Universidad Internacional de la Rioja (UNIR)

ESIT

Master in Artificial Intelligence

Breakfastclub - Agent-based model simulation of a virtual classroom

Master Thesis

Presented by: Manuel Pasieka

Directed by: Michael Kickmeier-Rust, Verdu Perez Elena

City:Valencia

Date: August 7, 2019

Additional Content

Figures

Tables

Abstract

Nota: Abstract here, max 150 words.

Keywords: Agent-based model, Big Five, classroom [**?**] blass

Chapter 1

Introduction

This document is describing the Master Thesis developed by Manuel Pasieka as part of the Master in Artificial Intelligence at UNIVERSIDAD INTERNACIONAL DE LA RIOJA, S.A. 2018-2019.

As part of the Thesis the student developed an Agent-based simulation named **Breakfastclub** (available at http://github.com/mapa17/breakfastclub) of a virtual classroom in order to study the effect of different Personality Traits on happiness and attention in the simulated class. This document is describing the development of the project and the results achieved.

The document is split into the following chapters.

- This first chapter introduces the reader into the motivation behind this work and its novelties.
- The second chapter will discuss the state of the art of the methods and technologies applied.
- The third chapter lays out the initials objectives as well as their adaption and final objectives of the Thesis.
- The fourth chapter describes in detail the implementation and technical solution to the proposed problem.
- The fifth chapter is focused on the Data Analysis of the results generated.
- The sixth chapter is providing a conclusion and summary of what has been presented.

1.1 Origin and Motivation

Human beings are social animals, and although this has been acknowledged by in psychology and the social studies for a long time [NOTE: References?], it is our claim that there is a lack of adequate tools and techniques to study in detail the behavior of humans in many different group settings.
In particular we are interested in the group dynamics of children in a classroom, engaged in a autonomous study group.

We have two goals for this work

1. Develop a flexible and extendable multi agent based simulation of a virtual classroom
2. Study how different personality traits effect attention and happiness of individuals and the group as a whole

Chapter 2

State of the art

In this chapter we will provide an overview of existing simulations software, and software particularly developed for virtual classrooms. We describe the deficiencies of those solutions, and why we believe that there is a necessity to develop our own simulation.

2.1 Social Simulations and Agent-based models

Social Simulations are a special type of Agent-based models (**ABM**) that are used to study the dynamics of groups based on the individual agents that respond to their own and other agents expectations[**?**].
NOTE[Have some definition + explanation of what is an agent-based model]

Social Simulations have been used since the early 1960s to study the dynamics of social segregation[**?**] or more recently on the spread of contagious diseases[**?**] within a city.
The intention of many of those simulations is to find emerging properties of the complete system (i.e. the group), which are absent in the individual agents[**?**]. This is of particular interest if the the found properties are empirically verified, but not understood.
Simulated systems have the great benefit to be easy to manipulate and produce almost no costs to run, making them an excellent tool to study complex dynamic systems.

2.2 Agent-based model Software

A series of open source and commercial distributed Agent Software[**?**] is available. Some of the most popular ones are NetLogo[**?**], Swarm [NOTE: Reference for Swarm], or Mesa[**?**], which provide a framework to develop multi-agent simulations, often including a simple Visualization and a GUI.

We decided against against using those existing frameworks, and instead develop our own solution based on the Unity3d [Note: REF???] Game and Simulation Engine.
In particular Unity3d provides us with the following features that are absent or underdeveloped in the other frameworks.

- **State of the Art Visualization:** Unity3D is used to develop triple A computer games and provides the possibility to build simulations with realistic appearing Visuals and even virtual reality environments.
- **User Interaction:** User interaction if present at all is implemented very poorly in the other simulation frameworks. As User Interaction is essential part in every computer game, Unity3d provides an excellent support for it.
- **Integration with Machine Learning tools:** In the last year Unity3d has been extending its capabilities as a Agent based modeling framework by including a Machine Learning Agent toolkit that provides easy interface between State of the Art machine Learning Tools like Tensorflow or Pytorch and the Unity simulation.
- **Actively Maintained:** Many simulation frameworks have been academic endeavors with a short lifespan, and on multiple occasions stopped to be maintained and to be usable after a short period of time. Relying on a commercial sustained framework like Unity3d ensures availability and eases future development of the project.

Those aspects are not essential to the present solution but provide use with the possibility to extend the system as is discussed in the Outlook, in the final chapter of the Thesis.

2.3 Simulations of virtual Classrooms

Of particular interest to us are Simulation Systems that focus on a virtual classroom. Several academic and commercial systems have been developed with different objectives in mind.

Some of those solutions (e.g. TLE TeachLivE[**?**][**?**] or simSchool [**?**]) focus on teacher education, providing a virtual classroom that can be used for new teachers to learn how to interact with a class and resolve issues. Others (e.g. Katana Sim:Classroom [?]) are used as a simulation environment for academic research, focusing on psychological studies.

Evaluating the different simulations we found that all of them lacked one or more of the following features, and therefore decided to develop our own solution.

- **OpenSource:** The Simulation should be open source and freely available for academic and commercial purposes, in order to support its adoption and support the sustainability of the project.
- **Model of Agent Logic:** The agent behavior should depend on an flexible agent logic that is based on empirical psychological studies.
- **Flexibility:** The simulation should be configure able to scale class size, student profiles and classroom environment.
- **Reproducibility:** The simulation outcome (except of user interaction) should be reproducible, in order to provide a framework to study particular group dynamics. If multiple runs of the same simulation produce different results, it is unclear how alterations of the simulation configuration effect the outcome.
- **Data Analysis:** The simulation should include methods and tools to study its results. In particular it should be possible to execute multiple instances of the simulation with slightly changed configurations in order to perform a statistical analysis of the outcome.

Chapter 3

Objectives

As it is typical for projects beyond certain size, the objectives and scope had to be adapted according to the progression of the project. This chapter describes the initial envisioned objectives as well as the objectives reached with the final version developed during as part of the thesis.
One of the initial objectives of the thesis was to develop a simulation environment that could be used interactively as well as a closed loop simulation (i.e. once defined and setup would run without any user interaction until a defined state is reached).
In addition as the simulation is based on Unity3d, the Machine Learning Package would have be used to implement agents trained using a Reinforcement Learning approach.
Because of time contains the interaction and ML-Agent features have not been included in the final version developed during the Thesis, but are described in the last chapter on the outlook of the project.

The objectives for the final version based on the resources and and time available are therefor the following:

- **Closed Loop Simulation:** Implementation of Unity3d based virtual classroom simulation, including a 2D top down visualization.
- **Psychological agent model:** Development of a psychological model governing the behavior of agents that is based on empirical and theoretical grounds.
- **Seedable and deterministic:** The closed loop simulation should be deterministic and the random components should be seedable, making it possible to reproduce results of previous simulations if the same seed is provided.

- **Simulation and Classroom configuration:** The simulation as well as well as the psychological profile of the classroom should be easily configurable and alterable without the need to modify the simulation software.
- **Agent and Classroom based analysis:** As part of the data analysis it should be possible to analyze the behavior of individual agents (i.e students) as well as the average of the complete classroom (i.e. group).
- **Comparison of pre-defined psychological classroom profiles:** Based on empirical pedagogical studies a defined set of psychological interesting classroom profiles are compared to each other.

Chapter 4

Development

This chapter describes in detail the main components of the simulation and how they where implemented. How and why certain decision have been taken and alternative solutions. The project was developed in an iterative fashion, following with a first Prototype after the initial literature research and evaluation of alternative simulation frameworks (see **??**).

4.1 Agent-based model

As described briefly in the chapter on the State of the Art (**??**) a agent-based model is a multi agent simulation with a special focus on the interaction and resulting group dynamics of its agents. The main components of a Agent-based model are the following:

- **the environment:** The environment is a strictly defined space in which the agents can move and interact with each other as well as with other objects that are part of the environment.
- **the agents:** The agents are autonomous dynamic systems with a set of sensors and actors that interact with each other and with the environment.
- **the simulation mechanics and agent logic:** The simulation mechanics controls how agents interact with each other, how the environment changes as of actions of the agent or external factors (e.g. a simulation protocol defining a change in the environment). The agent logic governs the dynamic cycle between the agent, other agents and the environment. It defines how internal states change, and which behavior the agent should perform.

4.1.1 Unity3d

Unity3d[NOTE: REF] is a Computer Game Engine that has been used to develop not only AAA (i.e. high quality [NOTE: definition]) computer games but as well is continuously applied more and more to build simulations for commercial and academic purposes. Unity3d is distributed under various licencees, including a Free use license, which enables its use for Indie Game Developers as well to be applied in Academia without costs. For our purposes Unity3d is a generic Simulation Framework that provides the tools to create virtual 3d environments, a physics engine, a User Interface and autonomous agents.

The simulation is implemented as a Unity3 application with a single scene that is dynamically generated based on the simulation and classroom configuration.

All objects (Agents and Tables) in the classroom are Unity GameObjects that are updated in defined sequence with a constant rate of 1 Hz. The agent logic is therefor running in discrete steps, although the underlying Unity3d engine is executed continuously (as much as this is possible on a discrete computer system).

Although not completely separated, Simulation Logic is split from game content like Sprites (i.e. Images), Animations and other visual elements. The Simulation Logic is implemented as C# scripts that interface the Unity3d Framework. During the development it was taken care to separate the Unity3d specific elements from the rest of the simulation logic, in order to reduce dependence and make it possible to port the Simulation Logic to other platforms.

4.1.2 The environment: a classroom

In our case the environment is classroom that contains multiple tables for students to study individually and in small groups. In addition the environment is modeling the noise that is accumulating in the classroom resulting from the different actions performed by the agents.

Agents are moving about in the classroom as part of the various actions they perform. The Unity3d Navigation Agent Infrastructure is used to control the movement of agents, including path finding and collision control.

The noise model implemented, is accumulating the noise produced by the different actions performed by all agents in the classroom. Different Actions produce different amount of noise depending on the simulation configuration.

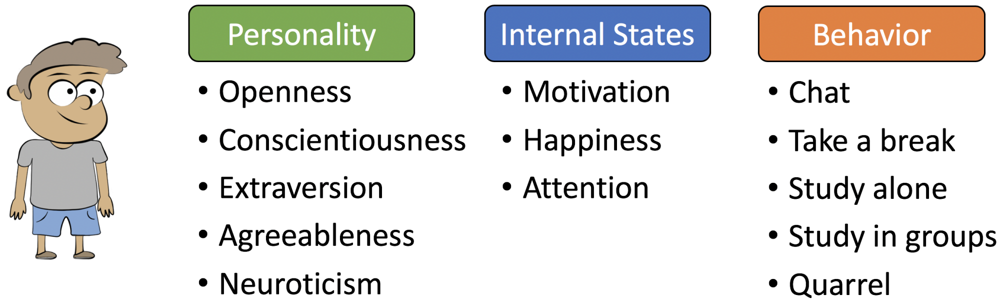

Figura 4.1: Agent Overview

4.1.3 The agents: school children

The agents are modeled to simulate school children of no specific age or physical property. Instead agents are characterized by their personality traits based on the Big Five Personality Traits model (see **??**). In addition agents have several internal states and a set of possible behaviors they can perform (see **??**).

The internal states modeled by the agent are **motivation** to study, **happiness** and **attention** during studies.

The behaviors available to the agents fall into one of three different types, being either educational, recreational or the conflict.

- **Chat:** Agents chat with another random selected agent in the classroom.
- **Take a break:** Agents take a break and start a random walk through the classroom.
- **Quarrel:** Agents start to quarrel with another random selected agent in the classroom.
- **Study alone:** Agents sit down on one of the individual tables and learn by themselves.
- **Study in groups:** Agents take a spot on a group table and study with the other agents on the table.

All possible actions, are in one of the following states at each moment in time

- **Inactive:** The action is not active at all (This is needed because of implementation details).

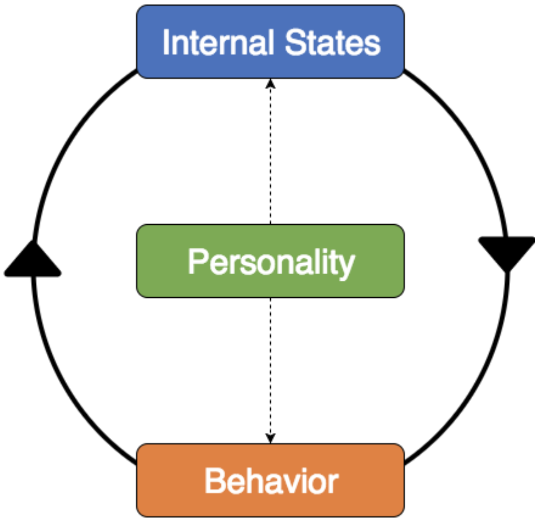

Figura 4.2: Agent Dynamics

- **Transition:** The agent is walking towards it needs to in order to perform the action.
- **Waiting:** The agents is waiting for either some response of another agent in order to perform the action.
- **Executing:** The agent is actively performing the action.

As the agents behavior depends on the internal states but as well is effecting them, each agent itself is a **dynamics systems** that is governed by the agent logic, based on the personality profile of the agent (see a visualization of this cycle in figure **??**).

Dynamic Systems

As mentioned in the introduction, agent-based models are focus on the interaction *between* components of the simulation. The complete system is therefore the result of the interaction of multiple dynamic systems (i.e. agents and environment) (see figure **??**).

This **multi level dynamic system** can express very sophisticated behavior and dynamics, making it one of the main reasons agent-based models are such a powerfully tool to study real world phenomena. One of the most curious aspects of complex dynamic systems are **emergent phenomena**[?] which describes aspects of the complete system (i.e. classroom), absent in the individual (i.e. children) components.

One examples of emerging properties is the *wetness* of Water that only appears in a ensem-

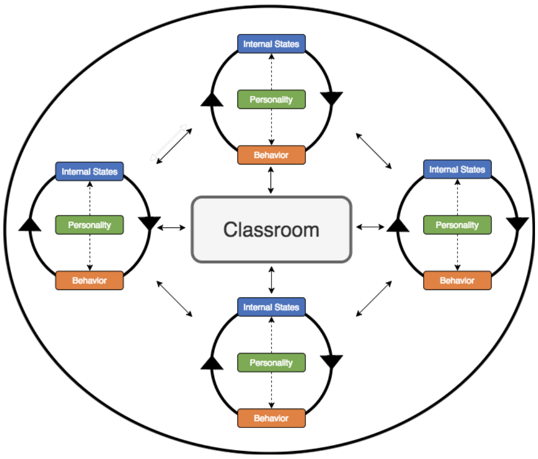

Figura 4.3: Group Dynamics

ble of many water molecules, not present in a single individual molecule. Another famous example is **Cowans Game of Life**[?] that shows the almost infinite complexity generated by a cellular automata simulating black or white cells on a infinite grid. Constructs generated by the simulation have emerging properties like *self replication*, *finite* and *infinite cycles* and many more. None of those behaviors are obviously deducible from the initial basic interaction rules. Instead those properties emerge in the interaction between the rules 'agents' (using agent in an amplified sense here) following basic rules.

Agent homogeneity

One axis along which to classify agent-based models is agent homogeneity[?]. In homogeneous agent models all agents share the same characteristic's and agent logic. Heterogeneous agent models on the other side can differ in the agents logic, its behavior or based on some parameters in its configuration.

In our case the simulation contains heterogeneous agents that differ, based on parameters in their Personality Traits.

4.2 Agent Personality Traits

As described earlier each agent is a dynamic system that in which the internal states interact with the agents behavior. Those interactions are governed by the agent logic witch is based on the **Personality Traits** of the agent.

It is not novel to use personality traits in ABM, but previous works[?] modeled very abstract personality traits that have no relation to the real world.

We therefore where careful, to chose an established and widely used personality traits model. The **OCEAN** personality trait model[?], common known as the **Big-Five**, has been developed in the 1960s and has since been used in applied and theoretical psychology. It is based on factor analysis of empirical studies (mostly self description of patients about their behavior and self image).

Its name is derived from the five orthogonal dimensions which are used to describe the personality of an individual, where the extremes of each dimension are associated with typical behaviors or thought patterns (see figure **??** for a graphical representation).

A short description of the different dimension has been taken from[?] (see table **??**).

Although the personality traits of a person could change over time, there is strong evidence ([?], [?]) that the personality traits of the big five model stay more or less constant over a long period of time or even the complete life of an individual.

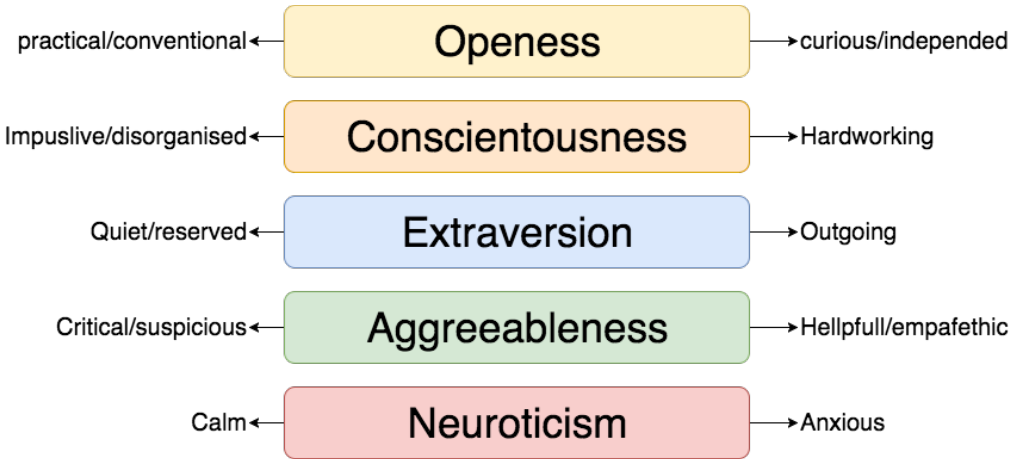

Figura 4.4: OCEAN Model

Big-five in the classroom

Various empirical studies have been performed in the past in order to investigate the association between Personality Traits, behavior and academic outcome in schools ([?], [?], [?]). We used those empirical found associations to define and tune agent logic as well as simulation parameters in order to reproduce agent behavior that is in agreement with those results.

Although it is well studies how the Big-Five behave on an individual level, we found very few studies that focused on group dynamics influenced by the Big-Five. One work

Personality Trait	Description
Openess	The general tendency to be curious about both inner and outer worlds. O includes the elements of an active imagination, aesthetic sensitivity, attentiveness to inner feelings, preference for variety, intellectual curiosity, and independence of judgment. A high O also includes individuals who are unconventional, willing to question authority, and ready to entertain new ethical and social ideas.
Conscientiousness	The general tendency to be able to resist impulses and temptations. The conscientious individual is purposeful, strong-willed, and determined. On the positive side, high C is associated with academic and occupational achievement; on the negative side, it may lead to annoying fastidiousness, compulsive neatness, or workaholic behavior, Low C's are not necessarily lacking in moral principles, but they are less exacting in applying them.
Extraversion	The general tendency to be outgoing. In addition, high E's prefer large groups and gatherings and are assertive, active, and talkative. They like stimulation and tend to be cheerful in disposition. They are upbeat, energetic, and optimistic.
Agreeableness	The general tendency to be altruistic. The high A is sympathetic to others and eager to help them, and believes that others will be equally helpful in return. By contrast, the low A is antagonistic and egocentric, skeptical of others' intentions, and competitive rather than cooperative.
Neuroticism	The general tendency to experience negative affects such as fear, sadness, embarrassment, anger, guilt, and disgust is the core of the N domain. However, N includes more than susceptibility to psychological distress. Perhaps because disruptive emotions interfere with adaptation, those who score high in N are also prone to have irrational ideas, to be less able to control their impulses, and to cope more poorly then others with stress.

Tabla 4.1: Ocean model factors taken from [?]

that we did find[?] studied how the Big-Five influence the forming of new friendships in adolescence, but limited the study to pair wise interactions.
We know of no other study besides ours that focuses on the modulation of group dynamics based on personality trait variations.

4.3 Agent Logic

The agent logic is identical for all agents, but the internal states and states related to the agent behavior is maintained separately per agent.
The Logic is implemented as a infinite loop, repeating the following steps

1. **Calculating action score**
2. **Action selection**
3. **Action execution**
4. **Handling interactions**
5. **Updating internal states**

Calculating action score

The agent can execute one of five actions (see the section about **??**). Independent of each other a score is calculated for each action. Tjat score is than used to select which action to perform. Section **??** covers the action score calculation in detail, for now it suffice to say that the score of an action depends on the the internal states of the agent and its psychological profile.

Besides the action score, the agent is calculating an **action score bias** that is added to the score of the current action and subtracted from the score of the previous action. This mechanism is used to keep the agent from switching between actions too quickly, and in case of the added bias models the tendency to continue with an ongoing task (similar to sustaining attention). The previous action score is reduced in order to keep the agent from looping between possible actions, and cause a more diverse action selection.

The action bias depends on the conscientiousness of the agent and is following an exponential decay curve, where time is the number of ticks the agent is performing the current action. The number of ticks is only counted as the action is executed, not in its other states, making sure that Transitions or Waiting do not effect the action score bias.

$$scorebias(a_i) = A * e^{-(1.0-c)*t} \tag{4.1}$$

with

- a_i is the action i
- where A is a simulation parameter defining the maximum bias
- where c is the agents conscientiousness
- where t is the number of ticks the current task is executed

Action selection

Once the actions are scored a single action is selected probabilistically. The probability for a action to be selected is defined by the square of the normalized action scores (see equation ??). Taking the squared action score makes sure that the highest rated score has a clear advantage over the other actions, but still gives other actions a chance to be selected.

$$p(a_i) = \frac{s_i^2}{\sum s_i^2} \tag{4.2}$$

with

- $p(a_i)$ is the probability of action i to be selected
- where s_i is the score for action i

Action execution

For the selected action it is tested if it can be performed, and if this is possible than the action is executed. Otherwise, the agent is taking a break, which works as the default action. In addition if it is not possible to perform an action, the agent keeps track of what its desired action is, and what the action that is actually executing.

Handling Interaction

Some of the agent behaviors like chat and quarrel depend on direct interactions between actions. Meaning that if agent A want to chat with agent B (that is randomly selected from the available agents), than Agent A depends on B *accepting* its invitation to chat. This mechanism is implemented by agent A is sending a request for chatting to agent B, and agent B decides to either accept or reject the invitation.
In case the request is accepted the agents perform the action (i.e. chat or quarrel), and if not, the sending agent A will retry either sending another request to the same agent B, or to another agent randomly selected.

The receiving agent B will interrupt any ongoing action and join the interaction. The decision if the agent B accepts the interaction os not depends on its personality traits. In case of chat the relevant personality trait is conscientiousness and in case of quarrel agreeableness. For each interaction a random number between 0.0 and 1.0 is generated. The that number is bigger than the corresponding personality trait of B, the interaction is accepted.
This mechanism makes sure to reflect the empirical findings that agents with a high level of conscientiousness are less likely to be distracted from the active task, and that high level of agreeableness is associated with less involvement in conflicts.

Updating internal states

The last step of the agent logic loop is to update the internal states of the agent.
From the free internal states motivation, attention and happiness, only two (attention and happiness) are updated at this step.

1. **Happiness:** is increased if the current action is not quarrel, and the executed action is identical to the desired action. In case the agent is executing a not desired action happiness is decreased by a factor that is scaled by the agents neuroticism. The happiness increment is a simulation parameter.

2. **Attention:** In case the agent is studying (either alone or in a group), its attention is calculated by calculating the sum of its motivation plus conscientiousness minus the noise in the classroom.

As for motivation, this internal state is altered by the action performed.

4.4 Agent Actions

As mentioned shortly before, the agents can perform one of five different actions (i.e. Chat, Quarrel, Break, Study alone and Study in groups), and each action can be in any one four states (Inactive, Transition, Waiting and Execution).
All actions follow a similar structure, and have three main functions:

- **Feasible:** Test if it is possible to execute and action. This includes to test for the availability for resources in the environment (e.g. for example if there is a free individual table for a agent to study alone) or the availability of other agents (e.g. in case of a group study if there are other agents at the table willing to study).
- **Action scoring:** Calculates a score for the action based on the internal states and the personality of the agent.
- **Action execution:** Makes the agent perform different behaviors based on the state of the action (e.g. in Transition it makes the agent walk toward its goal). In addition it modifies the agents internal states as defined by the simulation mechanics.

Before describing the exact mechanics for the different action we have a look at how to calculate the action scores.

4.4.1 Actions Scores

As mentioned before each action is scored independently, taking into account the internal states and the personality traits of the agent. The generated score is a continuos value between 0 and 1.0, where high values should be given if the actions is in correspondence with the simulation mechanics, and low values otherwise.
Although the scoring functions differ between the actions, they make use of the same basic building blocks.
The main building blocks is one exponential growing and one exponential decaying function and a scoring function calculating a wheight sum (see equation **??** and a visualization in figure **??**). The exponential functions have been defined to stay within a range of 0.0 to 1.0 for values of x between 0.0 and 1.0. In addition, the result of the scoring function is cutoff to stay within 0.0 and 1.0. The weights for each action are independent and are part of the simulation parameters.

All actions will effect happiness and motivation while being in the states waiting, transition and execution. For waiting and transition the strength with which happiness is altered depends on the agents neuroticism and agreeableness. While executing, the agents motivation and happiness are changed based on the simulation parameters.

$$E_{grw}(x) = \frac{e^{x^2} - 1}{e - 1} \tag{4.3a}$$

$$E_{dec}(x) = \frac{e^{(1-x)^2} - 1}{e - 1} \tag{4.3b}$$

$$score(\alpha, x, \beta, y, \gamma, z) = (\alpha * x) + (\beta * y) + (\gamma * z) \tag{4.3c}$$

with

- α the weight of personality
- β the weight of motivation
- γ the weight of happiness

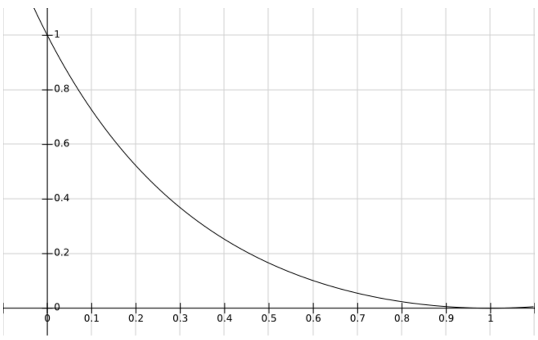

Figura 4.5: Exponential Decay

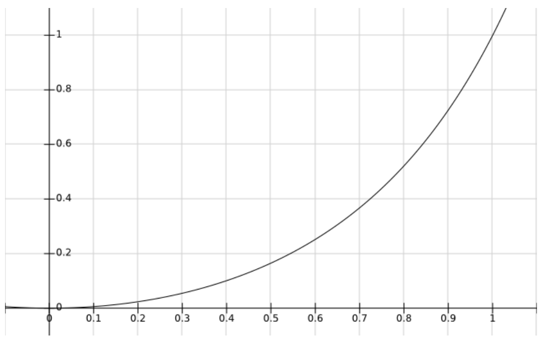

Figura 4.6: Exp Growth

Chat

The intention of this action is to recover motivation of the agent, for individuals with a high rate on extroversion. In case of chatting, the originating agent will randomly select another agent in the classroom to chat with, and approach that agent. If that other agent accepts the invitation, and both agents are next to each other, the agents start chatting. The agent will repeat a specific number of times to chat with the same agent, until giving up and choosing another agent randomly.

The action scored is calculated using the following scoring function

$$score(extroversion, w_{per}, E_{dec}(motivation), w_{mot}, E_{grw}(happiness), w_{hap}) \quad (4.4)$$

While two agents are chatting, their will produce a bit of noise adding the accumulated noise the classroom.

Takeing a break

The intention of this action is to recover motivation of the agent, for individuals with a low rate on extroversion, and therefor is an alternative to chatting. When taking a break, agents don't depend on any other agent, and as a behavior will start to walk around in the classroom randomly.
The action scored is calculated using the following scoring function

$$score(1.0 - extroversion, w_{per}, E_{dec}(motivation), w_{mot}, E_{grw}(happiness), w_{hap}) \quad (4.5)$$

Quarrel

Quarreling is the result of an agents happiness being very low but still having some motivation left. If the agent wants to quarrel it will find another agent at random, similar to Chat and starts to quarrel with that agent if accepted. If the other agent refuses, the initiating agent will repeat its request a fixed number of times before trying with another agent.
The action scored is calculated using the following scoring function

$$score(agreeableness, w_{per}, E_{grw}(motivation), w_{mot}, E_{dec}(happiness), w_{hap}) \quad (4.6)$$

Once the agents start quarreling, their motivation and happiness will fall drastically. In addition quarreling produces a lot of noise that is added to the accumulated noise in the classroom. Quarreling is special as when the agent stops quarreling, it gets an immediate boost in happiness by a value defined as part of the simulation parameters.

Studying alone

If the agent is motivated and happy, it will start to study, preferring to study alone in case of low levels of extraversion. In order to study alone the agent needs access to a free

individual table, and that the noise in the classroom is not too high.
The action scored is calculated using the following scoring function

$$score(1.0 - extroversion, w_{per}, E_{grw}(motivation), w_{mot}, E_{grw}(happiness), w_{hap}) \quad (4.7)$$

Studying in groups

As a counter part to studying alone, agents with high extraversion will start to study in case of high levels of happiness and motivation. Agents can begin to study if they have taken a seat on a group table with other agents ready to study.
The action scored is calculated using the following scoring function

$$score(extroversion, w_{per}, E_{grw}(motivation), w_{mot}, E_{grw}(happiness), w_{hap}) \quad (4.8)$$

Chapter 5

Data Analysis

In this chapter we will present how the simulation is run and how the results are analyzed.

5.1 Running the simulation

In order to run a simulation three parts are needed:

- **simulation software:** Which is a binary file of the simulation software available for Mac/Win/Linux
- **Simulation Parameters:** A JSON text file that defines Simulation Parameters
- **Classroom Profile:** A JSON text file that specifies the psychological profiles of students in the class

The simulation can be run interactively making it possible to observe the progression of the simulation, or in headless mode, where no visualization is generated. The later one is particularly useful in combination with a increased simulation speed. The combination of a headless mode with increased simulation speed can be used to run many different simulations in a **batch mode** like manner.

Independent of the way the simulation is run, it will generate a CSV file that documents the progress of the simulation. That CSV file can be opened in any arbitrary tabular data processing software (e.g Excel) for manual inspection, but is made to be analyzed by a set of python scripts developed for the purpose.
How those scripts are used and what results they generate is described in the following chapters.

5.2 Data Analysis Pipeline

The Data Analysis performed for the complete thesis is split into three parts, each having a distinct focus, answering a different set of questions.

1. **Simulation:** The goal of the simulation is to study the behavior of a particular classroom, ranging from the behavior of a single agent to the aggregated and average behavior of group as a whole.

2. **Experiment:** How much variation is there between multiple runs of the simulation for the same classroom, slightly changing the classroom profiles and random elements of the simulation?

3. **Study:** Having a expectation on how a specific classroom profile behaves, how two different profiles compare to each other, and how alterations of the personality profile effect group averages?

In the following chapters we will have a look at each step of the pipeline individually, as it is not necessary to always run the complete pipeline but based on the question one tries to answer only one or two of the first steps.

5.3 Simulation

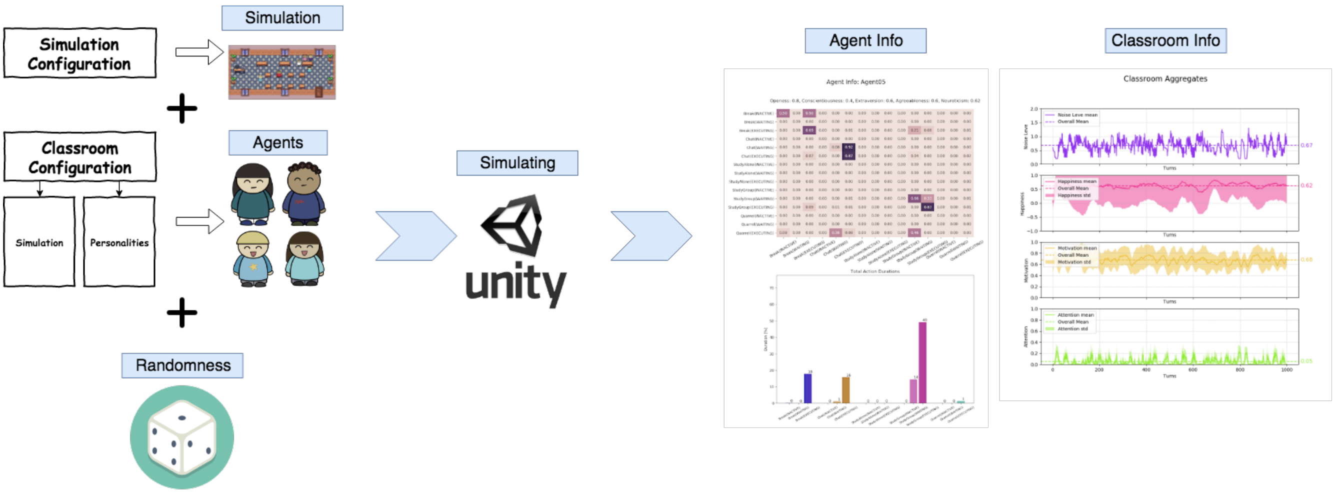

Figura 5.1: Simulation Overview

Simulation is the first step of the analysis, answering the question how a specific classroom of agents behaves.

The input to the simulation are three things. The simulation config, defining all simulation relevant constants that governing how the simulation mechanisms, the classroom configuration that defines the psychological profile of students, and a random seed that is used to initialize the random number generator used during the simulation.

Examples of the simulation config and classroom config can be found as part of the appendix (see **??** abd **??**).

The classroom configuration must contain a set of Personality Types and the number of students of each type. When the simulation is run, a classroom is dynamically generated with agents following the in the classroom configuration defined composition of profiles.

The python analysis scripts for the simulation will read the CSV file generated after the simulation is complete and generate a set of figures containing information about each individual agent, the classroom as a group and a new CSV file that contains aggregated information.

Agent Info

The analysis result specific to each agent is a **Agent Info** figure (see **??** showing three different figures) that contains information about the distribution and transitions between different behaviors performed by the agent. This figure is used to study how a specific instance of a psychological type behaves in the given classroom. One can observe how much of the overall time an Agent spends Studying alone or example and how long on average a single study session lasts.

It is interesting to observe how different the different Traits of the Big-Five effect the behavior distribution of the individual agent. So could one observe that agents high on conscientiousness for example on average have longer learning sessions that agents with a low level of conscientiousness.

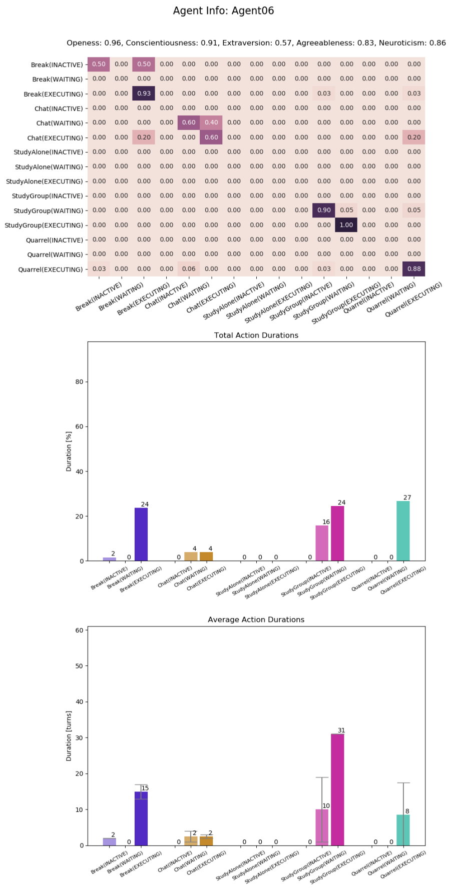

Figura 5.2: Agent Info 1

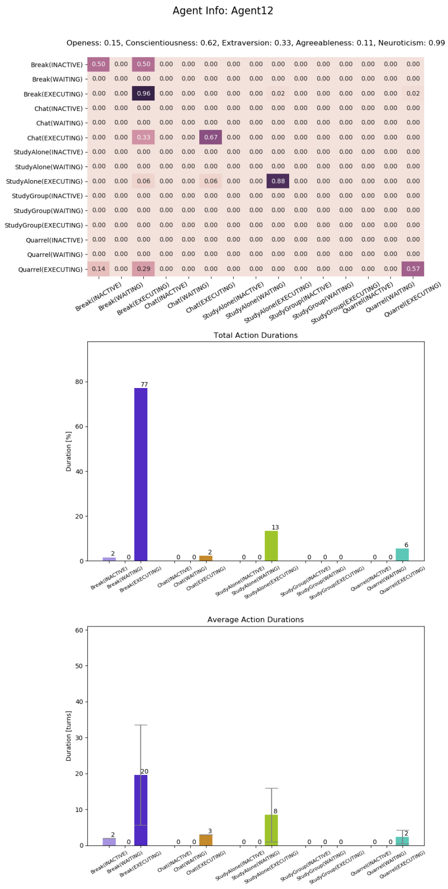

Figura 5.3: Agent Info 2

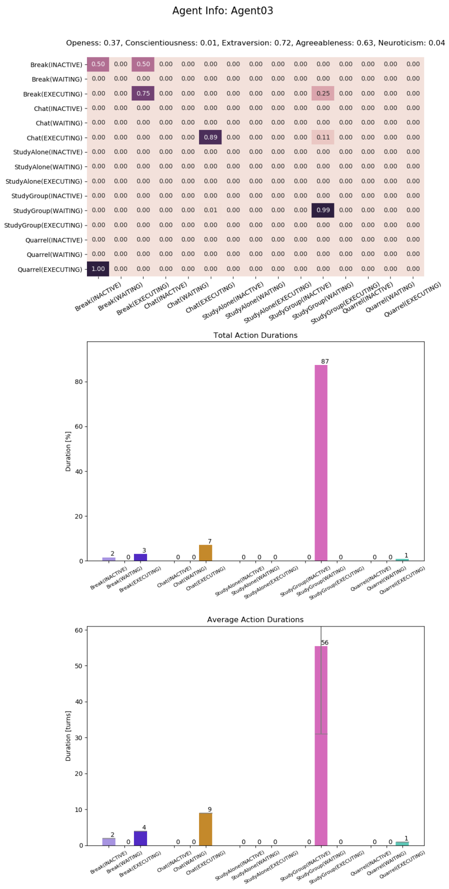

Figura 5.4: Agent Info 3

Classroom Aggregates

The second result of the simulation analysis is a figure showing classroom aggregated features over time (see figure **??** as an example). This figures contains information like the aggregated noise, the average happiness, motivation and attention of a class, in addition to information about how many of the students are studying or quarreling at a specific moment during the simulation.

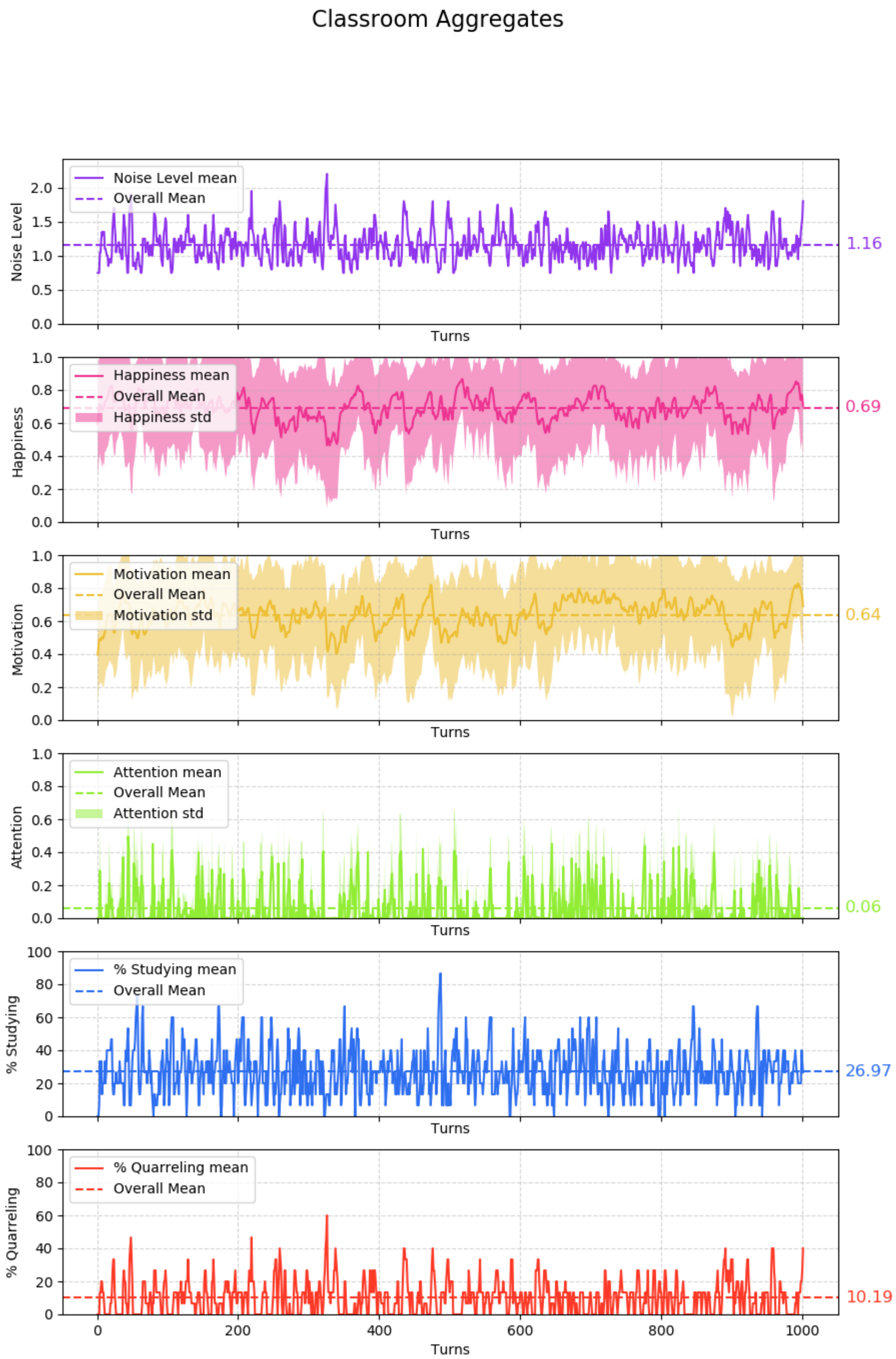

Figura 5.5: Classroom Aggregates

5.4 Experiment

The experiment is the second phase of the data analysis pipeline and is focused on evaluating the variance between simulations of the same classroom configuration.

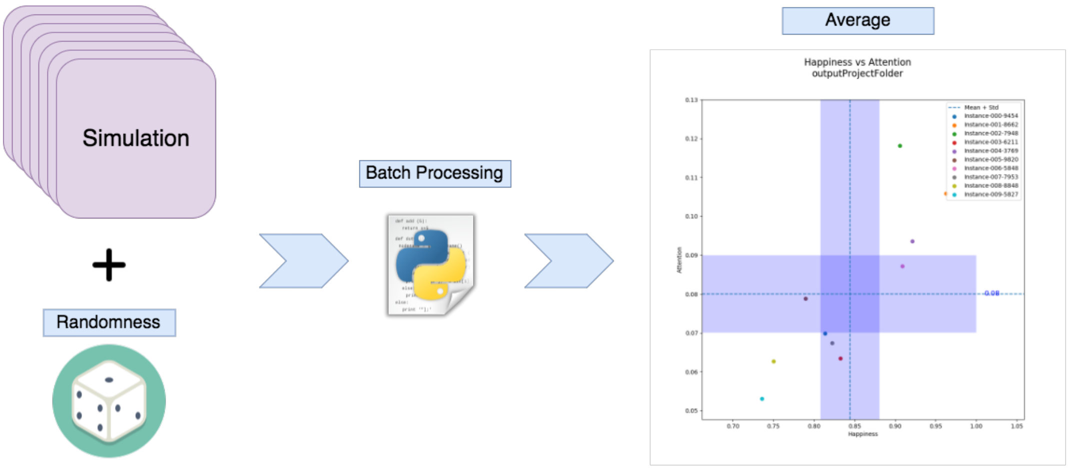

Figura 5.6: Experiment Overview

This is achieved by running multiple instances of the same simulation (identical simulation and classroom configs) but different seed values for the random number generator used during the simulation.

Depending on the classroom configuration, agent personality can be partially random, and the actions selection has a random element, and interactions are effected by the random number generator.

The results of the experiment phase is a single **HA-Plot** plot named after its axis Attention vs. Happiness. The plot shows a point for each simulation instance based on the average attention and happiness of the classroom over the complete simulation. In addition the overall happiness and attention average for all instances is indicated with two solid lines, and the corresponding standard deviation with semi transparent bars.

The plot visualizes the spread between different instances of the simulation, and could be used to detect outliers and estimate the stability of a specific classroom configuration.

In addition to the HA-Plot a CSV file is generated that contains the average happiness and attention values for each agent and classroom for all simulation instances. This dataset basically is the numeric equivalent to the generated HA-Plot.

5.5 Study

The last step in the data analysis pipeline is focused on comparing different classroom profiles.

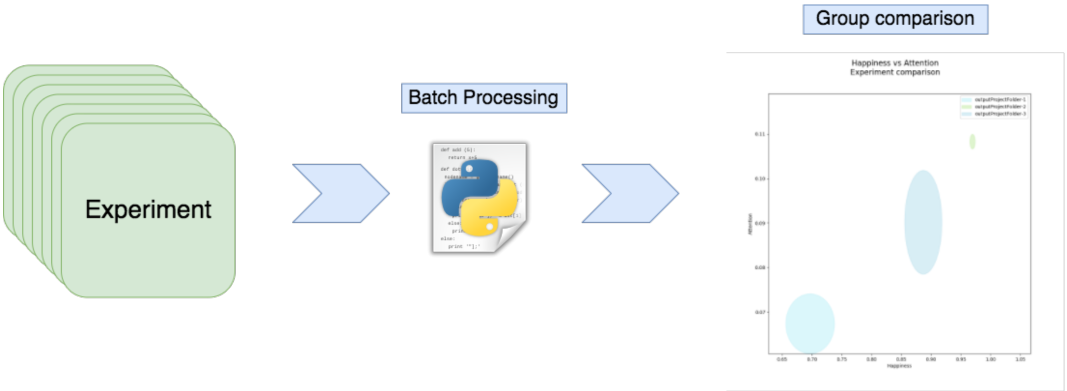

Figura 5.7: Study Overview

In the last step the CSV generated during the Experiment phase is anlayzed to generate a HA-Plot that contains one ellipse for each classroom profile simulated. The center of the ellipse are put to the overall mean attention and happiness of for the set of instances of the same classroom profile, indicating its standard deviation by the shape of the ellipse.

Chapter 6

Results, Conclusion and outlook

In the last chapter we will present the final results and draw a conclusion of the work. The chapter ends with an outlook on possible directions to continue and improve the work done so far.

6.1 Results

Here we will present hopefully the interesting results achieved

For the final analysis a set of classroom profiles have been selected

- **Class A:** Description
- **Class B:** ...
- **Class C:** ...

Comparing those classroom profile produces the following HA-Plot

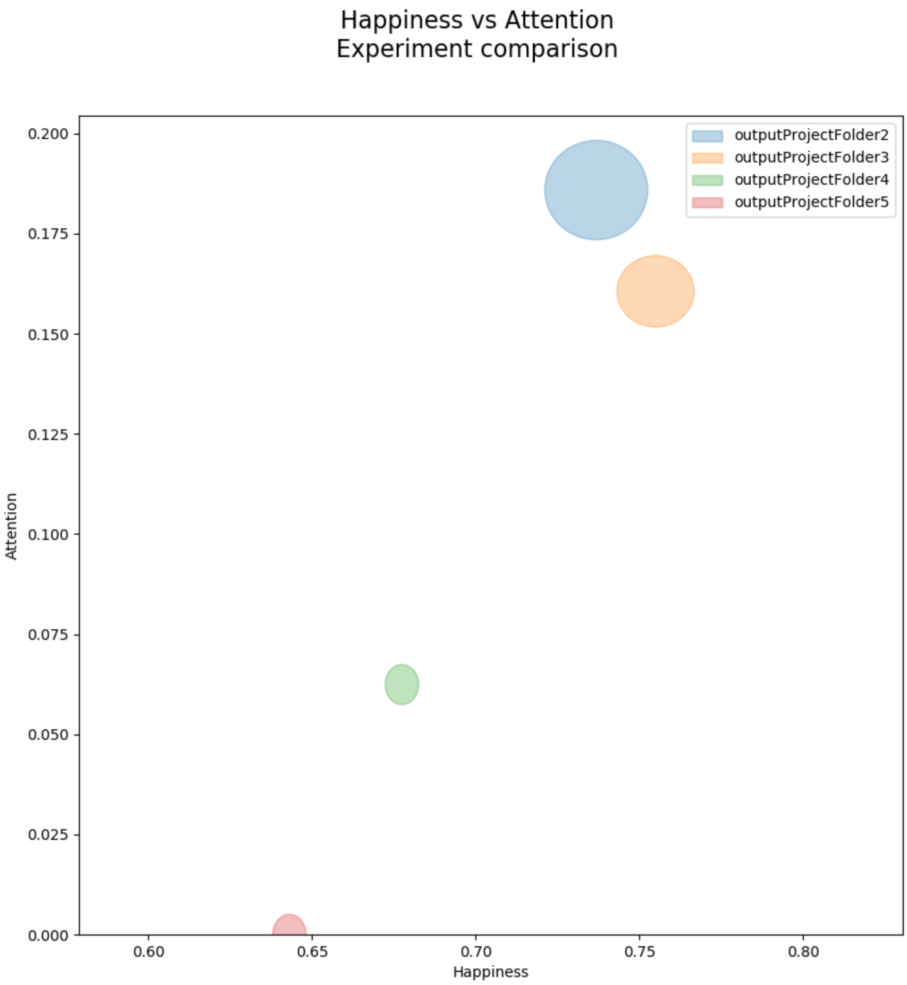

Figura 6.1: Final Study results

This result can be interpreted to be ....

Bla bla bla

. . .

6.2 Conclusion

As part of this Thesis we where able to develop a multi-agent model that is simulation a virtual classroom.

We have devised a Agent Logic that is based on the established Big-Five Personality Trait model and that produces agent behavior consistent with empirical studies.

We have developed a data analysis pipeline that makes it possible to efficiently run multiple

simulations in a batch mode, enabling a statistical analysis of the simulation results.
Based on the pedagogical literature about classroom dynamics we have selected a set of interesting classroom profiles and compared their simulation results.
Doing so we found that ...
The simulation software, the data analysis scripts and all content presented in this work is available under the MIT License from the github repository https://github.com/mapa17/breakfastclub.

6.3 Outlook

As mentioned shortly in the chapter on Objectives some initial objectives had to be dropped during the development of the thesis in order to stay within the available time frame.
The following is a list of possible improvements, as well as ideas for follow up projects.

- **Interactive Simulation:** Making the simulation interactive, by being able to force agents to perform certain actions, expulse students from the classroom, call for silence or similar interactions would make it possible to develop a teacher training program. Similar to commercial solutions like TLE TeachLive [?] or simSchool [?] but open source and with agent behavior that is based on a psychological model.

  In order to support learning and provide a novel visualization of the effect of user interaction with the simulation we envisioned a system that is generating a clone of the running simulation when ever the user is performing an action. That clone would continue in a fast forward like matter to a defined simulation end can than be compared to the all other clones generated in the same way. This would make it possible to evaluate the effect of each user interaction onto the final result of the simulation.

- **Reinforcement trained teacher:** At the moment the simulation is consisting of only students that behave like an autonomous study group. With the teacher interactions described in the previous point about an interactive simulation, one could train a virtual teacher using reinforcement learning (RL) and the objective to maximize happiness and attention of the class.

  As there is a fast amount of literature on different teaching methodologies, it would be interesting to study if the RL trained teacher applies any of the known methodologies or comes up with new ones. Another interesting aspect would be effect different classroom profiles have on the trained teacher. How different classroom profiles form and shape teacher behavior.

- **Screening of classroom profiles:** ...
- **Screening of classroom profiles:** ...

Appendix A

Appendix

A.1 Simulation Config


```
{
    "name":"BasicSimulation",
    "Classroom":
    [
        {"field":"SomeVariableName", "value": 1.0},
        {"field":"OtherSomeVariableName", "value": 0.5}
    ],
    "Agent":
    [
        {"field":"HAPPINESS_INCREASE", "value": 0.05},
        {"field":"HAPPINESS_DECREASE", "value": 0.05},
        {"field":"ACTION_SCORE_BIAS", "value": 5.00}
    ],
    "AgentBehavior":
    [
        {"field":"WAITING_HAPPINESS_INCREASE", "value": -0.04},
        {"field":"WAITING_MOTIVATION_INCREASE", "value": -0.04},

        {"field":"TRANSITION_HAPPINESS_INCREASE", "value": -0.02},
        {"field":"TRANSITION_MOTIVATION_INCREASE", "value": -0.02},

        {"field":"NEUROTICISM_WEIGHT", "value": 1.0},
        {"field":"AGREEABLENESS_WEIGHT", "value": 0.5}
    ],
    "Chat":
```

```
[
    {"field":"NOISE", "value": 0.10},

    {"field":"HAPPINESS_INCREASE", "value": 0.00},
    {"field":"MOTIVATION_INCREASE", "value": 0.05},

    {"field":"PERSONALITY_WEIGHT", "value": 0.50},
    {"field":"MOTIVATION_WEIGHT", "value": 0.25},
    {"field":"HAPPINESS_WEIGHT", "value": 0.25},

    {"field":"RETRY_THRESHOLD", "value": 3}
],
"Break":
[
    {"field":"NOISE", "value": 0.05},

    {"field":"HAPPINESS_INCREASE", "value": 0.00},
    {"field":"MOTIVATION_INCREASE", "value": 0.05},

    {"field":"PERSONALITY_WEIGHT", "value": 0.50},
    {"field":"MOTIVATION_WEIGHT", "value": 0.25},
    {"field":"HAPPINESS_WEIGHT", "value": 0.25}
],
"Quarrel":
[
    {"field":"NOISE", "value": 0.20},

    {"field":"HAPPINESS_INCREASE", "value": -0.20},
    {"field":"MOTIVATION_INCREASE", "value": -0.20},

    {"field":"PERSONALITY_WEIGHT", "value": 0.25},
    {"field":"MOTIVATION_WEIGHT", "value": 0.25},
    {"field":"HAPPINESS_WEIGHT", "value": 0.50},

    {"field":"RETRY_THRESHOLD", "value": 3},
    {"field":"HAPPINESS_BOOST", "value": 0.3}
],
"StudyGroup":
[
    {"field":"NOISE", "value": 0.05},
```

```
        {"field":"HAPPINESS_INCREASE", "value": 0.00},
        {"field":"MOTIVATION_INCREASE", "value": -0.05},
        {"field":"NOISE_THRESHOLD", "value": 4.00},

        {"field":"PERSONALITY_WEIGHT", "value": 0.50},
        {"field":"MOTIVATION_WEIGHT", "value": 0.25},
        {"field":"HAPPINESS_WEIGHT", "value": 0.25}
    ],
    "StudyAlone":
    [
        {"field":"NOISE", "value": 0.05},

        {"field":"HAPPINESS_INCREASE", "value": 0.00},
        {"field":"MOTIVATION_INCREASE", "value": -0.05},
        {"field":"NOISE_THRESHOLD", "value": 3.00},

        {"field":"PERSONALITY_WEIGHT", "value": 0.50},
        {"field":"MOTIVATION_WEIGHT", "value": 0.25},
        {"field":"HAPPINESS_WEIGHT", "value": 0.25}
    ]
}
```


A.2 Classroom Config


```
{
    "name":"Profile-3",
    "seed":42,
    "ticks":1000,
    "timescale": 100.0,
    "agent_types":
    [
        {"name":"Hustler", "openess":0.6,"conscientousness":0.8,"extraversion
            ":0.7,"agreeableness":0.2,"neuroticism":-1.0},
        {"name":"Doublechecker", "openess":0.7,"conscientousness":-1.0,"
            extraversion":-1.0,"agreeableness":-1.0,"neuroticism":0.7},
        {"name":"Politician", "openess":0.6,"conscientousness":0.5,"
            extraversion":0.8,"agreeableness":0.6,"neuroticism":0.3},
        {"name":"Mover", "openess":0.8,"conscientousness":-1.0,"extraversion"
            :0.9,"agreeableness":0.6,"neuroticism":0.5},
        {"name":"Artist", "openess":0.6,"conscientousness":0.7,"extraversion"
            :0.6,"agreeableness":-1.0,"neuroticism":0.8},
```

```
        {"name":"Engineer", "openess":0.7,"conscientousness":0.7,"
            extraversion":-1.0,"agreeableness":-1.0,"neuroticism":-1.0},
        {"name":"Normal", "openess":0.6,"conscientousness":0.5,"extraversion"
            :0.6,"agreeableness":0.4,"neuroticism":0.5}
    ],
    "nAgents":[2,2,2,2,2,2,2]
}
```

A.3 Article

Contact

cia, España

el.pasieka@protonmail.ch

b.com/mapa17

din.com/in/manuelpasieka

scount

Skills

tware Development

chine Learning

a Analysis

blem Solving

nguages

German

Python

English

I am eager to work with other machine learning experts to solve problems of automatic decision making and general artificial intelligence (e.g. learning, search and pattern recognition).

Experience & Education

Scientific Software Engineer

present - 2012

Biocomp / Vienna Biocenter Core Facilities

Developing serval data analysis applications used by neuroscience researchers to study the amygdala circuit in the mouse brain.

In particular application to automatically quantify and analyze animal behavior, and software to process and analyze neuronal activity.

- Programming in Python, Matlab, R
- Data Visualization: matplotlib, seaborn
- Data Processing: numpy, pandas
- Data Analysis: scipy, scikit-learn

Master in Artificial Intelligence

present - 2018

Universidad Internacional de La Rioja

Master Thesis: "The effect of personality traits on attention and happiness in a virtual classroom"

- Cognitive Neuroscience
- Automatic reasoning and planing
- Natural Language Processing
- Deep Learning

Research Assistent

2012 - 2011

Universidad Politécnica de Valencia

Developing a simulation environment controlled by a stationary replica of a autonomous vehicle.

- Programming in Python, C
- Developing simulation software

Master in Parallel and Distributed Computing

2012 - 2010

Universidad Politécnica de Valencia

Master Thesis: "Peer selection and Bandwidth allocation methods in BitTorrent Systems"

- Distributed Systems
- P2P Networks
- Parallel Computing
- High performance computing

Embedded System Engineer

2010 - 2007

Adaptivia GmbH

Programming of 16 bit low power SoC devices for wireless underground sensor networks.

- Embedded system engineering
- Programming in C
- System and Network design

Bachelor of Technical Informatics

2009 - 2005

Technical University Vienna

Bachelor Thesis: "Course Timetabling using Constraints satisfaction programming"

- Software Development
- Embedded system engineering
- Computer Theory

Various Internships

2007 - 2000

Database Developer
System Administrator
Infrastructure Maintances
Warehouse clerk
Project Assistant

Publications

Pliota, P., Böhm, V., Grössl, F., Griessner,J., Valenti, O., Kraitsy, K., Kaczanowska, J., **Pasieka, M.**, Lendl, T., Deussing, J. M. and Haubensak, W. (2018) 'Stress peptides sensitize fear circuitry to promote passive coping', *MolecularPsychiatry.*

Dr. Johannes Griessner , **Manuel Pasieka** , Mr. Vincent Boehm , Mr. Florian Grössl , Mrs. Joanna Kaczanowska , Dr. Pinelopi Pliota , Mr. Dominic Kargl, Ms. Barbara Werner , Dr. Nadia Kaouane , Ms. Sandra Strobelt , Dr. Silke Kreitz , Prof. Andreas Hess and Haubensak, W. (2018) 'Central amygdala circuit dynamics underlying the benzodiazepine anxiolytic effect', *MolecularPsychiatry.*