

Understanding Dialogue: Sentiment and Topic Analysis of Dialogue Transcripts

by

Anjali Muralidhar

B.S. Electrical Engineering and Computer Science, MIT 2012

Submitted to the Department of Electrical Engineering
and Computer Science

in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

May 2013

[JUNE 2013]

Copyright 2013 Anjali Muralidhar. All rights reserved.

The author hereby grants to M.I.T. permission to reproduce and
to distribute publicly paper and electronic copies of this thesis document in whole
and in part in any medium now known or hereafter created.

Author:

Department of Electrical Engineering and Computer Science
May 24, 2013

Certified
by:

[Catherine Havasi] Thesis Supervisor
May 24, 2013

Accepted
by:

Prof. Dennis M. Freeman, Chairman, Masters of Engineering Thesis
Committee

Understanding Dialogue: Sentiment and Topic Analysis of Dialogue Transcripts

by

Anjali Muralidhar

Submitted to the

Department of Electrical Engineering and Computer Science

May 24, 2013

In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

The field of natural language processing has had success in analyzing sentiment and topics on written text, but similar analysis on dialogue is more difficult due to the fragmented and informal nature of speech. This work explores sentiment and topic analysis on data from the Switchboard dialogue corpus, as well as a dataset of recorded dialogues between parents and children while reading an interactive e-book. The goal was to be able to identify the emotion and mood of the dialogue in order to make inferences about what parents and children generally talk about when reading the book because conversations between an adult and child while reading a book can greatly contribute to the learning and development of young children.

Table of Contents

- Introduction 4
- Machine Learning Tools from the Digital Intuition Group 6
 - Open Mind Common Sense Knowledge Base 6
 - ConceptNet..... 7
 - AnalogySpace 8
 - Divisi 10
- Related Work 11
 - Luminoso 11
 - Colorizer 12
 - The Glass Infrastructure 14
 - Opinion Summarization in Conversations 15
 - Behavior Trajectories of Cancer Patients in Online Support Groups 17
- Previous Work 19
 - Textual Tinkerability 19
 - Initial Topic Recognition for E-book Data..... 21
- Analysis on Tinkerbooks Data 24
 - Emotions and Sentiment 25
 - Sentiment Analysis on Tinkerbooks Data 29
 - Magnitude Discrepancies for Different Emotions 31
 - Using Topic Words Instead of Full Dialogue..... 33
 - Conclusion: Tinkerbooks Results 37
- Analysis on Switchboard Corpus 39
 - Sentiment Analysis on Switchboard Data 40
 - Topic Analysis of Switchboard Data 41
 - Topic Analysis Using SVM 43
 - Conclusion: Switchboard Results 45
- Conclusion 47
- Future Work 50
- Acknowledgements 51
- References 52

Introduction

Sentiment and topic analysis have been well studied in the field of natural language processing. Currently, there is technology in place to derive information about sentiment and topics from written natural language text. However, the same technology does not work as well for dialogue. People write and speak very differently. Transcripts of speech show that people often speak in fragments, with pauses and breaks, and do not always follow the rules of grammar. This is in contrast to written text, where the point being made by the writer is usually explicit and the writing is grammatical. Transcripts can be recorded from many sources such as customer service call centers, interviews, lectures, and phone conversations. However, analysis of the emotion and topic of such transcripts has not been heavily explored for general speech. For example, sentiment analysis of call center transcripts has the potential to give an indication if the call is particularly hostile and the customer is having a negative experience. There has been success in analysis of dialogue in very specific domains, where the input and output can be largely anticipated due to the narrow scope of the domain, and in sentiment analysis when taking into account prosody. The scope of this work is more general and is not limited to set input and outputs, nor does it consider prosody.

Similar analysis of conversations between parents and children as they read an e-book on a tablet has the potential to reveal emotive experiences that can supplement a child's learning. My initial focus was to work with a dataset of parent-child dialogues reading a e-book aloud together and engaging in peripheral conversation as they read. Reading aloud to children has been known to foster cognitive, language, and literacy skills. Reading

on a tablet adds a level of interaction and engagement that has the potential to augment a child's learning and expand their vocabulary. Sentiment analysis of this dataset should show fluctuations of emotion as the parents and children discuss the story, which would reflect the additional engagement of a child reading with a parent from a tablet instead of from a traditional book. I additionally analyzed transcripts phone conversations between adults to look for similar trends. Various sentiment and topic detection techniques were tested to develop a system that would recognize the emotion or topic of a conversation.

Machine Learning Tools from the Digital Intuition Group

Open Mind Common Sense Knowledge Base

Using common sense reasoning to evaluate sentiment and topics can provide more insight than using other machine learning techniques that do not include such information. Humans find it trivial to recall background knowledge when learning new things. They use this background knowledge to process new information and infer the proper response. For example, people recall the social etiquette they learn growing up to behave properly when they meet new people. They know that it is appropriate to extend their hand and shake the other person's hand as a polite greeting.

Computers have no such background knowledge. They have not spent a lifetime learning concepts and procuring knowledge to inform their decisions. Computers need a sense of concepts and the relations between concepts to form a knowledge base to create artificial intelligence systems. The Open Mind Common Sense project seeks to give computers a common sense knowledge base from which they can find relations and make inferences.

For years, the Open Mind Common Sense project has been acquiring information from volunteers from the general public via the Internet. Since 2000, more than 700,000 commonsense statements have been collected by over 16,000 contributors [1]. Natural language statements are presented to users, and users are asked to define concepts and relations, also as natural language. Thus, the knowledge stored by the Open Mind Common Sense project is foremost in natural language format so that the system is rooted in the way

people actually write and speak. Although the knowledge base inputs and outputs information as natural language, it needs a more structured format in order to be processed by computers. This format is provided by ConceptNet.

ConceptNet

Currently, the Digital Intuition group, as a part of the Open Mind Common Sense project, has been developing a system called ConceptNet. ConceptNet [2] is a semantic network that defines concepts and graphs relations between concepts based on the Open Mind Common Sense corpus. It is a representation that links concepts, relations, and the polarity of assertions in the common sense knowledge base. Using a parser, it transforms natural language information into a format for the computer to understand, in keeping with the ConceptNet architecture.

Each node of the ConceptNet graph is a concept while edges are assertions of relations between two concepts. ConceptNet selects assertions from a predetermined limited set of relations, based on relations seen in the Open Mind Common Sense corpus. Assertions are assigned a frequency value based on how often the public say the corresponding relation holds. Assertions also have a confidence value based on how confident the system is about that assertion. This score can be increased if many users make the same assertion.

The end result is ConceptNet, a network of concepts and relations such as “isA” or “part of” that connect each word to words and concepts it is related to. ConceptNet is useful to map a word to the understanding of that word and related words and concepts.

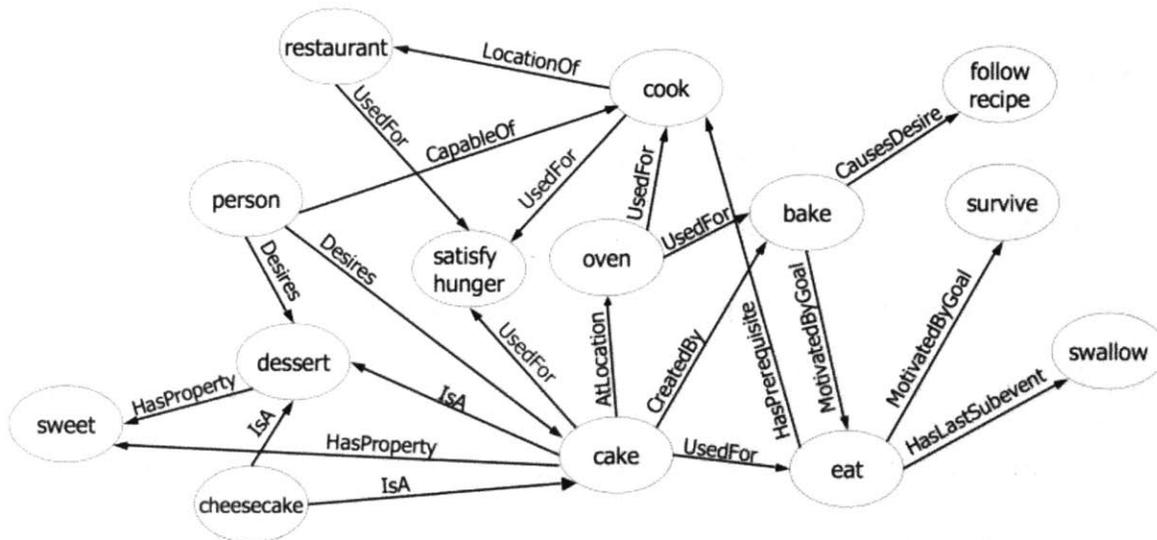


Figure 1: A small section of the ConceptNet network [1]

AnalogySpace

The information in the common sense knowledge base is for the most part correct, but there is still a substantial amount of noise. The computer must still be able to reason using the information in the knowledge base and ConceptNet despite this noise. In order to do this, the computer must not rely completely on correct information and assertions. Instead, the system must be able to make inferences based on similarities, using AnalogySpace [1].

AnalogySpace helps a reasoning system make conclusions about common sense data. AnalogySpace configures source information into a knowledge matrix of objects and features

and tries to reduce the dimensionality of this matrix to make conclusions. It combines symbolic reasoning and statistical methods to learn from source data.

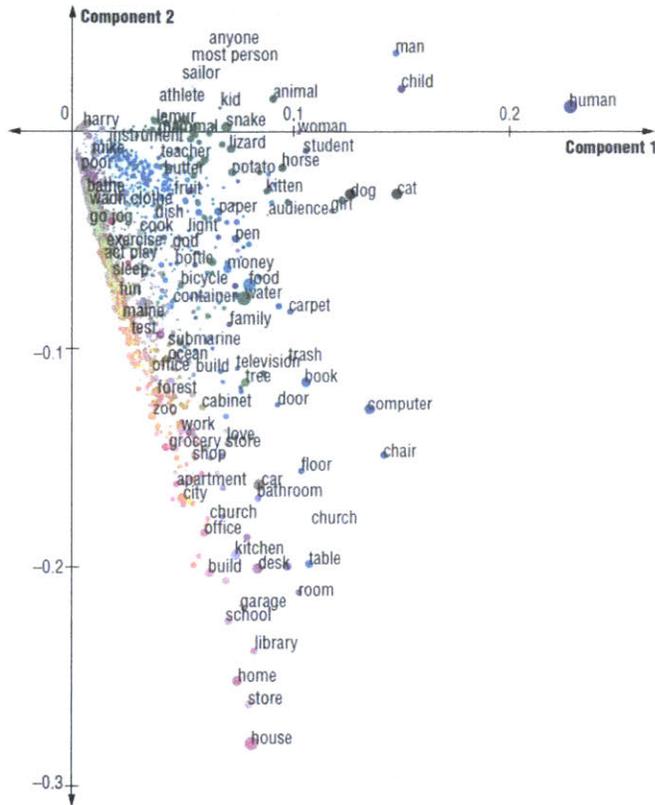


Figure 2. Concepts from ConceptNet are plotted on axes of two principal components, providing a two dimensional visualization of AnalogySpace [3].

The matrix is formed from the ConceptNet graph. The graph is transformed into a matrix by assigning concepts as rows, features as columns, and the value corresponds to the truth value of an assertion [3]. A feature is defined as an assertion with one concept left blank. For the assertion, “a trunk is part of a car,” the (partOf, car) feature is added to “trunk” and (trunk, partOf) is added to the “car” concept. The truth value represents how many users support the assertion

based on their input. The dot product of rows represents the similarity between two concepts. These dot products are approximated by truncated singular value decomposition (SVD) which reduces the dimensionality of the matrix. This results in a more compact space in which to calculate similarity.

Concepts and features form the rows and columns of the matrix on which the SVD is performed, projecting the concepts and features on the same space. In this new space, the

concepts and features can be described as linear combinations of the principal components identified by the SVD. Thus, they can be plotted on axes of these components (see Figure 2).

AnalogySpace can take information from ConceptNet to make inferences and predict new assertions. It can then make inferences to give feedback to contributors to the Open Mind Common Sense knowledge base. Thus, AnalogySpace can help fill in gaps in the original knowledge base by asking questions back to Open Mind Common Sense users. By having these questions answered by users, the Open Mind Common Sense project learns new information in a targeted way. AnalogySpace identifies where the information was previously lacking so that it can ensure that information is procured via the Open Mind Common Sense project.

The more compact AnalogySpace makes it much easier to find similarities between concepts in ConceptNet. It is possible that a concept does not have a truth value for a given feature. In that case, the system would look up if similar concepts have a truth value for the feature. If they do, then the system can infer that the original concept probably has that feature as well. In this way, AnalogySpace can make inferences about information that is not explicitly in the common sense knowledge base.

Divisi

In order to efficiently compute the aforementioned SVDs, a software package called Divisi was used [4]. Divisi integrates common python packages such as Numpy [5] and PySparse [6], as well as a wrapper of SVMLIBC [7] to compute sparse SVD. Divisi also makes it

easy to import the ConceptNet graph to be used in the SVD analysis of concepts and features. Divisi was used in computation for this thesis.

Related Work

Luminoso

With the information from ConceptNet and AnalogySpace as a base, Luminoso [8] was developed as a toolkit to help users process natural language data. Luminoso uncovers

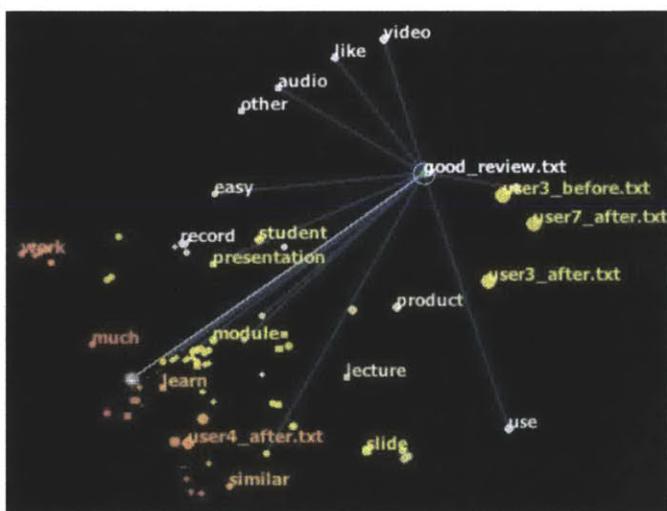


Figure 3. The point selected represents the content of a good review. It is connected to the origin with a gray line. Related points are represented in white and yellow, while unrelated points are in orange [8].

and visualizes connections and common sense understanding that allows natural language data to be accurately interpreted. Luminoso is a tool that visualizes the information in the multi-dimensional ConceptNet graph and reduced AnalogySpace. Users can feed in natural language text and see a visual representation of the common sense analysis of the data. They can closely analyze a single point in the data and see all the related information, inferences, and associations corresponding to that point.

Luminoso allows a human user to navigate through all of the connections and similarities found by ConceptNet and AnalogySpace. Using the Luminoso visualization, users can analyze input data for a targeted purpose. If they want to discern what parts of the data

correspond to good or bad things, they can visualize the data points plotted along axes representing good or bad (see Figure 3). Luminoso allows a user to elicit the information that is truly interesting from the ConceptNet data.

Colorizer

Colorizer is a system that maps words and phrases to a color that is associated with that word either physically or emotionally, also using common sense reasoning [9]. An example of a physical color association is the word “beach.” This word would be associated with the colors blue and beige, with the blue representing the water and the beige representing the sand. However, the word “envy” has no physical color since it is an abstract concept. Colorizer would assign a color based on an emotional association, so “envy” would be associated with the color green.

The color information was gathered from three sources: NodeBox [10], ConceptNet, and an online survey by Rnadall Munroe[11]. NodeBox is a Python library with a color visualization toolkit that maps a set of words to eleven basic colors, such as blue, red, green, and brown. ConceptNet also has some color information about concepts built in as assertions, specifically in the *hasProperty* relation. Using NodeBox’s concept to color mapping, as well as ConceptNet’s mapping, an association matrix can be built with (color, word) pairs. Robert Munroe, the author of XKCD, sent out a survey that showed participants a random color and asked them to describe it. Descriptions could be as simple as “blue” or “light yellow” and as creative as “dusty prune” or “eggplant.” Munroe’s color names were

mapped to concepts by using ConceptNet to parse the description and associate it with concepts. For example, the color “lemon yellow” will be associated with the concepts “yellow” and “lemon.” In this way, a mapping of concepts to colors can be made.

Often a concept is matched to multiple colors. Munroe’s color survey also gives rise to many different colors. To figure out the true value of a color that participants agreed on, color values and names were plotted in the CIE Lab color space which measures differences in color by Euclidean distances [12]. Color descriptions associated with concepts from ConceptNet and NodeBox were also included as points in Lab space. The statement “money is green” generates an observation that maps the concept of “money” to the color green. Color centroids are also found in this space, showing consensus of true color values (see Figure 4).

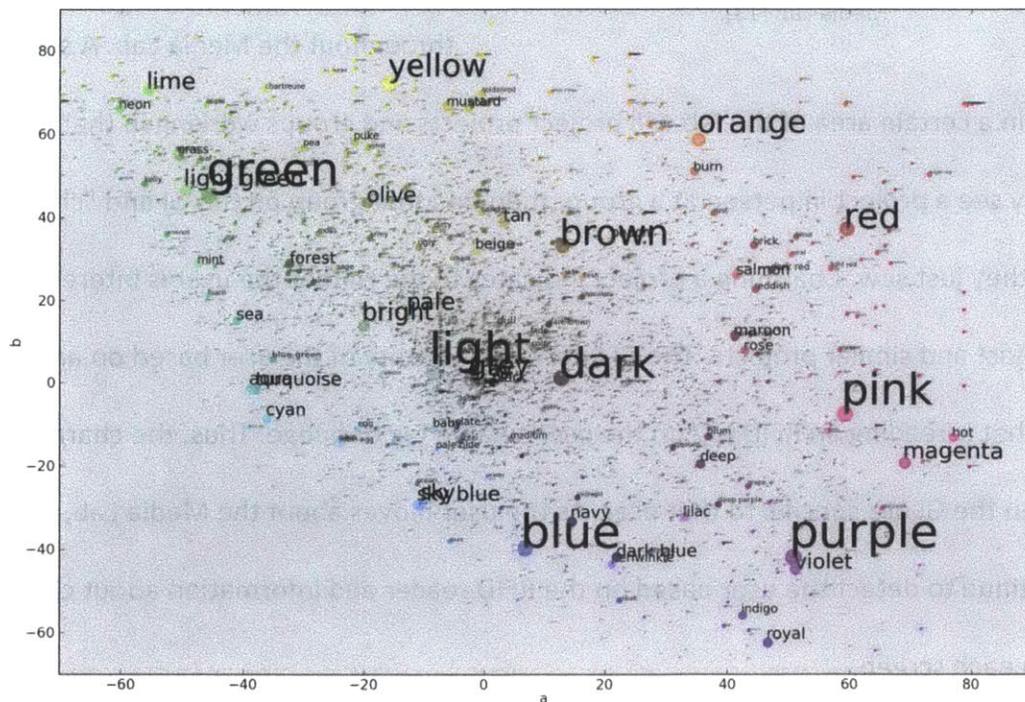


Figure 4. Color centroids of all concepts plotted in Lab coordinates (a vs. b). Names represented with larger text have more information [9].

The Glass Infrastructure



Figure 5. Users interacting with a GI screen at the MIT Media Lab [13].

The Glass Infrastructure, or GI, is a system of touchscreen surfaces at the MIT Media Lab with which users can interact and discover related projects in that lab that interest them, using ConceptNet [13]. This is especially useful for visitors to the Media Lab who want to see projects that are specific to their area of expertise or interest (see Figure 5).

Screens of the GI are set up throughout the Media Lab. A screen located in a certain area of the lab will project projects and groups working in that area. A user may see a project in person at a group, then find that group on the GI and “charm” the project they just saw. Charming a project indicates to the GI that the user is interested in that project and similar projects. The GI knows the identity of the user based on an RFID reader that is reading an RFID tag on the user’s name tag or badge. Thus, the charms a user makes on the GI are specific to that user. As the user moves about the Media Lab, GI screens will continue to detect the user based on the RFID reader and information about charms will move to each screen.

Based on the projects that a user has charmed, the GI will recommend other projects that the user may be interested in exploring. This recommendation is done using ConceptNet. All project descriptions in the Media Lab are stored in the Project List Database, or PLDB. Using ConceptNet, these project descriptions are read into a system and similarities are found between projects. This allows the GI to figure out that projects related to “human computer interaction processing” are similar to projects related to “user interfaces.” The GI uses ConceptNet to then recommend projects that are similar to projects that the user has charmed. This technology has proven to be especially useful during sponsor visits to the Media Lab. Sponsors fund projects in the lab and can use this tool to discover new or similar projects that may be of use to their company.

Opinion Summarization in Conversations

There has also been previous work done analyzing text and dialogue that does not use common sense reasoning, but instead uses other machine learning techniques. Researchers at The University of Texas at Dallas came up with two methods [14] to determine the sentiment, or opinion, of a spoken conversation from the Switchboard Corpus [15]. Both methods have been widely used for text summarization and were applied to spoken dialogue and analyzed. The first method used sentence ranking. A score was assigned to each utterance, or dialogue act (DA), and the most highly ranked were selected based on the probability of the DA for a given topic and sentiment. The second method was a graph-based method that took into account the dialogue structure in order to score the

[Extractive Summary]

I think I've seen some statistics that say that, uh, it's more expensive to kill somebody than to keep them in prison for life. committing them mostly is, you know, either crimes of passion or at the moment or they think they're not going to get caught but you also have to think whether it's worthwhile on the individual basis, for example, someone like, uh, Jeffrey dahlmer, by putting him in prison for life, there is still a possibility that he will get out again. I don't think he could ever redeem himself, but if you look at who gets accused and who are the ones who actually get executed, it's very racially related – and ethnically related

[Abstractive Summary]

B is against capital punishment except under certain circumstances. B finds that crimes deserving of capital punishment are "crimes of the moment" and as a result feels that capital punishment is not an effective deterrent. however, B also recognizes that on an individual basis some criminals can never "redeem" themselves.

[Overall Opinion]

Somewhat against

DAs. The graph-based method takes into account dialogue structure factors such as whether DAs were from the same speaker and within the same turn or whether DAs form a question-answer pair. These methods take the conversation dialogue as input and output an extractive summary of the dialogue. An extractive summary takes lines from the original dialogue and selects them for the summary based on a score. An abstractive summary forms new sentences that make up a summary. The sentence-ranking and graph-based methods explored were used to form an extractive summary that would give rise to the overall opinion of the conversation (see Figure 6). Both methods performed better than the baseline method which simply extracted the utterances of longest

Figure 6. Examples of extractive and abstractive summaries of dialogue data, as well as an assessment of overall opinion of the topic [14].

length. In particular, the graph-based method worked well because it incorporated the structure of the conversation into the score.

Behavior Trajectories of Cancer Patients in Online Support Groups

Additional work has been done to evaluate sentiment based on online forum posts, again not using common sense reasoning but rather other machine learning tools. It is difficult to track the behavior of medical patients as they go through diagnosis, treatment, and/or remission of a disease such as cancer. Often, qualitative measures are necessary to discern the behavior and emotional state of a patient. This makes it difficult to efficiently track the progress of many patients. However, results have shown that the actions of cancer patients on online health support groups can be accurately and computationally tracked by analyzing posts in discussion threads in online forums [17].

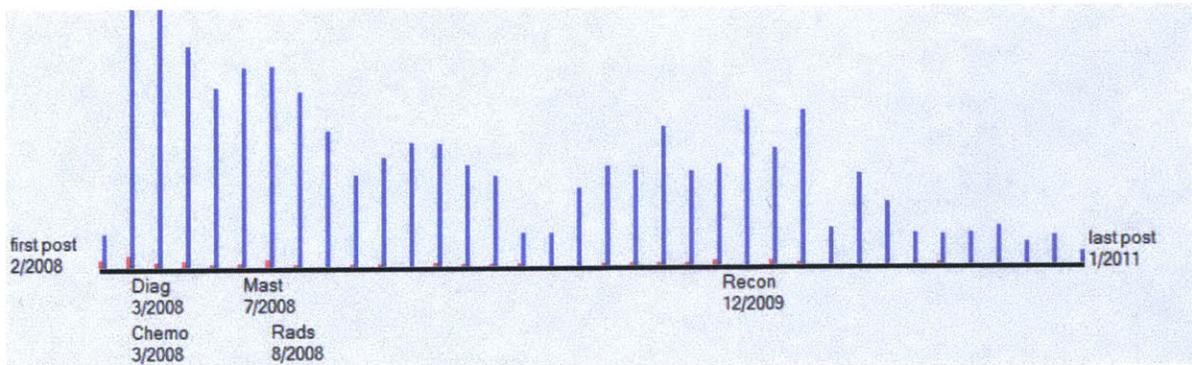


Figure 7. The post frequency of a specific cancer patient on an online support forum throughout cancer event months and non-event months [17].

The activity of individual users was tracked in an online forum for cancer patients. They had the option of tagging each post with a substantial cancer event such as diagnosis, chemotherapy, and lumpectomy. Figure 7 shows the frequency of posts for a specific user

over time that is marked with important cancer events. The cancer events are abbreviated (e.g. "Diag" represents diagnosis and "Chemo" represents chemotherapy). It was found that cancer patients post more than twice as often in the forums in stressful months marked by cancer events than in non-event months. The topics of their posts change depending on the cancer event, and are detected using Latent Dirichlet Allocation (LDA) topic modeling [18]. This topic modeling showed that spiritual and religious topics came up when discussing the metastasizing cancer. It also revealed that topics related to hair loss and physical appearances were discussed during event months marked by Chemotherapy. By evaluating online posts of cancer patients, it was shown that a behavioral trajectory can be constructed that represents the stressful cancer events an individual patient went through and what topics he or she were specifically concerned about.

Previous Work

Textual Tinkerability

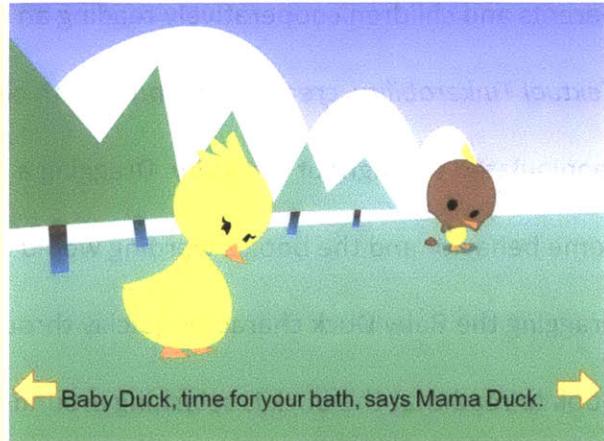
One of the corpuses of data I used came from previous work at MIT that recorded parents and children cooperatively reading an e-book on a tablet [19]. The project, known as *Textual Tinkerability*, created an e-book with characters and props that could be manipulated throughout the story. Dragging a character around the screen would cause some behavior and the book's wording would change to describe the behavior. For example, dragging the Baby Duck character quickly through the pond (see Figure 8, Page 5) causes e-book to read that the duck is splashing around the pond. However, dragging the duck slowly through the pond causes the book to read that the duck is simply swimming around the pond. The physical appearance of the duck can be changed by altering the color of the duck. The goal of the story is to make the Baby Duck character clean. If the user does not make sure that the duck goes into the pond, the story will never reach the final page where Mama Duck tells Baby Duck, "Good job." This adds an additional layer of interaction and manipulation of the story because the character's behavior will not only affect how they interact with the page (e.g. swimming or splashing in the pond or waddling either slowly or quickly) but also what the final outcome of the story will be (e.g. whether or not Mama Duck congratulated Baby Duck).

By allowing parents and children to interact with the book and manipulate the characters, this project hoped to foster engagement with the story and promote literacy skills. The project also shows how textual tinkerability fosters additional dialogue between

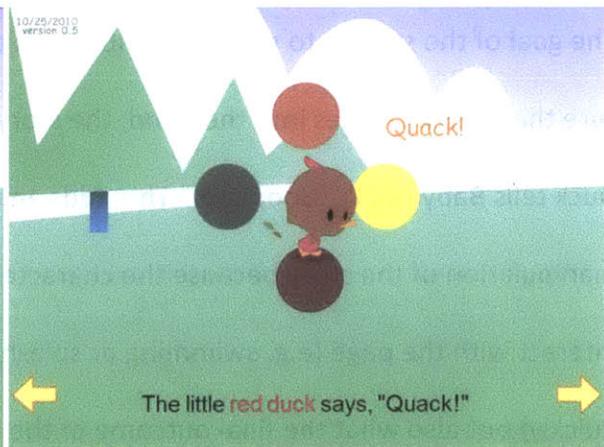
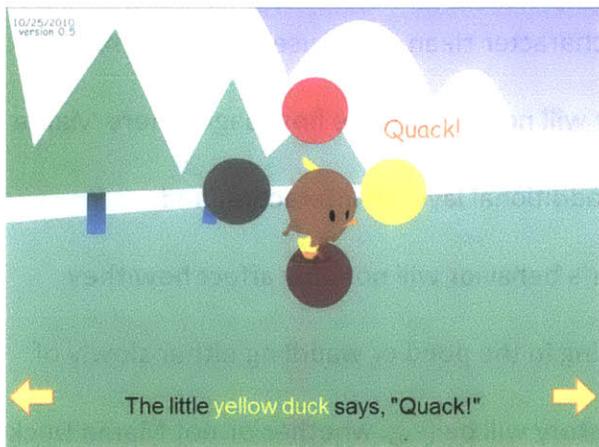
the parents and children. All conversations between parents and children interacting with the book were recorded and transcribed for later analysis. These dialogue transcripts make up the dataset I will refer to as the Tinkerbooks dataset.



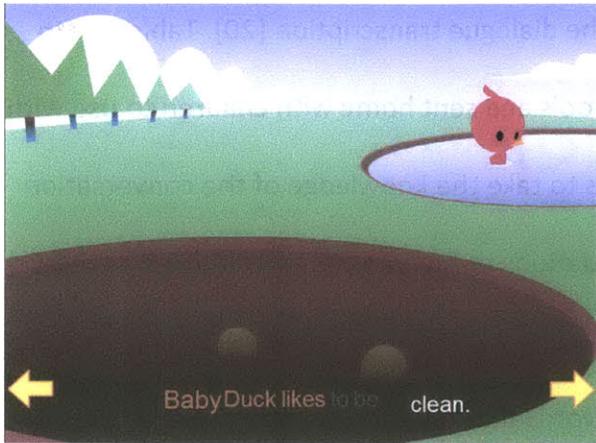
Page 1. Start screen.



Page 2. Time for Baby Duck's bath, Baby Duck is initially dirty.



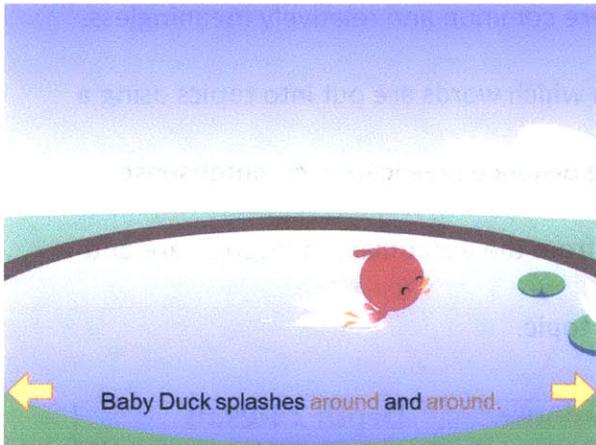
Page 3. User can interactively choose color of Baby Duck, duck shown in yellow and red.



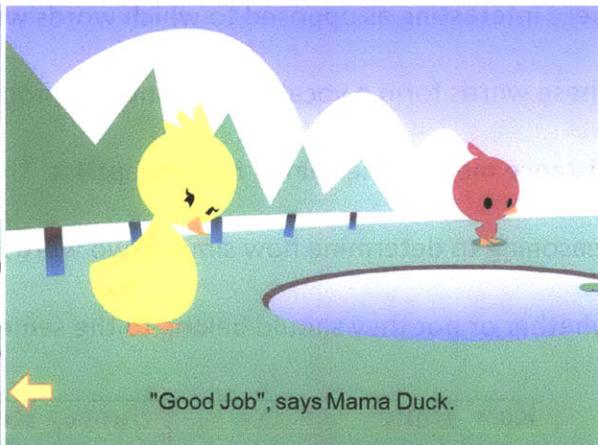
Page 4. User can make duck clean or dirty



Page 5. Duck waddles to the pond



Page 6. Duck plays in pond



Page 7. End page, can only be reached if duck is clean at the end of page 3.

Figure 8. The 7 pages of the Tinkerbooks e-book. The words written in the e-book change as the user manipulates the page [19].

Initial Topic Recognition for E-book Data

Another version of an e-book similar to the Tinkerbook described above, called CloudPrimer, was used to gather recordings of parent-child dialogue. This version of the e-book had a similar story about a duck taking a bath, but the plot and interactions were slightly different. The goal of this project was to gather these recordings, transcribe them to

text, and extract topics of the discussion from the dialogue transcription [20]. Tablets with the e-book loaded onto them were given to schools and sent home with children to use with their parents. The ultimate goal of this project is to take the knowledge of the conversation topics and leverage it to make topic recommendations to the parents and children as they read the e-book on a tablet.

ConceptNet was used to come up with words that roughly describe topics in the transcripts. An interest metric helped designate which words from the overall vocabulary were interesting as opposed to which words were common and relatively meaningless. These words form a vocabulary of interest from which words are put into topics using a distance metric determined by ConceptNet. The distance metric uses commonsense reasoning to determine how similar two words from the vocabulary of interest are and whether or not they should belong to the same topic.

Raw Topic	Survey Response	Depth 2 Result
brown old long hello thing green yellow okay yes whole white red	brown green yellow white red	brown green yellow white red
chew tuck stop put let lay touch follow hello	chew tuck lay touch	(no topic)
seven ten six three next four thing cant nine five eight hello	seven ten six three four nine five eight	three four nine five eight

Figure 9. Raw topics before topic refinement in the leftmost column. Survey response of humans refining the topic from the raw topic in the middle column. Results of topic refinement from the raw topic using a maximum search depth of 2 in the ConceptNet graph [20].

The words in these topics are further refined by remaining in the topic only if they are connected to each other in the ConceptNet graph by a number of nodes that is less or

equal to the length of a given search depth. Best results were found with a search depth of 2. This topic refinement technique performed similarly to manual topic refinement when initially given a rough set of words that somewhat imprecisely model a topic. It successfully extracted topic words from the raw topic when there seemed to be a clear underlying topic in the raw topic words (see Figure 9).

Analysis on Tinkerbooks Data

Analysis was initially done on the Tinkerbooks data collected by Angela Chang [19]. The data consisted of dialogue transcriptions between parents and children reading an interactive story about Baby Duck taking a bath in a pond, and there were 8 conversations total. The conversations were carried out in a supervised manner, where the researcher went to the houses of the parents to make the recordings. This ensured that the parents and children were actively engaging with the e-book. This is in contrast to the manner of data collection for the other version of the e-book [20] that was sent home with children and collected data by recording dialogue as the children used the tablet at home, without any supervision of a researcher. Throughout the conversations, parents would read the words of the book aloud to the child and often ask questions to the child to ensure comprehension. My goal was to evaluate the conversations and figure out how the sentiment changed and evolved as the conversation progressed. I hypothesized that, as parents read the book to their children, they would get sidetracked from the content of the book itself and discuss topics related to, but not specifically about, the book. For example, if the book discussed Baby Duck taking a bath, I hypothesized that a parent would start a conversation about how the child just completed his bedtime routine and brushed his teeth. The sentiment of such a conversation could possibly be angry if the parent was scolding the child for not being cooperative while getting ready for bed earlier that evening.

Emotions and Sentiment

In order to evaluate a score for a particular emotion during the conversation, I used ConceptNet common sense reasoning through the Divisi package [4]. The emotions that I explored are enumerated in Figure 10. Words in the column labeled “Positive” represents concepts in ConceptNet that express a positive sentiment, while the columns labeled “Negative” represents concepts in ConceptNet that have a negative sentiment. Words in the same row of Figure 10 represent the positive and negative concepts of the same emotion. For example, “happy” and “sad” represent the two shades of the same emotion, but “happy” has a positive sentiment and “sad” has a negative sentiment.

Positive	Negative
Happy, joy, love	Sad
Empathy	Apathy
Hope	Despair
Agree	Disagree
Compliment, praise	Criticize
Calm	Concern, fear, anxiety
Surprise, Awe	
Satisfaction	
	Frustration, anger
	Annoyance

Figure 10. Positive and negative sentiments used for analysis of dialogue.

In order to evaluate how the transcript of a given dialogue scores for a particular emotion (e.g. “happy/sad”), vectors must be created to represent the dialogue and the emotion. Divisi uses ConceptNet to come up with a vector representation of a word, and the vector is of length 150.

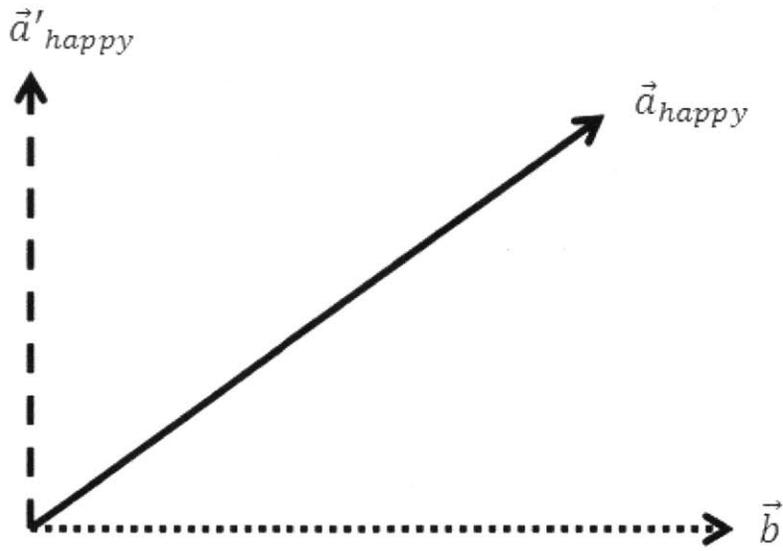
The dialogue is split into sections for analysis. In the case of the Tinkerbooks data, the dialogue is split by page so that all conversation dialogue that was spoken while the parents and children were on a certain page is put into a single section for analysis. This was done because there is a different event of the story on each page, which would subsequently prompt a different dialogue. The dialogue must be evaluated in sections because the conversation is likely to evolve and change while reading the e-book. Evaluating the dialogue as a whole may average out sentiments, preventing the analysis from showing the evolution of an emotion from page to page. We wanted to be able to see a spike in the “happy” emotion on a certain page that particularly engages the child and makes him or her very gleeful. Thus, sentiment analysis needed to be done on a page by page basis.

For each page, a vector \vec{v} is constructed by Divisi based on the sum of the vectors of individual words. This vector is compared with a vector representing a particular emotion in order to calculate a score for that emotion. However, in order to find a vector representation of an emotion, the emotion must be isolated from other emotions, and balanced by its positive and negative components. This is because the concepts “happy” and “sad,” while seemingly opposite, are actually very similar because they are the positive and negative sides of the same emotion. A concept that would truly be the opposite of “happy” would be a completely unrelated concept such as “sofa.” The vector \vec{a} represents the Divisi vector computed for the positive concepts of a particular emotion, called \vec{p} , minus the Divisi vector computed for the negative concepts of that emotion, called \vec{n} . In order to extract the emotionality of a concept such as “happy/sad,” the positive and negative sides of the concept must be separated from the overall emotionality. The \vec{p} vector is constructed from

the positive concepts of the emotion (e.g. happy, joy, and love for “happy/sad”) while the \vec{n} vector is constructed from the negative concepts of the emotion (eg. sad for the “happy/sad” emotion) Overall emotionality is defined as a vector \vec{b} , which is the sum of all emotion concepts, positive and negative. Figure 11 shows the \vec{b} vector as a sum of the positive and negative components of many emotions including “happy” emotion and “annoy” emotion.

To distinguish the particular sentiment of the emotion we are interested in analyzing, it is necessary to find the vector rejection of \vec{a} from \vec{b} , denoted as \vec{a}' for a given emotion. This vector separates the particular emotion being analyzed from all emotions represented by the overall emotionality vector. The sentiment score for that emotion is then simply the dot product between \vec{a}' and \vec{v} .

The sentiment score is a reflection of how highly the words in the page, represented by \vec{v} , scores for a certain emotion. A page containing many words like “wonderful,” “exciting,” and “great” should score highly for the “happy” emotion. Figure 11 visualizes and shows how these vectors would be calculated for the “happy” emotion.



$$\vec{v} = \sum_{i=1}^N \text{assocmat}(\text{term}_i)$$

where term_i is a word in a page of N words, which also exists in ConceptNet. The function $\text{assocmat}(\text{term}_i)$ generates the association vector for term_i using Divisi.

$$\vec{a}_{happy} = \vec{p}_{happy} - \vec{n}_{happy}$$

$$\vec{b} = \vec{p}_{happy} + \vec{n}_{happy} + \vec{p}_{annoy} + \vec{n}_{annoy} + \dots$$

where \vec{b} is a vector that represents all emotions in the space.

$$\vec{a}'_{happy} = \vec{a}_{happy} - (\vec{a}_{happy} \cdot \hat{b})$$

$$\text{score}_{happy} = \hat{a}_{happy} \cdot \vec{v}$$

Figure 11. Evaluation of sentiment score for the emotion “happy”

Sentiment Analysis on Tinkerbooks Data

Sentiment scores for each emotion and each page were calculated for every individual conversation (see Figure 12) and averaged across all conversations (see Figure 13). The results from a single conversation show that there is not much variation in sentiment throughout a conversation. That is, each line in Figure 12 represents the trajectory of an emotion throughout a particular conversation in the Tinkerbooks dataset, and each line is relatively flat. This suggests that there is not great fluctuation in sentiment throughout the conversation. This trend was seen across all conversations in the dataset. The sentiment scores for each emotion, when averaged across all conversations in the dataset, were similarly static.

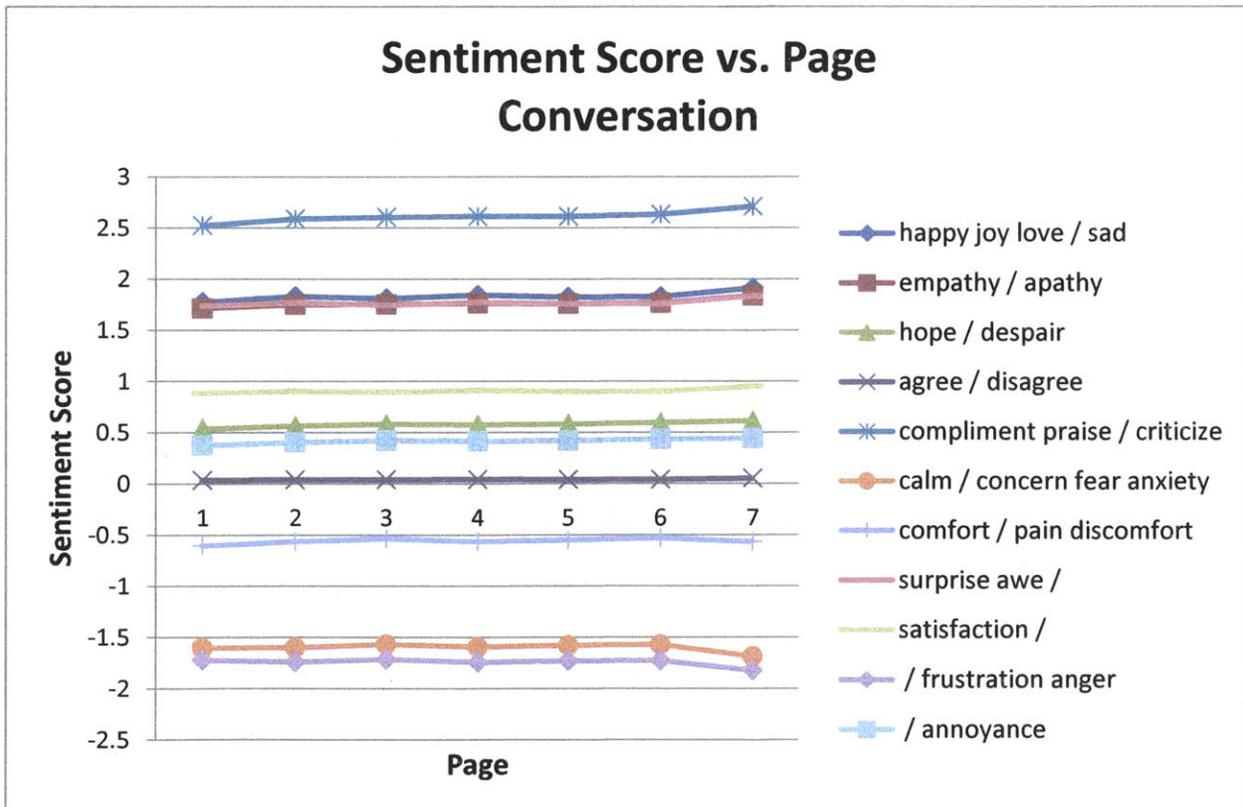


Figure 12. Sentiment scores per page for a single conversation in the Tinkerbooks dataset. There is little fluctuation of sentiment throughout the conversation.

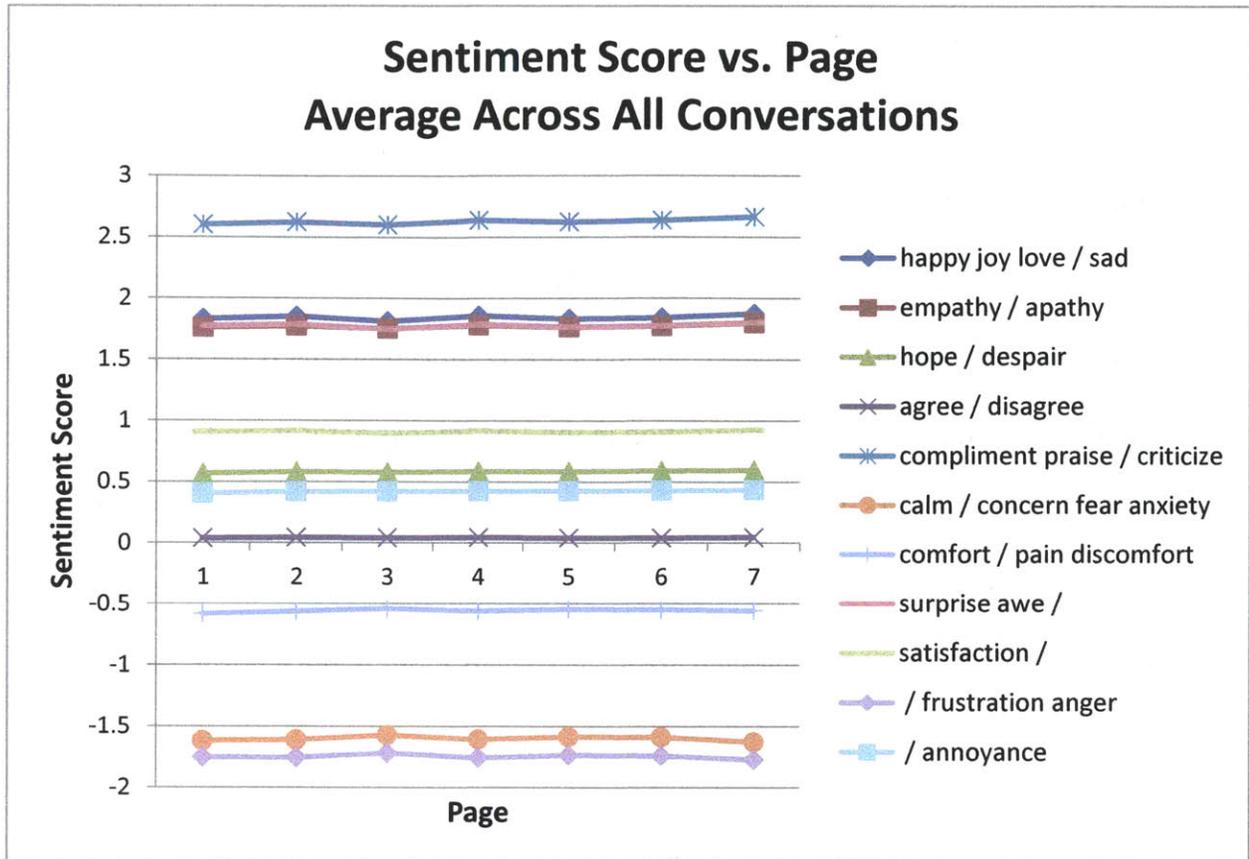


Figure 13. Sentiment scores per page, averaged across all conversations in the Tinkerbooks dataset. Similar to the trend in a single conversation, there is little fluctuation of sentiment throughout the averaged conversations.

The flatness of the trajectories of the sentiment score is most likely due to the lack of vivid language used in the e-book and dialogue. Children’s books are very simple in their language and plot, and thus would not use descriptive and highly emotive language. Similarly, parents may not be using particularly emotive language when discussing the book with their children. Qualitative analysis of the dataset shows that most of the dialogue in the conversation transcripts is made up of the parents reading lines of the e-book, or simply rephrasing the words of the e-book to ensure that the child comprehends the story. Since the conversation does not veer far from the original story, and the story itself is very simple, there is a lack of strong emotion and sentiment.

Magnitude Discrepancies for Different Emotions

Figures 12 and 13 reveal differences in the magnitude of certain emotions relative to other emotions. For example, the lines for “happy/sad” and “compliment/criticize” are particularly higher than the lines for “agree/disagree.” Normalization of the vectors was performed to account for this when calculating scores, but the difference in magnitude persists. Since the sentiment vector \vec{a}' is partially constructed from the difference between the positive and negative components of an emotion, it is possible that having too many concepts in the positive or negative sides of an emotion may be skewing the magnitude of the resulting score. In order to remove this factor, the sentiment scores were recalculated considering just one concept (either positive or negative) per emotion. That is, instead of representing the happy emotion as the difference between the sum of the “happy,” “joy,” and “love” vectors and the “sad” vector, it would be represented by only the “happy” vector.

Figure 14 shows the results of finding sentiment scores with just a single concept representing each emotion. Positive emotions are represented in the legend by a slash after the concept name, while negative emotions are represented by a slash before the concept name. As expected, most positive concepts have a positive score and most negative concepts have a negative score. However, “criticize” and “annoy” are negative concepts and have positive scores. This causes us to cast some doubt on the significance of the relative magnitude of the vectors returned by Divisi for a given word. It seems that when only considering a single concept or emotion, Divisi can accurately reflect relative fluctuations in sentiment. However, the concept vectors returned by Divisi are not as accurate in reflecting

the magnitude of one concept relative to other concepts in the space. Thus, Divisi uses ConceptNet to find the trajectory of a sentiment, but does not do as good of a job with the relative magnitude of various sentiments.

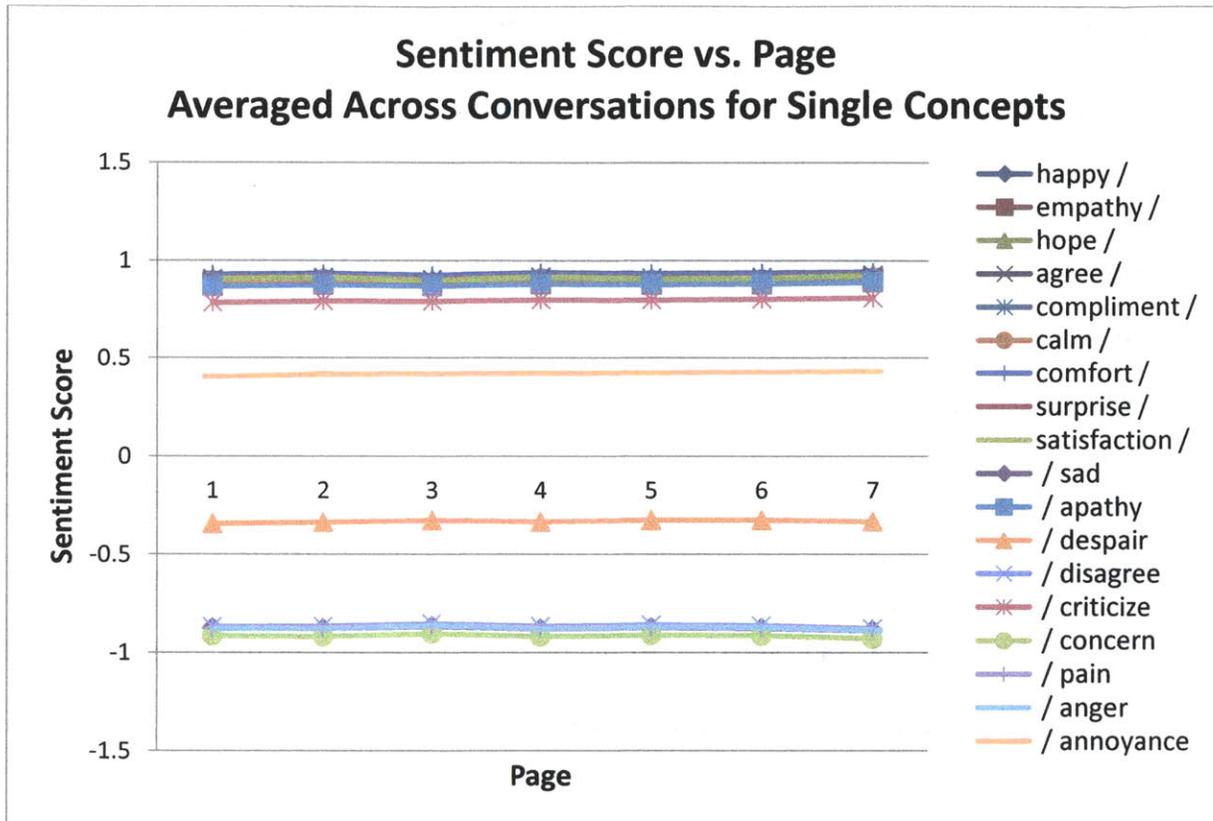


Figure 14. Average sentiment score vs. page using a single concept to represent an emotion.

In order to take a deeper look into the magnitude discrepancies for various emotions, I analyzed how a single word from the Tinkerbooks dataset would score for an emotion represented by a single concept. I also analyzed what words scored highest for an emotion in order to ensure, for example, words like “hate” and “rage” scored highly for the “anger” emotion. This analysis shows that words scored on average lower in magnitude for concepts like “despair” and “annoyance” and higher for concepts like “happy” and “compliment.” However, the top words for a concept like “happy” were “happy,” “smile,” “like,” “fun,” and

“love.” The top words for a concept like “despair” were “hurt,” “difficult,” “wrong,” “ignore,” and “disappoint.” This shows that although the relative magnitudes of the scores of words for a certain emotion are specific to that emotion, the scoring of words for that emotion reflects the nature of the emotion. The top words for an emotion are reasonable and make sense, but the score relative to another emotion does not carry as much significance.

It is also important to note that after the first few top scoring words for a particular emotion, the next set of words that score highly do not make as much sense for that emotion. Words like “network” and “prove” are among the top 50 scoring words for the “happy/sad” emotion. While the top scoring words make sense for an emotion, it seems that other words that do not make as much sense can score almost as highly for that emotion and skew the sentiment score. The number of words per page is very high compared to the number of emotive words that should score high for a particular emotion. If words other than the emotive words are also receiving high sentiment scores, the overall sentiment score for a page cannot truly reflect the sentiment of that page. This is likely contributing to the lack of fluctuation of sentiment scores because both emotive and non-emotive words can score highly for a particular emotion.

Using Topic Words Instead of Full Dialogue

Due to the lack of fluctuation of sentiment throughout the conversations between parents and children in the Tinkerbooks dataset, I decided to try to use the topics of each page to calculate the sentiment scores, rather than the full dialogue of the page. I suspected

that there was not a great deal of emotion in the original results because all words in the dialogue were weighted equally and contributed to the sentiment score. However, this meant that many words that do not carry much sentiment were included in the analysis. In order to mitigate this issue, I used the topic words of a sentence to calculate the score for an emotion, rather than all of the dialogue. This meant that a page in which the dialogue was about a bath would have topic words like “bath,” “clean,” and “dirty.” These topics words would then be used to calculate the sentiment score.

Topic Summary	Topic Words
interact with water	babyduck pond swims dive make swim splash splashes water dives swimming wanna splashing cross finger mouse backwards side put
tablet interactions	duck wanna move back head words change push don walk feet touch babyduck start making click arrow tap yeah
choose color	quack yellow duck colour brown grey red make change wanna word tap good find choose hold job page pick
baby wants to be clean/dirty	clean dirty likes babyduck mud water back make wanna put gonna yellow page finger yeah uh drag coming leave
mom wants baby to be clean	mamaduck babyduck good job wanna bath time dirty mama mommy press yeah ll today water clean big drag page
waddle/move	pond babyduck waddles waddle slowly make waddling swim wait fast quickly quack drag finger called slow put walking touch

Figure 15. Topic words defined by the MALLET toolkit and the associated topic summary.

The topic words per page were found using the MALLET toolkit [21] to do Latent Dirichlet Allocation (LDA) topic modeling [18]. I found that the best distribution of topics words to topics is when 6 topics were used for topic modeling. Figure 15 shows the topic words for 6 topics that were manually assigned a summary phrase for ease of reference. MALLET uses a unique topic code to identify the topic, but the topic summaries that were manually determined are shown here to more intuitively correlate with the topic key words. The topic summaries were written based on the theme, event, or behavior described by the

topic words. For example, the topic with color words “yellow,” “brown,” and “grey” as well as the words “change,” “choose,” and “pick” suggest that this topic is about choosing the color of the duck. Thus, the topic summary for these topics words was chosen to be “choose color.”

The topic modeling toolkit also rates how much each topic correlates with each page of the conversation. In Figure 16, the actual topic of each page is compared to the predicted topic summary as outputted by MALLET. The entries for the actual topic are summarizations of the words of the e-book on the corresponding page. The entries for the predicted topic are the topic summaries of the topic that MALLET said most correlated to that page. Figure 16 shows the impressive accuracy of the predicted topic in comparison to the actual topic of each page.

Page	1	2	3	4	5	6	7
Actual Topic	Start	Babyduck time for your bath, says Mama Duck	Set duck color, babyduck says quack	Put babyduck in water or mud	Babyduck waddles to pond/mud	Babyduck splashes/swims in pond/mud	Good job, says Mama Duck
Predicted Topic	tablet interactions	mom wants baby to be clean	choose color	baby wants to be clean/dirty	waddle/move	interact with water	mom wants baby to be clean

Figure 16. MALLET toolkit accurately predicts the topic of each page. The topic words of each topic are used for sentiment analysis rather than the full dialogue of a page.

Instead of using the text of the dialogue transcription for a given page to calculate a sentiment score, the topic words of the predicted topic for that page will be used. This will ensure that only meaningful words that are significant to the events of the page will be used

for sentiment analysis. Figure 17 shows the sentiment score when using the topic words per page instead of the full dialogue. For most of the pages, there is a lack of fluctuation in emotion as also seen in previous analysis. This means that there is truly not much emotion in

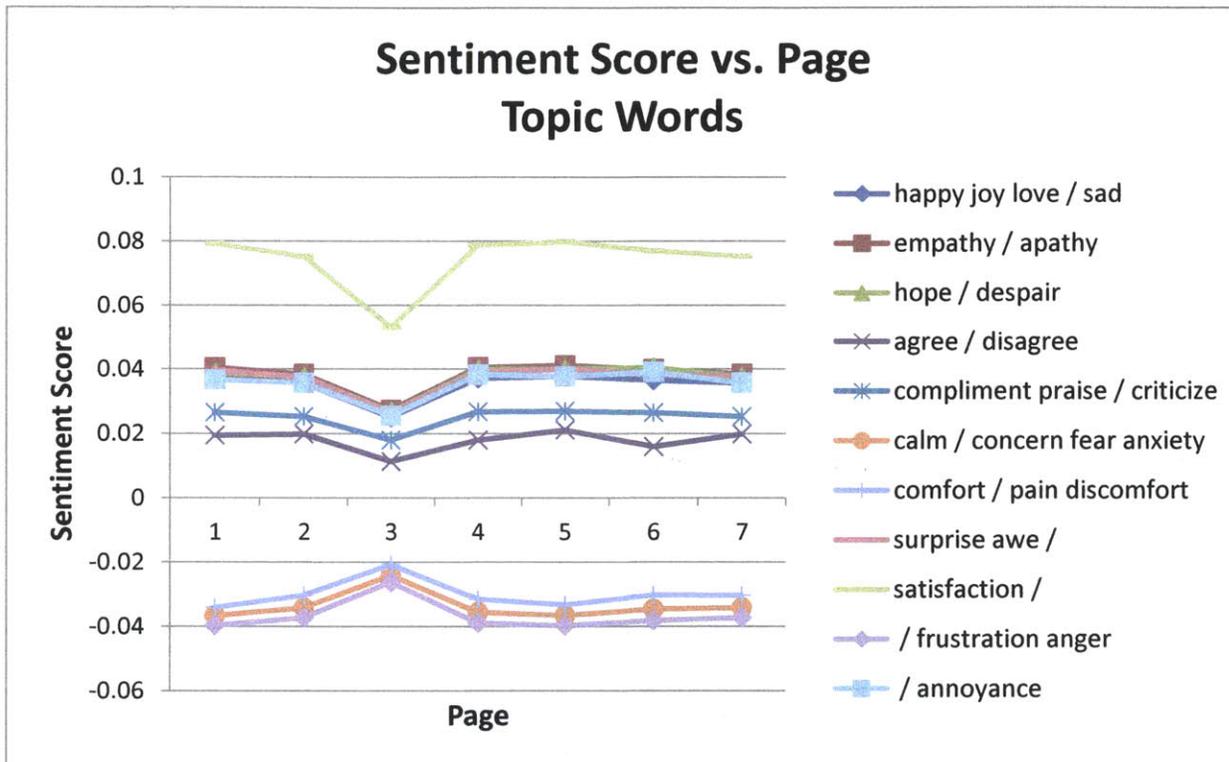


Figure 17. Sentiment score vs. page when using the topic words to calculate sentiment scores. There is little fluctuation in sentiment scores with the exception of a dip in all emotionality on page 3, which discusses changing the color of the duck in the e-book.

the book and dialogue because both the dialogue text and dialogue topics throughout the book failed to yield varying sentiment scores as the conversations progressed. There is a dip for all emotions at page 3, which is the page where the children choose the color of the duck. This is likely because of the lack of emotion for color words. In order to see more emotion for the topic words that correspond to the “choose color” topic summary, a system like the one used by Colorizer would be necessary to match emotions and colors. Such a

system could plug into the current sentiment score calculations and contribute to the sentiment score for color words. This would mitigate the lack of emotion seen on page 3.

Conclusion: Tinkerbooks Results

Analysis of the trajectory of emotions using the full dialogue text and topic words showed very little change in emotion throughout the conversations between parents and children reading the e-book. This is likely because of the simple wording of the book itself, as well as the fact that parents do not diverge much from content of the book in their dialogue. That is, most of the dialogue is made up of parents reading the text of the book, asking the child to repeat the text of the book, instructing the child how to interact with the tablet, and simply rephrasing the events of the book to make sure their child is following the story. None of these aspects of the dialogue contribute greatly to the emotionality of the text and thus, the sentiment scores throughout the conversation remain relatively flat.

The Tinkerbooks dataset I used was collected by supervising conversations between parents and children reading an e-book so that they were actively interacting and engaging with the tablet. The alternative version of the book that was analyzed [20] had data collected in an unsupervised manner where children would take the e-book home with them and were expected to read it with their parents. The result of these recordings had even less dialogue than the supervised conversations that I used. Children would often read the e-book without a parent so no dialogue was recorded other than sounds and words pronounced by the e-book itself. This study was also carried out in a rural part of Georgia, where the participants of the study were less familiar with technology. Consequently, it

appears that many recordings consist of the parents themselves interacting with the tablet to test out the interactions and explore the new technology (D. Nunez, personal communication, May 7, 2013).

The recordings from Georgia were not transcribed, making it impossible for me to use for analysis. However, as mentioned above, the recordings had little to no conversation and mostly picked up background noise or the sounds of the tablet itself. This suggests that participants of the study are not interacting with the tablet in any way that is substantially different from a normal book. In fact, there may be less interaction between parents and children due to the e-book because parents view it as a substitute for them reading aloud to the child, hence the very silent recordings when the tablet is in use. The e-book is not fostering much extra conversation, and the conversation it does invoke is not particularly emotive or informative. There is no evidence from my results, or the data collected from the alternative e-book, that the interactive Tinkerbook is any better than a normal book at promoting a child's literacy, cognitive, and language skills by fostering additional dialogue.

In order to see a more interesting trajectory of sentiment throughout a conversation, the dialogue needs to cover more interesting topics that can draw out more emotion. For this reason, the next dataset I decided to analyze was the Switchboard Corpus [22] which consists of telephone conversations between adults, who talk about a variety of topics that are either assigned or spontaneous.

Analysis on Switchboard Corpus

Analysis on the Tinkerbooks dataset did not show much fluctuation in sentiment when analyzing the full dialogue text or simply the topic words of the dialogue per page of the e-book. This is possibly due to the simple nature of the dialogue and lack of emotion throughout the conversations, since the e-book is targeted towards children and is therefore very simplistic. Also, parents talking to children will speak simplistically so that the children can understand what the adults are saying to them. For this reason, I decided to consider a corpus of dialogue between adults, who would be much more likely to speak to each other with more emotion and about more varied and complex topics.

The corpus I used was made up of transcripts of dialogue that made up the Switchboard corpus [22] collected by the University of Pennsylvania from late 1999 to early 2000. Two hundred fifty of the audio conversations recorded as a part of the Switchboard corpus were transcribed [23] and make up the dataset I used for my analysis. Participants were asked to call a switchboard where they would be connected to another random participant in the study. Once connected on a call with each other, the participants were asked to speak for at least 5 minutes and were prompted with a topic to talk about. The topics ranged from “What’s your favorite vacation you’ve ever been on?” to “Is flag burning protected under the 1st Amendment?” Participants often used this topic or question to get their conversation started but usually diverged from this topic as the conversation naturally progressed.

Sentiment Analysis on Switchboard Data

I performed a similar sentiment analysis on the Switchboard dialogue transcripts as I did on the Tinkerbooks dataset. Each conversation was split into even segments to parallel the pages of the Tinkerbook. The text in each of these segments was used to calculate a sentiment score with all of the emotions that were also analyzed in the Tinkerbooks dataset (see Figure 10 for a summary of these emotions). I found that there was slightly more fluctuation in sentiment in the conversations from the Switchboard corpus, and the difference in magnitude of the scores between emotions persisted (see Figure 18).

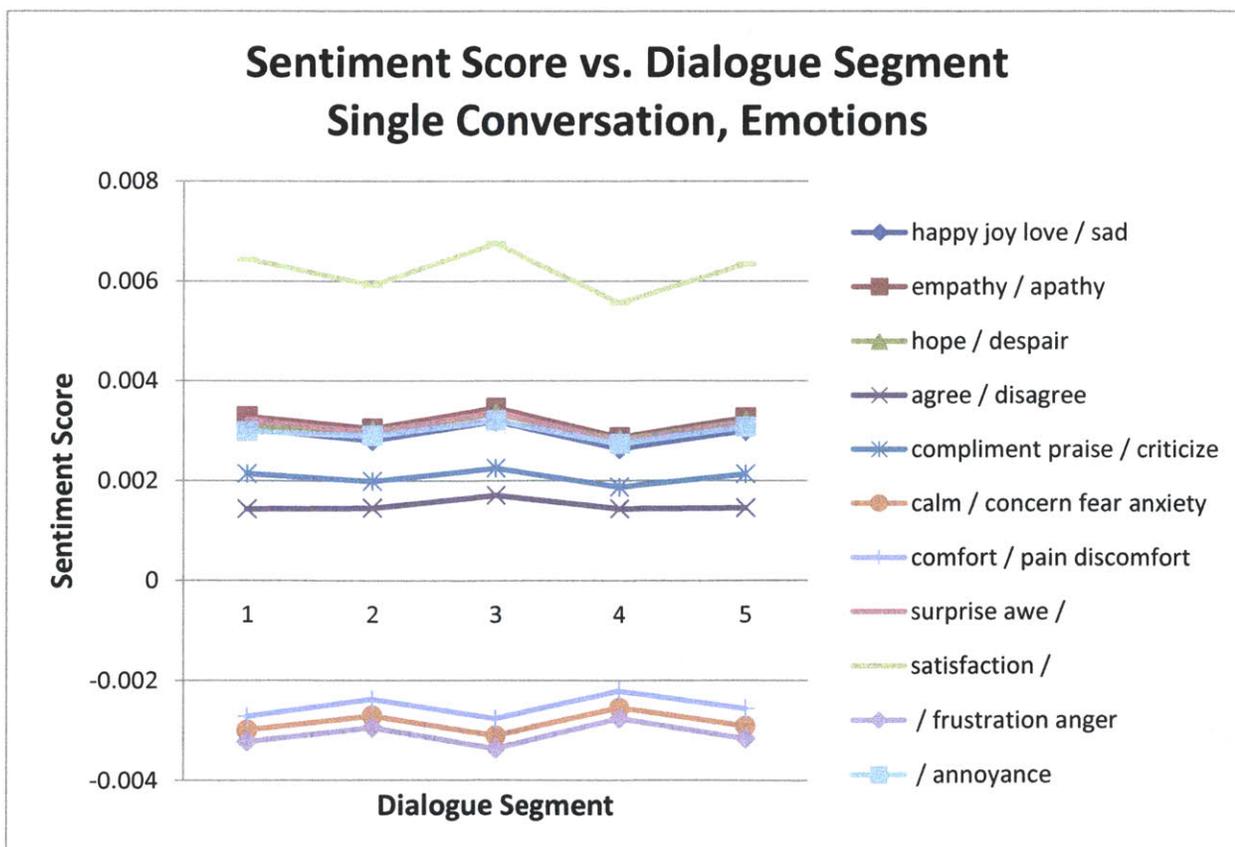


Figure 18. Sentiment score vs. dialogue segment for a conversation in the Switchboard corpus. Sentiment scores are calculated for various emotions.

The fluctuation in sentiment, while slightly more pronounced in conversations in the Switchboard corpus than in the Tinkerbooks dataset, was still not substantial. Analysis of sentiment scores for single words in the Switchboard corpus revealed results similar to the parallel analysis done for the Tinkberbooks dataset. Individual words like “agree,” “thankful,” “thank,” and “allow” are words that score high for the “agree/disagree” emotion. However, words like “okay “ and “ought“ score relatively high for that emotion as well, suggesting some inaccuracy of the vectors returned by ConceptNet that are used to calculate the sentiment score for a particular emotion based on the text of the transcripts.

The sentiment scores continued to remain relatively flat for the Swithboard corpus, although not as flat as they were for the Tinkerbooks dataset. The analysis of the sentiment scores for individual words suggests that ConceptNet sometimes ranks words that are not very related to an emotion highly for that emotion. This prevents the truly emotive words from standing out when calculating the sentiment scores, and consequently smoothes out possible fluctuations in sentiment scores.

Topic Analysis of Switchboard Data

Because both emotive and non-emotive words had the potential to score highly for a particular emotion, and the ration of emotive to non-emotive words was low, the sentiment scores for emotions did not have much variation where there truly may have been some emotion. However, if we consider topics instead of emotion, the ratio of topic related words and non-topic related words is most likely higher than the ration of emotion to non-emotive

words. This is because the participants in the conversations are often speaking specifically about an assigned topic, or a spontaneous topic. This analysis can be done by replacing the emotions with words that describe a topic. An emotion was previously defined by its positive and negative components. The “happy/sad” emotion was defined by the positive components (“happy,” “joy,” and “love”) and the negative components (“sad”). A topic simply needs to be identified by a set of topic word because topics, unlike emotions, do not have positive and negative components that need to be considered. With this small change, the analysis can be run by replacing emotions with topics and calculating a “sentiment score.” The sentiment score is now defined as how highly a segment scores for a particular topic, rather than an emotion. The topic words were found by using the MALLET toolkit to find 15 topics. The topics were manually refined into 19 topics and the topic words used for analysis were selected from topic key words of the final 19 topics.

The results of this analysis (see Figure 19) show that the sentiment score trajectories of the topics are all within the same magnitude range and do not have the same differences in magnitude between topics as the sentiment analysis had between emotions. The sentiment score trajectories for topics also have the same general shape as the sentiment trajectories for emotions. This is due to the fact that many non-emotive words are also non-topic related and are still influencing the score.

Figure 19 also shows that no topic particularly stands out for the conversation that was analyzed. In reality, the *sports*, *money*, and *family* topics were discussed in the conversation. This distinction cannot be detected from the sentiment scores calculated for that conversation.

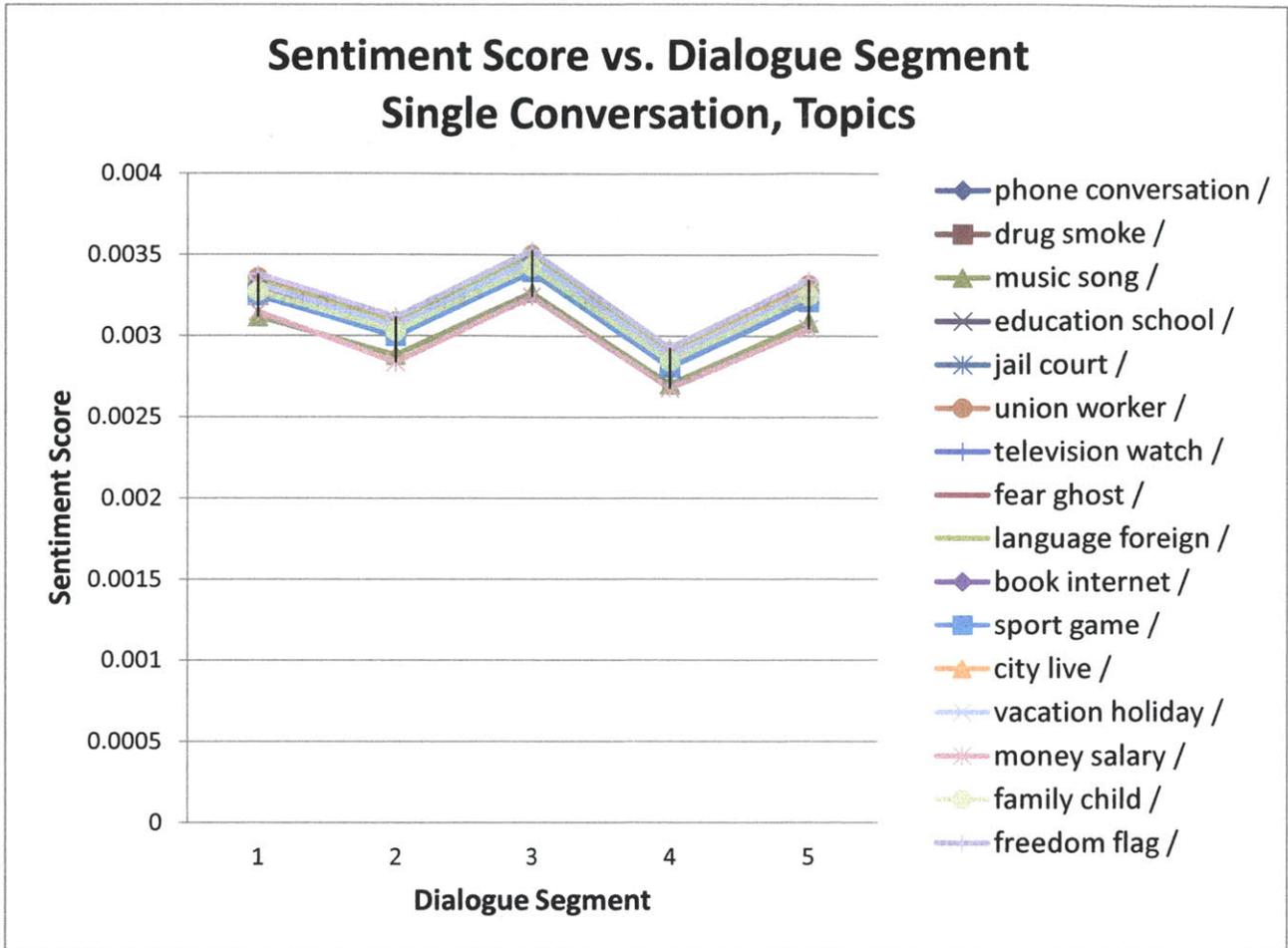


Figure 19. Sentiment score vs. dialogue segment for a conversation in the Switchboard corpus. Sentiment scores are calculated for various topics rather than emotions.

Topic Analysis Using SVM

In order to better detect the topic of a conversation in the Switchboard corpus, a classification method that did not use ConceptNet or any commonsense reasoning was explored. The library LIBSVM [24] was used to implement Support Vector Machine (SVM) classifications [25]. In particular, LIBSVM was used to implement the SVM formulation C-Support Vector Classification (C-SVC) [26].

Binary classification was used for each topic. A conversation would be classified as +1 if the topic was discussed during the conversation and as -1 if the topic was not discussed. Two sets of feature vectors were tested. In one model, the feature vectors were made up of word frequencies for 14 words that were high indicators of the topic. In the second model, the feature vectors were made up of the sentiment score of that conversation for a given topic.

The 14 words that were chosen for the word frequency feature vectors were found by using the topic words returned by the MALLET toolkit, as found during the initial topic analysis of the Switchboard corpus described previously. ConceptNet sentiment score feature vectors were calculated as the sentiment score of the entire text of a conversation for the topic being analyzed. The topics of the 250 conversations in the corpus were manually designated. Often a single conversation covered many topics. 188 random conversations made up the training set and 62 random conversations made up the test set.

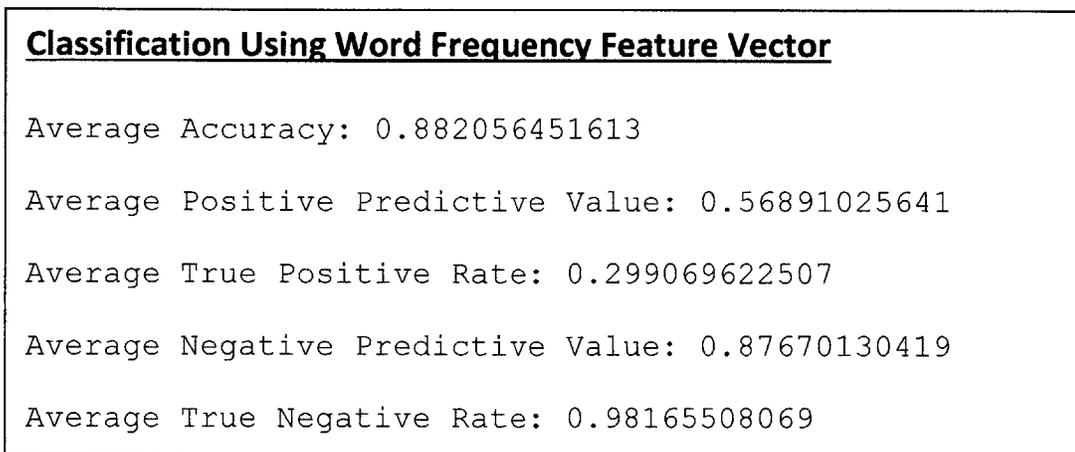


Figure 20. Average performance metrics of classification of each conversation in the test set using word frequency feature vectors.

Figure 20 shows the results of using the LIBSVM library with feature vectors defined by the word frequencies of 14 key words per topic. Figure 21 shows the results with feature vectors defined by the sentiment score of the conversation for each topic. The results were averaged across all conversations in the test set.

<u>Classification Using Sentiment Score Feature Vector</u>
Average Percentage: 0.828629032258
Average Positive Predictive Value: 0.0
Average True Positive Rate: 0.0
Average Negative Predictive Value: 0.828629032258
Average True Negative Rate: 1.0

Figure 21. Average performance metrics of classification of each conversation in the test set using sentiment score feature vectors.

The results of the model using the sentiment score feature vector indicate that every conversation was classified as -1 for every topic. That is, the model predicted there was no topic discussed in any of the conversations in the test set.

Conclusion: Switchboard Results

Sentiment and topic analysis using sentiment scores, calculated using Divisi, did not show any emotions or topics standing out for any conversations. Sentiment scores for emotions remained relatively flat, as they did in the analysis of the Tinkerbooks data.

Sentiment scores for topics were similar regardless of the topic, so no topic scored particularly higher or lower than any others in any of the conversations.

Analysis of how individual words in the corpus score for emotions revealed the same results as they did for the Tinkerbooks dataset. The highest scoring words for an emotion made sense, but many top scoring words also did not make sense. This is a likely contributor to the flatness of the sentiment score trajectories and the lack of distinction of any emotion or topic that may be prevalent in the conversation.

The LIBSVM library was used to evaluate the conversations based on two different types of feature vectors. One was based on word frequency of words related to the topic being classified. The other was based on the sentiment score of a conversation for the topic being classified. The model using the word frequency feature vector performed substantially better than the model using the sentiment score feature vector, which always classified a -1 for the topic. Both models had accuracy above 80% because of the prevalence of -1 classifications due to the sparseness of topics amongst conversations. There were 250 conversations and 16 topics, but a conversation would usually only be classified as +1 for one to three of the topics. Thus, the challenge for these models was to correctly identify the sparse +1 classifications for a topic. The word frequency model classifies quite a few false negatives, thus yielding a true positive rate of only about 30%. It does not wrongly classify too many positives, yielding a positive predictive rate of about 57%. The sentiment score model did not identify any +1 classifications. Thus, its accuracy was the same as its negative predictive value and its true positive rate was 100%.

Conclusion

Two datasets of dialogue transcripts, the Tinkerbooks dataset and the Switchboard corpus, were analyzed to determine salient sentiments and topics throughout the conversations. The first dataset, known as the Tinkerbooks dataset, came from transcripts of parents and children reading an e-book about a duck taking a bath. Sentiment scores were calculated for a variety of emotions using Divisi, a Python library that uses ConceptNet to construct concept vectors. These scores were calculated for dialogue spoken on each page of the e-book, but the sentiment score trajectories for all conversations were very flat. The scores were also calculated for the topic words, as found using an LDA topic modeling toolkit, for each page rather than the full text of dialogue for the page. The sentiment scores for the conversations based on topic words also remained relatively flat.

Further investigation into the sentiment scores of individual words for a given emotion revealed that sentiment scores would be high for many words that were obviously related to the emotion. However, many other seemingly unrelated words also scored relatively high for the emotion. This indicated that the vectors returned by Divisi and ConceptNet did not exactly reflect the correlation between a word and an emotion. The flatness of the sentiment score trajectories can be somewhat attributed to this, because many words contributed a skewed score to the overall sentiment score.

Another component that may have led to the flatness of the sentiment score trajectories is the fact that the dialogue in the Tinkerbooks dataset was very simplistic. Parents spoke in an uncomplicated and straightforward way to ensure the child's

comprehension. The parents usually read the simple words of the book and rephrased them to help the child understand the plot of the book, rather than diverging into a side conversation that could be emotional and dynamic. Thus, the lack of sentiment fluctuations seen in the sentiment analysis may simply be a byproduct of the lack of sentiment in the dialogue itself.

The lack of emotionality in the Tinkerbooks dataset has broader implications for early childhood literacy. Reading aloud to children promotes language, literacy, and cognitive skills (Allington & Cunningham, 1996; Hall & Moats, 1999; Holdaway, 1979). Using an interactive e-book such as the Tinkerbook did not reveal any substantial emotionality in the conversation. The parents and children read the e-book in the same way they would read a regular book, and simply discussed more instructions on how to interact with the tablet. Thus, there did not seem to be much gained from the extra conversation that was hypothesized to occur due to using an e-book because, in reality, there was very little extra conversation between the parents and children.

The second dataset, known as the Switchboard corpus, came from transcripts of 5 minute phone conversations between adults who were asked to speak about various topics. This dataset was analyzed with the hope that adults speaking to each other would have more emotionality than parents reading with children. However, results showed that the sentiment scores for various emotions were flat, similar to the results of the Tinkerbooks dataset. Topic modeling was performed to extract the main topics of the conversations, that were further refined manually. The topic words were then used instead of emotions and the sentiment score was redefined to be the score between the dialogue and the topics, rather

than the emotions. However, the sentiment score trajectories for the topics were similar for all topics, and no salient topic stood out from the rest of the topics in any of the conversations.

A c-SVC classifier was then used to classify the topics of a conversation. All conversations were classified as discussing one or more of the topics found from the topic modeling. The classifier used a feature vector based on word frequency or based on the sentiment score for that topic. The word frequency vector was made up of the frequency of several topic key words for the topic being classified. Results showed that the model using the sentiment score performed poorly and did not detect a topic for any conversation. This is likely due to the fact that earlier analysis showed that sentiment scores were similar for all topics in a given conversation, making it difficult for the classifier to use this score to distinguish a topic for a conversation. The word frequency model performed substantially better than the sentiment score model. It was able to identify topics for each conversation, but predicted less than 1/3 of the correct topics. However, this was an improvement on the sentiment score model that could not predict any of the topics.

A c-SVC classification model using a word frequency feature vector performed better than the sentiment score feature vector model. This suggests that it is better to use word frequency to predict topics than the commonsense reasoning of Divisi and ConceptNet.

Future Work

Additional work can be done to make a deeper investigation into why the scores calculated using Divisi for individual words did not completely reflect the relation of the word to the emotion or topic. It is possible that the space in which calculations are done must be further refined to consist of only emotions so that differences between emotions are more pronounced and sentiment scores may be more varied. Emotions, while having semantically different meanings, are very similar concepts. “Happy” and “sad” seem like opposites but are actually two shades of the same underlying emotion. Further analysis in such a modified space may yield better results.

The c-SVC classification performed using LIBSVM could have improved performance for the word frequency vector model. Currently, the model tended to favor not classifying a topic for a conversation. This still results in fairly good accuracy due to the sparseness of topics across all conversations. However, this imbalance can be configured into the classification model to promote the model classifying more topics for a conversation. There is currently a very large number of false negatives classified by the model, which is a result of this imbalance. Correctly the imbalance would ideally convert many of these false negatives into true positives, meaning the model would more often correctly identify topics for conversations.

Acknowledgements

This work was partially supported by grants 1117584 and 1138986 awarded by the National Science Foundation. I would like to thank Rob Speer and the Personal Robots group at the MIT Media Lab for their advice and collaboration. I would also like to thank to my supervisor, Catherine Havasi, for her guidance throughout this project.

References

- [1] Speer, Robert, Catherine Havasi, and Henry Lieberman. "AnalogySpace: Reducing the dimensionality of common sense knowledge." *Proceedings of AAAI*. 2008.
- [2] Liu, Hugo, and Push Singh. "ConceptNet—a practical commonsense reasoning toolkit." *BT technology journal* 22.4 (2004): 211-226.
- [3] Havasi, Catherine, et al. "Digital intuition: Applying common sense using dimensionality reduction." *Intelligent Systems, IEEE* 24.4 (2009): 24-35.
- [4] Speer, Rob, Kenneth Arnold, and Catherine Havasi. "Divisi: Learning from Semantic Networks and Sparse SVD." *Proc. 9th Python in Science Conf.(SCIPY 2010)*. 2010.
- [5] Oliphant, Travis E. *A Guide to NumPy*. Vol. 1. USA: Trelgol Publishing, 2006.
- [6] Geus, Roman, Daniel Wheeler, and Dominique Orban. "Pysparse Documentation." (2008).
- [7] Rohde, Doug. "SvdLibC." (2007).
- [8] Speer, Robert H., et al. "Finding your way in a multi-dimensional semantic space with luminoso." *Proceedings of the 15th international conference on Intelligent user interfaces*. ACM, 2010.
- [9] Havasi, Catherine, Robert Speer, and Justin Holmgren. "Automated color selection using semantic knowledge." *Proceedings of AAAI CSK, Arlington, USA* (2010).
- [10] Smedt, T. D., and Bleser, F. D. *Nodebox Prism*. NodeBox web site. <http://nodebox.net/code/index.php/Prism>. 2010.
- [11] Munroe, Robert. "Color survey results". XKCD blog. <http://blog.xkcd.com/2010/05/03/color-survey-results>. 2010.
- [12] János Schanda. *Colorimetry*. Wiley-Interscience, 2007. p 61.
- [13] Havasi, Catherine, et al. "The Glass Infrastructure: Using Common Sense to Create a Dynamic, Place-Based Social Information System." *AI Magazine* 33.2 (2012): 91.
- [14] Wang, Dong, and Yang Liu. "A pilot study of opinion summarization in conversations." *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL-2011)*. 2011.

- [15] Godfrey, John J., Edward C. Holliman, and Jane McDaniel. "SWITCHBOARD: Telephone speech corpus for research and development." *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92.*, 1992 IEEE International Conference on. Vol. 1. IEEE, 1992.
- [17] Wen, Miaomiao, and Carolyn Penstein Rose. "Understanding participant behavior trajectories in online health support groups using automatic extraction methods." Proceedings of the 17th ACM international conference on Supporting group work. ACM, 2012.
- [18] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." *the Journal of machine Learning research* 3 (2003): 993-1022.
- [19] Chang, Angela, et al. "Textual tinkerability: encouraging storytelling behaviors to foster emergent literacy." Proceedings of the 2012 ACM annual conference extended abstracts on Human Factors in Computing Systems Extended Abstracts. ACM, 2012.
- [20] Boteanu, Adrian, and Sonia Chernova. "Modeling discussion topics in interactions with a tablet reading primer." Proceedings of the 2013 international conference on Intelligent user interfaces. ACM, 2013.
- [21] McCallum, Andrew Kachites. "MALLET: A Machine Learning for Language Toolkit." <http://mallet.cs.umass.edu>. 2002.
- [22] Godfrey, John J., Edward C. Holliman, and Jane McDaniel. "SWITCHBOARD: Telephone speech corpus for research and development." *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92.*, 1992 IEEE International Conference on. Vol. 1. IEEE, 1992.
- [23] David Graff, Kevin Walker, and David Miller. "Switchboard Cellular Part 1 Transcription." Linguistic Data Consortium, Philadelphia. 2001.
- [24] Chang, Chih-Chung, and Chih-Jen Lin. "LIBSVM: a library for support vector machines." *ACM Transactions on Intelligent Systems and Technology (TIST)*2.3 (2011): 27.
- [25] Cortes, Corinna, and Vladimir Vapnik. "Support-vector networks." *Machine learning* 20.3 (1995): 273-297.
- [26] Boser, Bernhard E., Isabelle M. Guyon, and Vladimir N. Vapnik. "A training algorithm for optimal margin classifiers." Proceedings of the fifth annual workshop on Computational learning theory. ACM, 1992.