From Children’s Play to Intentions: A Play Analytics Framework for Constructionist Learning Apps

by

Mina Soltangheies

Master of Science in Interactive Arts and Technology, Simon Fraser University (2015)

Submitted to the Program in Media Arts and Sciences in partial fulfillment of the requirements for the degree of

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2017

© Massachusetts Institute of Technology 2017. All rights reserved.

Signature redacted

Author .................................................................

Program in Media Arts and Sciences

August 21, 2017

Certified by .......................................................  

Signature redacted

Deb Roy

Associate Professor

Program in Media Arts and Sciences

Thesis Supervisor

Accepted by ....................

Signature redacted

Pattie Maes

Academic Head

Program in Media Arts and Sciences
Abstract

Educational games and digital learning environments provide opportunities to collect fine-grained data on how learners engage with these technologies. The number of technologies targeted at literacy learning for children is increasing. However, the majority of them are structured and reward-based. Therefore, the users’ behavior and data collected from them have the same limits. In this thesis, however, we assess children’s engagement with a constructionist literacy learning app. The open ended nature of play in such an environment gives us the opportunity to analyze children’s play not only through what they made while playing but also how they did it. This thesis provides an analytics pipeline from data acquisition to modeling behavioral patterns. This systematic way of capturing significant events in children’s play can be used to inform stakeholders such as parents, peers and teachers and engage them with the learning process. It also gives the learning environment more intelligence on when and what to provide scaffolding on.

To collect data, we ran two pilot studies and gathered audio and video recordings of play sessions. In addition, all of the children’s interactions within the app were automatically logged. The fine-grained longitudinal data collected during the pilot studies provides a rich yet raw corpus. To reveal the patterns hidden in the data, the analytics pipeline parses logs of low-level interactions into abstract representations for sequences of actions in a word construction process. Next, it visualizes the process for each play session and the entire play history. Using the visualizations, I identified and annotated repeated motifs for more intentional sequences of actions during play and used supervised learning models to capture those patterns.

The results of this analytical pipeline are currently being used by literacy experts to provide feedback to parents and suggest activities based on the child’s process.
Acknowledgments

The past two years of my graduate studies were full of enlightening moments that pushed the boundaries of my thoughts and my self further. This journey was simply impossible without my supervisor Prof. Deb Roy. Thank you for your guidance, understanding, and patience. Thank you for trusting me with this opportunity to learn and grow.

Thank you Prof. Sepandar Kamvar and Prof. Mitchel Resnick for all the inspiring conversations, kind support and sharing your smiles and liberating thoughts.

Heather Pierce, Linda Peterson, and Keira Horowitz your persistent and honest care throughout these years gave me the strength to keep going forward.

This dissertation would not have been possible without the contribution and support of my collaborators, Ivan Sysoev, Susan Fine, gracious teachers of the Children’s Center at Northeastern University and also, comments, thoughts and continuous support from my dear friends and colleagues Dr Soroush Vosoughi, Neo Mohsenvand, and Laya Anasu.

Also, especial thanks to Ivan Sysoev for designing and implementing the wonderful SpeechBlocks and doing so with so much discipline and genuine care for children. I can see a near future when all your hard work will be paid off and SpeechBlocks will bring opportunities to those children who need it the most. Also, thank you for your great help and contribution in development of Play Trees. I will always remember the late nights we stayed together working in the lab while reading the great book of Tao of the Pooh. I also want to express my gratitude for the hard work of the team who conducted the first pilot of SpeechBlocks: Ivan Sysoev, Anneli Hershman and Susan Fine.

Many friends at the Laboratory of Social Machines thank you for being generous with your thoughts and time and helping me learn from you: Sophie Chou, James Kondo, Perng-hwa “Pau” Kung, Raphael Schaad, Luke Guolong Wang, Martin Saveski, Juliana Nazare, Ann Yuan, Nazmus Saquib, Sneha Priscilla, Nabeel Gillani, Andrew Heyward, and Allen Gorin.
Media Lab felt like a big yet close family. My friends here always inspired me in surprising ways and humbled me by their kind love. They lifted me up and kept me standing when I thought I lost my ground: my homie Eric Chu, my now brother Prashanth, The king of the twin moons Soroush Vosoughi, The Moral Eddy Awad, my girl Manisha Mohan, my long-late-night-walks companion and one of the kindest people I know Laya Anasu, Heather Pierce(<3), Russell Stevens, Eugene Yi, Preeta Bansal, William Powers, Tod Machover, Joi Ito, Seymour Papert and Marvin Minsky.

Mom, it’s because of you that I am proud. Thank you for never being less than the champion who you are.
Thesis requirement for the degree of Master of Science in media Arts and Sciences completed

by

Mina Soltangheis

The following people served as readers for this thesis:

Signature redacted

Thesis Reader ...........................  Mitchel Resnick
LEGO Papert Professor of Learning Research  MIT Media Lab
Thesis requirement for the degree of Master of Science in media Arts and Sciences completed

by

Mina Soltangheis

The following people served as readers for this thesis:

Signature redacted

Thesis Reader .......... /

Sepandar Kamvar
Associate Professor
MIT Media Lab
## Contents

1 Introduction ........................................ 17

2 Relevant Work ...................................... 21

3 SpeechBlocks ....................................... 25
   3.1 SpeechBlocks V.1 .................................. 25
   3.2 SpeechBlocks V.2 .................................. 27

4 Data Collection ..................................... 29
   4.1 Study Set-up ...................................... 29
   4.2 Participants ...................................... 30
      4.2.1 Pilot 1 ........................................ 30
      4.2.2 Pilot 2 ........................................ 31
   4.3 Complimentary Materials and Scaffoldings ............. 33
      4.3.1 Pilot 1 ........................................ 33
      4.3.2 Pilot 2 ........................................ 34
   4.4 Multi-modal Data Collection ......................... 37

5 Play Trees: a Visualization Engine .................. 39

6 Data Annotation ..................................... 47
   6.0.1 Annotation Tool .................................. 47
   6.0.2 Annotation Framework ............................ 49
   6.0.3 Stats on Annotation .............................. 52
7 Modeling

7.1 Data Preparation ................................................. 59
    7.1.1 Feature Selection ............................................ 60
7.2 Play-Specific analysis ............................................. 63
7.3 Additional Contextual Analysis ................................. 65

8 Conclusion ............................................................. 69
List of Figures

3-1 multiple layers of mapping between spoken words and letters. Phonemes are the building blocks of spoken words and can be encoded with one or a group of letters. ................................................. 26

3-2 SpeechBlocks app (a) Play area: an empty canvas that children can drag different blocks in and manipulate them. (b) Letter shelf: a shelf that can be pulled up and holds all English letter blocks. (c) Word Shelf: a shelf that holds a collection of pre-made words that can be dragged into the play area. ................................................. 27

3-3 Examples of interactions within SpeechBlocks. top: dragging a word from the shelf. bottom: splitting a word block by pulling the blocks apart using fingers ................................................. 28

3-4 New design of SpeechBlocks ................................................. 28

4-1 Study Set-up. ................................................................. 30

4-2 Gender of participants in each pilot. ................................................. 31

4-3 Age distribution of participants in each pilot. Average age for participants in pilot1 is 5.02 years and average age for participants in pilot2 is 4.85 years. ................................................. 31

4-4 Pre- and post- screenings for CTOPP-2 Sound Matching Subtest. The scores are percentile ranks for these screenings in pilot1 .......... 32

4-5 Pre- and post- screenings for CTOPP-2 Sound Matching Subtest. The scores are percentile ranks for these screenings in pilot2. .......... 32
4-6 Examples of the material provided in pilot studies. a) articulation cards b) character cards c) .............................................................. 34

4-7 Wizard of Oz experiment. a) Tapping the “question mark” at the top right corner starts scaffolding from SpeechBlocks. It provides visual cues to help the children find the appropriate letters and place them in order to make the word they want. b) The oversize letter A in the letter drawer helps bring the attention of the child to that letter. Also, an almost transparent block, here block A, appears at the correct position in the play area. Children can follow the cues or continue with their own choice of letters and words. ................................. 36

5-1 log sample ................................................................. 40

5-2 Overview of the play tree visualization pipeline. a) a sample log file, every timestamp shows the time passed since the last logged frame and deletions marked in the logs are different than deletions made by user by removing blocks from the play canvas. b) The Abstract Action Extractor (AAE) parses the fine-grained data into meaningful actions and passes it on to Play Trees Generator that builds a representation that gets visualized through layout algorithms build in the visualization engine(c) An example of a Play Tree: every node represents a letter that was brought in the play canvas (e.g. “P” and “L”). When two letters are put together, a merge happens between two branches of the tree (e.g. “P” and “L” merge into “PL”). Time progresses downward and red marks on branches represent tap or touch on the associated block. ................................. 42

5-3 Play Trees visualization of a child’s interactions in SpeechBlocks. 43
5-4 The process of making the same word in the first week and ten weeks after. As you can see the final result of the word construction has improved and there is no misspelling in the last week. Also, the process and “how” the word is made changes over time. The process is represented here by the structure of Play Trees. Similar structures emerged amongst children in sessions later in the study and when they were presented with more scaffolding on “how” to spell a word.

6-1 An overview of the annotation tool. At the top, a user can choose any play log. It automatically loads the Play Trees visualization of the selected log. Users can zoom in and out in the visualization and save the result of the annotation process using the save option.

6-2 Examples of interactions and user scenarios designed in the annotation tool. The purple areas show the sequences that are already marked as intentional. Once a user chooses the beginning of a new sequence, a horizontal line marks the beginning and the tool highlight the area that can potentially be selected as the intentional period. This especially saved time and effort when the sequences are longer.

6-3 The annotation tool provides the ability to zoom in and out in the play session. By zooming out, a user can get an overview of the context of actions in a glance and then zoom in and make judgments on each sequence. Here for example, by looking at the entire process, the user can see that the child is writing the sentence “my family goes to California”. In this context, the short constructs such as “to” and misspellings such as “myfamymy” can be interpreted as intentional words.
Example of an intentional sequence with a shift in the end product. Although the intentionality of building the word remains the same throughout this sequence, halfway through the process, the child starts a rearranging process. The tool allows us to annotate the shift in intentions within a sequence and it automatically generates and saves the related subsequences. The orange line marked the end of a subsequence and beginning of the next one.

An example of incidental intention. The end result evolves as the child plays and the rhymes inspire new words.

An example of children indicating the intention for spelling a word by collecting a character card and using it for spelling.

An example of pure exploration of letters and sounds.

Examples of misspelling I and L.

Two examples of words being misspelled. a) The word “Apple” is spelled just using the speech sounds. Therefore, the second P and the silent e at the end are not spelled. b) The word “lollipop” is spelled as spoken. These words, although not amongst the most frequently misspelled words, are spelled as they sound to the child and will be considered intentional words.

Normalized intentional play duration per session in pilot 1. Each child is represented with a unique color and the intentional play duration is normalized over the play time per session for each child.

Normalized intentional play duration per session in pilot 2. Each child is represented with a unique color and the intentional play duration is normalized over the play time per session for each child.

intentional nodes per child in pilot 1.

intentional nodes per child in pilot 2.
### List of Tables

6.1 Duration of play and annotated intentional play ....................................... 53

7.1 Actions annotated in both pilots ................................................................. 60

7.2 Play-specific Features and the result of Linear Regression analysis .......... 64

7.3 Actions annotated in both pilots ................................................................. 65

7.4 performance of different models on F2,F6,F7,F8,F3 .............................. 65

7.5 Features for modeling Play Trees ............................................................... 67

7.6 Decision Tree for all features ................................................................. 68

7.7 performance of different models on F2,F6,F7,F8,F3 .............................. 68
Chapter 1

Introduction

The literacy skills acquired in kindergarten and first grade set the development path for subsequent reading skills [8] and large scale studies have confirmed such effects on later literacy learning [39, 38]. Early childhood education and early literacy learning have gained more attention [17], motivating researchers and policy makers to develop monitoring systems that enable better assessments of these skills [35]. The majority of these systems use a variety of measures that are focused on early literacy skills such as phonemic awareness, alphabetic understanding and other components of critical literacy [37, 7]. The data collected from these assessments can be used to identify the instructional needs of a child and can also help diagnose early indications of difficulty with literacy acquisition.

However, these assessments miss the insights that can be gathered from the fine-grained longitudinal data of a learner’s process. With the creation of technologies targeted at literacy learning for children under eight years old, new possibilities for collecting data are arising. However, even the handful of apps that are designed with an eye toward how children learn at the more granular level are designed by small start-ups and larger companies that provide very little transparency about how they use the data. More importantly, the learning environments are mainly structured and reward-based [15]. Adopting a more instructionist approach to learning by these environments inherently limits and biases the learning processes we can collect data on.
Therefore, Laboratory for Social Machines has built an early literacy technology that supports child-driven learning. Additionally, this technology collects data that can be used to study the progression of a child’s literacy skills and provide updates to different stakeholders (e.g. parents) throughout the learning process. These stakeholders can engage in the child’s learning process and also have the data available to seek out informed advice from literacy experts through feedback provided from the app. At the same time, the data can be used to choose and improve the types of scaffolding necessary for a child given the history of his/her learning process. Studying the process can also help to scaffold children’s emotions when there are signs of them being in a “frustrated” or “stuck” state.

In this thesis, I assess children’s engagement with an early literacy learning technology called SpeechBlocks[36]. Based on lessons learned from deploying a technology without a culture to support it, SpeechBlocks is envisioned to be built as a piece of a larger, more encompassing system. With that same notion in mind, my thesis is also a significant and necessary step to empower the learning medium and humans involved in the system. My system also personalizes the experience of the app and allows the learning to be more social. The personalized feedback generated through looking at the process of play of a child can initiate social engagement from caregivers or parents. Those unique social interactions might not happen without feedback from the data. Therefore, although my work is looking at ways to generate more personalized views into the learning process, it also gives way to more and newer social interactions around the learning experience. The same concept holds through at more macro levels when communities of learners get connected to each other.

In this thesis, I develop an analytics pipeline with the objective of not only looking at what children made in our learning app but also how they made it. It is an end-to-end effort from designing a learning experience tailored for a classroom using a literacy learning app, to instrumenting the environment for data collection and developing analytics for play. The fine-grained longitudinal data collected during pilot studies provides a rich yet raw corpus. The goal of this thesis is to unearth the stories hidden in the log files. Moreover, it aims to create a systematic way of
capturing indications of a child’s literacy learning process and subsequently detect significant events in this process over time. I strive to find indicators and patterns that predict the intentionality of play and also attempt to capture significant words constructed within SpeechBlocks. To do so, I will focus on “what” children built while playing with SpeechBlocks and “how” they did it.

It is also important to note that the results of the machinery described here are already in use by literacy coaches to detect and report on significant events that happened during a child’s play given the play history of that child. Those updates are used to inform parents and help them engage in the learning process of their child. We will continue to study the effects of such feedback in future studies as well.

The organization of this thesis follows the analytics pipeline’s step. Each chapter explains a step of the pipeline and lays the foundation for the future chapters.

Chapter 2 provides an overview of the relevant technical background and literature review.

Chapter 3 introduces SpeechBlocks and different designs that we used throughout the studies.

Chapter 4 describes the data collection process. It starts by describing the design and development of two separate studies we ran with children. It also provides details on different forms of data collected during each pilot study.

Chapter 5 addresses the problem of play data exploration. It gives a detailed description of our proposed visualization engine that includes multiple computational units. The first unit turns the logs into meaningful sequences actions. The next unit turns those sequences into a network of actions we refer to as Play Trees. The last unit is a bundle of rendering algorithms that creates a visual representation of Play Trees. This chapter also explains some of the patterns from the data associated with intentional play.

Chapter 6 focuses on a framework to define and annotate intentional play. It also describes a visual analytics tool that I created for annotating sequences of intentional actions.

Chapter 7 provides a step by step description of modeling the process for inten-
tional play. It starts by a feature selection process and ends with two separate sets of models: the first set focuses only on features relevant to the data gathered from Play Trees. The second set adds more contextual features such as gender of participants into the models.

Chapter 8 provides a summary and remarks on the scope of this work and suggestions for the future directions.
Chapter 2

Relevant Work

Today, smart phones and touch screen tablets are widely used by young children. A survey by Rideout[22] has shown that by 2011 more than half of zero to eight years olds in America had access to touch screen devices at home. Moreover, more than 70% of the paid apps in the App Store are targeted towards young children [34].

Findings suggest that smart devices can help enhance the early literacy skills such as letter name and sound knowledge and phonological awareness[23]. These early literacy skills are crucial for developing future reading and writing ability [20, 6]. However, there are very few high quality apps that promote these literacy skills. In addition, the majority of educational apps are structured, reward-based and goal-directed [15, 16] and there are very few literacy learning applications implemented within the constructionist paradigm. In this work, I follow Papert and Harel’s [26] situating of Constructionism. Constructionist literature cited in this work refers to constructionism as the idea of building knowledge structures through constructing artifacts. In this view, learners are the builders of their own intellectual constructs and understanding happens through building artifacts. This gives learners opportunities to learn in personally meaningful and deeply engaging ways [25]. Additionally, through this method, technologies such as computers can provide means of free exploration and self expression. The personal nature of the experience in turn motivates richer understanding and increased intrinsic motivation and self-efficacy [28, 13, 40]. Constructionism as a powerful framework has given rise to large scale projects such as
Scratch[29]. The lack of literacy applications that follow a Constructionist approach motivates the early literacy learning app called SpeechBlocks [36]. SpeechBlocks is an app implemented for Android smart phones and is a game to explore spelling principles with no extrinsic goal or reward mechanism.

The development of more constructionist learning environments on one hand and more tools for data collection, sensing, and data mining technologies, has given rise to an opportunity to develop methods for exploring the unique types of data that come from these educational settings. Using these methods, we can better understand the learners and environments they learn in. This thesis is an early effort to merge the two fields with a focus on literacy learning.

Although collection and analytics of players behavior in the domain of early literacy learning similar to SpeechBlocks haven’t been explored yet, game-play analytics and play-learner analytics in serious games have a longer, richer history. Serious games became popular in 2002 through the work of Sawyer and Rajeski [31]. Sawyer defines serious games as "any meaningful use of computerized game/game industry resources whose chief mission is not entertainment" [32]. The collection and analysis of analytics incorporated into serious games provide researchers with objective data on player behavior related to serious game design elements and learning. Such analytics offer insights about play-learners’ engagement in serious games that is not possible to capture through traditional techniques [19].

The collection methods for data used for serious games analytics can be separated into two categories: in-situ and ex-situ. In-situ collection occurs in the game itself (for example, logging game-play events), whereas ex-situ is data collected outside of the game. Focus group discussions and pre- and post- test surveys are examples of ex-situ data [19].

These approaches have their own limitations. For example, the in-situ data collection can happen automatically and objectively without distracting the players. However, it might not capture the context, environment and social dynamics around the player. Pre- and post- testing as an example of ex-situ approach, is frequently used to understand the player’s behaviors[3]. However, they are less objective. More-
over, in longitudinal studies players can go for days before they interact with the game again. Therefore, it’s more difficult to control and identify the contributing factors to the differences in pre- and post- scores.

Often, serious games are intended for learning [9] and are referred to as educational and learning games. Using data analytics in the context of educational games provides opportunities for researchers to go beyond the current assessment instruments and their limitations as discussed in [2]. Play analytics help researchers to not only look at the final products that play learners made in the learning environment but also move towards more fine-grained analysis of the processes that learners went through and gain insight from observing the development of final results [4].

The majority of educational serious games are developed for training purposes and have clear goals for activities in the game. However, there are open-ended games that allow for free exploration; although there is a final goal to the game, there are numerous solutions and paths to get to the goal state [14]. Although these environments are not constructionist, the exploratory aspect of these open-ended games makes them more relevant to my domain of focus.

Examples of analyzing users’ behaviors in these open-ended games can provide insights for analysis of more open-ended environments. Amershi and Conati [1] use cluster analysis to uncover the strategies used by students who were probing a variety of common search and other AI algorithms. In another study, Pedro et al. [30] used a supervised approach to analyze student experimentation skills in a physical science simulation environment. Lynch et al. [21] study the argumentation strategies used by students in an online legal reasoning system. They use decision trees to find attributes that are predictive of students’ eventual scores on a legal reasoning test.

The process of extending such methods to constructionist data is challenging, but the work done by Blikstein [5], for example, can be seen as an early attempt in this domain. Blikstein uses data from a variety of synchronized sources including video, audio and gesture tracking to study how students use everyday materials to build simple structures such as stable towers. To analyze the data, Blikstein’s team built a coding system for annotating semantic actions and they used a range of features
such as occurrence of each code to classify the scores representing the students’ level of expertise.

Although, there are few examples of analyzing play-learners’ behaviors in open ended games, they are mostly focused on older users and not children. Working with adults makes it easier for collecting data on the intentions of player through protocols such as think aloud and interviews. In addition, these games implement clear end-goals which provide more context for analyzing user’s behavior and interpreting the intentions and goals of actions. Moreover, the previous work is not focused on literacy learning process. This thesis provides a very unique contribution to the literature by focusing on children’s learning behavior in a constructionist literacy learning environment. In addition, the current work provides a unique comprehensive data collection process which poses challenges in the data analysis process. The unified data analytics framework proposed here addresses those challenges and provides new ways to understand children’s play in constructionist, child-driven learning environments.
Chapter 3

SpeechBlocks

An important step in literacy learning is to understand the mapping between written words and spellings to their pronunciations. First, learners need to understand that the building blocks of spoken words are smaller sounds, called phonemes, and that those sounds can be represented in written forms using a letter or a group of letters, known as graphemes (Figure 3-1 [18]. Learning how speech sounds can be encoded in letters is the key to decode those letter combinations in new words and read new words[10]. Readers gradually begin to recognize more regularities in common spellings and their matching sounds. Hence, they create a more sophisticated phoneme-grapheme mapping and are able to break words into likely sounds and blend letters to chunks that are associated with speech sounds.

In this thesis, I assess children’s engagement with an early literacy learning technology called SpeechBlocks, which provides ways to explore how a) words can be broken down into its sound blocks b) sounds can blend into a word and c) sounds are encoded with visual representations(letter). Next, I briefly introduce SpeechBlocks and review the features of two versions of the app we used in our pilot studies.

3.1 SpeechBlocks V.1.

SpeechBlocks is a smart phone app that turns letters and words into manipulable talking blocks. The main function of a block is to produce a certain sound. For
example, if a child doesn’t know the letters, s/he can simply tap on any letter blocks in the app and hear the sound and gradually build the association between the visual representations (letters) and sounds. The blocks in the game environment are magnetic. Therefore, children can hear the blocks and mix the ones they like together to build meaningful or nonsense words. These magnetic blocks can also be taken apart as shown in figure 3-3 to let children explore how words can be broken down into sounds. Children can build word-blocks through simple interactions in the game including including dragging, attaching and detaching letter blocks. Once blocks are tapped, put together or pulled apart, the app provides auditory feedback by generating speech from the content of the block and playing it back to the child. This feedback is instant and allows for further tinkering and fixing as the child is making a word.

In SpeechBlocks, any combination of letters is possible and children can create nonsense words as well as real ones. There is no extrinsic reward incorporated in the app to allow child-driven and persistent explorations in the game.

In the first pilot, we used an early version of SpeechBlocks as shown in Figure 3-2. SpeechBlocks has a single page interface, and upon starting the app, there are two default words such as “BALL” and “CAT” on the screen as an invitation to explore
Figure 3-2: SpeechBlocks app (a) Play area: an empty canvas that children can drag different blocks in and manipulate them. (b) Letter shelf: a shelf that can be pulled up and holds all English letter blocks. (c) Word Shelf: a shelf that holds a collection of pre-made words that can be dragged into the play area.

The game. The play area covers almost the entire screen in the default mode of play. There are two "material shelves" that hold letter blocks and word blocks. Children can drag each shelf open and access the blocks on them. Any block can be dragged off the shelves and brought to the play area for further mixing as shown in Figure 3-3. This version of SpeechBlocks does not provide social or personalized scaffolding features. However, it provides basic scaffolding such as pre-made words on the "word shelf" in the app and also provides audio pronunciation of any word (real or invented).

3.2 SpeechBlocks V.2.

The design and implementation of SpeechBlocks has been an iterative process. During each pilot, as play sessions progressed, we observed bugs in the software and drawbacks in the design of the interface. We tried to remedy the software crashes and bugs and make the dynamics of the game easier to control. However, the overall design of the interface and basic semantics of the game remained unchanged within each pilot.
Figure 3-3: Examples of interactions within SpeechBlocks. top: dragging a word from the shelf. bottom: splitting a word block by pulling the blocks apart using fingers.

Based on the lessons learned from the first pilot, SpeechBlocks' design changed as shown in Figure 3-4 and new features were added. In the new design, there is a login page where each child can log in by clicking on their assigned icons labeled by their name. In the play sessions, I logged in every participant to avoid login mistakes.

Additionally, children can save the words they make in the word shelf in the new interface. To do so, they can simply drag the words they made to the word shelf.
Chapter 4

Data Collection

To study children’s interactions with SpeechBlocks, we have conducted three pilot studies in the Greater Boston area. The first two studies were focused on preschool children in a classroom setting. The third pilot was in collaboration with Boston826’s after school program and focused on the use of SpeechBlocks at home with the engagement of the children’s parents.

In this thesis, I only use the data gathered from the first two of these pilots because a) they are both conducted at the same children’s center with the same teachers and are held in similar classroom settings and b) the age of participants in both studies are limited to the range between four and five years old.

4.1 Study Set-up

Both pilots were conducted at a daycare in Northeastern University. Teachers at this daycare split children in groups of four and rotated them through different activities that lasted 15 minutes each. Our studies ran for two months, twice a week, as one of the activities focused on literacy learning skills. As shown in Figure 4-1, we set up a desk station where children could choose to come and play with SpeechBlocks. At least one researcher was always present at the station and would be there to introduce activities, answer questions, or help with the app. Although the activities lasted for 15 minutes, children could choose to leave whenever they desired.
4.2 Participants

The participants of the study were mainly the children of staff and faculty members of the university, which made them more likely to be exposed to literacy-rich environments. The participants were neurotypical with no speech and hearing disorders. English was their first language.

To assess the participants’ literacy skills, we conducted a variety of standardized pre-literacy assessments for pre- and post- screenings including RSI, FLUHARTY, and CTOPP. Among these evaluations, CTOPP test is the only one that provides a comprehensive assessment of the phonological processing skills that are most relevant to interactions within SpeechBlocks. Therefore, we will only use CTOPP scores for the rest of our analysis.

4.2.1 Pilot 1

We had 16 participants between the ages of four and five: twelve girls and four boys. Figure 4-2 and 4-3 show the breakdown of the participants’ gender and age for each pilot.

Figure 4-4 shows the results of the screenings for the CTOPP-2 Sound Matching
Subtest. This data is collected for all but one child who missed the pre-screening session.

Figure 4-2: Gender of participants in each pilot.

Figure 4-3: Age distribution of participants in each pilot. Average age for participants in pilot 1 is 5.02 years and average age for participants in pilot 2 is 4.85 years.

4.2.2 Pilot 2

We had 15 participants split in three groups of four plus a group of three. Participants were between the ages of four and five: ten girls and five boys. Two of the participants left the daycare mid study. Figure 4-2 and 4-3 shows the breakdown of the participants’ gender and age.

Figure 4-5 shows the results of the CTOPP-2 pre- and post- screenings. This data is collected for all but two children who missed the pre-screening session. The
results of CTOPP tests in both studies shows the increase in post-scores. We can not contribute the increase in the literacy scores with SpeechBlocks because our study was run over two months and children were exposed to other programs for literacy learning. However, the increase in the literacy skills can be used for further analysis of patterns that we might find in our analysis of play history.

In both studies, there is one child who shows a decrease in his/her score. This can be due to many factors such as the physical and emotional state of the child before
starting the test or sources of distractions in the environment. The aforementioned challenges further show the limits of current common assessments and highlights the need for more passive longitudinal assessment of children’s learning/play process.

4.3 Complimentary Materials and Scaffoldings

During both studies, we experimented with different materials and scaffoldings. To prepare and design the materials for each session, we incorporated observations from previous sessions together with feedback from the children and teachers. In both pilots, we started by introducing SpeechBlocks and letting children play freely with it and later moved towards introducing more scaffolding that will be discussed in detail in the following sections.

4.3.1 Pilot 1

We started by introducing the game and its features to the children in the first few sessions. The children were able to spend time exploring the game and playing freely with the app to spell words from their own intuitions and imaginations.

As the study progressed, we introduced a variety of prompts during the sessions to scaffold the play. In those cases, the play session starts with the researcher introducing the complementary materials and matching activity and giving the children the option to explore with those materials or continue with their free play.

The first prompt was introduced in the second session. We provided a set of themed pictures related to foods, animals, etc.

To spark the children’s ideas on what to spell, we introduced, in the next two sessions, a set of articulation cards. Pictures of objects or places or characters are paired with their spellings on these cards, as shown in Figure 4-6a.

During the fifth and sixth sessions, we provided stories with blanks in them so that children could use their own words to make their personal versions of those stories, similar to the game MadLibs.

For session nine, we designed a new set of cards we refer to as character cards.
Similar to the articulation cards, the character cards displayed a picture and its spelling. However, the character cards feature a picture of a well known cartoon character such as Olaf or Sven from Frozen. As shown in Figure 4-6b the words are spelled with blocks that look similar to SpeechBlocks.

![Figure 4-6: Examples of the material provided in pilot studies. a) articulation cards b) character cards c)](image)

We also added action cards that resembled character cards in design but represented simple verbs such as fly or run. These could easily pair with the character cards to make sentences such as: Olaf Sings or Buzz Flies.

Children couldn’t save the words they made in SpeechBlocks in this pilot, so each child was assigned a physical notebook in which they could note down the words they made in SpeechBlocks. We also turned some of the sentences they made into sentence cardsto encourage them to make more sentences and stories.

4.3.2 Pilot 2

I designed and developed the second pilot a year after the first pilot and it was build based on the lessons learned from the first pilot. In the second pilot, I used the new version of SpeechBlocks which had more features than the basic version. In the first session, I started by introducing only the basic features of the game to the children.
I did not provide the children with any instructions on what to spell, and they were free to explore the game's features and spell whatever words they wanted to. For the next two sessions, I started by reviewing the basic features and introducing some new features of the game. Children continued with their free-play in the sessions.

In the fourth session, I introduced a set of character cards similar to the first pilot. However, during play, children asked for particular characters such as Pikachu from Pokemon or words that were not in the set but inspired by the words. For example, one of the characters in the cards was Hook from Peter Pan, and a child asked how to spell the word “Captain” so he could spell “Captain Hook”. With this in mind, for the fifth session, I designed sticker boards—single pieces of construction paper personalized with their names that the children could take home. I also provided sticker versions of the letters of SpeechBlocks. I used the sticker boards mainly to spell out words not found in the provided cards that the children wanted to spell but didn’t know how to. I also added the most frequently requested words to the character cards, such as dragon and shark.

In the next session, I introduced action cards similar to the ones in the first pilot to encourage storytelling in the children’s play. In the session after, I added a new set of verbs that would match the character cards best. For example, fly for dragon.

In the last four sessions, we ran a new feature that we called “Wizard of Oz”, which refers to a condition in our study where children could ask SpeechBlocks: “How do I spell ...?” The researcher could remotely load the word that the child specifically asked to spell. Once the words were loaded in the game, child was informed by the researcher that the word is ready. Children then would proceed with tapping on the question mark as shown in Figure ??, to ask speechBlocks to help them spell the word. SpeechBlocks provides step by step visual cues to help them spell the word. In session 4, we introduced the save option in SpeechBlocks. Once children started saving the words they made, they wanted to take the phones home to show their saved words to their friends and family. As a result, I gave them booklets of the words they had saved in previous sessions at the start of the following sessions. Some of the children used the booklets as reminders of the words they had already spelled so that they
Figure 4-7: Wizard of Oz experiment. a) Tapping the “question mark” at the top right corner starts scaffolding from SpeechBlocks. It provides visual cues to help the children find the appropriate letters and place them in order to make the word they want. b) The oversize letter A in the letter drawer helps bring the attention of the child to that letter. Also, an almost transparent block, here block A, appears at the correct position in the play area. Children can follow the cues or continue with their own choice of letters and words.

could try new cards to spell. Some also took the booklets to the daycare’s designated reading area to read the words and the stories they had made.

In the last session, we gave them a personalized booklet of all the words they had made throughout the pilot.
4.4 Multi-modal Data Collection

During both studies, we conducted standardized pre-literacy assessments for pre- and post- screenings. Moreover, the play sessions were audio and video recorded using two separate cameras and researcher(s) present during the play session would take observation notes. In addition, all of the children’s interactions within the app was automatically logged and saved in text files. We will study the log files in detail in Chapter 5.
Chapter 5

Play Trees: a Visualization Engine

As discussed in a previous chapter, SpeechBlocks was designed to allow passive collection of data during children’s play. All the children’s interactions within the app are automatically recorded into text files we refer to as play logs. These logs include all the information necessary to reproduce a play session. Figure 5-1 shows an example of those logs.

Each log file is named so that it shows the date and time when the game starts on the device. All logs begin with the identity of the player who is currently logged in and the specification of the device being used, such as screen dimensions, followed by a description of the game’s environment including the size of blocks, the positioning of the shelves and the shelves’ knobs, as well as a list of all the blocks on them. The list contains a unique id for each letter or word block together with its content and initial position.

As players interact within the game, time-stamped touch points on the screen, center location, and rotation of the blocks as well as position of shelves are recorded in the logs. Additionally, logs record the time and content of auditory feedback made by SpeechBlocks during play.

Collecting data at such a fine-grained level ensures that no potential data is lost in the logging process. As a result, the collected data is not limited to a particular research question and can be used for future analysis as questions evolve. This comprehensive data collection is even more critical in the case of studies such as
Player’s and Phone’s specifications
  e.g. screen size
specifications of game’s environment
  e.g. block size, shelves’s position

Blocks on the word shelf
Blocks on the letter shelf

Touch interactions
Location change for blocks and shelves
Deletion and creation of blocks

Content of synthesized voice

Figure 5-1: log sample

this thesis that are done in a classroom setting and with groups of subjects. These studies are harder to conduct as they need significant amount of time, coordination and preparation. It is also more difficult to note for all the interactions in the group with a limited number of researchers. Moreover, the dynamic environment makes it hard to foresee and control all the variables. Therefore, with a more comprehensive approach of collecting data we can address some of these challenges better.

Data that is transcribed at this level of granularity, however, needs to be later converted into meaningful actions that a player exerts on the game environment. To be able to explore the data, the first step is to build a computation unit that I call "Abstract Action Extractor" hereafter referred to as AAE. This unit parses raw logs
into meaningful events and actions. It abstracts sequences of mere touch points and
block movements into operant actions [33] of drag, merge, split, tap, and speak. These
actions are chosen to mirror the interactions within the game.

Merge encapsulates sequences of actions where children put two or more letter
or word blocks together to create a new construct. Split encapsulates sequences of
actions where children drag apart a word block which can result into two or three
separate word blocks. Merges and splits normally appear in logs as sequences of block
deletions and creations within a short period of time. Drags can be inferred from the
coordinates of finger touches at each given time together with the position of the
center of the blocks, block size, and the length of the content.

Abstractions made by AAE create a new representation of play session that we
use for further analysis. To explore and reveal the stories in the data, I start by
creating a visualization pipeline as illustrated in Figure 5-2.

The sequence of actions generated by AAE is turned into a semantic/symbolic
data structure called Play Trees. Play Trees encapsulates the sequence of actions of
merge, split, and speak and is used by our layout algorithms to visualize the process
of word construction. Figure 5-3 shows a snapshot of the Play Trees of a four year-
old’s play in the first day of pilot study 5-3. In this visualization, time is shown by
horizontal yellow lines marked at 30 second increments and starts from 0 at the top
and progresses downward. Auditory feedback given by SpeechBlocks during play is
visualized by a speech bubble and blue lines marking the content and time of the
feedback. In this example, we zoom in the time between 10.5 minutes after the start
of play session. Each node represents a block that was dragged by the child from the
letter or word shelf to the play area. The label of the node shows the content of the
associated block.

Here, the child starts by bringing letter H to the play area. Next, he brings the
letter O and puts the two blocks together to create OH. When two blocks are put
together a merge happens between the branches of the tree. He continues by adding
another O and K to the beginning which results in the word KOOH and SpeechBlocks
reads it back to him shown by the blue line and a speech bubble attached to it that
Figure 5-2: Overview of the play tree visualization pipeline. a) A sample log file, every timestamp shows the time passed since the last logged frame and deletions marked in the logs are different than deletions made by user by removing blocks from the play canvas. b) The Abstract Action Extractor (AAE) parses the fine-grained data into meaningful actions and passes it on to Play Trees Generator that builds a representation that gets visualized through layout algorithms build in the visualization engine. c) An example of a Play Tree: every node represents a letter that was brought in the play canvas (e.g. "P" and "L"). When two letters are put together, a merge happens between two branches of the tree (e.g. "P" and "L" merge into "PL"). Time progresses downward and red marks on branches represent tap or touch on the associated block.

reads KOOH. From the video, we see that he has a card in front of him that spells HOOK, inspired by the character Captain Hook. After hearing the feedback, he realizes the difference between what he made and what he has in mind. Therefore, he continues by pulling the parts away. He pulls the K apart from KOOH. Splits are shown by a fork in the tree. He continues the splits and restarts by adding the letters to the right side of the previous letters. As a result, he makes the word HOOK and he hears it back from SpeechBlocks. Whenever he interacts with a block, such as dragging or tapping on a block, the associated branch on Play Trees will be marked red to highlight those interactions. As the interactions with blocks fade away, the color on the branches fade from red to gray.
He finally spells the word correctly but he struggles with directionality of text which is an early literacy learning skill. Play Trees opens a window to look at "how" he created the word and makes it possible to see that struggle. Consequently, we can make a more thorough story of the play and provide more contextual feedback and scaffolding based on a child's process.

Most importantly, the visual representation gives an overview of the play in a glance which facilitates running such analysis at scale. Additionally, these visualizations can be helpful as we move towards technologies that can be used at settings like home, where we can't afford to record the video and audio of every child during play.
Currently, our literacy experts are using these visualizations to analyze children’s play with SpeechBlocks at home to provide feedback to parents.

Moreover, the visualization engine generates a history view of play that allows you to zoom out of a single play session and look at the entire history of a child’s play. This multi-level view into the data can lead to discovery of new patterns and allows researchers to understand the story of each play session in the context of the entire history of play.

By visualizing the entire history of each child’s play in the study, I noticed that the structure of trees changes over time. Towards the last days of the pilot, similar structures emerge among many of the children (Figure 5-4). I also noticed that a particular structure is more frequent when children have a specific word in mind that they want to spell and are personally invested and engaged in the word making process. They also actively ask others for help with the parts they can’t spell in a way they want. These parts of play are significant and can be used to provide in-time feedback and scaffolding and help engage parents, peers and teachers. Therefore, exploring ways to detect those patterns in play became the focus of the rest of analysis in this thesis.
Figure 5-4: The process of making the same word in the first week and ten weeks after. As you can see the final result of the word construction has improved and there is no misspelling in the last week. Also, the process and “how” the word is made changes over time. The process is represented here by the structure of Play Trees. Similar structures emerged amongst children in sessions later in the study and when they were presented with more scaffolding on “how” to spell a word.
Chapter 6

Data Annotation

In order to analyze the data from children’s play, I went through multiple iterations of annotating the data manually to synthesize an annotation protocol that a) accommodates the idiosyncrasies existing in different children’s play processes and b) notes for scenarios that are imposed by the design of our medium, SpeechBlocks. The iterative manual annotation of the data is tedious and error prone. To facilitate and address the challenges of a manual annotation process, I developed a visual annotation tool that I will describe next.

6.0.1 Annotation Tool

The annotation tool is designed to facilitate extracting and highlighting sequences of actions that have occurred in children’s play logs. It allows for choosing a specific log and provides a simplified visualization of Play Trees as shown in Figure 6-1.

The visual annotation tool allows a user to highlight the boundaries of any sequence of nodes. Once a node is selected as the start of a sequence, it automatically highlights the possible areas that can be selected as shown in Figure 6-2 and the user can modify the selected boundaries at any time.

It also provides multiple levels of zooming to give users an easy way to zoom out from any selected sequence and be able to assess that sequence in the context of the entire play. For example, by zooming out of the entire play history one can
Figure 6-1: An overview of the annotation tool. At the top, a user can choose any play log. It automatically loads the Play Trees visualization of the selected log. Users can zoom in and out in the visualization and save the result of the annotation process using the save option.

Figure 6-2: Examples of interactions and user scenarios designed in the annotation tool. The purple areas show the sequences that are already marked as intentional. Once a user chooses the beginning of a new sequence, a horizontal line marks the beginning and the tool highlight the area that can potentially be selected as the intentional period. This especially saved time and effort when the sequences are longer.

see the whole story in one glance and make a judgment on the intentionality of each piece faster. Here in Figure 6-3 the child is writing the story of her family trip
to California. Having the entire context makes it easier and faster to annotate the intentional sequences in this case which includes the entire play session.

Figure 6-3: The annotation tool provides the ability to zoom in and out in the play session. By zooming out, a user can get an overview of the context of actions in a glance and then zoom in and make judgments on each sequence. Here for example, by looking at the entire process, the user can see that the child is writing the sentence “my family goes to California”. In this context, the short constructs such as “to” and misspellings such as “myfamymy” can be interpreted as intentional words.

In some cases similar to Figure 6-4 the child is making multiple words within the same sequence. In fact, the intention is shifted from one word to another although the overarching intention includes the bigger construct. To annotate such intermediate goal achievements that are intentional, the tool allows for cutting the sequence in subsequences as shown in Figure 6-4.

6.0.2 Annotation Framework

During my observations of children’s play with SpeechBlocks, I observed children indicating objectives or intentions for the final result of their word-making process.
Figure 6-4: Example of an intentional sequence with a shift in the end product. Although the intentionality of building the word remains the same throughout this sequence, half way through the process, the child starts a rearranging process. The tool allows us to annotate the shift in intentions within a sequence and it automatically generates and saves the related subsequences. The orange line marked the end of a subsequence and beginning of the next one.

This objective could gradually develop during the play as they brought some letters or words into the play area and listened to the sounds and were inspired to turn those letters into a word such as the one shown in Figure 6-5. This objective can also be indicated prior to the actual spelling process. For example through verbal cues such as "I want to write ...." or "how do I spell ...". The intention can also be indicated by picking one or more character cards and using the cards to see the spelling of the words as shown in 6-6.

To contrast, there are moments in the play where the child is merely exploring the application itself instead of building a word. For example, the child is bringing as many letter blocks as possible to the play area just to explore how many letters can fit on the screen as shown in 6-7 or randomly dragging a few blocks in and letting them click together just to listen to the sound of the result. Although these moments are playful and important to investigate in future studies, I focus on moments where the intention is on creating sensible words.
Figure 6-5: An example of incidental intention. The end result evolves as the child plays and the rhymes inspire new words.

Figure 6-6: An example of children indicating the intention for spelling a word by collecting a character card and using it for spelling.

The sensible words in the context of this work are limited to a) English words b)
frequent misspellings such as examples shown in figures 6-8 or 6-9 c) subtrees that were a part of an actual word but could not be finished because of the application crashing or a shortage of time. Using the video recordings, however, we see indications of implicit or explicit intentions for spelling the actual word and only a part of the word could be spelled. We were also sensitive to the words they were practicing at school during their reading time and looked closely for occurrences of misspelling or words close to their practice words. Before reviewing the result of annotation process, it is important to note that for validity of my framework I used the intuitive indications of intentionality. However, for future extension of this work, it is suggested that one or multiple external analysts go through some or all of the data and annotate the data.

6.0.3 Stats on Annotation

From the data collected from all children in both pilots, I have annotated 36 hours of play time out of which 12.1 hours are labeled as intentional play. Figure 6.1 shows the breakdown of intentional play time in each pilot.

In the second pilot, children have spent more time in the intentional play phase. There are many factors that could explain this difference. For example, from Figure ?? it can be seen that the duration of intentional play in the beginning of the study
Figure 6-8: Examples of misspelling I and L.

<table>
<thead>
<tr>
<th>Pilot</th>
<th>Duration of intentional play</th>
<th>Duration of play</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.2</td>
<td>17.8</td>
</tr>
<tr>
<td>2</td>
<td>9.9</td>
<td>18.4</td>
</tr>
<tr>
<td>Total</td>
<td>12.1</td>
<td>36.2</td>
</tr>
</tbody>
</table>

Table 6.1: Duration of play and annotated intentional play

is low and then increases after session 8.

The early sessions of the first pilot were set up for free play and in session 8 character cards were presented as a form of scaffolding. After providing character cards, the duration of intentional play increases. In the second pilot, we spent only three sessions for free play and then we introduced the character cards. Therefore, children had more sessions with the cards and that could explain the increase in hours of intentional play in the second pilot. Moreover, Figure 6-11 illustrates the increase of the duration of intentional play in the second pilot after introducing the character cards.

Although the duration of intentional play can be an important indicator, the duration can simply increase by a small number of actions performed far apart in time.
Figure 6-9: Two examples of words being misspelled. a) The word “Apple” is spelled just using the speech sounds. Therefore, the second P and the silent e at the end are not spelled. b) The word “lollipop” is spelled as spoken. These words, although not amongst the most frequently misspelled words, are spelled as they sound to the child and will be considered intentional words.

These delays between actions can be caused by distractions and interruptions in the environment. Therefore, although the child has shown the indications of intentional word construction, the child is not fully engaged in the process. This could also be caused by poor interface design where random actions are easy to do but intentional actions take much longer time to do. For example, the objects in the game are arranged so that searching for a particular object is not intuitive and takes a long time. As a result, the longer intentional play durations don’t necessarily translate into more intentional actions and engagement within the app.
Figure 6-10: Normalized intentional play duration per session in pilot 1. Each child is represented with a unique color and the intentional play duration is normalized over the play time per session for each child.

Figure 6-12 and 6-13 illustrate how the number of intentional actions have changed by presenting the character cards. Actions in this context are referred to sequences of interactions in the app that are for a) bringing a new letter into the play area, b) merging two strings, and c) splitting a word into two or more parts.

In both pilots, after introducing the character cards, the number of intentional actions also increases. Therefore, the process of word construction becomes more intentional and children are engaged in the process. These findings further highlight the significance of detecting and reporting these intentional sequences of play which is the goal of Chapter 7.
Figure 6-11: Normalized intentional play duration per session in pilot 2. Each child is represented with a unique color and the intentional play duration is normalized over the play time per session for each child.

Figure 6-12: intentional nodes per child in pilot 1.
Figure 6-13: intentional nodes per child in pilot 2.
Chapter 7

Modeling

In this chapter, I describe my attempt to model intentional play. For our models, we took a supervised learning approach that is described in the following sections. Such modeling is needed to build more flexible and personalized learning mediums that provide scaffolding for play. In addition, such models can be used to generate more fine-grained reports about the child’s play process and to inform and engage teachers and literacy coaches, parents and caregivers.

7.1 Data Preparation

To prepare the data, I started by manually labeling the sequences of actions during play that are intentional. In this context, intentional play refers to the sequence of actions that results in the creation of a word that could be determined as intentional through explicit or implicit indications. The explicit indications are verbal expressions such as “I want to write my name”. Implicit indications include actions such as spelling a) English words b) commonly misspelled words c) phonetically accurate sounding words d) words that are not spelled entirely or correctly but are from a character card chosen by a child.

Using the annotation framework described in Chapter 6, I annotated 16660 nodes in total in both pilots from which 9831 of them are labeled as intentional. Figure 7.1 shows the breakdown for each pilot.
<table>
<thead>
<tr>
<th>Pilot</th>
<th>Intentional actions</th>
<th>Unintentional</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6298</td>
<td>1330</td>
<td>7628</td>
</tr>
<tr>
<td>2</td>
<td>3533</td>
<td>5499</td>
<td>90322</td>
</tr>
<tr>
<td>Total</td>
<td>9831</td>
<td>6829</td>
<td>16660</td>
</tr>
</tbody>
</table>

Table 7.1: Actions annotated in both pilots

In my dataset, I have more positive (intentional) examples than negatives (unintentional). To remedy that bias in the data, I sample from the positive examples to create sets with the same size. Consequently, the baseline for performance of models is 0.5. Furthermore, I split each set by the ratio 90:10 and use 10% of each set for evaluations. For analysis in this chapter, I use Scikit-Learn[27] which is a widely used open source Machine Learning toolkit in Python.

7.1.1 Feature Selection

To describe each sequence of play, I use the following features representing the player, settings and interactions within the game. In this context, actions are limited to merges, splits or bringing new letters or words to the play area. The time associated with each action is extracted from an activity log. Each session can have multiple logs and time restarts for each log. Based on this notion of time, I defined new measures for time of actions for further analysis.

F1*: The content of the block that is the focus of the play at each time. It is either a number between 0 to 25 for each letter in the English alphabet or 100 for blocks containing more than one letter. During play, some letters were used more than others in the exploration phase. For example, letters such as Z or X were used more often or repeatedly when children were making funny sounding constructs.

F2*: The length of the content of each block which is the total number of letters in each block. When children are playing freely with the sounds they tend to add many letters together which happens during a short time when they bring
many random letters with fast drags. However, building longer words with longer engagement in the app can indicate intentionality of the process.

F3*: A binary value for splits. 1 shows that the word block is being split next. Splits are normally an indication of debugging which can show intention towards a certain result.

F4*: The number of parts created from splitting a word block. Split is a more difficult interaction because it requires use of both hands normally. If the children want to break the construct in more than one place, then they need to put in more focus and attention. Therefore, it’s more likely that splits happen in more intentional sequences of play.

F5*: The number of single letter parts created from splitting a word block. If children split a word construct by specifically removing a single letter from the word, the action can indicate an intentional fix or remix.

F6*: The number of parts merged to create the current word block. Merging more than two parts is a demanding interaction. Thus, in more intentional parts of play, we might see more complicated interactions within the app.

We observed from the Play Trees visualizations that when children are spelling a word that they know how to spell they follow a step-by-step process through which they add one letter at a time to the core word that they are building. Normally, they add to the right side of the word block. However, when they are not yet familiar with the concept of directionality in text, they might add to the left side. Sometimes they start by using the letters that are available on the screen and then add the extra ones to the appropriate side.

F7*: The number of single letter blocks merged to the right of the word block.

F8*: The number of single letter blocks merged to the left of the word block.

F9*: The time in milliseconds representing when an action has happened. As time progresses, children might get bored or tired. Some children, however,
need sometime to adjust to the new activity and get inspired by their peers.

F10*: Normalized time of action for each activity log. Children go through different phases of engagement with the app such as a) exploratory phase where they are playing freely with the sounds and words and don’t have a clear word in mind to build towards b) planning phases when children are not interacting in the app directly but are developing stories or words in their mind and planning on what to make next using SpeechBlocks c) intentional activities within the app, etc. Therefore, this feature can capture signs of variation of play phases in the play.

F11*: Normalized time of actions for each session.

F12*: Adjusted time for each action so that the time of action is assigned in respect to the entire time a child spent playing in the app during the entire pilot. As children progress during the pilot they become more familiar with the app and their skill level changes which affects their process.

F13: Normalized time of actions for duration of pilot. As the features and conditions of the study progresses the word making process changes, therefore this feature is important to look at.

F14: A binary value marking whether character cards were present in each session during the activity or not. We observed that scaffolding the play with character cards motivated children to spell more English words. Often, they would indicate their intentionality for making a specific word by searching for a specific card in the collection and gathering the ones they liked to spell and laying those in front of them.

F15: Gender of each participant. According to various studies including [12, 24, 11], there are differences in play based on gender.

F16: Pre-screening raw score. Literacy skills also affect the engagement with SpeechBlocks and the kind of play that happens in the app. Therefore, we used
the CTOPP pre-screening scores as our features.

F17: Pre_ screening score showing the percentile rank.

F18: Pre_ screening scaled score.

These features can be categorized in two sets. The first category is extracted from a play session within the app. Features F0 through F11 belong to this category. They are defined based on the interactions within the app without any extra knowledge of the player or the settings. These features are highlighted by a star in the list above.

The second category of features is defined based on characteristics of the player and the setting. Child’s gender (F15) or presence of scaffolding (F14) are examples of features in this category.

I start by developing models based on the first category of features only. These features are play-specific and always available from the data gathered from the app and can be used for analysis of children’s play at home when we have no further knowledge of the player. Then, I add the second category of features and evaluate the models built based on the broader range of features.

For feature selection, I start by testing the significance of features using regression analysis. From the significant features, I find the most predictive ones using a Decision Tree model. Next, I evaluate the performance of multiple models for the selected features.

7.2 Play-Specific analysis

The first category of features, F0 to F11, are extracted from the data gathered through the app alone. These features are built based on the basic abstract actions we introduced in earlier chapters such as merge or split. In addition, after investigating the patterns of play closely through the Play Trees visualization, I developed more specific actions to describe interactions in the game such as merges to the right or left end of the word block or merges that involve adding a single letter to the previous word block.
Table 7.2: Play-specific Features and the result of Linear Regression analysis

Table 7.2 shows the significance of these features using linear regression analysis. I proceed with features with p-value greater than 0.005.

Next, I feed these potential features to a Decision Tree model incrementally to find the features that are most predictive. In each iteration, based on the performance of the model, the feature which leads to the best performance gets added to the existing list of features in the model.

Figure 7.3 shows the result of this iterative process. The performance of the model increases the most by addition of F2,F6,F7,F8 and F3. It shows that the combination of the existence of a split and progression to the left or right by adding a single letter are the most predictive features for modeling intentionality.

Based on my observations during the pilot studies and reviewing the visualization of play logs, the existence of splits together with progression to the right or left is normally a sign of “debugging”. For example, when a child brings letters into the play area and puts the letters together one by one, the splits happen to fix an error in spelling by adding a missing letter or replacing a wrong letter. Another reason for splitting is to reuse a letter or a part of a word for a new word. This is especially seen
when children are building words that rhyme: for example, cat to hat, then hat to chat. In all these cases children are more intentionally involved in the word making process than simply enjoying the game dynamics and features.

Using these features, we trained different models. Table 7.4 shows the evaluation of these different models. The Decision Tree performs better than the others. In addition, it can provide further insights into how the value of each feature changes the predictions of our model.

<table>
<thead>
<tr>
<th>Model</th>
<th>performance</th>
<th>Average precision</th>
<th>Average recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision Tree</td>
<td>0.674</td>
<td>0.68</td>
<td>0.67</td>
<td>0.67</td>
</tr>
<tr>
<td>SVM</td>
<td>0.654</td>
<td>0.65</td>
<td>0.65</td>
<td>0.65</td>
</tr>
<tr>
<td>NaiveBayes</td>
<td>0.593</td>
<td>0.61</td>
<td>0.60</td>
<td>0.59</td>
</tr>
</tbody>
</table>

Table 7.4: performance of different models on F2,F6,F7,F8,F3

7.3 Additional Contextual Analysis

The features used in the previous section are purely based on the interactions in the app. However, we can train our models using a broader set of features including
player and settings characteristics.

Table 7.5 shows the result of linear regression analysis using features F1 to F18. As highlighted in the table, all features have a p-value less than 0.005 except F9, F17 and F18. Therefore, we exclude these features from our future analysis. Next, we identify the most predictive features using a Decision Tree model as shown in the Table 7.6. Finally we compare the performance of different models in Table 7.7. Although the majority of the features show significance in characterizing intentional play, the scaffolding provided in the play together with some play-specific features such as splits appear to be most predictive. This finding also matches the observations we made during both pilot studies. Children seemed to get ideas from character cards and action cards to make more words, stories and sentences. Additionally, during their play sessions, they kept asking for more characters and words that go with those characters so that they could spell out the stories they had in mind.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Coefficient</th>
<th>P-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>const</td>
<td>1.3080</td>
<td>0.000</td>
</tr>
<tr>
<td>F1</td>
<td>-0.0047</td>
<td>0.000</td>
</tr>
<tr>
<td>F2</td>
<td>0.0294</td>
<td>0.000</td>
</tr>
<tr>
<td>F3</td>
<td>0.1624</td>
<td>0.002</td>
</tr>
<tr>
<td>F4</td>
<td>-0.2293</td>
<td>0.000</td>
</tr>
<tr>
<td>F5</td>
<td>0.1098</td>
<td>0.000</td>
</tr>
<tr>
<td>F6</td>
<td>-0.0584</td>
<td>0.000</td>
</tr>
<tr>
<td>F7</td>
<td>-0.2041</td>
<td>0.000</td>
</tr>
<tr>
<td>F8</td>
<td>-0.1110</td>
<td>0.000</td>
</tr>
<tr>
<td>F9</td>
<td>-5.174e-06</td>
<td>0.014</td>
</tr>
<tr>
<td>F10</td>
<td>0.0577</td>
<td>0.000</td>
</tr>
<tr>
<td>F11</td>
<td>-0.0583</td>
<td>0.000</td>
</tr>
<tr>
<td>F12</td>
<td>-5.905e-06</td>
<td>0.000</td>
</tr>
<tr>
<td>F13</td>
<td>0.5881</td>
<td>0.000</td>
</tr>
<tr>
<td>F14</td>
<td>-0.5059</td>
<td>0.000</td>
</tr>
<tr>
<td>F15</td>
<td>0.0287</td>
<td>0.000</td>
</tr>
<tr>
<td>F16</td>
<td>7.84e-05</td>
<td>0.000</td>
</tr>
<tr>
<td>F17</td>
<td>0.0002</td>
<td>0.284</td>
</tr>
<tr>
<td>F18</td>
<td>-0.0002</td>
<td>0.211</td>
</tr>
</tbody>
</table>

Table 7.5: Features for modeling Play Trees
Table 7.6: Decision Tree for all features

<table>
<thead>
<tr>
<th>Features</th>
<th>performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>F14</td>
<td>0.881</td>
</tr>
<tr>
<td>F14, F7</td>
<td>0.882</td>
</tr>
<tr>
<td>F14, F7, F3</td>
<td>0.882</td>
</tr>
<tr>
<td>F14, F7, F3, F8</td>
<td>0.882</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5</td>
<td>0.880</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15</td>
<td>0.876</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6</td>
<td>0.865</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6, F2</td>
<td>0.848</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6, F2, F4</td>
<td>0.825</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6, F2, F4, F16</td>
<td>0.809</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6, F2, F4, F16, F12</td>
<td>0.891</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6, F2, F4, F16, F12, F13</td>
<td>0.913</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6, F2, F4, F16, F12, F13, F11</td>
<td>0.926</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6, F2, F4, F16, F12, F13, F11, F10</td>
<td>0.925</td>
</tr>
<tr>
<td>F14, F7, F3, F8, F5, F15, F6, F2, F4, F16, F12, F13, F11, F10, F1</td>
<td>0.911</td>
</tr>
</tbody>
</table>

Table 7.7: performance of different models on F2,F6,F7,F8,F3

<table>
<thead>
<tr>
<th>Model</th>
<th>performance</th>
<th>Average precision</th>
<th>Average recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision Tree</td>
<td>0.91</td>
<td>0.90</td>
<td>0.88</td>
<td>0.88</td>
</tr>
<tr>
<td>SVM</td>
<td>0.88</td>
<td>0.90</td>
<td>0.88</td>
<td>0.88</td>
</tr>
<tr>
<td>NaiveBayes</td>
<td>0.862</td>
<td>0.89</td>
<td>0.86</td>
<td>0.86</td>
</tr>
</tbody>
</table>
Chapter 8

Conclusion

Throughout this thesis, I developed a play analytics framework to study children’s engagement in a constructionist literacy learning technology. To do so, I designed a wholistic analytical pipeline from data acquisition to modeling behavioral patterns. The pipeline includes 1) data gathering 2) semantic action extraction from raw data 3) action network formation 4) visualization of play 5) data annotation 6) modeling of behavioral patterns.

As part of the first step of the analytics pipeline, collecting data, I designed and developed a longitudinal pilot study for a constructionist early literacy technology. The study was designed to be integrated into the classroom environment. During each session of the study, I collected feedback from the teachers and children and integrated that feedback into the development and design of materials and scaffoldings for future sessions. In other words, each session was seen as a rapid design cycle from idea to user testing and evaluation. I didn’t constrain the study with predefined controlled conditions and the study evolved as a part of the feedback loop. Also, the longitudinal nature of the study made it impossible to track all the variables affecting the participants during the course of the study. therefore the pre- and post-test scores can’t be associated directly with what happens in the study.

In addition, the open-ended nature of play, the dynamic environment of a classroom setting and rich social interactions within the group of children required a comprehensive data collection strategy. Our participants were children between the
ages of four and five which made it challenging to ask for direct feedback and required a more passive approach to data collection.

We instrumented the classroom with cameras and collected audio and video recordings of the play sessions. Furthermore, we collected fine-grained interactions within the app in the form of play logs. The fine granularity of data ensures that no data is lost and allows us to go back and ask new questions as analysis evolves (Step 1). The data gathered in Step 1 is rich, however, it is raw and hard to analyze. In this work, we developed an ensemble of algorithms to provide ways to analyze children’s play not only through what they made during play but also how they did it. These algorithms also help us unearth stories hidden in the data by looking into the process over time.

I started by developing the computational unit that parses the play logs into abstract actions that are contextually meaningful in the app and easier to interpret semantically. For example, turning sequences of time-stamped finger positions into the action of dragging an object in the game (Step 2). Using these abstractions, we could provide statistics such as frequency of each action in a play session. However, these actions are not simply disjointed and independent. Normally, they are a part of a sequence of actions involved in a word making process. To explicitly consider this relation between the actions, we built a network of consequent actions that we call a Play Tree. Each play session can be represented by multiple Play Trees (Step 3). Forming networks out of sequences of actions turned the logs of each session into computable structures and enabled us to: 1. Move beyond the simple statistics of actions and use the structure of the network to study the patterns that emerge from the connections between actions. 2. Use the existing toolboxes for analyzing networks 3. Develop network visualization algorithms to explore our data visually.

To visualize each play session and the history of play, we developed a visual representation for Play Trees and implemented the layout algorithms so that there are no unwanted edge crossings and the order of actions dictates the placement of nodes in the visualization (Step 4). Using the Play Trees visualization, we could get a closer look into the play process and tell a more complete story of the play. We
noticed a difference in the structure of Play Trees between periods where children are simply exploring the game and where they are building more intentionally towards a certain word they have in mind. These intentional parts of the play are significant because: 1. They are more personally meaningful to the children. 2. Children are consciously engaged in the process and are more likely to remember these parts.

By detecting and highlighting these parts of play we can: 1. provide scaffolding for the children’s learning process and emotions. 2. further engage teachers, parents and peers in children’s learning process and help them create activities relevant to the child’s process. To develop models for intentional play, I developed a framework for annotating the data (Step 5). I annotated for sequences of actions resulting in a) English words, b) words with common misspellings, c) phonetically correct sounding words, d) any of the previous sensible words that are spelled backwards, e) any construct accompanied with explicit verbal cues for the intention of spelling a word or implicit cues such as holding a card with the word’s spelling as captured in the video and audio recordings. In these cases, even if they don’t finish the word-for example if they run out of time-the existing portion was annotated as intentional.

To facilitate and accelerate the annotation process, I developed a visual annotation tool that enabled me to:

1. Annotate each sequence of play in the context of the entire play by allowing users to zoom in and out in time.

2. Revise a previously annotated sequence which was necessary for our iterative process.

3. Turn the result of the annotation process into computable data structures that can be immediately used for statistical analysis

4. Develop further analysis by overlaying the annotation on the Play Trees structures.

Next, I used the annotated data to develop models to detect sequences of intentional play (Step 6). I used two categories of features: 1. Features characterizing
the process of play within the app. Features related to players (e.g., literacy skills and gender), the settings and scaffolding available to the player. Using the first category alone, the features related to debugging the spelling and also to the addition of a single letter to the end of the previous construct are among the most predictive features of intentional play. Using both categories of features together showed us that presence of scaffolding (character cards) is also a predictive feature.

This analytics pipeline provided ways to analyze children’s play not only through what they made during play but also how they did it. The ability to look into the process over time revealed patterns in the data that we didn’t know existed before and provided us with insights into how to analyze for more behavioral patterns such as intentionality in play. Literacy experts are already using the results of this pipeline to provide feedback for parents and suggest activities based on children’s play.

This work was built based on limited annotation of the videos and data due to time constraints. However, with more extensive annotation of the data, more sophisticated patterns can be fed into our models. Also, our data was collected over the course of two months. Literacy skills develop gradually and collecting data for longer periods can give us more opportunities to look at the progression of play as children develop literacy skills over time.

This study was done in a classroom setting. In the future, as we move towards continuous data collection both at school and at home, we can study the effect of the context on play. In the current study, we looked at the effect of a limited number of scaffolding on play and we did not focus yet on the effect of social interactions in play. In the future, as more social features get integrated into the app, this analytics pipeline can be extended to bring more intelligence to the learning environment to moderate the engagement of parents, peers, and teachers in the child’s process to provide support when and as needed.
Bibliography


73


[34] Carly Shuler, Zachary Levine, and Jinny Ree. ilearn ii an analysis of the education category of apple’s app store, 2012.


75

