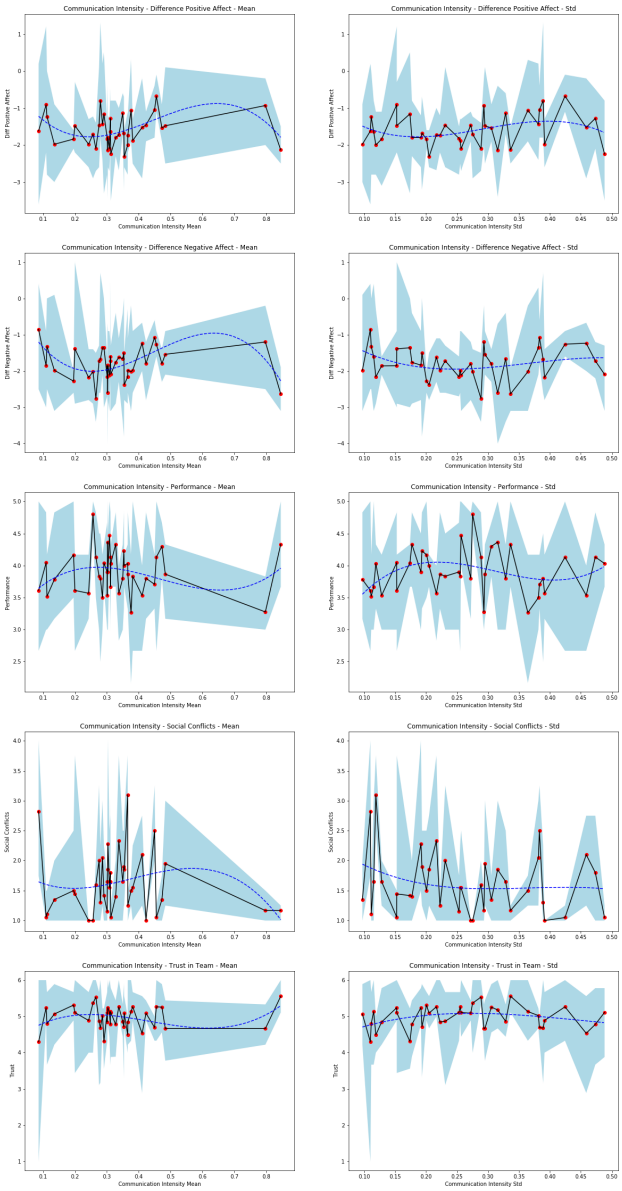

Abbildung G.3: Communication Intensity - Gesamtauswertung

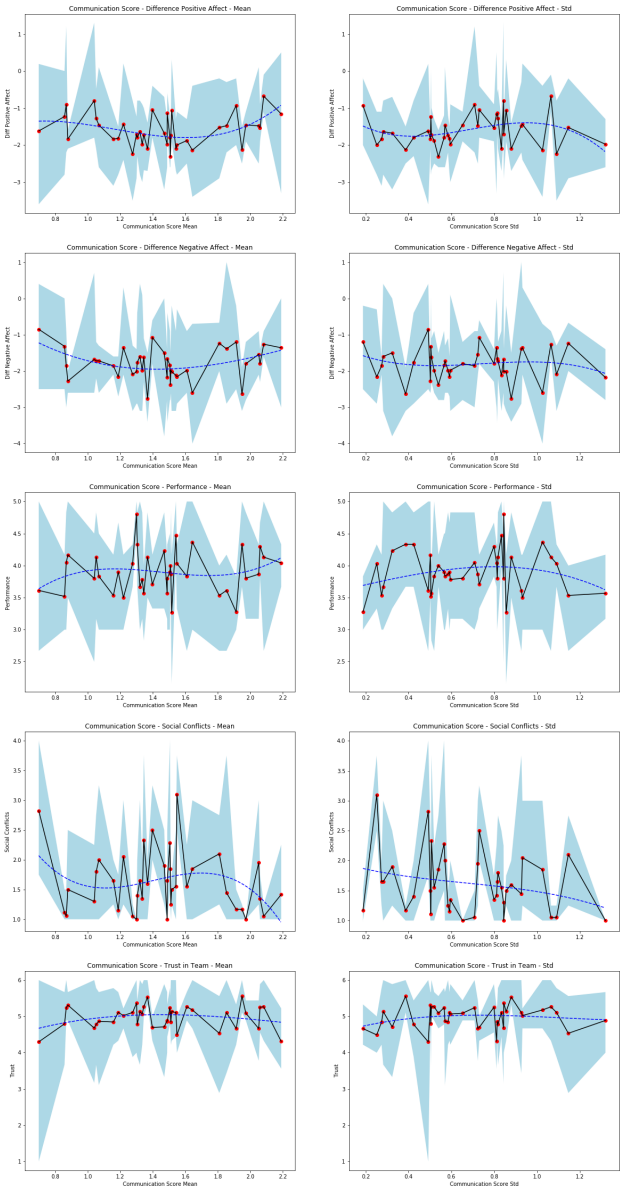

Abbildung G.4: Communication Score - Gesamtauswertung

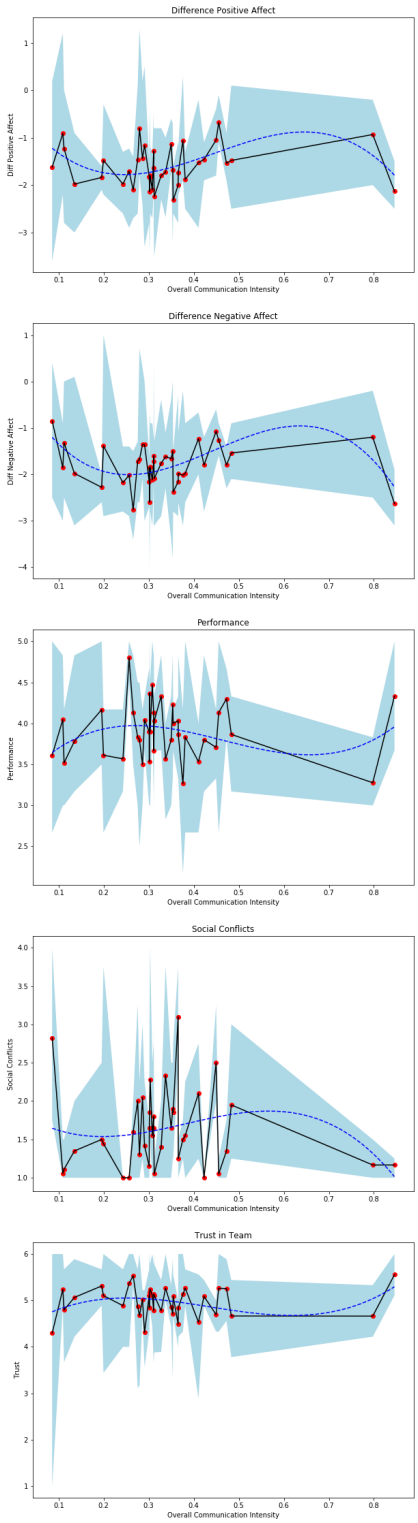

Abbildung G.5: Overall Communication Intensity - Gesamtauswertung

Anhang H

Korrelations-Koeffizienten

Tabelle H.1: Koeffizienten für die Gesamtanalyse

Referenz - größe	Grad - Zentralität	Closeness - Zentralität	Commu - nication Score	Commu - nication Intensity	Overall Commu - nication Intensity
Difference Positive Affect	$r_p = -0.02$ $r_s = -0.07$	$r_p = -0.3$ $r_s = -0.05$	$r_p = -0.03$ $r_s = -0.08$	$r_p = 0.0$ $r_s = -0.05$	$r_p = 0.0$ $r_s = -0.05$
Difference Negative Affect	$r_p = -0.14$ $r_s = -0.09$	$r_p = -0.15$ $r_s = -0.1$	$r_p = -0.14$ $r_s = -0.11$	$r_p = -0.14$ $r_s = -0.01$	$r_p = -0.14$ $r_s = -0.1$
Per - formance	$r_p = -0.06$ $r_s = -0.02$	$r_p = -0.03$ $r_s = -0.0$	$r_p = -0.06$ $r_s = -0.02$	$r_p = -0.06$ $r_s = 0.03$	$r_p = -0.06$ $r_s = 0.03$
Social Conflicts	$r_p = 0.15$ $r_s = 0.34$	$r_p = -0.29$ $r_s = -0.28$	$r_p = 0.12$ $r_s = 0.3$	$r_p = 0.14$ $r_s = 0.27$	$r_p = 0.14$ $r_s = 0.27$
Trust	$r_p = 0.11$ $r_s = 0.05$	$r_p = 0.03$ $r_s = -0.03$	$r_p = 0.12$ $r_s = 0.07$	$r_p = 0.14$ $r_s = 0.1$	$r_p = 0.14$ $r_s = 0.1$

Tabelle H.2: Koeffizienten für die Interaktionsanalyse

Referenz - größe	Grad - Zentralität	Closeness - Zentralität	Commu - nication Score	Commu - nication Intensity	Overall Commu - nication Intensity
Difference Positive Affect	$r_p = 0.06$ $r_s = -0.01$	$r_p = 0.15$ $r_s = 0.21$	$r_p = 0.05$ $r_s = -0.02$	$r_p = 0.07$ $r_s = 0.02$	$r_p = 0.07$ $r_s = 0.02$
Difference Negative Affect	$r_p = -0.12$ $r_s = -0.08$	$r_p = 0.15$ $r_s = 0.22$	$r_p = -0.12$ $r_s = -0.08$	$r_p = -0.1$ $r_s = -0.04$	$r_p = -0.1$ $r_s = -0.04$
Per - formance	$r_p = -0.07$ $r_s = -0.01$	$r_p = -0.015$ $r_s = -0.1$	$r_p = -0.07$ $r_s = -0.00$	$r_p = -0.09$ $r_s = 0.04$	$r_p = -0.09$ $r_s = 0.04$
Social Conflicts	$r_p = 0.05$ $r_s = 0.18$	$r_p = 0.0$ $r_s = -0.04$	$r_p = 0.07$ $r_s = 0.24$	$r_p = 0.06$ $r_s = 0.24$	$r_p = 0.06$ $r_s = 0.24$
Trust	$r_p = 0.11$ $r_s = 0.13$	$r_p = 0.01$ $r_s = -0.06$	$r_p = 0.12$ $r_s = 0.13$	$r_p = 0.14$ $r_s = 0.1$	$r_p = 0.14$ $r_s = 0.1$

Tabelle H.3: Koeffizienten für Unterbrechungen

Referenz - größe	Grad - Zentralität	Closeness - Zentralität	Commu - nication Score	Commu - nication Intensity	Overall Commu - nication Intensity
Difference Positive Affect	$r_p = -0.08$ $r_s = -0.25$	$r_p = 0.8$ $r_s = 0.02$	$r_p = 0.43$ $r_s = 0.14$	$r_p = 0.43$ $r_s = 0.14$	$r_p = 0.43$ $r_s = 0.14$
Difference Negative Affect	$r_p = -0.09$ $r_s = -0.23$	$r_p = -0.01$ $r_s = -0.08$	$r_p = 0.18$ $r_s = 0.05$	$r_p = 0.18$ $r_s = 0.03$	$r_p = 0.18$ $r_s = 0.03$
Per - formance	$r_p = -0.09$ $r_s = 0.06$	$r_p = -0.14$ $r_s = -0.05$	$r_p = -0.28$ $r_s = -0.06$	$r_p = -0.26$ $r_s = -0.07$	$r_p = -0.26$ $r_s = -0.07$
Social Conflicts	$r_p = 0.3$ $r_s = 0.44$	$r_p = -0.24$ $r_s = -0.18$	$r_p = 0.1$ $r_s = 0.31$	$r_p = 0.09$ $r_s = 0.17$	$r_p = 0.09$ $r_s = 0.17$
Trust	$r_p = 0.09$ $r_s = 0.03$	$r_p = 0.05$ $r_s = 0.06$	$r_p = -0.06$ $r_s = -0.1$	$r_p = -0.03$ $r_s = -0.04$	$r_p = -0.03$ $r_s = -0.04$

Tabelle H.4: Koeffizienten für Negative Interaktion

Referenz - größe	Grad - Zentralität	Closeness - Zentralität	Commu - nication Score	Commu - nication Intensity	Overall Commu - nication Intensity
Difference Positive Affect	$r_p = 0.08$ $r_s = 0.16$	$r_p = 0.22$ $r_s = 0.25$	$r_p = 0.52$ $r_s = 0.6$	$r_p = 0.56$ $r_s = 0.7$	$r_p = 0.56$ $r_s = 0.7$
Difference Negative Affect	$r_p = -0.32$ $r_s = -0.13$	$r_p = 0.23$ $r_s = 0.3$	$r_p = -0.26$ $r_s = -0.08$	$r_p = 0.19$ $r_s = 0.32$	$r_p = 0.19$ $r_s = 0.32$
Per - formance	$r_p = 0.23$ $r_s = 0.11$	$r_p = -0.18$ $r_s = -0.36$	$r_p = 0.29$ $r_s = 0.18$	$r_p = 0.23$ $r_s = -0.07$	$r_p = 0.23$ $r_s = -0.07$
Social Conflicts	$r_p = -0.63$ $r_s = -0.37$	$r_p = -0.38$ $r_s = -0.12$	$r_p = -0.51$ $r_s = -0.37$	$r_p = -0.34$ $r_s = -0.38$	$r_p = -0.34$ $r_s = -0.38$
Trust	$r_p = 0.42$ $r_s = 0.13$	$r_p = -0.41$ $r_s = -0.52$	$r_p = 0.5$ $r_s = 0.3$	$r_p = 0.35$ $r_s = 0.3$	$r_p = 0.35$ $r_s = 0.3$

Tabelle H.5: Koeffizienten für Positive Interaktion

Referenz - größe	Grad - Zentralität	Closeness - Zentralität	Commu - nication Score	Commu - nication Intensity	Overall Commu - nication Intensity
Difference Positive Affect	$r_p = 0.02$ $r_s = -0.05$	$r_p = 0.13$ $r_s = 0.2$	$r_p = 0.02$ $r_s = -0.05$	$r_p = 0.05$ $r_s = -0.03$	$r_p = 0.05$ $r_s = -0.03$
Difference Negative Affect	$r_p = -0.09$ $r_s = -0.08$	$r_p = 0.15$ $r_s = 0.22$	$r_p = -0.09$ $r_s = -0.08$	$r_p = -0.1$ $r_s = -0.08$	$r_p = -0.1$ $r_s = -0.08$
Per - formance	$r_p = -0.07$ $r_s = 0.01$	$r_p = -0.06$ $r_s = 0.01$	$r_p = -0.09$ $r_s = 0.01$	$r_p = -0.09$ $r_s = -0.01$	$r_p = -0.09$ $r_s = -0.01$
Social Conflicts	$r_p = 0.11$ $r_s = 0.24$	$r_p = -0.12$ $r_s = -0.11$	$r_p = 0.14$ $r_s = 0.29$	$r_p = 0.11$ $r_s = 0.23$	$r_p = 0.11$ $r_s = 0.23$
Trust	$r_p = 0.06$ $r_s = 0.04$	$r_p = -0.02$ $r_s = -0.06$	$r_p = 0.08$ $r_s = 0.06$	$r_p = 0.11$ $r_s = 0.08$	$r_p = 0.11$ $r_s = 0.08$

Tabelle H.6: Koeffizienten für Sprecherreihenfolge

Referenz - größe	Grad - Zentralität	Closeness - Zentralität	Commu - nication Score	Commu - nication Intensity	Overall Commu - nication Intensity
Difference Positive Affect	$r_p = -0.11$ $r_s = -0.13$	$r_p = 0.27$ $r_s = 0.18$	$r_p = -0.16$ $r_s = -0.16$	$r_p = -0.07$ $r_s = -0.1$	$r_p = -0.07$ $r_s = -0.1$
Difference Negative Affect	$r_p = -0.15$ $r_s = -0.1$	$r_p = 0.04$ $r_s = 0.08$	$r_p = -0.15$ $r_s = -0.13$	$r_p = -0.15$ $r_s = -0.12$	$r_p = -0.15$ $r_s = -0.12$
Per - formance	$r_p = -0.05$ $r_s = -0.05$	$r_p = 0.04$ $r_s = 0.02$	$r_p = -0.03$ $r_s = -0.06$	$r_p = -0.03$ $r_s = 0.02$	$r_p = 0.03$ $r_s = 0.02$
Social Conflicts	$r_p = 0.27$ $r_s = 0.42$	$r_p = -0.33$ $r_s = -0.34$	$r_p = 0.27$ $r_s = 0.37$	$r_p = 0.24$ $r_s = 0.35$	$r_p = 0.24$ $r_s = 0.35$
Trust	$r_p = 0.08$ $r_s = -0.0$	$r_p = -0.06$ $r_s = 0.02$	$r_p = 0.09$ $r_s = 0.02$	$r_p = 0.1$ $r_s = 0.04$	$r_p = 0.1$ $r_s = 0.04$

Literaturverzeichnis

[1] A. Ben-Ze ev. *The subtlety of emotions, A Bradford book*. MIT Press;, Cambridge, Mass. u.a., 2000.

[2] B. Breyer and M. Bluemke. Deutsche version der positive and negative affect schedule panas (gesis panel), 03 2016.

[3] R. S. Burt, M. Kilduff, and S. Tasselli. Social network analysis: Foundations and frontiers on advantage. *Annual Review of Psychology*, 64(1):527–547, 2013. PMID: 23282056.

[4] M. Cataldo and J. D. Herbsleb. Communication networks in geographically distributed software development. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, CSCW '08, page 579–588, New York, NY, USA, 2008. Association for Computing Machinery.

[5] M. E. Conway. How do committees invent. *Datamation*, 14(4):28–31, 1968.

[6] K. H. Delhees. *Redekommunikation*, pages 199–232. VS Verlag für Sozialwissenschaften, Wiesbaden, 1994.

[7] T. M. Emmers-Sommer. The effect of communication quality and quantity indicators on intimacy and relational satisfaction. *Journal of Social and Personal Relationships*, 21(3):399–411, 2004.

[8] N. Gamero, V. González-Romá, and J. M. Peiró. The influence of intra-team conflict on work teams' affective climate: A longitudinal study. *Journal of Occupational and Organizational Psychology*, 81(1):47–69, 2008.

[9] N. P. Garg, S. Favre, H. Salamin, D. Hakkani Tür, and A. Vinciarelli. Role recognition for meeting participants: An approach based on lexical information and social network analysis. In *Proceedings of the 16th ACM International Conference on Multimedia*, MM '08, page 693–696, New York, NY, USA, 2008. Association for Computing Machinery.

[10] D. Graziotin, X. Wang, and P. Abrahamsson. Do feelings matter? on the correlation of affects and the self-assessed productivity in software engineering. *Journal of Software: Evolution and Process*, 27(7):467–487, 2015.

[11] A. Hagberg, P. Swart, and D. S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[12] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. Fernández del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with NumPy. *Nature*, 585:357–362, 2020.

[13] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(3):90–95, 2007.

[14] W. H. Institute. Interactive network visualizations. `https://pyvis.readthedocs.io/en/latest/index.html`, 2018.

[15] D. Jansen. *Einführung in die Netzwerkanalyse : Grundlagen, Methoden, Forschungsbeispiele*. VS, Verl. für Sozialwiss.;, Wiesbaden, 2006.

[16] C. G. Jung. Personality types. *The portable Jung*, pages 178–272, 1971.

[17] S. Kauffeld and N. Lehmann-Willenbrock. Meetings matter: Effects of team meetings on team and organizational success. *Small Group Research*, 43(2):130–158, 2012.

[18] M. Keup. *An Meetings teilnehmen*, pages 62–82. Gabler Verlag, Wiesbaden, 2010.

[19] M. Keup. *Special: Projektmanagement international*, pages 178–200. Gabler Verlag, Wiesbaden, 2010.

[20] J. A.-C. Klünder. *Analyse der Zusammenarbeit in Softwareprojekten mittels Informationsfluessen und Interaktionen in Meetings*. PhD thesis, Gottfried Wilhelm Leibniz Universität Hannover, Hannover, Februar 2019.

[21] J. Klünder, N. Prenner, A.-K. Windmann, M. Stess, M. Nolting, F. Kortum, L. Handke, K. Schneider, and S. Kauffeld. Do you just discuss or do you solve? meeting analysis in a software project at early stages. In *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*, ICSEW'20, page 557–562, New York, NY, USA, 2020. Association for Computing Machinery.

[22] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, and C. Willing. Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides and B. Schmidt, editors, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press, 2016.

[23] P. L. Krapivsky, G. J. Rodgers, and S. Redner. Degree distributions of growing networks. *Phys. Rev. Lett.*, 86:5401–5404, Jun 2001.

[24] E. Lazega, S. Wasserman, and K. Faust. Social network analysis: methods and applications. *REVUE FRANCAISE DE SOCIOLOGIE*, 36(4):781–782, 1995.

[25] N. Lehmann-Willenbrock, R. A. Meyers, S. Kauffeld, A. Neininger, and A. Henschel. Verbal interaction sequences and group mood: Exploring the role of team planning communication. *Small Group Research*, 42(6):639–668, 2011.

[26] A. Linke and J. Schröter. *Sprache in Beziehungen – Beziehungen in Sprache Überlegungen zur Konstitution eines linguistischen Forschungsfeldes*, pages 1–32. De Gruyter, 2017.

[27] J. C. McCroskey. Willingness to communicate: The construct and its measurement., Oct 1985.

[28] W. McKinney et al. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, volume 445, pages 51–56. Austin, TX, 2010.

[29] A. Mehler, T. Sutter, and B. Frank-Job. *Die Dynamik sozialer und sprachlicher Netzwerke : Konzepte, Methoden und empirische Untersuchungen an Beispielen des WWW.* Springer Fachmedien Wiesbaden;, Wiesbaden, 2013.

[30] K. Patterson, A. Switzler, J. Grenny, and R. McMillan. *Heikle Gespräche: Worauf es ankommt, wenn viel auf dem Spiel steht.* Linde Verlag GmbH, 2012.

[31] PySimpleGUI. Pysimplegui repository. `https://github.com/PySimpleGUI`, 2018.

[32] G. Sabidussi. The centrality index of a graph, psychometrika. *Psychometrika*, 31(4):581–603, 1966.

[33] H. Salamin, S. Favre, and A. Vinciarelli. Automatic role recognition in multiparty recordings: Using social affiliation networks for feature extraction. *IEEE Transactions on Multimedia*, 11(7):1373–1380, 2009.

[34] N. C. Sauer and S. Kauffeld. The ties of meeting leaders: A social network analysis. *Psychology*, 06(04):415–434, 2015.

[35] N. C. Sauer, A. L. Meinecke, and S. Kauffeld. Networks in meetings: How do people connect. In *The Cambridge handbook of meeting science*, pages 357–380. Cambridge University Press New York, NY, 2015.

[36] M. Schaub. Gespräche und beziehungen. In *Psychologie, Soziologie und Pädagogik für die Pflegeberufe*, pages 47–59. Springer Berlin Heidelberg, 2001.

[37] K. Schneider, J. Klünder, F. Kortum, L. Handke, J. Straube, and S. Kauffeld. Positive affect through interactions in meetings: The role of proactive and supportive statements. *Journal of Systems and Software*, 143:59–70, 2018.

[38] K. Schneider, O. Liskin, H. Paulsen, and S. Kauffeld. Media, mood, and meetings: Related to project success? *ACM Trans. Comput. Educ.*, 15(4), Dec. 2015.

[39] A. Sharma. *Text Book of Correlations and Regression.* DPH mathematics series. Discovery Publishing House, 2005.

[40] C. Stegbauer. *Netzwerkanalyse und Netzwerktheorie*, volume 2. Springer, 2010.

[41] D. Surian, D. Lo, and E.-P. Lim. Mining collaboration patterns from a large developer network. In *2010 17th Working Conference on Reverse Engineering*, pages 269–273, 2010.

[42] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[43] H. Zhang, R. Dantu, and J. W. Cangussu. Socioscope: Human relationship and behavior analysis in social networks. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 41(6):1122–1143, 2011.

[44] M. Zhang. *Social Network Analysis: History, Concepts, and Research*, pages 3–21. Springer US, Boston, MA, 2010.