

**Humour-in-the-loop:
Improvised Theatre with
Interactive Machine Learning Systems**

by

Kory Wallace Mathewson

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Computing Science

University of Alberta

© Kory Wallace Mathewson, 2019

Abstract

Improvisation is a form of live theatre where artists perform real-time, dynamic problem solving to collaboratively generate interesting narratives. The main contribution of this thesis is the development of artificial improvisation: improvised theatre performed by humans alongside intelligent machines. First, I present background underlying the art of improvisation and the scientific fields of interactive machine learning and dialogue generation. Then, I present *Pyggy*, the first experiment on live stage human-machine improvisation and *A.L.Ex.*, the Artificial Language Experiment which addresses several key technical improvements over *Pyggy*. *Improbatics* is then presented which details audience evaluation of Turing test-inspired live improvised performance using *A.L.Ex.* Two novel contributions to machine-assisted narrative generation are then presented and discussed. The first of these contributions, *Shaping the Narrative Arc*, is a model incorporating an underlying narrative arc to improve response generation. The second contribution, *dAIrector*, synthesizes a plot graph with contextual information to generate contextual plot points and serve as director. The thesis concludes by discussing public reflections on live artificial improvisation performances from around the world and interesting future directions to explore. My work presents fundamental advances in human-machine interaction through the lens of improvised theatre which is the ideal test bed for collaboration between humans and intelligent machines.

Preface

This thesis is an original work by Kory Wallace Mathewson.¹ Several chapters of this thesis are based on published or submitted papers written in collaboration with other researchers. For each chapter contained herein, details on these publications are included as footnotes. Submission and author contribution details are summarized below for completeness. Unless otherwise stated, no parts of this thesis have been previously published. Several of the studies included in this thesis are part of the research project which received research ethics approval from the University of Alberta Research Ethics Board. Project Name: *Developing an Engaging Artificial Intelligence for Conversational Performance Art*, ID Number: Pro00079215, Date: February, 23 2018. Mayank Rehani and Dr. Patrick Pilarski provided support compiling and submitting the ethics application.

The background and literature review in Chapters 1 and 2 as well as the discussion and conclusions in Chapters 7 and 8 are original works composed by Kory Wallace Mathewson and are previously unpublished. The only exception is that some of the initial conceptualization and progress as described in Chapter 3 is published on Kory Wallace Mathewson’s website as: Mathewson KW (2016) **Building an Artificial Improvisor**.²

Chapter 3 is adapted from a paper submitted and accepted at the 13th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2017). The original technical apparatus (*Pyggy*) described in Chapter 3 was designed and built by Kory Wallace Mathewson. The second iteration of the technical apparatus (*A.L.Ex.*) was conceived of by Dr. Pi-

¹Through the thesis the narrator will prefer the plural voice over the singular.

²<https://korymathewson.com/building-an-artificial-improvisor/>.

otr Mirowski (HumanMachine), and designed and built for this work through collaboration between Kory Wallace Mathewson and Dr. Mirowski. The experimental design, data collection, and concluding analysis are the authors shared original work. It is published as: Mathewson KW and Mirowski P. (2017) **Improvised Theatre Alongside Artificial Intelligences**. 13th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE'17). Snowbird, Utah, United States. October 2017. Additionally, a preliminary version of this work was presented as a workshop extended abstract and poster: Mathewson KW and Mirowski P. (2017) **Artificial Improvisation: Improvisational Theatre with Deep Neural Networks and Robots**. 2017 Multidisciplinary Conference on Reinforcement Learning and Decision Making. Ann Arbor, MI, United States, 2017.

Chapter 4 is adapted from a paper accepted to the 14th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2018) and a workshop paper submitted and accepted at the 2017 Workshop on Machine Learning for Creativity and Design at the 2017 Conference and Workshop on Neural Information Processing Systems (NIPS) conference. The technical apparatus (*A.L.Ex.*) was designed and built in collaboration with Dr. Piotr Mirowski (HumanMachine), as described above. The experimental design was developed in collaboration with Dr. Mirowski. The data analysis and concluding analysis are the authors shared original work. It is published as Mathewson KW and Mirowski P. (2018) **Improbatics: Exploring the Imitation Game using Machine Intelligence in Improvised Theatre**. 14th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE'18). Edmonton, Alberta, Canada. November 2018. Additionally, a preliminary version of this line of experimentation was presented as: Mathewson KW and Mirowski P. (2017) **Improvised Theatre as a Turing Test**. NIPS 2017 Workshop on Machine Learning for Creativity and Design.

Chapter 5 is adapted from a paper entitled **Shaping the Narrative Arc: An Information-Theoretic Approach to Collaborative Story Generation through Dialogue**. The technical apparatus was designed and built by Kory Wallace Mathewson in collaboration with Pablo Castro and Marc G.

Bellemare. Pablo Castro assisted with the data collection and contributed to manuscript edits. Colin Cherry and George Foster provided valuable conceptual insights and manuscript edits. Marc G. Bellemare was the supervisory author and was involved with concept formation, experimental design, and manuscript composition.

Chapter 6 is adapted from Eger M. and Mathewson KW. (2018) **dAIrecor: Automatic Story Arc Generation through Knowledge Synthesis**. Joint Intelligent Narrative Technologies / Intelligent Cinematography and Editing Workshop at AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE). November 2018. Edmonton, Alberta, Canada. This work is a collaborative effort with Markus Eger (NC State University). Both authors collaborated on the conceptualization, development of the experimental apparatus, data collection, manuscript preparation.

I include references for 11 other research contributions supporting human-machine interaction that I collaborated on:

- Dziri N, Kamaloo E, **Mathewson KW** and Zaiane O (2019) Evaluating Coherence in Dialogue Systems using Entailment. Annual Conference of the North American Chapter of the Association for Computational Linguists (NAACL-HLT). Minneapolis, USA.
- Winters T and **Mathewson KW** (2019) Automatically Generating Engaging Presentation Slide Decks. 8th International Conference on Computational Intelligence in Music, Sound, Art and Design (EvoMUSART'19). Leipzig, Germany.
- Dziri N, Kamaloo E, **Mathewson KW** and Zaiane O (2018) Augmenting Neural Response Generation with Context-Aware Topical Attention. arXiv preprint arXiv:1811.01063.
- Travník JB, **Mathewson KW**, Sutton RS and Pilarski PM (2018) Reactive Reinforcement Learning in Asynchronous Environments. *Frontiers in Robotics and AI*, Vol 5, p.79.
- Pilarski PM, Sutton RS, **Mathewson KW**, Sherstan C, Parker AS and Edwards AL (2017) Communicative Capital for Prosthetic Agents. arXiv preprint arXiv:1711.03676.

- **Mathewson KW** and Pilarski PM (2017) Concurrent Human Control and Feedback Shaping for Robot Training with Actor-Critic Reinforcement Learning. Multidisciplinary Conference on Reinforcement Learning and Decision Making (RLDM). Ann Arbor, USA.
- **Mathewson KW** and Pilarski PM (2017) Reinforcement Learning based Embodied Agents Modelling Human Users Through Interaction and Multi-Sensory Perception. 2017 AAAI Spring Symposium on Interactive Multi-Sensory Object Perception for Embodied Agents. Palo Alto, USA.
- Vega R, Sajed T, **Mathewson KW**, Khare K, Pilarski PM, Greiner R, Sanchez-Ante G, Antelis JM (2017) Assessment of feature selection and classification methods for recognizing motor imagery tasks from electroencephalographic signals. Artificial Intelligence Research, Vol 6:1.
- **Mathewson KW** and Pilarski PM (2016) Simultaneous control and human feedback in the training of a robotic agent with actor-critic reinforcement learning. 2016 IJCAI Workshop on Interactive Machine Learning. New York, USA.
- Pilarski PM, Sutton RS, Edwards AL, Sherstan C, **Mathewson KW**, Parker ASR, Hebert JS (2016) Towards strong prosthetic machine intelligence. Cybathlon Symposium, Zurich, Switzerland.
- Pilarski PM, Sutton R, **Mathewson KW** (2015) Prosthetic Devices as Goal-Seeking Agents. Second Workshop on Present and Future of Non-Invasive Peripheral-Nervous-System Machine Interfaces: Progress in Restoring the Human Functions (PNS-MI). Singapore.

This page is unintentionally left blank.

– Demetri Martin

A whole is that which has a beginning, a middle, and an end. A beginning is that which does not itself follow anything by causal necessity, but after which something naturally is or comes to be. An end, on the contrary, is that which itself naturally follows some other thing, either by necessity, or as a rule, but has nothing following it. A middle is that which follows something as some other thing follows it. A well constructed plot, therefore, must neither begin nor end at haphazard, but conform to these principles.

– Aristotle in *Poetics* (Part VII), c. 335 BC.

Supposing, for instance, that the fundamental relations of pitched sounds in the science of harmony and of musical composition were susceptible of such expression and adaptations, the engine might compose elaborate and scientific pieces of music of any degree of complexity or extent . . .

– Ada Augusta, Countess of Lovelace, notes on L. F. Menabrea’s *Sketch of the Analytical Engine Invented by Charles Babbage*, Scientific Memoirs, 1843.

The script places constraints on dialogue and interaction, and it defines concise test scenarios. The stage constrains the environment, especially if it is equipped with special sensing, communication or computational infrastructure. More importantly, the intelligent stage, with its embedded computing and sensing systems, is a resource that autonomous robotic performers could use to bolster their own ability to perceive and interact with people within the environment.

– Cynthia Breazeal, MIT Media Lab in *Interactive Robot Theatre*, 2003.

Acknowledgements

In June 2015, less than one year into my Ph.D. program at the University of Alberta, I engaged in a dialogue with the brilliant UK theatre teacher and director Adam Meggido that would start me on a journey to develop an artificial intelligence that I could perform improvisational theatre alongside.

As I reflect on this work, as one is want to do in the preparation of a dissertation, I remember many dialogues such as these. It is these conversations that have shaped, and in a way become, my academic progression. I am overwhelmed with gratitude for the exuberance my friends, colleagues, and collaborators show when discussing this work. An absolutely critical feature to any research trajectory is shared passion and enthusiasm. It was common for me to bring this work up casually with friends, and I was always amazed at how quickly they offer their time, energy, and ideas for creative and scientific progression. Perhaps this is the nature of the flock of humans that I fly through life with; we improvisors are ready to accept and expand on the smallest gust of motivation. We can create something amazing, if only for an instant. Thank you to especially Colin Mochrie, for being an enthusiastic sounding board, excited at the prospect of having a robot take your job.³

I would like to share gratitude explicitly with all those who have helped. Thank you so much to The Professors (Julian Faid and Joe Vanderhelm) and to Paul Blinov, Sarah Ormandy, Matt Schuurman, Joel Crichton, Nikki Hulowski, Paul Blinov, Leif Ingerbritsen, Paul Blinov, and Stuart Hoye. Thank you to Rapid Fire Theatre for being the perfect place for new ideas to grow and flourish (especially to Karen Brown Fournell) and to the world-wide cast of Improbots—in the United Kingdom, Sweden, and across Canada. Thank you to Alessia Pannese, Stephen Davidson, Luba Elliott, Katy Schutte, and Steve Roe. Thank you to Etan Muskat and the cast of “Yes Android” and all of the guests who have joined an artificial improviser on stage. Thank you to Arthur Simone and Dr. Heather Knight whose work I connected with at the perfect moment in time. Thank you to every audience member who has shouted ‘bathroom’ as a suggestion.

With the excellent support from my friends and colleagues, the days of my doctoral studies have been some of the most personally fulfilling. I want to express my gratitude to my supervisor Dr. Patrick M. Pilarski for supporting

³Personally, I think you and a bot will do a better show than just a bot.

me as a mentor and friend throughout. I am fortunate that we have spent many days talking about talking, thinking about thinking, and discussing discussion. I am deeply grateful to the members of my candidacy and final examining committees including: Dr. Pierre Boulanger, Dr. Dale Schuurmans, Dr. Patrick Pilarski, Mike Kennard, Dr. Mark Riedl, Dr. Greg Kawchuk, and Dr. Richard Sutton. The time that I spent with them during exams was memorable and clarifying. It is seldom that you have a room full of individuals who care so deeply about what you are saying and how you say it. I feel grateful to have shared that room with all of you.

I would like to express gratitude to Dr. Russ Greiner for introducing me to machine learning and for sharing honest reflections on family and connections when I needed them. As well, to Dr. Vadim Bulitko for teaching a memorable class for which I am grateful to have been a teaching assistant. You taught me the importance of focus, flow, and deliberate interaction design. Thank you to Dr. Matthew Taylor and Dr. Kevin Swersky for believing in me and my skills early on. I can honestly say that I have enjoyed every conversation I have had with both of you. I owe many thanks to my closest scientific collaborators who were delightful to work with. I feel very lucky to have had a chance to work with individuals across many different companies through my industrial experience. Thank you to Arjun Maheswaran (Twitter); Douglas Eck, Jessie Engle, Adam Roberts, Ian Simon, Curtis Fjord Hawthorne, and David Ha (Google Brain); Ruslan Salakutdihnov, Charlie Tang, and Jian Zhang (Apple) and to Pablo Castro, Colin Cherry, George Foster and the rest of the team at Google Montreal. I am indebted to Marc G. Bellemare for his first-rate mentorship. The best mentors are those that amplify your time and talent; nothing less can be said of Marc.

I would like to thank the Alberta Innovates Center for Machine Learning (AICML), the Alberta Machine Intelligence Institute (Amii), the Bionic Limbs for Improved Natural Control lab (BLINC), and the University of Alberta for supporting my research. I am grateful for financial support from the Natural Sciences and Engineering Research Council (NSERC) and the Alberta Innovates Technology Fund (AITF).

I am thankful to my family members for their support and belief. You all told me I would never be a comedian, *well who's laughing now?* Thank you to John Agapiou for introducing me to a scholar and a gentleman who I am proud to call my close colleague, Dr. Piotr Mirowski. Piotr, thank you for relentlessly challenging me. Your humour, calibre, expectations, and honesty motivate me without end. Thank you most to Lana. You can try to find someone in the world like Lana Cuthbertson, it will not happen. Lana, you are the best teammate. Thank you for your support in basecamp. For all those that I have not mentioned, there is no way that I merely forgot you. I spent months writing and editing this. It is a mathematical certainty that I did it on purpose, with maleficence, and it was completely intentional.⁴

⁴I hope.

Contents

1	Introduction	1
2	Background	5
2.1	Interactive Machine Learning	6
2.1.1	Shared Communication	9
2.1.2	Shared Perception	10
2.1.3	Evaluation	11
	Human-Centric Evaluation	12
	Machine-Centric Evaluation	13
2.1.4	IML in Summary	13
2.2	Conversational Dialogue Systems	14
2.2.1	Conversational Modelling	16
2.2.2	Information Retrieval for Chatbots	17
2.2.3	Language Modelling	18
2.2.4	Sequence-to-Sequence Modelling	26
2.2.5	Neural Model Improvements	33
2.2.6	Datasets	34
2.2.7	Evaluating Dialogue Systems	35
2.2.8	Conversational Dialogue Summary	39
2.3	Improvisational Theatre	40
2.3.1	Theatre Technology and Automata	40
2.3.2	Cyborg Theatre: Stimulation Over Simulation	42
2.3.3	Live Theatre with Chatbots	43
2.3.4	Interactive Machines in Theatre and Comedy	44
2.3.5	Creative Interactive Machine Learning	46
2.4	Interaction, Interlocution, Improvisation	47
3	Improvised Theatre Alongside Artificial Intelligences	49
3.1	Introduction	51
3.2	Methods	52
3.2.1	Version 1: Pyggy – Rule Based Generation	52
3.2.2	Version 2: A.L.Ex. – Corpus Based Generation	55
	Automatic Language Generation in Improvised Theatre	55
	Text Generation with Neural Language Models	56
	Dataset	59
	System Architecture	60
3.3	Results	61
3.3.1	Version 1: Pyggy	63
3.3.2	Version 2: A.L.Ex.	64
3.3.3	Human-AI Collaboration through Games	65
3.3.4	Justification Game	65
3.3.5	Multiple-choice Human-mediated Dialogue	66
3.3.6	Multiple-person Games	66

3.3.7	Comparison with ELIZA	67
3.4	Discussion and Summary	68
4	Improbotics: Exploring the Imitation Game	71
4.1	Introduction	72
4.2	Background	73
4.2.1	Related Work	75
4.2.2	Motivation	76
4.3	Methods	77
4.3.1	Architecture of the Artificial Improvisor	77
4.3.2	Typical Structure of AI-based Improvised Shows	80
4.3.3	Technical Configuration	82
4.3.4	Turing Test-based Theatrical Improvisation Games	83
4.4	Observations	84
4.4.1	Evaluation	85
4.5	Results	86
4.5.1	Quantitative Evaluation	87
4.5.2	Qualitative Evaluation	89
4.6	Discussion and Summary	91
4.6.1	Reflections from Professional Improvisors	91
4.6.2	Deception and Problem Solving	91
4.6.3	Lack of Contextual Consistency	92
4.6.4	Handling Timing in Improvisation	93
4.6.5	Human Curation of Dialogue	94
5	Information-Theoretic Narrative Generation	96
5.1	Introduction	97
5.2	Shaping the Narrative Arc	99
5.2.1	The Conversation Model	99
5.2.2	The Universe Model	100
5.2.3	Recursive Universe Belief Propagation	100
5.2.4	The Narrative Arc	103
5.2.5	Generating Dialogue with the Narrative Arc	104
5.3	Evaluation	105
5.3.1	Narrative Arc of Existing Dialogues	105
5.3.2	Shaping the Narrative Arc	106
5.3.3	Predicting the Next Best Line	108
5.3.4	Interactive Collaborative Dialogue	111
5.4	Related Work	111
5.5	Discussion and Summary	113
6	dAIrector: Contextual Story Arc Generation	114
6.1	Introduction	115
6.2	Background and Related Work	115
6.2.1	Automated Story Generation	115
6.2.2	Digital Storytelling	116
6.2.3	Improvised Theatre	117
6.3	Approach	118
6.3.1	Plotto	118
6.3.2	TV Tropes	119
6.3.3	Paragraph Vectors	120
6.3.4	Plot Tree Generation	121
6.3.5	Contextual Tilts	122
6.3.6	Stage Presence	122
6.4	Evaluation	124

6.4.1	Evaluation of Tilts	124
6.4.2	Sample Stories	126
6.4.3	Qualitative Evaluation by Professional Improvisor	128
6.5	Discussion and Conclusion	129
7	Discussion	131
7.1	Human-Machine Improvisation	132
7.2	Critical Reception	134
7.2.1	From Audience and Improvisors	134
7.2.2	From Reviewers	135
7.2.3	From News Media	136
7.3	Future Directions	138
7.3.1	Improv in Any Language	138
7.3.2	Starting and Finishing Conversations	140
7.3.3	Tuning Language Models with Rewards	140
7.3.4	Adapting Language Models in Real-Time	142
7.3.5	Evaluating Conversational Dialogue	143
7.4	Summary	144
8	Conclusion	145
	References	147
	Appendix A IML Research and Future Directions	175
A.1	Active IML Research	175
A.2	How IML Continues to Evolve	177
	Appendix B A Brief History of Chatbots	181
B.1	ELIZA from Joseph Weizenbaum 1966	181
B.2	PARRY from Kenneth Colby 1968	182
B.3	GUS 1977, Dr. Sbaitso 1991, and Jabberwocky 1997	183
B.4	ALICE 1995, AIML, The Loebner Prize	184
B.5	ActiveBuddy Inc. SmarterChild 2001	185
B.6	The Rise of Corporate Bots	185
B.7	Social Bots, Cleverbot, Mitsuku	186
B.8	Chatbot Competitions	187
B.9	The Future of Chatbots	188
	Appendix C Neural Model Research and Future Directions	189
C.1	Additional Evaluation Methods	189
C.2	Reinforcement Learning in Dialogue Generation	190
C.3	Vocabulary Size and Approximating the Softmax	191
C.4	Word Embeddings	193
C.5	Topic Model	193
C.6	Batching and Bucketing	193
C.7	Editing Outputs	194
C.8	Training on Logs	194
	Appendix D The History of Improvising Intelligences	195
	Appendix E The Shapes of Stories	198

Appendix F	Supplementary Material for Narrative Arc	200
F.1	Data Processing Details	200
F.2	Rejection Sampling	201
F.3	Additional Narrative Arcs Underlying Existing Dialogues . . .	202
F.4	Improviser Transcripts	204
F.5	Improviser Assessments	212
	F.5.1 Subject/Model Specific Comments/Assessment	212
	F.5.2 General Interaction Comments/Assessment	213
Appendix G	Listing of Media Coverage	215

List of Figures

2.1	General dialogue system diagram	15
2.2	Figure of the encoder / decoder framework.	32
3.1	High level system diagram of Pyggy.	53
3.2	High level system diagram of A.L.Ex.	55
3.3	Context and input to Artificial Language Experiment and un-rolled single output decoding.	56
3.4	User interface with example input (top) and response (bottom).	61
3.5	Human Improvising with Pyggy	63
3.6	Human Improvising with A.L.Ex.	64
3.7	Multiple Improvisors and A.L.Ex.	67
4.1	Rehearsal for Improbatics	73
4.2	Artificial Language Experiment (A.L.Ex.) System Diagram	74
4.3	Embodiment of an Artificial Improvisor	79
4.4	Human performers alongside an Artificial Improvisor	80
4.5	Simplified Improbatics Interface	82
4.6	Audience and Performer Presence Analysis across Cities	83
4.7	Performer Presence Analysis across Systems	86
4.8	Lexicographical Analysis across Sources	89
5.1	Example of a Narrative Arc	101
5.2	Narrative Arc of the Opening of Romeo and Juliet	104
5.3	Concealing, Neutral, and Revealing Narrative Arcs	107
5.4	Revealing and Concealing across Universe Models	108
5.5	Narrative Arc Boosting Predictive Accuracy	110
6.1	Plotto Sub-graph	119
E.1	Detailed Story Arc	199
F.1	Narrative Arc of “Blade Runner”	203
F.2	Narrative Arc of “The Great Dictator”	203
F.3	Narrative Arc of “Dead Parrot Sketch”	203

List of Tables

2.1	Example Unigram Language Model	20
2.2	Example Bigram Language Model	24
5.1	Next Line Prediction Task Results	110

List of Algorithms

1	Simplified interactive machine learning algorithm.	8
2	Generating dialogue with rejection sampling.	106

List of Code Samples

6.1	Pseudo-code for dAIrector plot generation.	123
-----	--	-----

Chapter 1

Introduction

*A good improviser looks great on stage;
the best improvisors make everyone else look great.*

– Jacob Banigan, 2004

Our uniquely human experience is defined by knowledge sharing and storytelling. The main means by which we share our experiences is through spoken dialog. Spoken dialogue has evolved to become one of the most complex functions of human communication. We can engage in multiple conversations over a short period of time, quickly adapting to our conversational partners. Through these conversations we share our perspective, seek advice, share stories, and find common ground. These interactions collectively define the human condition.

Many commonplace technologies place humans in contact with computerized dialogue systems—computer systems that interpret and generate speech (e.g., cell phones, home assistants). These systems are limited in their capacity to continue engaging conversations with humans. To help address this limitation, dialogue systems may be improved through the ability to adapt to different speakers, contexts, and modes of operation. Rapid adaptation is a unique characteristic of human dialog.

A major domain of study for dialogue research is that of curated dialogue in cinematic and theatre scripts. This script research is constrained, as the data is meticulously authored and edited by its very nature. Rather than using curated, written dialog, we propose improvised theatre dialogue as an

exceptional domain for experimentation toward improving dialogue generation systems. Improvised theatre is characterized by adaptive performers spontaneously responding to each other, and the environment, to collaboratively create a narrative.

This thesis aims to test and evaluate novel dialogue system improvements by through experimentation with improvisational theatre performers and audiences. The hypothesis is that these improvements will allow for systems to continue longer, more engaging interactions than current hand-coded performance systems. Further to this aim, this thesis aims to investigate interactions between humans, as participants and observers, and these systems. These interactions are surveyed under conditions when human believe they are interacting with, or watching another human interact with, an artificial intelligence rather than a human, and vice-versa.

Several researchers have designed experiments to understand the interactions between humans and artificial-intelligence-powered dialogue systems. To date, these systems have not been investigated by expert collaborative storytelling humans. This thesis accomplishes this testing by recruiting improvisational theatre artists and audiences to engage with the system and provide quantitative and qualitative feedback.

Understanding the interaction between humans and artificial-intelligence-based dialogue systems offers enormous potential to significantly improve how humans express themselves and share their experiences. The findings herein offer significant insights into interactions when humans and machines perform together on the theatrical stage. With experimentation and future research, similar configurations could be used in a wide range of human-machine collaborative tasks; this general impact should not be understated. We interact with technology, and specifically machine-learning systems, at an increasing rate in our every day lives. By studying how these interactions occur, from both sides of the collaboration, we can make the entire experience safer, easier, and more effective

Artistically, the main contribution of this thesis is the formal development and dissemination of the art-form of “artificial improvisation”. **Artificial im-**

provisation is improvisational theatre performed by humans alongside intelligent machines. The research and development in this thesis has been used in over 75 shows, by 5 professional theatre companies around the world. This work has ignited a community to technology-augmented improvisational theatre performers to present and share their work.

This dissertation contains four contributions supporting the first application of interactive machine learning-based dialogue systems for improvised theatre. The contributions are separated into chapters. Each chapter presents progress in open-domain dialogue research in improvised theatre.

- Chapter 3 presents the first formalization of human-machine improvised theatre experiment as a suitable, challenging test bed for interactive open-domain dialogue research. It presents the development and evaluation of two artificial improvisors: 1) Pyggy, a rule-and-retrieval-based conversational system, and 2) A.L.Ex. (Artificial Language Experiment), a generative language modelling-based conversational system.
- Chapter 4 presents evaluation of artificial improvisors, contributing the first quantitative and qualitative analysis of artificial improvisation. It presents results and analysis of the systems and experiments introduced in Chapter 3.
- Chapter 5 contributes a novel method extending artificial improvisation. “Shaping the Narrative Arc” is a method which models the underlying arc of conversation. Augmenting conversation systems with a notion of story arc enhances narrative generation. Results from surveys with expert human improvisors are presented in support of the methods capabilities.
- Chapter 6 contributes the “dAIrector”: a novel method for generating coherent, contextual plot structures by fusing information contained in two knowledge graphs.

Chapters 3 and 4 present development and evaluation of complete artificial improvisor from a systems approach. Chapters 5 and 6 each present a novel method augment any artificial improvisor. The methods for “Shaping the Narrative Arc” and “dAIrector” are system agnostic, and evaluated in isolation,

disconnected from the artificial improvisors in the earlier chapters.

In addition to these science-directed efforts, this dissertation has made additional contributions toward the artistic goal of integrating machine learning-based dialogue systems into theatrical productions. Artificial improvisation is a new interactive multimedia art-form which signifies a novel innovation in field of improvisation as supported by reflections from critics, public media, expert performers, and audiences (Section 7.2). Artists describe the system as inspiring, engaging, challenging, and that it provides them a new medium to enhance and augment their own artistic creations.

The thesis exists at the intersection between three worlds of research—interactive machine learning, dialogue systems, and improvisational theatre. In the Chapter 2, we present background on these fields with historical context and supporting literature. We introduce topics and methods, and discuss how the fields braid together and intertwine and support each other. It is our hope that all background materials necessary to understand the experiments through this thesis are contained in this chapter. The contributions in Chapters 3-7 are detailed previously in this section. Chapter 7 discusses the scientific and artistic innovations of artificial improvisation, and explores how this work has progressed the frontier, and presents several research directions for future endeavour. Finally, Chapter 8 provides concluding remarks and hope for the future.

Chapter 2

Background

We are blessed with a language, a system of communication, so powerful and so precise that we can share what we've learned with such precision that it can accumulate in the collective memory. And that means it can outlast the individuals who learned that information, and it can accumulate from generation to generation. And that's why, as a species, we're so creative and so powerful, and that's why we have a history. We seem to be the only species in four billion years to have this gift.

– David Christian

TED 2011

This thesis focuses on advancing interactive machine learning for improvised theatre. Section 2.1 synthesizes research in interactive machine learning (IML), Section 2.2 provides relevant background literature for appreciating and understanding the conversational dialogue system research, and Section 2.3 provides background on the artistic field of improvised theatre. This thesis is positioned at the intersection between these three fields. The chapters contain new methods and analysis which represent our contributions to this intersection by connecting ideas, methods, and research from IML, dialogue systems, and improvised theatre. At a high level, these fields connect at the intersection of continual learning through interaction, trial and error.

2.1 Interactive Machine Learning

This section presents and synthesizes research and applications allowing a human with no assumed application-specific knowledge or technical ability, to interact with machine learning (ML) models. Interactive Machine Learning (IML) research techniques and approaches are outlined and applications of IML research in creative applications and artistic presentations are presented. This section concludes with promising prospects for future development.

Developments in IML combine the unique talents of humans and machines to produce new capabilities. Human-machine teams are prevalent in the lives of many humans. We live in an interactive world; examples include: bicycles for transportation, medical prosthetic devices, web search engines, voice-based assistants, and connected home technology. Many of these systems already incorporate ML. For instance, many systems can learn to personalized performance based on unique individual interaction.

Fundamentally, IML systems predict appropriate responses to interactions. That is, a human provides an input and a system provides a response. This model of the interaction can be generalized by the *motivating question*:

“When the human does this, what should the machine do?”

We interact with the expectation that the system will be able to translate our inputs into meaningful outputs. While we may understand the limitations of these systems, we may also overestimate the capabilities. This overestimation can lead to fear. Artificial intelligence and robots have been presented as dangerous platforms and used to mislead the public.¹ Naïvety can also transform into understanding, appreciation, and excitement for the future. This requires rigorous science devoted to effective knowledge transfer; science which is both fundamental research and clear communication.

ML is a field of computing science which explores automated model learning. Learning is the improvement in performance on a certain task with respect to a given objective. Models are functions which learn to map inputs to outputs through mathematical transformations. ML problems often start with a

¹<https://goo.gl/gAWwAK>

set of data (diverse input/output pairs). Training progresses as the model is incrementally modified until the diverse inputs are consistently transformed into the desired, respective outputs. Thus, ML concerns itself with how to program machines to learn a mapping, as opposed to explicitly defining the mapping function.

Generalization is a secondary, albeit no less important, objective in ML. The trained model should be general enough to transform unseen inputs to their corresponding outputs. These held-out data-points should be separated and delineated prior to training. They are referred to as the validation data (used for development and tuning) and test data (used for final evaluation). An objective performance mismatch between training, validation, and test data can indicate a variety of issues in the training process.

All ML is interactive. Thus, IML systems should be designed using frameworks for interaction with humans [284]. IML is a term coined by Fails and Olsen (2003) to describe their novel system where a human and an ML algorithm shared control to perform image segmentation [79]. Conventionally, IML is defined as a subfield of ML which explores the addition of a human, or multiple humans, in the learning loop. IML is sometimes referred to as human-in-the-loop learning. The human(s) may observe the learning and provide input to improve training toward the performance objective.

In many ML systems a human spends time preparing data, designing objectives, and tuning hyperparameters. These systems could rightly be considered IML. Most ML methods, while agnostic to the fact that the task is defined by a human, are ultimately optimizing toward an objective function which is defined by the subjective opinion of goodness of that human. If the human is only defining the objective and preparing the data, the interaction is loosely coupled. In many IML examples, the interaction can be more tightly coupled. For instance, in prosthetic robotics where human nerves are connected to machine wires [113].

The learning problem underlying IML is: *how can machines and people interact to solve tasks more efficiently and effectively than either could on its own?* [6]. It is helpful to consider several task examples. For instance, if a

Algorithm 1 Simplified interactive machine learning algorithm.

Given a model M which maps to input I_t to output $O_t \Leftarrow M(I_t)$
Given a time horizon T , and a test of goodness $\{good, bad\} \Leftarrow G(O_t|I_t)$
while $t \leq T$ **do**
 obtain $I_t \sim \mathbf{I}$
 $O_t \Leftarrow M(I_t)$
 if $G(O_t|I_t) == good$ **then**
 modulate M to increase probability of $O_t \Leftarrow M(I_{t'}) \forall t' > t$
 else $\{G(O_t|I_t) == bad\}$
 modulate M to decrease probability of $O_t \Leftarrow M(I_{t'}) \forall t' > t$

human provides a line of dialogue, the machine might discriminate between a set of underlying topics, predict a sentiment score value, or generate an intelligible response. Often tasks require adaptable collaboration over the course of incremental, continuous, and sequential decision making.

While there are many implementation details, teaching ML systems to perform these tasks is straightforward. A high-level algorithm is shown in Algorithm 1.

Algorithm 1 helps us to frame most ML. It also connects to the *motivating question*. To address this question, IML designers and teachers must define the goodness function, $G(\cdot)$, and use a method to modify the outputs of the model, $M(\cdot)$, toward maximizing accumulated goodness. $M(\cdot)$ maps from the space of inputs, $I_t \in \mathbf{I}$, to a space of outputs, $O_t \in \mathbf{O}$. It attempts to map from from all of the things an interacting human can input to all of the things the ML system can output. This interaction between human inputs and model outputs constrains IML systems [259].

While we have explored the basis of IML thus far, it is important to rephrase an expanded *motivating question*: “given human inputs, what are desirable model outputs?” IML systems should use information available by leveraging both the human- and the machine-in-the-loop. The machine should work on what it is best at, and so should the human, and interaction bandwidth capacity should used for sharing knowledge. Collaboration between multiple humans has historically been easier than with machines due, in large part, to shared communication.

2.1.1 Shared Communication

For effective human-machine collaboration, both intelligent systems—human and machine—should have a means by which to communicate their intentions, actions, goals, and understanding of the others. Human teaching allows agents to achieve arbitrary user-centric goals, improve an agent learning speed, and beneficially direct agent exploration [159].

There exists a continuum of IML systems, from loosely coupled to tightly coupled. This continuum is defined by the explicitness of interaction and the bandwidth of shared communication [261]. Tightly coupled humans might provide input at each step of model prediction. Loosely coupled humans might only provide input prior to model learning prediction as in supervised learning. Optimal learning machines might have a adaptive levels of autonomy. This sliding autonomy is deployed in Autopilot on new Tesla vehicles.² In rural highway driving the model could act more independently. In busy urban driving, the model requires more input from a human.

There are a number of explicit communication channels the human may use for teaching a learning machine. Direct teaching signals take the form of scalar inputs like buttons, switches and/or dials [261], verbal and auditory signals [323], tactile inputs, gestures [264], visual inputs for facial emotions [337], natural language [90], [165] and extend to neural interfaces capturing intention and signal directly from the human brain [143]. A 2015 review article discusses additional interactions [214].

Through direct ML-human interaction, the human often adapts their input to be more appropriate. This co-adaptation is echoed in the Joseph paradigm: “to teach is to learn twice over”.³ In teaching a machine, the human is learning both how to teach the machine and how the machine learns. Extending this idea, we can imagine an ML system that can teach a human how effective they are at teaching. This transparency of learning can allow humans to better understand how to teach agents [325].

As two learning systems interact and communicate, they build a shared

²https://www.tesla.com/en_CA/autopilot

³<https://www.bartleby.com/354/18.html>

language and a model of the understanding of the other. This *communicative capital* is built up over many back and forth interactions [259]. Reducing the number of interaction is a common goal of IML. Like capital investment, in the economic sense, *communicative capital* can require significant upfront cost in terms of time and effort to establish and maintain a communication channel. These costs may be incurred during the normal interactions of a partnership, or through preliminary effort external to the task-specific goals.

Through all of this communication, the human is attempting to share with the learning machine a sense of “when I do this, I want this to happen”. The human is attempting to define a mapping from inputs to outputs and communicate this mapping with the machine. The human co-adapts by refining which inputs they are providing to the machine, based what they have learned about how the machine learns from different inputs.

This co-adaptation may be tightly coupled (e.g. the human pushes buttons to program a thermostat to turn on every morning), or loosely coupled (e.g. the thermostat builds a model of human interaction by finding patterns in an abstract data stream). While these two methods of co-adaptation are similar, they differ on the perception stream of the learning machine.

2.1.2 Shared Perception

Perception is the process of an intelligent system becoming aware of something through its senses. The perception stream is the flow of continuous data input to the sensors and processed by the computational machinery. Classic, non-learning, machines perceive inputs from humans through specially programmed sets of instructions represented in an abbreviated format (e.g. buttons on a remote control). Learning machines have expanded perception and capacity to process abstract data.

Perception data streams are quantized into single binary digits (i.e. bit) streams of ones and zeros. Data can stream into learning machines from a wide variety of communication channels or perception sensors. Ongoing sensor innovation allows for arrays of sensors to be embedded in IML systems. These sensor arrays form perceptual streams for learning machines. When adding

sensors, there is a balance to be struck. With no perception it is impossible to communicate teaching and task signals. With too many sensors it is difficult to find meaningful correlations in inputs to help predict outputs. That is, it is hard to model patterns in data across all the sensor dimensions. The reduced predictive power due to a high dimensional space is sometimes referred to as the *curse of dimensionality* [19]. A narrow stream of data can only contain a limited amount of information [304], and in a stream too vast meaningful information is difficult to find.

From the human perspective, we imagine interacting with learning machines in a variety of ways. We can imagine that machines can learn from us through indirect sensors (passively observing parts of what we do), and/or through direct sensors. Humans are quick to build a mental model of teaching and learning through both their direct and indirect actions.

We can perceive these learning machines through our five main senses (e.g. taste, sight, touch, smell, and hearing), through our additional sensor streams (e.g. balance, temperature, and proprioception), and critically through our sensing of time-passing, familiarity, recollection, and agency. We desire IML systems which respond immediately, in a familiar fashion, remembering our specific preferences, and allow us to take actions in the interaction.

It is with this background on communication and perception we are able to discuss evaluation in IML. Several exciting current research directions in IML and additional considerations can be found in Appendix A.

2.1.3 Evaluation

Evaluation of any ML system is the very essence of optimization toward an evaluation metric (i.e. measure of goodness), that drove the simple learning algorithm previously presented (Algorithm 1). Evaluation is based on an objective function, this function is most often defined by a human. To properly evaluate an IML system, care should be given to define this objective.

There is a distinction to be made between evaluating the performance of the ML system on a task and evaluating the amount of physical effort, cognitive effort, and input is required from the human. Whereas task performance can be

measured and compared to optimal task performance, human factors will often be user-specific and may not correspond to the learning systems performance on a given task. This could be due to a misunderstanding of the communication channel by the human, by differing uses of the communication channel, and even by the human shifting how they communicate over the course of the task [120]. Boukhelifa *et al.* (2018) provide a comprehensive overview of quantitative and qualitative evaluations of IML systems [31]. Brief details are included below, as they deserve consideration when reading the following chapters.

Human-Centric Evaluation

Measuring the quality of the interaction between humans and ML systems often requires a human-centric design approach and solid user experience testing. Human-centred evaluations should explore the mental models of the IML system that the human develops.

Questions focused on engagement, such as “how well could you concentrate on the task rather than on the mechanisms used to perform those tasks?”, and on perceived ability, such as “how much were you able to control events in the task performance?” can be asked to explore these important human-centric opinions. These questions are explored in an IML context in Chapter 4.

The amount of training, and interaction, the training strategies, should be assessed. These assessments may reveal interesting patterns in the strategies of different groups of human users. Share-ability of a users trained model should be measurable (additional discussion in Appendix A). Can they effectively train a system and then share that training with someone else to catch that user’s system up with their own. Finally, privacy of the user’s data should be quantifiable and clearly communicated. Both, what is being captured in the perceptual stream and how that data is transformed, stored, transmitted, and shared need to be explicitly stated on any deployed IML system, similar to a websites privacy policy [7].

Machine-Centric Evaluation

Evaluating from the machine side is more straightforward, but there are several important features measurable which are specific to IML. The balance between the number of actions taken by the machine for each action by the human, for instance, correlates with the amount of agency each has in task performance. The machine must learn to find an objective balance between appeasing the goal of the human and maximizing task performance. Classic ML evaluations should be designed, executed, and reported for IML experimentation. Measures of algorithm robustness, convergence, and diversity analysis, and task performance (e.g. number of training steps to asymptotic performance) are as vital in IML as they are in classic ML problems.

Finally, one interesting evaluation metric is the correlation between user evaluation metrics and predictions of those values. Given that the IML system should be modelling the human user, and have some features of the human user in the perception stream, systems should make predictions about how a user would evaluate a system. Learned models of human evaluation are presented in Section 2.2 as they related to automated dialogue systems evaluation.

2.1.4 IML in Summary

Learning machines are gaining popularity with advances in communication and machine perception. These machines use IML which combines the skills and abilities of humans and machines to produce new capabilities. Interactive learning algorithms should be applied with an understanding that different humans interact with these systems differently; individual-specific dynamics should be explicitly modelled. Through open-source code and open data, scientists, creative technologists, and industry developers can quickly iterate by sharing their ideas and developments around the globe. We will soon develop, deploy, and disseminate IML systems with similar ease.

This section has provides background on the field of IML. Collaboration toward optimal task performance can tightly coupled interaction. Insufficient communication between systems can be due to limits in information capacity

or transmission rate and/or misaligned models of perception. The next section concerns itself with one of the most information dense and rapid transmission modes of interaction which humans employ: communication through conversational dialogue.

2.2 Conversational Dialogue Systems

The ability of a machine to generate meaningful open conversational dialogue, which can not be distinguished from a human participant has been one proposed test of a successful AI [330]. The imitation game, was proposed by Alan Turing (1950) and is thus often referred to as the Turing test [330]. Passing the Turing test refers to the ability of a computer program to impersonate a human in a real-time conversational dialogue, using written communication. Impersonation is sufficient when a human judge is unable to distinguish, on the basis of the conversation alone, between the program and a human.

The term dialogue comes from the Greek word *dialogos*, its roots are *dia: through* and *logos: speech or reason* [141]. In relation to the work of this thesis, we define dialogue as text-based, English language conversation between two participants as a means to share information. This conversation need not occur via speaking and listening, nor need it be grounded in an English language-based textual representation [42], [43]. The simplified and constrained domain permits focus on improvements to specific sub-components of conversation. Conversations can be categorized, based on participant goals, into one of four types: debate, dialogue, discourse, and diatribe.⁴ Dialogue is separated from the other three as it is a co-operative, two-way conversation where the goal is to exchange information and build relationships with one another. Open dialogue is the sharing of opinions on subjects and topics that are brought up and/or thought of during conversation.

Early work on text-based dialogue has expanded to include spoken dialogue on personal mobile devices due, in large part, to advances in automated speech recognition and automatic speech generation. These advances include

⁴<https://goo.gl/2G4nbh>

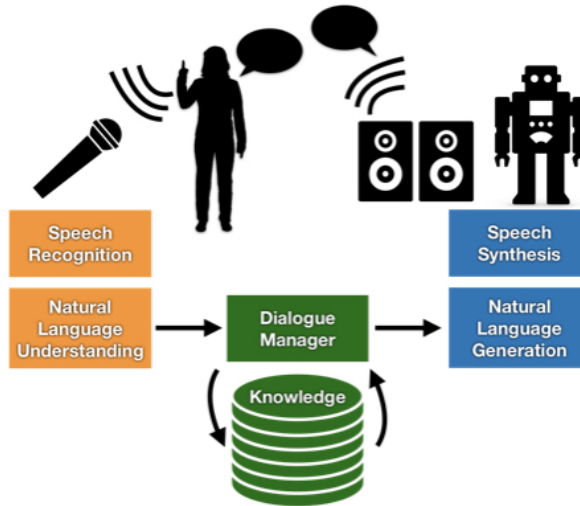


Figure 2.1: A general dialogue system diagram. The entire dialogue loop contains (from top-left, counter-clockwise) a human interacting with the system, a microphone voice interface and speech recognition to convert audio waveforms to digital signals, natural language understanding to parse the recognized input, a dialogue manager optionally connected to external knowledge bases, natural language generation and speech synthesis to generate appropriate output audio from the generated response. The audio is output using speakers and the robot embodies response dialogue through emotive actions and movement.

increases in speed and reductions in error rates [332]. These improvements are due in large part to replacing the conventional acoustic-feature modelling with supervised deep learning models [116]. These deep learning models are large networks of many parameters which define the mapping function from input to output. With enough data, these models can learn underlying patterns and correlations in high-dimensional data and generalize well to unseen input data.

A dialogue system is comprised of three major components (as illustrated in Figure 2.1: 1) an input interface, 2) an output interface, and 3) a processing module. The processing module in a dialogue system is often called the dialogue manager, and it processes the input text, models the conversation as a whole, and generates a text response based on the input.

Machines capable of conversation, sometimes referred to as conversational agents, are of growing importance in IML systems. These agents enable humans to use natural, smooth, efficient communication to interact with their devices [233], [242]. These systems allow ML systems to collaborate and in-

teract with humans to perform simple tasks (e.g. booking flights or playing music), enable adaptive user-directed tutoring systems [187], provide social assistance through machine assisted therapy—including supporting cognitive-behavioural therapy for treating depression [85], helping patients with chronic pain [230], and supporting reminiscence [245].

Several advances in computational linguistics (i.e. natural language processing) have led to the growth in popularity of these systems. These advances include: an increase in computing power⁵, increased availability of large amounts of clean training data [298], development of successful ML algorithms [30], [297], and a richer understanding of the structure of human language [117]. There are major challenges in designing, building, and deploying conversational agents [233]. Problems of robustness, scalability, data biases, domain adaptation, engineering infrastructure, privacy, and stability continue to limit the wide spread adoption of these systems.

The focus of this thesis is on improvised theatre, thus the focus of this background material is on conversational modelling for generating interesting and engaging responses to inputs over the course of a conversational dialogue.

2.2.1 Conversational Modelling

The basic problem of conversational modelling involves taking a natural language input from a human and outputting an appropriate natural language response. As the system is engaged in an conversation, the system takes in human inputs, and outputs responses, in alternating sequence. This assumption imposes a turn-taking model of conversation on the system, a common feature in open conversational dialogue.

A chatbot (also known as a chatterbot, bot, messenger bot, interactive agent, conversational dialogue agent, or artificial conversation entity) is a computer program which engages in turn-taking conversational dialogue. Chatbots can be differentiated based on their capabilities and/or method for conversational modelling. For instance, taxonomies include open- vs. closed-domain dialogue systems (i.e. differentiating the space of knowledge of the system) and

⁵<https://cloud.google.com/tpu/>

rule-based vs. corpus-based dialogue systems (i.e. differentiating based on the underlying model for conversational modelling). Inputs are often text or auditory signals, but, may also include visual information and/or side-context relevant to the current conversation. Outputs are often text or auditory signals, but, may also include multi-modal generation and embodiment actuation. For instance, a dialogue system could actuate a physical embodiment or modify the state of the environment alongside the generated dialogue.

Appendix B contains a brief history of chatbots by detailing the evolution from rule-based, to corpus-based, to unified conversational models. It also covers dialogue system competitions which have driven innovation and the rise of social and corporate bots.

Rule-based chatbots, have a conversational model based on hard-coded rules. These hand-crafted systems are often heavily engineered to match a set of pre-defined inputs to pre-defined outputs. Corpus-based chatbots use a large corpus of data to generate outputs in response to inputs. These corpora are often too large to fit in the memory of a computer. Corpus-based bots are further subdivided into two categories: information retrieval (IR) methods and generative language modelling (LM) methods. IR methods take as input a query sentence and then look for the best responses in a corpus of possible dialogue outputs. LM methods, sometimes referred to as statistical LMs or generative LMs, aim to model the statistics of the language used in a corpus and then generate responses to a given input based on the underlying probabilities.

2.2.2 Information Retrieval for Chatbots

Given a large set of possible responses (i.e. textual sequences of words), a conversation model could select a likely response from the set. The appropriateness of the selected response given an input is based on a human-defined objective function. This function is often based on a distance metric, measured by modelling input-response pairs of the corpus. A variety of metrics have been used including linguistic similarity based on overlap of words, characters, and parts of speech [339]. Another metric used is distance of nearest

neighbours in a distributed semantic vector space [16].

As a representative example, Banchs *et al.* (2012) suggested the use of movie scripts to build chat oriented dialogue systems [16], [297]. By conditioning on one or more utterances, their model searched a database of movie scripts and retrieved an appropriate response. Ameixa *et al.* (2014) showed that movie subtitles could be used to provide appropriate responses to out-of-domain questions using IR [5]. Another approach, using unstructured documents for chatbot engines, is presented in Yan *et al.* (2016) [357].

2.2.3 Language Modelling

Information retrieval methods are limited to a predefined corpus of responses. We may desire to generate new responses. This can be done using a statistical language model (LM) which is a probability distribution over a sequence of words. Using this LM, generating new responses corresponds to sampling from a probability distribution. The LM can also be used to compare the relative likelihoods of different sentences.

To understand much of the following material on LMs, several definitions and a brief introduction to probability theory are required. Language modelling is best communicated with a working example sentence. Consider the following sequence of words:

‘now i live in cortona italy’

Note that capitalization and punctuation are excluded from this example for clarity.⁶ This sentence is composed of six unique sub-elements or words. There is an order to the words, and the words in another different order may, or may not, convey the same meaning (e.g. consider the order “*in cortona italy i now live*” and the ambiguous mis-ordering “*i cortona live now in italy*”).

If each of the words, has an associated probability of occurring, $p(w_m)$, then the probability of the sentence, $p(sentence)$, can be expressed as a ‘chain’ of probabilities, $p(w_1, \dots, w_m)$, given a sequence of length m . For our example

⁶Capitalization and punctuation can be added and removed with invertible rule-based functions or learned with predictive models.

sentence, this corresponds to calculating the probability of the sequence of 6 words, $p(\textit{sentence}) = p(w_1, w_2, w_3, w_4, w_5, w_6)$. An LM can tell us the probability of this sentence occurring and how it compares to the probability of other sentences occurring. For generation, each next word is sampled from the LM based on probabilities of words which tend to follow the current word and context in a given structured collection of sentences (i.e. corpus).

We will start with a naïve LM similar to rolling a fair die. Imagine on each face of a fair six-sided die is written one of the words from our working example sentence. Each of the words has an equal chance of occurring (i.e. independent, uniform probability). The likelihood of generating our example would be calculated as the product of the independent probabilities: $p(\textit{sentence}) = \prod_{i=0}^m p(w_i) = (1/6)^6 = 1/46656 \approx 0.000021$.

There are 46,656 possible permutations⁷ of the 6 words in our vocabulary. Generating the example sentence we desire is quite unlikely. Also, it is equally as unlikely as generating our mis-ordered sentences.

From this naïve model, one improvement might be to note that the words ‘now, i, in’ might occur more often in a subset of English text than the words ‘live, italy, cortona’. So, we might weigh the probabilities of each word based on the relative likelihoods of each word occurring in a corpus. That is, we can calculate the ratio between how often a given word occurs and the total number of words in the corpus. Consider a hand-crafted, 3 sentence corpus:

‘now i live in italy’
‘specifically in italy now i live in cortona’
‘cortona italy’

It is now clear that the relative likelihood of each word occurring in the corpus is not uniform, as in the first example. In this corpus, there are 15 words, 7 of which are unique. Table 2.1 tracks the number of occurrences and relative likelihood of each unique word in the corpus.

The data in Table 2.1 defines our unigram model. This model can be used to compute our desired probability $p(\textit{sentence})$ by considering the probability

⁷Recall that order matters.

Word	Occurrences	Relative Likelihood
now	2	2/15
i	2	2/15
live	2	2/15
in	3	3/15
italy	3	3/15
specifically	1	1/15
cortona	2	2/15

Table 2.1: Unigram language modelling table of occurrences and relative likelihoods in the 3 sentence corpus.

of a term occurring given a context. Before continuing, it is helpful to reiterate Bayes' theorem and the chain rule of conditional probabilities.

Consider random words sampled from a vocabulary $w \sim \mathbf{V}$. Then, the probabilities of observing w_1 and w_2 are $p(w_1)$ and $p(w_2)$, respectively. But, the likelihood of observing w_2 might be different if we have observed w_1 ; the probability might be conditional on earlier observations.

Bayes' theorem states that the posterior probability is equal to the prior probability multiplied by the likelihood dividend by the evidence. The posterior probability is the conditional probability of w_2 occurring given w_1 . Often times the evidence (i.e. the marginal probability of the w_1) must be collected over all possible $w_2 \in \mathbf{V}$. Formally, Bayes' theorem states:

$$p(w_2 | w_1) = \frac{p(w_2) \cdot p(w_1 | w_2)}{p(w_1)} \quad (2.1)$$

$$= \frac{p(w_2) \cdot p(w_1 | w_2)}{\sum_{w_2 \in \mathbf{V}} p(w_1 | w_2) \cdot p(w_2)} \quad (2.2)$$

The chain rule of conditional probabilities states that any joint probability over many random variables may be decomposed into conditional distributions over only one variable:

$$p(w_i, w_j) = p(w_i | w_j) \cdot p(w_j) = p(w_j | w_i) \cdot p(w_i) \quad (2.3)$$

and, over multiple variables as:

$$p\left(\bigcap_{k=1}^n w_k\right) = \prod_{k=1}^n p\left(w_k \mid \bigcap_{j=1}^{k-1} w_j\right) \quad (2.4)$$

Thus, decomposition over three random words w_1, w_2, w_3 can be performed as follows:

$$p(w_1, w_2, w_3) = p(w_3 \mid w_2, w_1) \cdot p(w_2, w_1) \quad (2.5)$$

$$= p(w_3 \mid w_2, w_1) \cdot p(w_2 \mid w_1) \cdot p(w_1) \quad (2.6)$$

We might assume a “naïve” conditional independence for the current unigram model. That is, we assume that each word w_i is conditionally independent of each other word w_j for $j \neq i$. Conditional independence implies that knowledge of one word occurring provides no information about the likelihood of another word occurring. We can thus calculate our desired probability as:

$$p(\textit{sentence}) = p(w_1, w_2, w_3, w_4, w_5, w_6) \quad (2.7)$$

$$= p(w_1) \cdot p(w_2 \mid w_1) \cdot p(w_3 \mid w_2, w_1) \cdot p(w_4 \mid w_3, w_2, w_1) \dots \quad (2.8)$$

$$= p(w_1) \cdot p(w_2) \cdot p(w_3) \cdot p(w_4) \cdot p(w_5) \cdot p(w_6) \quad (2.9)$$

The model has a probability distribution over the vocabulary \mathbf{V} . The sum of the probabilities in Table 2.1 sum to 1. We indicate the unigram model in the subscript of the probability notation. Then, the probability of a target sentence under the unigram LM is calculated as:

$$p_{\textit{unigram}}(\textit{sentence}) = \prod_{w \in \textit{sentence}} p_{\textit{unigram}}(w) \quad (2.10)$$

The likelihood of our example sentence ‘now i live in cortona italy’, given our source corpus, can be calculated using likelihoods from Table 2.1 as $p_{\textit{unigram}} = (2/15) * (2/15) * (2/15) * (3/15) * (2/15) * (3/15) \approx 0.000012$. We can compare this to the naïve uniform distribution model using the expanded 7-word vocabulary of the 3 sentence corpus, $p_{\textit{uniform}} = (1/7)^6 \approx 0.0000085$.

This difference, $p_{unigram} > p_{uniform}$, represents a higher likelihood of this new LM generating sentences with a similar distribution to the corpus. The unigram model is a relatively minor and rather obvious improvement from the uniform LM.

In the unigram model, the probability of each word only depends on that word's probability of occurring in the corpus, as opposed to a uniform probability. Given Equation 2.10, an LM with a large vocabulary size, $|\mathbf{V}|$, will assign a low probability to most sentences. Minor probability, due to relative frequency, can make significant differences in language modelling.

How might our unigram LM assign a probability to the sentence 'i live in edmonton'? As the word 'edmonton' was never seen in the input corpus, what probability should the model assign? Just because the input never occurred in the training data, that does not mean it will not occur in the testing data. Several solutions have been proposed to handle words that are out-of-vocabulary. One common solution is to use smoothing by assigning a low probability to every out-of-vocabulary word [51], [198]. Another solution is to limit the vocabulary of the model by replacing each out-of-vocabulary word with a special token 'OOV' [354]. Then, as the model may see multiple 'OOV' tokens, it is possible to estimate the probability of sentences involving out-of-vocabulary words. Alternative methods include using characters, sub-word units, or byte-pair encoding methods to improve coverage by modelling the corpus at a more granular level [296].

We can extend the unigram model to an n -gram model by considering the preceding n words of context in our probability calculation. The concept is to model the structure of a corpus as the probability of words occurring alone and with context. The simplest of the n -gram models, unigram or $n = 1$, considers words in isolation (i.e. with no consideration of word order). Bigrams ($n = 2$), and higher order n -gram models ($n = 3, \dots$), consider the probability of a word occurring in the context of the preceding $n - 1$ words.

N -gram models require 2 additional important tokens, 'START' and 'END', at the start and end of the corpus sentences, respectively. These tokens help to model the fact that some words are more likely to occur at the start or end

of sentences. We modify the corpus by appending these tokens:

‘START now i live in italy END’
‘START specifically in italy now i live in cortona END’
‘START cortona italy END’

We also append these tokens to the working example target sentence which becomes: *‘START now i live in cortona italy END’*. The joint probability of the set of words can be expressed using conditional probability:

$$p(\text{‘START now i live in cortona italy END’}) \tag{2.11}$$

$$= p(\text{‘now’}|\text{‘START’}) * p(\text{‘i’}|\text{‘now’}) * \dots * p(\text{‘END’}|\text{‘italy’}) \tag{2.12}$$

$$\tag{2.13}$$

The conditional probability is calculated by dividing the number of occurrences of the sequence $|w_{i-1} w_i|$ by the total number of occurrences of the context word $|w_{i-1}|$. Absolute value notation is used to denote the number of occurrences of a word in the source corpus:

$$p(w_i|w_{i-1}) = |w_{i-1} w_i|/|w_{i-1}| \tag{2.14}$$

This calculates the relative likelihood of a pair of words, and maintains a measure of likelihood of two words occurring in a specific order. The model addresses two important issues. First, the unigram model contained no knowledge of ordering, with our new model the probability of the word pair ‘now i’ is higher than the probability of the word pair ‘i now’. This appropriately reflects the corpus statistics. Also, the unigram model may generate repetitions of high-probability words (e.g. ‘now now now ...’), our improved model accounts for this by assigning these repetitions a low probability.

The probability of the first word is conditional on being preceded by the ‘START’ token, $p(w_1|\text{‘START’})$, and similarly for the last word, $p(\text{‘END’}|w_m)$ for a sentence with m words. Table 2.2 presents word pair occurrence counts, note that only bigrams from the corpus are included. All others bigrams are of frequency 0, thus a smoothed relative likelihood near 0 might be assigned. Note that given a 3 sentence corpus, there are 3 occurrences of both the ‘START’ and

Bigram	Occurrences
START now	1
now i	2
i live	2
live in	2
in italy	2
italy END	2
START specifically	1
specifically in	1
italy now	1
in cortona	1
cortona END	1
START cortona	1
cortona italy	1

Table 2.2: Bigram language modelling table of occurrences and relative likelihoods of the modified 3 sentence corpus.

‘END’ tokens. The rest of the necessary occurrences for probability calculation are in Table 2.1. It is now straightforward to calculate the probability of the example sentence under the bigram model:

$$\begin{aligned}
& p(\text{‘START now i live in cortona italy END’}) \\
&= p(\text{‘now’}|\text{‘START’}) \cdot p(\text{‘i’}|\text{‘now’}) \cdot p(\text{‘live’}|\text{‘i’}) \cdot p(\text{‘in’}|\text{‘live’}) \\
&\quad \cdot p(\text{‘cortona’}|\text{‘in’}) \cdot p(\text{‘italy’}|\text{‘cortona’}) \cdot p(\text{‘END’}|\text{‘italy’}) \\
&= \left(\frac{|\text{‘START now’}|}{|\text{‘START’}|}\right) \left(\frac{|\text{‘now i’}|}{|\text{‘now’}|}\right) \dots \left(\frac{|\text{‘italy END’}|}{|\text{‘italy’}|}\right) \quad (\text{Eq. 2.14}) \\
&= (1/3) \cdot (2/2) \cdot (2/2) \cdot (2/2) \cdot (1/3) \cdot (1/2) \cdot (2/3) \\
&= 1/27 \approx 0.037\dots
\end{aligned}$$

There are still limitations to the bigram model. Depending on smoothing values, simple and correct sentences may have very low probability (e.g. ‘cortona in italy’). Also, there is a non-zero probability that, when sampling from the model for sentence generation, the sentence never reaches the ‘END’ token (e.g. ‘now i live in italy now i live in italy now ...’). Truncation at a certain length can help address the second problem, and comparisons between smoothing techniques are discussed in past research [51]. With these examples we have

covered the core mathematical concepts for n-gram LMs. The probability of observing w_i in context can be approximated by the probability of observing it in the shortened context history of the preceding $n - 1$ words. The probability $p(\textit{sentence})$ of observing the sentence ‘ $w_1 w_2 \dots w_m$ ’ is approximated as:

$$\begin{aligned} p(\textit{sentence}) &= p(w_1, \dots, w_m) \\ &= \prod_{i=1}^m p(w_i \mid w_1, \dots, w_{i-1}) \\ &\approx \prod_{i=1}^m p(w_i \mid w_{i-(n-1)}, \dots, w_{i-1}) \end{aligned}$$

The conditional probability of a sentence is then calculated using n-gram occurrences and frequency counts:

$$p(w_i \mid w_{i-(n-1)}, \dots, w_{i-1}) = \frac{|w_{i-(n-1)} \dots w_{i-1} w_i|}{|w_{i-(n-1)} \dots w_{i-1}|}$$

It may be helpful to contextualize this model by comparison with other sequence memory techniques in ML. Specifically, what is the relationship between Hidden Markov Models (HMMs) and n-gram models? The n-gram model is a probabilistic LM for predicting the next item in a sequence in the form of a $(n - 1)$ -order Markov model [268]. That is, independence assumptions are made so that each word depends on only the last $(n - 1)$ words. This is a Markov model approximation of the true underlying language. Thus, an n-gram calculates $p(w_i | w_{i-(n-1)}, \dots, w_i)$. In an HMM model, the states of the Markov chain are words and transition probabilities govern the transitions from word to word. These probabilities are derived based on relative frequencies of n-grams in the corpus. Only the outcome of the transition is output, and thus the state is said to be hidden from the observer.

Given a suitable source corpus, these models can perform well for some applications. HMM models are used in performance⁸ and narrative script modelling, due to their speed of training and consistent, appreciable results [247].

This section covered how to build a conditional probabilistic LM given a source corpus. This LM can be used to generate sentences and calculate the

⁸<http://botnik.org>

probability of a given test sentence under the LM. As n-grams and HMMs have limited contextual history with which to generate responses, researchers have turned to conversational modelling techniques with increased capacity.

2.2.4 Sequence-to-Sequence Modelling

Rule-based methods can provide adequate responses to a constrained set of inputs, but the output space is limited. Neural models can generate new and interesting responses, which a controllable amount of randomness, by modelling sequences of words in a large input corpus.

Text-based conversational response generation and translation both involve the transduction of information in sequences of sentences. They both involve understanding the meaning and context of a sequence of words and the generation of a meaningful response. In dialogue, the response forwards the conversation, in translation the response contains the same information in a new language. Previous work has drawn this analogy in greater detail and used machine translation techniques for dialogue response generation [172], [280].

Ritter *et al.* (2011) defined the problem of generating locally coherent, purposeful dialogue as generating an appropriate response for any conversational input [280]. The authors describe how generating appropriate conversational responses is more difficult than translation between languages due to two main factors: 1) the large range of plausible responses, and 2) the potential lack of semantic alignment between input and output pairs [188]. Given that translation models attempt to capture input sequence information in the output, it can lead to the echoing problem [82]. The authors took two measures to avoid the system from paraphrasing the input sequence in the output sequence. They discourage lexical similarity by removing substring phrase-pairs from the training data, and penalize their model by Jaccard similarity of the input and output strings. Jaccard similarity is a rapidly calculable measure of similarity defined as the size of the intersection divided by the size of the union of two sample sets [136].

Backpropagation in Feed-forward and Recurrent Neural Networks

A feed-forward neural network is a function mapping from inputs to outputs. It is composed of computational units called artificial neurons (or nodes). It may be single layered to multi-layered, depending on the composition of the computational units. Each node a network has an associated weight and non-linear activation function. This nonlinearity allows these networks to distinguish data that is not linearly separable. A perceptron is an artificial neuron which uses the Heaviside step function as the nonlinear activation function. Multi-layered perceptrons (MLPs) consist of at least three layers of nodes: input, hidden, and output. MLPs use a supervised learning technique called backpropagation for training [282].

Backpropagation is used by the gradient descent optimization algorithm to adjust the weight of neurons by calculating the gradient of the loss function. Gradient descent is a first-order optimization algorithm for finding the minimum of a function. Backpropagation, or the backwards propagation of errors, is a method to calculate the gradient needed to update the weights (or parameters) associated to the nodes in a neural network.

Feed-forward networks have connections between the nodes which do not form cycles, or loops in the network [292]. Information flows in one direction: from input nodes, through the hidden layer nodes, to the output nodes. This property differentiates it from a recurrent neural networks (RNN) which is a sequence compatible generalization of a feed-forward network where connections between nodes form a directed graph along a sequence. At high level, RNNs have an internal state, or memory, which allows for information to persist. An RNN can be thought of as multiple copies of the same network, each passing a message to a successor network. It may help to imagine *unrolling* an RNN to reveal a chain-like sequence for feed-forward networks.

With this background, a mathematical description of an RNN is now provided. An RNN computes the conditional probability of a output sequence of length n given an input sequence of length m , or $p(y_1, \dots, y_n | x_1, \dots, x_m)$. The affine transformation is defined by the multiplication of a linear transformation, weight matrix W , in the input space and subsequent addition of a

bias vector, b , in the output space, $(W \cdot x + b)$, in the equations below bias vectors for the hidden and output layers, b_h and b_y respectively, are excluded for simplicity.

Let subscripts denote the time step (or sequence index) and superscripts denote the layer in the network. Assume states are n -dimensional, thus $h_t^l \in \mathbb{R}^n$ is the hidden state in layer l at step t . Then, given a source sentence S of length m , represented by the sequence of input vectors $S = (x_1, \dots, x_m)$, a single hidden layer RNN computes a sequence of hidden states (h_1^l, \dots, h_m^l) and a target sequence of output vectors $T = (y_1, \dots, y_n)$ of length n by iterating over the following equations for $t = 1 \dots n$:

$$h_t^1 = \sigma(W^{hx} x_t + W^{hh} h_{t-1}^1) \quad (2.15)$$

$$y_t = W^{yh} h_t^1 \quad (2.16)$$

In these equations, h_t^1 is the hidden state vector at time step t , and the activations h_t^1 predict y_t . W^{hx} is the input-to-hidden weight matrix, W^{hh} is the hidden-to-hidden weight matrix, and W^{yh} is the hidden-to-output weight matrix. h_{t-1}^1 is initialized at time $t = 1$ with an initialization vector [317]. Note the use of the sigmoid activation function, $\sigma(\cdot)$. This non-linear activation function is bounded, differentiable, real function defined for all real input values and has a non-negative derivative at every point. Other, activation functions may be used—the hyperbolic tangent, *tanh*, or the rectified linear unit (or ReLU), for instance—and may provide training speed and performance improvements [240].

An RNN model such as this was used by Cho *et al.* (2014) with a fixed-sized vector, and Karpathy (2015) has illustrated the capacity of these models to generate compelling outputs.⁹ One of the limitations of RNNs is capturing long term dependencies. This limitation was directly addressed when Hochreiter and Schmidhuber (1997) introduced the Long Short Term Memory cell, or LSTM [123]. The LSTM was designed to avoid the long-term dependency

⁹<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>

problem by using a more powerful update equation and improved backpropagation dynamics. The cells were designed to ‘remember’ information over many steps.

The key idea behind LSTMs is that each individual LSTM cell has a notion of state. The state allows for information to flow through the cell unchanged, it also has capacity to remove or add information to the cell state by using information gates. An LSTM can be compared to the basic RNN cell by comparing the mathematical equations underlying the information flow of the input-output mapping functions. Breaking the LSTM down into three components (or gates) clarifies cell dynamics; we have the 1) forget gate, f , 2) input gate, i , and 3) output gate, o . c represents the memory cell state. Below, matrix W^q represents input weights and U^q represents recurrent weights where $q \in (f, i, o, c)$. Bias vectors, $b_q \in \mathbb{R}^h$, are excluded for clarity.

$$f_t = \sigma(W^f x_t + U^f h_{t-1}) \quad (2.17)$$

$$i_t = \sigma(W^i x_t + U^i h_{t-1}) \quad (2.18)$$

$$o_t = \sigma(W^o x_t + U^o h_{t-1}) \quad (2.19)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W^c x_t + U^c h_{t-1}) \quad (2.20)$$

$$h_t = o_t \odot \tanh(c_t) \quad (2.21)$$

Initial values are defined as $c_0 = 0$ and $h_0 = 0$ and the operator \odot denotes the Hadamard or element-wise product. Also, $x_t \in \mathbb{R}^d$, $f_t, i_t, o_t, h_t, c_t \in \mathbb{R}^h$, $W \in \mathbb{R}^{h \times d}$, and $U \in \mathbb{R}^{h \times h}$. Weights matrices W^q and U^q , and bias vectors b_q are the parameters learned during training.

The parameters of an RNN with LSTM units can be trained, on a set of training data, using gradient descent with backpropagation. This update iteratively modifies the weights of the network in proportion to the derivative of the error with respect to the weight. Using gradient descent for standard RNNs can result in vanishing or exploding gradients [122]. With LSTM units, as errors propagate backwards from the output layer, the error remains in the LSTM unit’s cell. This maintains the propagation of error back to each of the

LSTM gates [123].

RNNs with LSTM cells, often called LSTMs for short, can be used for both the encoder and decoder (Figure 2.2). They may be unidirectional or bidirectional (scanning over input in one or both directions) [294], single or multi-layered (network depth). We now provide details on designing, training, and using sequence-to-sequence neural network models.

Sequence-to-Sequence Models Neural response generation models use a neural network to map input (or source) sequences of words, $S = x_1, x_2, \dots, x_m$, to output (or target) sequences of words, $T = y_1, y_2, \dots, y_n$. We will refer to these models handling sequences of words, but these models can work on other text units (e.g. subwords, characters, word pairs). Also, length of these sequences (n and m) need not be equal. Sequence-to-sequence models (SEQ2SEQ) are one form of neural response model which have found success in machine translation, speech recognition, text summarization, and dialogue generation [15], [52], [302], [313], [318]. SEQ2SEQ models maximize the probability of generating a sequence of target words given a sequence of source words. SEQ2SEQ models make it possible to condition the response on a longer historical context than previous models [280].

These models learn conversational dialogue response generation by predicting responses and using a maximum-likelihood estimation (MLE) objective function [257], [318]. The model is trained to maximize the probability of the target sentence given the source. As these probabilities can be small in practice, it is more numerically stable to work with the logarithm of the likelihood (log-likelihood).

A SEQ2SEQ model learns parameters, θ^* , which maximize the conditional log-likelihood of producing the correct output T given input S for all pairs in a set of training data \mathcal{D} :

$$\theta^* = \arg \max_{\theta} \frac{1}{|\mathcal{D}|} \sum_{(S,T) \in \mathcal{D}} \log p_{\theta}(T | S) \quad (2.22)$$

As the model is often differentiable, gradient-based training such as mini-batch stochastic gradient descent can be used to estimate model parameters [52]. A modified version of backpropagation, Backpropagation through time (BPTT), is used for RNNs as parameters are shared across time steps in the network. The gradient calculated at each output depends not only on the current time step, but also previous time steps.

The trained model can be used to assign a perplexity to a given source-target pair (S, T) . It can also be used for generation by predicting the most likely target, \hat{T} , according to the model, formally:

$$\hat{T} = \arg \max_T p_\theta(T|S) \quad (2.23)$$

The most likely targets are found using a left-to-right beam search decoding scheme. Beam search maintains a number, B , of partial target hypotheses [318]. Beam search decoding is an approximate, greedy algorithm with uses a best-first heuristic search to explore a possibility space by expanding the most promising hypotheses.¹⁰ At each time step the prefix hypotheses are extended with each word in the vocabulary. Using the log probability calculated with the model, all but the B most likely hypotheses are discarded. When ‘END’ token is appended to a hypothesis, that hypothesis is removed from the beam search and added to a complete candidate list. This complete hypothesis list may then be re-scored using an additional scoring function [318].

This MLE objective may provide likely responses, but, it is likely not the optimal surrogate objective for a dialogue agent [176]. Analogous to the repetition problem due to loops in n-gram models described above, MLE-based SEQ2SEQ models may not account for repetition. As well, SEQ2SEQ models tend toward generic responses (e.g. ‘no’ or ‘i dont know’) given their relatively high frequency in many source corpora [176], [177], [301].

Encoder - Embedding - Decoder Model The objective of sequence prediction models is to predict an output sequence given an input sequence.

¹⁰<http://www.dtic.mil/dtic/tr/fulltext/u2/a049288.pdf>

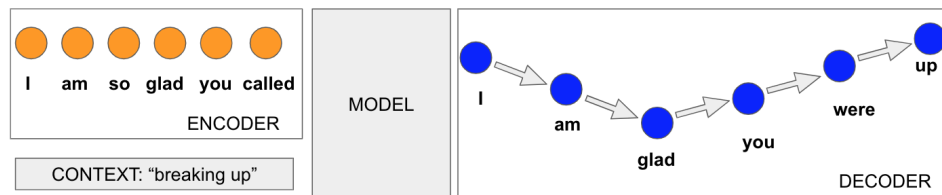


Figure 2.2: Figure of the encoder / decoder framework.

SEQ2SEQ models serve as the basis for many current chatbots [299], and these models use similar architectures (illustrated in Figure 2.2). At a high level, SEQ2SEQ models are composed of an input encoder (which maps words into fixed length embeddings), an embedding processing model (generates responses embeddings to inputs), and a output decoder (which maps embeddings to words). Additional context information might be appended to the inputs. There are many classes of sequence modelling problems based on the relative shapes of input and output sequences.¹¹ Generally, there is an input sequence of multiple tokens mapped to an output sequence of multiple tokens. We may want to map a sequence input to a single class, or alternatively, map a single token to a sequence of output tokens. These are special cases of the general sequence modelling problem.

The encoder maps a variable-length input sequence (source) to a fixed-length vector. The decoder maps a fixed-length vector to a variable-length output sequence (target). Often the encoder and decoder are separate functions, but they may share weights to speed-up learning. This technique is particularly useful when using deep neural network networks with many parameters [52], [168].

Similar to the n-gram model, a vocabulary list defines the words that the model can encode and decode. In addition to the special tokens introduced prior (‘START’, ‘END’, ‘OOV’) neural models may also utilize the ‘PAD’ token. This token allows for batches or source/target pairs to be padded to constant width. This processing technicality improves performance of models at test time. Fixed dimensionality embeddings allows for the use of deep

¹¹<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>

neural networks (DNNs), which have been shown to be powerful models on many language tasks [70], [216], [318].

LMs are partly inspired from theories of context-free grammars and generative grammars, introduced by Chomsky (1959) coupling syntax and semantic information in sentences [53]. Grammar is a set of rules governing the composition of words in a language. Each language is based on a grammar. The order of words has influence on the meaning of the sentence. Discriminating the grammar underlying sentences is paramount when parsing sentences for language understanding [267], [287]. In fact, neural language models have been used to track abstract hierarchical syntactic structure [102] and learn context-free grammars [295]. Learning semantic and syntactic information will improve language modelling. These works support the notion that neural models can learn to track grammatical structure and meaning; a hypothesis which motivated early neural network research [77], [283].

2.2.5 Neural Model Improvements

In the SEQ2SEQ model, the last source state is passed from the encoder to the decoder when starting the decoding process. The single fixed-size hidden state becomes an information bottleneck for long sentences. Similar in motivation to the LSTM, the concept of ‘attention’ was introduced to give the decoder more direct access to the information in the input [15], [197]. The key idea of attention mechanisms are to establish direct short-cut connections between the target output and the source input by attending to relevant source content during translation. Instead of discarding all hidden states computed in the source RNN, attention allows the decoder to peek at them (treating them as a dynamic memory of the source information). The attention mechanism improves the translation of longer sequences [15], [197]. Attention mechanisms have been applied to many sequence modelling tasks including image caption generation, speech recognition, and text summarization.¹²

The Transformer model is a sequence modelling neural network architecture based on self-attention [335]. The self-attention mechanism directly mod-

¹²<https://ai.googleblog.com/2017/07/building-your-own-neural-machine.html>

els relationships between all words in a sentence, regardless of their respective position. The Transformer model assigns every word in the sentence with a relative position encoding. These positions are added to the embedded representation. The benefits of the Transformer model include increased quality of sequence generation, better parallelization on modern hardware, better generalization, and less time to train [334], [335]. The Transformer model has outperformed recurrent and convolutional networks a diverse set of sequence modelling tasks [335]. The open-source Tensor2Tensor Transformer framework¹³ was used for the experiment in Chapter 5.

Thus, we come full circle. From naïve rule-based chatbots, to n-gram and Hidden Markov Models (HMMs), to finding inspiration in the methods of statistical machine translation, and advancing neural response generation with the long-short term memory (LSTM) and Transformers. Additional considerations for corpus based chatbots and neural model training are included in Appendix C. Many neural modelling techniques and experiments would not be possible without two crucially important additional factors: suitable datasets and rigorous evaluation.

2.2.6 Datasets

There is an adage in computing science that *garbage-in leads to garbage-out* [14], and so it is with conversational dialogue systems. These systems and continue to rely on large corpora of high-quality data collected, indexed, and accessible by the model at training and inference time. There are many large datasets available to train dialogue systems [298]. This survey from Serban *et al.* (2015) attempts to collect these datasets and profile their characteristics to ease in selection and use. Data collection and sharing is an active area of research in the field, and with the continued deployment of conversational systems, these datasets will continue to grow in scale and diversity of language, task, and structure.

Datasets used in this thesis work include the Cornell Movie-Dialog Corpus (CMDC) and OpenSubtitles (OS). The CMDC [65] contains 220,579 conver-

¹³<https://github.com/tensorflow/tensor2tensor>

sational exchanges between 10,292 pairs of movie characters 9,035 characters from 617 movies for a total of 304,713 utterances. The OS dataset¹⁴, while considerably larger at 446,612 movie subtitle files, covering more than 110,000 unique movies identifiers (or IMDB IDs)¹⁵ for a total of 441.5 million lines, is noisier data (e.g. duplicates, spelling and grammar mistakes) [185]. It consists of user contributed movie and TV subtitles which are then processed with an automated processing pipeline. OS constitutes a resource for the conversational dialogue. From a linguistic perspective, movie subtitles cover a wide range of genres, from colloquial language or slang to narrative and expository discourse as in documentaries [185].

2.2.7 Evaluating Dialogue Systems

How do we go about evaluating dialogue systems when the task is not related to a binary outcome, as is common in goal-oriented or non-conversational dialogue? This question is of particular interest for the open-domain dialogue systems discussed in the following chapters.

Chatbots can be separated into open- and closed-domain systems. Open-domain dialogue systems (i.e. chit-chat bots, chatter bots) can discuss a diverse set of topics and domains. Closed-domain systems (i.e. domain specific, task- or goal-oriented bots) limit conversation to a specific domain or task. Evaluation of task-directed agents is considered more straightforward than for open-domain bots [270], [338].

Interlocutors engaged in chit-chat conversation often do have goals.¹⁶ Cognitive science and conversational linguistics has explored the many underlying goals of conversational dialogue [161], such as: generating shared knowledge, obtaining information, delivering action instruction, understanding motivation, and/or building relationships. These goals are more abstract than an closed-domain, outcome-centric, booking task (e.g. “*reserve a table for 4 at 6:00pm at the Italian restaurant down the block*”). Goals also change over the

¹⁴<http://opus.nlpl.eu/OpenSubtitles2018.php>

¹⁵<https://www.imdb.com/>

¹⁶Conversational partners.

course of a dialogue [161].

The goals listed above are a step towards evaluation, but they are still underspecified. What we desire is a testable condition for success; a measure allowing us to compare one model against another. In the chapters that follow, we describe how specific concepts from ‘improvised theatre dialogue’ can provide these measures. Before discussing these, it is helpful to provide background on current evaluation techniques.

Early work in dialogue generation used human interactions to test systems (Appendix B). When automated optimization of these models became necessary, due in part to the large number of parameters and hyperparameters, the community looked to statistical machine translation (SMT) for metrics [280]. As Liu *et al.* (2016) show, these metrics correlate weakly with human judgments [188]. Additional work has shown weak correlations between humans and other automatic metrics [193].

Automated metrics from SMT [296], including overlap-based methods such as Jaccard similarity, BLEU [251], METEOR [17], ROUGE [183], CIDEr [336], are used for machine translation, summarization, and image captioning tasks. These methods assume a valid, aligned, output for a given input has significant word overlap with the ground truth response. For instance, bilingual evaluation understudy (BLEU) evaluates the quality of text which has been machine-translated from one natural language to another. While this inspired model innovation and development for previous research [280], word overlap need not exist in conversational dialogue [100].

Training data specifies one notion of naturalness, so we can measure how well the output data matches the input data. Evaluating an LM is often done by measuring the perplexity [304], a measurement of how well a probability distribution predicts a given sample. This is a useful extension of the likelihood calculations from Section 2.2.3. Recall that an LM represents a probability distribution over a given corpus. For a given LM, low perplexity indicates a high likelihood, conversely, high perplexity indicates a low likelihood sample.

Entropy is a measure of the expected number of bits required to encode the outcome of a random variable. Assume we have a sequence of words,

w_1, w_2, \dots, w_m from a vocabulary \mathbf{V} . The probability of a given word, w_i , is dependent on the previous words, w_1, \dots, w_{i-1} . The per-word entropy, H , represents the new information (in bits) provided by each word on average:

$$H = - \lim_{m \rightarrow \infty} \frac{1}{m} \sum_{w_1, \dots, w_m} \left(p(w_1, \dots, w_m) \cdot \log p(w_1, \dots, w_m) \right) \quad (2.24)$$

Where the sum is over all word sequence permutations. If we assume the LM to be ergodic and stationary, then for large values of m , the approximate entropy is [3]:

$$\hat{H} = -\frac{1}{m} \log p(w_1, \dots, w_m) \quad (2.25)$$

An ideal LM would have a per-word entropy of H . We use a measure related to entropy to compare language models. The perplexity of a probability distribution, p , is related to the entropy, H , of a distribution (in bits):

$$\text{perplexity} = 2^{\hat{H}} \quad (2.26)$$

$$= \hat{p}(w_1, \dots, w_m)^{-\frac{1}{m}} \quad \text{by Eq. 2.25} \quad (2.27)$$

$$= \sqrt[m]{\frac{1}{\hat{p}(w_1, \dots, w_m)}} \quad (2.28)$$

$$= \sqrt[m]{\prod_{i=1}^m \frac{1}{\hat{p}(w_i | w_1, \dots, w_{i-1})}} \quad \text{by chain rule} \quad (2.29)$$

$$(2.30)$$

where $\hat{p}(w_1, \dots, w_m)$ is the probability estimate of the sequence of words, w_1, \dots, w_m , estimated by the LM.

While perplexity provides a measure of the likelihood of a response based on a training corpus, and a means to compare trained language models, there are limitations. Plausible responses with out-of-vocabulary words might be assigned a low likelihood. Also, while perplexity is meaningful and available for probabilistic LMs (i.e. generative models capable of assigning a probability to a candidate response), some models are not applicable to this metric. Thus,

it is difficult to use this metric for model comparison. Additionally, an naïve echoing model (i.e. which repeats inputs) would generate ‘good’ responses under this notion [82]. These outputs are well within the input distribution, but the conversation lacks a quality familiar to anyone who has conversed. The responses do not add additional information to the conversation. We need to account for the fact that good conversation incrementally, collaboratively builds information at each conversational step.

Humans (i.e. crowd-sourced) have generated ‘good’ responses and/or provided scores for the model generated responses. For instance, raters may be shown context and then pick their preferences between candidate outputs [188]. This technique is costly in both time and money. Also, human raters can be subjectively biased (e.g. toward short responses) and can have high variance between raters based on personal differences (e.g. reading ability).

Additionally, it is difficult to ask the right questions to get meaningful, reproducible measures from crowd raters [73]. Several studies have asked questions related to overall appropriateness of response, whether the response is on-topic, whether the response is specific, and whether the response requires additional background information [178], [272], [300], [338]. Researchers have also asked raters if a given conversation would be one they want to continue [154].

With enough humans agreeing on ratings for an input/output pair, it is perceivable that a model could be built to predict human ratings on unseen inputs [73], [193]. These works allow for the prediction of human ratings, which can allow for automated testing and comparison of model improvements without costly human-participant studies.

It is paramount to test models with real-life human interaction developing models designed for human interaction. Competition-based evaluations (discussed in Appendix B) include ‘wild’ evaluations which involve human conversations with bots. Following the conversation, the human converser scores the interaction based on various measures which, depending on the competition, which might include measures of ‘adequacy’, ‘informativeness’, ‘fluency’, ‘consistency’, and ‘engagingness’ or more abstract measures of ‘how much a person liked the conversation’ or ‘how well the bot stuck with a consistent

persona' [127], [250], [364]. Additional evaluation methods are presented and discussed in Appendix C.1.

Systems can model the personality [364], mood [93], and emotion [47] of generated sentences as well as human affect in interactive settings [362]. By modelling the psychological state of the user, and adapting accordingly (much like human interlocutors do) conversational systems will continue to evolve and personalize [50]. Ongoing appraisal and adaptation required significant language and non-verbal interaction understanding. The transition from building systems which reflect the underlying training dataset, to systems which adapt pre-trained models to ongoing interaction, is an ongoing area of research and discussed in detail in Chapter 7.

It is important to evaluate from both human-centric and machine-centric perspectives. The evaluation of dialogue systems continues to be explored as an active area of research [188]. Dialogue systems should be measured and evaluated based on the interaction that they facilitate. In the work that follows in future chapters, we lean towards evaluating the dialogue system by evaluating the application where they are used. By measuring qualities of the application of a model down-stream, we gain valuable insights to compare models (Chapter 3).

2.2.8 Conversational Dialogue Summary

This section provides an review of the history, the implementation, and evaluation of conversational dialogue systems. This chapter has covered interactive machine learning (Section 2.1) and dialogue systems (Section 2.2), but they have remained loosely coupled. There is a missing connection in the loop. That missing piece is a real-time interactive test bed for conversational dialogue systems. The next section provides background and motivation for an ideal testing ground for IML-based conversation systems: the applied Turing test of live improvisational theatre.

2.3 Improvisational Theatre

Improvisational theatre (or improv) is a form of theatre in which some or all components of the performance are unplanned or unscripted. Improvisation is performing without any preparation or planning [109]. Improv is theatre conceived, written, directed, and performed in the same moment often in front of a live audience. An abridged history of improvising intelligences is included in Appendix D.

The ‘offer’ is a common term across improvisational theory [147]. It defines the basic unit of collaboration between individuals. Offers can be physical, mental, or verbal. For the majority of the work in this thesis, I focus on verbal offers with physical embodiments (e.g. robots and projections), extensions to additional means of delivering offers are discussed in Chapter 7. All offers can be characterized by how much they accept and how much they extend [147]. These properties are sometimes summarized with the concept of “Yes, and...”, common in improvisation.

When given an offer in an improvisational scene, an improviser will respond with an offer of their own. The responding offer can either ‘accept’ or ‘reject’ the initial offer. Additionally, the response can either ‘extend’ (build on) or ‘advance’ (move past) the initial offer. Improv scenes are composed of sequences of offers, each verbal offer is a sequence of words with additional features of character: intonation, speed, and directness. These sequences of words, and sequences of offers, will be what interactive artificial improvisors model and generate using the techniques from Section 2.2. Concepts of accepting and extending are further contextualized in the framework of generative language models in Chapter 5 when they are used to model underlying narrative arcs.

2.3.1 Theatre Technology and Automata

The theatrical stage is a reflection of our natural world. Much like the rest of our world, theatre is intrinsically enabled by technology [115]. Technology with which we augment our lives, including screens, computers, lights, and

videos, have thus found their way into modern theatrical productions. Technical theatre is a field which includes stage and set design, lighting controls and cues, sound design, and visual projection design. In live performance, stagecraft is as old theatre itself [108], [316]. To Shakespeare, the theatre was a space-time machine driven by the energy of imagination [115], [164]. Not only is the incorporation of technical theatre into productions functional, it is as artistic as the performance itself.

As natural as humans and technology sharing the stage is, it is important to inspect the digital-augmentation from two perspectives: 1) how can humans best be served by the technology of the production? And, 2) how can the technology inspire as a source of creativity, or co-creator, on stage? Answers to these broad questions, in regards to performative consequences of incorporating technology in live improvisation, are explored in this section. As above, it is important to appreciate history to gain perspective of the future.

The word automaton comes from Greek origins and means roughly “*acting of one’s own will*”. It was used by Homer to describe an automatic door opener [333], and has now come to refer to non-electronic moving machines. Automata are found throughout history. Often described as statue-like, but with an element of motion, these automata serve as inspiration for the robots and cyborgs of current culture. These automata stretch across culture from Greek mythology, to Jewish legend¹⁷, to ancient Chinese texts [244]. There is an almost universal appeal to these figures at the intersection between human and machine, a thread through history that connects us all.

Leonardo da Vinci, the great artist-scientist, made several automatic machines. He built a lion which automatically lurched several steps forward and opened its breast to reveal a collection of lilies. He also built a self-propelled cart (not unlike the self-driving cars of today), and a mechanical knight.¹⁸ da Vinci is a notable connection in this context as these automata are not the only automatic theatre technology he designed and built. His journals are full of engineering notes on stage craft and preparations for large theatrical

¹⁷<https://goo.gl/F1K6Fd>

¹⁸<http://history-computer.com/Dreamers/LeonardoAutomata.html>

displays for parties [135].

2.3.2 Cyborg Theatre: Stimulation Over Simulation

Cyborg theatre places robots, automata, and learning machines on the stage during a production [252]. While there are many delightful inspirations which stem from the world of cyborg theatre, there is much to suggest that robots on stage, like children or animals, attract all the attention of audiences and might steal the show from human actors when performing on the same stage [192]. Placing a cyborg on the stage confronts an audience with several challenging themes, most notably *the benefits and consequences of creation*. The creation theme progresses toward two paths: 1) the replacement of ourselves and each other, and 2) creating something that can not be contained.

Thus, cyborg theatre finds itself in a cultural context envisioning human-machine encounters as problematic and dangerous [252], likely due to fear. The fear of not being good enough, not being apt for the future, and the fear of being replaced by someone or something superior [99]. The creation of a human from clay is a theme that occurs across religions and cultures (e.g. Prometheus [111] and the golem from Jewish folklore [133]). The descriptions of the golem could just as easily be referring to a computer: ‘if commanded to perform a task, they will follow instructions literally’. As Ada Lovelace described in 1842, “the Analytical Engine has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform [32].” The notion of obedient machines, and the hubris which accompanies them, also appear in Mary Shelly’s “Frankenstein; or, The Modern Prometheus” [307], James Cameron’s “The Terminator” [38], and in Karel Čapek’s “R.U.R (Rossum’s Universal Robots)” [40]. Notably, R.U.R. is the 1921 play which coined the term robot.

There is a tension between provoking and pleasing the creator, and while cyborg theatre is filled with fear, it is also filled with a unique sense of love for metamorphosis. Such is the tension underlying the ancient Greek myth of Pygmalion who fell in love with one of his sculptures [234]. Pygmalion, and the George Bernard Shaw play of the same name [306], are the inspiration

for our first attempt at creating an artificial improviser, Pyggy, described in Chapter 3. This adoration of transformation is hopeful. It is hopeful that we humans can affect change and, in doing so, may be changed ourselves. Modern stories explore the themes of bringing a creation to life, the notion of bringing one of these systems onto the stage is another question.

2.3.3 Live Theatre with Chatbots

What does live theatre mean? Does it mean that the performers on stage are alive, that the audience is alive, or both? How much liveness must there be for it to be live theatre? A ventriloquist act would be considered live-theatre performed by a human and a puppet. But, what if the puppet acted independently with no ventriloquist? What if there is a voice that can interact with the performers on stage? What if there is nothing but a voice on stage that can interact with an audience? Imagine a stage is full of acting bots and an audience full of reacting bots. With no living beings present, would this be considered live theatre?

It is the extremes of this line of thought permeated Auslander’s mind when he stated “the existence of chatterbots reopens and re-frames the question of liveness at a fundamental level” [13]. The notion that a dialogue system, capable of conversation has a unique liveness to it that eludes definition. In a way, it is creating something from nothing. Similar to teaching a machine to do something it previously could not. These thoughts serve as the undertone for cyborg theatre with conversational dialogue generation.

As described in the history of chatbots in Appendix B, ‘ELIZA’ is often attributed as the first chatbot. It was the creator, Weizenbaum, who described ‘ELIZA’ as a machine that simulates humanity by encouraging the dialogue partner to improvise: “in a sense, ELIZA was an [actress] who ordered a set of techniques but who had nothing of her own to say. The script, in turn, was a set of rules which permitted the actor to improvise.” [345]. The astute reader will notice the name ‘ELIZA’ was chosen because it was also the name of the lead character in G.B. Shaw’s “Pygmalion” [306]. As Weizenbaum cleverly said “the program could be taught to ‘speak’ increasingly well, although, also like

Miss Doolittle, it was never quite clear whether or not it became smarter” [110].

An important questions to ask while on this line of thought is ‘*why?*’ Why would we bring a machine onto the stage? Why would we want a cyborg co-creator in creative endeavours? This section aims to address the ‘*why?*’ It is human nature to create in our own image. It challenges our thoughts and feelings about what is real. Most importantly, the machine’s “creativity” is seen in the kind of suggestions the computer makes to humans; the system serves the creator, it is both pleasing and provoking [151]. As described in Section 2.1, the ultimate goal is to bring together human and machine contributions synergistically. The performance on the objective function will be better working together than either contributor could reach alone. Creativity emerges through from the process of working interactively [182].

Theatre represents an ideal experimental setting for robotics and machine intelligence research [33], [157], [195]. The theatrical stage is a middle ground between controlled environments and complex novel environments. Routines and rehearsal are natural in the theatre, but no performance is strictly the same twice. Improvisors are masters of co-creation, specialists in processing misunderstandings, re-contextualizing to find middle ground, covering communication breaks, and making ideas connect [147], [192]. Magerko *et al.* (2009) formulated the theoretical processes needed for a computational system for improvisation [200]. That study reports several important aspects of cognition in improvisation: basic cognition, shared mental models, narrative development, and referent use, and frames improvisational theatre as real-time dynamic problem solving [146].

2.3.4 Interactive Machines in Theatre and Comedy

There is a growing intersection of researchers and artists exploring interactive machines in theatrical productions. Interactive robot theatre has been explored as a test-bed for sociable robotics [33]. Theatre robots are unique in their performances and often tailored to the audiences and productions in which they appear. That said, there is much in common between the systems. They are often built of three major components: 1) a main control software,

2) a perception stream (i.e. visual, auditory, tactile, etc.), and 3) an expressive embodiment. With these three components, we can describe many different systems which have been used in performances and research papers of the last few decades.

Robotic performances have been explored previously [33] and are discussed in Section 2.3.4. In 2000, Tom Sgorous performed “Judy, or What is it Like to Be A Robot?”.¹⁹

The silicon-based comedy, robot entertainer work of Dr. Heather Knight has been featured in many high profile media outlets [158].²⁰ Knight used a visual perception stream to track audience responses and a joke control system selected the next joke to be similar to those that the audience had responded well to. The expressive embodiment was the Aldebaran (SoftBank) Nao robot.²¹

Katevas et al. (2014) programmed and scripted a life-sized humanoid robot (the RoboThespianTM from Engineered Arts²²) as the embodiment to perform a stand-up routine [153]. During the performance gesture and gaze were modulated and the affects on the audience were measured. Notably, they found that simulated eye contact between human and robot increases positive human audience response. They also present findings that people responded more positively to human comedians than the RobotThespianTM.

RoboThespianTM also made an appearance in Jon Welch’s “Spilikan, a love story” (2015) by the Pipeline Theatre. All activities of the robot were pre-programmed and controlled by software running backstage. Timing was flexible based on cues from a controller.

Hiroshi Ishiguro’s lifelike android embodiments were featured in two theatre pieces: “I am a worker” (2008) and “Sayonara” (2010) [263]. Both collaborations between Ishiguro and director Oriza Hirata, they aimed for the Japanese concept of “son-zaikan” or the feeling of a presence of a person. Ishiguro excels at lifelike embodiments and the control of the robots was pre-programmed,

¹⁹<http://sgouros.com/judy/>

²⁰https://www.ted.com/talks/heather_knight_silicon_based_comedy

²¹<https://www.softbankrobotics.com/emea/en/robots/nao>

²²<https://www.engineeredarts.co.uk/robothespian/>

rehearsed, and timed. It is unlikely these androids had any perceptual stream.

Duncan *et al.* (2010) performed Shakespear’s Midsummer Nights Dream with several of the fairies played by robots and surveyed the audience about believability [71]. In 2014, Carnegie Mellon University’s Personal Robotics Lab and School of Drama collaborated to stage a production of David Ives’ “Sure Thing” with HERB [361]. The robot was controlled live on-stage by a human in plain sight of the audience. More interactively, the robot Myon made an appearance in “My Square Lady”, a 2015 opera at the Komische Oper Berlin.²³ Myon was able to improvise by responding to the live situation with small gestures [171].

These systems serve as a backdrop for the chapters to follow and each of them provide valuable lessons about bringing a robot to the stage. Through engineering efforts, robotic platforms are becoming less expensive, more robust, responsive, and powerful. These qualities are inspiring a generation of creative artists and scientists to build novel interactions.

2.3.5 Creative Interactive Machine Learning

Creative interactive machine learning (IML) is an applied subfield focused on with creative artistic expression. This growing field has seen artists from diverse disciplines adopt and adapt cutting edge ML science to extend human creativity. There has been incredible leaps made in machine generated visual art. Notably, the “Portrait of Edmond Belamy” (2018) created by GAN (Generative Adversarial Network) was sold for \$432,500 a Christie’s auction in October 2018.²⁴ While visual art can dominate coverage in creative IML, there exists a growing number of creative technologists integrating these systems into their non-visual contemporary work.

These include designing real-time expressive interactions in a variety of media including: story, joke and poem co-creations²⁵; musical interaction²⁶

²³<https://www.youtube.com/watch?v=jWBN9627mSg>

²⁴<https://goo.gl/QW6Dy9>

²⁵<http://botnik.org>

²⁶<https://magenta.tensorflow.org/demos/>

and production²⁷; interactive kinetic light projections [191]; interactive story tellers [275]; dynamic video game storyline generation [327]; expressive movements in stage robots [33]; and playful swarm robots [68]. There is also work on tuning trained models using interactive RL for musical melody and facial recognition to enhance interactive creative doodle generation [137], [138]. There is exciting work which introduces tracking of emotional and narrative arcs in text, audio, and video [55]. This tracking can be used to shape sequence generation as we present in Chapter 5.

Creativity is a subjective quality assigned by the observing audience based on societal values [149]. As Margaret Wolfe Hungerford said in 1878, ‘beauty is in the eye of the beholder’. Creative sparks from these IML systems ignite when curated by humans. Artists continue to embrace these technologies to further develop and present explorations of the human condition. When asked what the end goal is for work in *artificial improvisation*, we joked: “when two robots are on stage, performing theatre for an audience of robots, we’ll know all this work was done vain.”²⁸ These systems do not aim to remove the human. They exist to augment human capabilities and creativities, and to inspire humans to create novel interactions.

2.4 Interaction, Interlocution, Improvisation

This background section provides relevant supporting material and information in the three areas at which thesis intersects. It covers an overview of the fields of IML, conversational dialogue systems, and improvisational theatre. It details how creative, interactive machine learning will continue to develop. It posits that the setting of improvisational theatre is the ideal test bed for experimentation with conversational IML systems.

Through the following chapters, we formalize the challenge of improvising alongside artificial intelligences, explore the imitation game using machine learning, improve collaborative narrative generation using information-

²⁷<https://www.flow-machines.com/>

²⁸<https://blog.mastermaq.ca/2017/02/09/recap-democamp-edmonton-34/>

theoretic methods, and present a method to synthesize knowledge to generate coherent directed plots. In these chapters, we evaluate how collaborative systems are perceived by performers and observing audiences. We present challenges and improvements along the iterative process. Finally, we synthesize and analyse critical reception and present future directions to use humour-in-the-loop to improve human-machine interaction for improvised theatre with interactive machine learning systems.

Chapter 3

Improvised Theatre Alongside Artificial Intelligences

I conceived of an instrument that would create sound without using any mechanical energy, like the conductor of an orchestra. The orchestra plays mechanically, using mechanical energy; the conductor just moves his hands, and his movements have an effect on the music artistry.

– Lèon Theremin, 1989

In this chapter, we present the formalization of artificial improvisation. This chapter builds on the background material in the preceding chapter by addressing the grand challenge of human-machine conversational dialogue interaction in improv. Human-machine improvised theatre is a suitable and challenging testing environment and application for interactive conversational dialogue experimentation. As a preliminary evaluation of this application, this chapter details two intelligent machines trained to perform live shows alongside human actors for human audiences. First presented is Pyggy, a rule-based system which uses fuzzy string matching and information retrieval methods optimized for speed and efficiency. Second is A.L.Ex. (the Artificial Language Experiment), a corpus based neural response generation system using recurrent neural networks and optimized for maximum likelihood estimation. Once

This chapter is adapted from **Mathewson KW** and Mirowski P. (2017) Improvised Theatre Alongside Artificial Intelligences. 13th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE'17). Snowbird, USA. October 2017.

these systems are trained they are evaluated through performance of theatrical improvisation alongside human performers. This chapter presents details on the methods underlying both systems. The evaluative results and discussions and conclusions support the innovations in the following chapters.

This chapter presents the iterative development of artificial improvisation, or improvisational theatre performed live, on-stage, alongside an artificial intelligence-based improvisational performer. As introduced in the previous chapter, the artificial improviser is a form of artificial conversational agent, or chatbot, focused on open domain dialogue and collaborative narrative generation. Using machine learning techniques spanning from natural language processing and speech recognition to reinforcement and deep learning, chatbots have become more capable of common conversational tasks. Work in conversational agents has been focused on goal-directed dialogue focused on closed domains such as appointment setting, bank information requests, question-answering, and movie discussion [10]. Natural human conversations are seldom limited in scope, jump from topic-to-topic, and are often laced with metaphor and subtext [145]. Face-to-face communication is supplemented with non-verbal cues [214]. Live improvised performance can be framed as a natural conversation task where multiple performers collaborate to generate narrative in real-time front of an audience. In improvisation the topic of the conversation is often given by the audience several times during the performance [147]. These suggestions inspire actors to perform novel, unique, and engaging scenes. During each scene, actors must make rapid fire decisions to collaboratively generate coherent narratives. We introduce Pyggy and A.L.Ex. (Artificial Language Experiment), the first two artificial improvisors, each with a unique composition and embodiment. This chapter highlights research and development, successes and failures along the way, celebrates collaborations enabling progress, and presents discussions for future work in the space of artificial improvisation.

3.1 Introduction

Improvisational theatre, or improv, is the spontaneous creation of unplanned theatrics, often performed live on-stage in front of an audience. Improv is a form of collaborative interactive storytelling, where performers work together to generate novel narratives. It is grounded in the connections between the performer(s) and the audience. Improv requires the performers to work as a team. The actors must rapidly adapt, empathize, and connect with each other to achieve natural, fluid collaboration. To truly excel at the art-form, performers must think and react to audiences reactions quickly, and work together to accept and amplify each other's offers—an act that can be seen as *real-time dynamic problem solving* [200]. Improv demands human performers handle novel subject matter through multiple perspectives ensuring the audience is engaged while progressing narrative and story. Due to the incredible difficulty, improvisors must embrace failure and surrender to spontaneity [147].

Improvised theatre has been a platform for digital storytelling and video game research for more than 20 years [112], [256]. Past research has explored several knowledge-based methods for collaborative storytelling and digital improvisation [199], [246], [310], [363]. Similar work explores how humans interact with system which improvise music and dance [125], [326]. Computer aided interactive storytelling has been considered for applications in video games with an aim to create endless narrative possibilities in video game universes for user engagement [275].

Scripted robotic performances, in comedy and drama, have been explored previously and are described in detail in Section 2.3.4. In previous performances, robots were precisely choreographed, deterministic, or piloted on stage. These shows required the audience to suspend disbelief and embrace the mirage of autonomy. These performances verge ever closer to the deep cliffs surrounding the uncanny valley—the idea that as the appearance of a human-like robot approaches, but fails to attain, human likeness, a person's response would abruptly shift from empathy to revulsion [236].

This chapter presents a bridge between the artificial intelligence labs and

improvisational theatre stages. The bridge crosses over the uncanny valley, toward a future where humans and autonomous agents converse naturally together. Our work is partially inspired by the narratives behind George Bernard Shaw’s “Pygmalion” [306], Mary Shelly’s “Frankenstein” [308], and Alan Jay Lerner’s “My Fair Lady” [62]. In these stories, creators attempt to design and build reflections of themselves, fabricating their respective ideal images of perfection. These themes are discussed in detail in Section 2.3.2.

We present methods underlying the systems which power the first two artificial improvisors. We concisely report on findings, and discuss future work at the intersection of artificial intelligence and improvisational theatre.

3.2 Methods

This section describes the methods underlying *Version 1: Pyggy*, using classic machine learning and deterministic rules, and *Version 2: A.L.Ex.* which uses deep neural networks, advanced natural language processing, and a much larger training dataset. While improvisational theatre is a complex art-form mixing dialogue, movement, and stagecraft, and there exist many improvisational rules for the novice improviser [147], in this chapter we focus on a single component: training the dialogue system. An artificial improvisor dialogue system is composed of three major building blocks (Figure 2.1): 1) speech recognition and natural language understanding (NLU), 2) speech synthesis and natural language generation (NLG), and 3) a dialogue management system with access to external knowledge bases. The three modules comprise a simplified framework, inspired by the *General Architecture of Spoken Dialogue Systems*, for extemporaneous dialogue systems [258]. We detail these components for both Pyggy and A.L.Ex.

3.2.1 Version 1: Pyggy – Rule Based Generation

Pyggy, short for Pygmalion [306], is the first version of a chatbot-based artificial improvisor.¹ Pyggy was built using speech recognition powered by

¹<http://korymathewson.com/building-an-artificial-improvisor/>

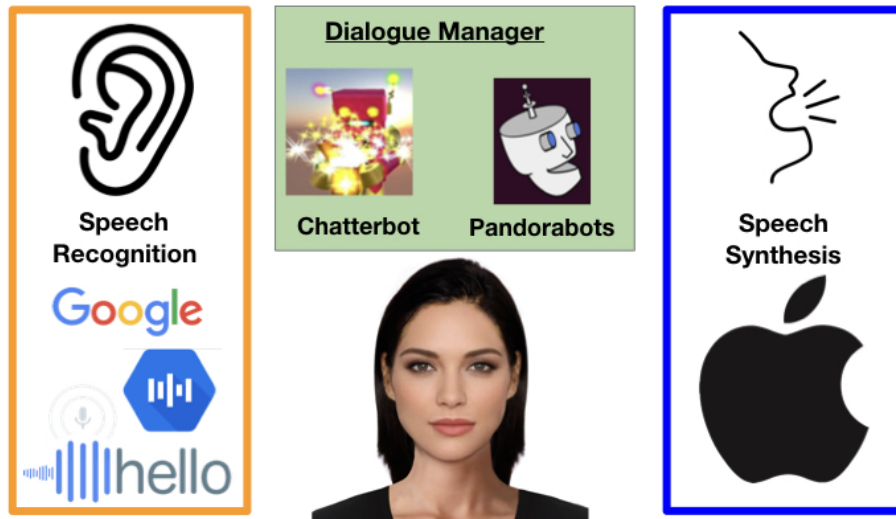


Figure 3.1: High level system diagram of Pyggy.

Google Cloud Speech.² Speech recognition translates sound waves from human voice, to text through a network-dependent application programming interface. Pyggy used Apple Speech Synthesis for translated output text to sound. This allowed for multiple synthesized voices to be used. Dialogue management was handled with Pandorabots³ and Chatterbot open-source software packages.⁴

For each human improviser’s verbal offer, each of these systems would generate a response, which were then concatenated and output to the user. Pandorabots handled hard-coded rules and deterministic responses. For example, when the human offered: “Let’s start improvising”, the system would *always* respond: “Ok”. Pandorabots also handled saving named entities. For example, if the human said: “My name is Lana” then the system could answer the recall question “What is my name?” with the correct answer: “Your name is Lana.” Chatterbot was introduced to handle open dialogue generation and add randomness to the system. Chatterbot was pre-trained on a set of dialogue, as described below, and then “learned” based on responses the human gave back to the system (similar to Cleverbot, described in Appendix

²<https://cloud.google.com/speech/>

³<http://www.pandorabots.com/>

⁴<https://github.com/gunthercox/ChatterBot>

B.7). It works by fuzzy matching the input string with strings in its dictionary and then producing the closest match to that input. Fuzzy matching uses the Levenshtein distance (or edit distance) as a metric to measure the distance between two sequences.⁵ The Levenshtein distance is the minimum number of single-element edits (i.e. insert, delete, substitute) required to change one sequence a into another b . Formally, $lev_{a,b}(n, m)$ is the distance between the first n elements of a and the first m elements of b :

$$lev_{a,b}(n, m) = \begin{cases} \max(n, m) & \text{if } \min(n, m) = 0, \\ \min \begin{cases} lev_{a,b}(n-1, m) + 1 \\ lev_{a,b}(n, m-1) + 1 \\ lev_{a,b}(n-1, m-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases} \quad (3.1)$$

Where $1_{(a_i \neq b_j)}$ is an indicator function equal to 0 when $a_i = b_j$ and otherwise equal to 1.

Pre-training of Pyggy was done through an interactive website where individuals could directly interact in basic chit-chat dialogue. Unfortunately, when the general public had the ability to interact with Pyggy many of them started to act adversarially and mischievously, training the system to say rude and inappropriate things. Once the compiled training set was cleaned and filtered, it was small (less than 100 lines). Thus, additional clean training data was appended from the Cornell Movie Dialogue Corpus [65]. The dataset is composed of 220579 conversational exchanges from 617 movies and provided the system a larger corpus of novel, interesting, and appropriate dialogue to pull from.

Pyggy is embodied by a visualization as seen in Figure 3.1 and Figure 3.5. The dynamic image-based visualization of Pyggy was accomplished with Magic Music Visualizer.⁶ The simple animation system controlled the movement of the mouth. Mouth movement was modulated by the amplitude of the synthesized sound during speech generation system. This physical embodiment gave Pyggy an animated face on stage.

⁵<https://github.com/seatgeek/fuzzywuzzy>

⁶<https://magicmusicvisuals.com/>

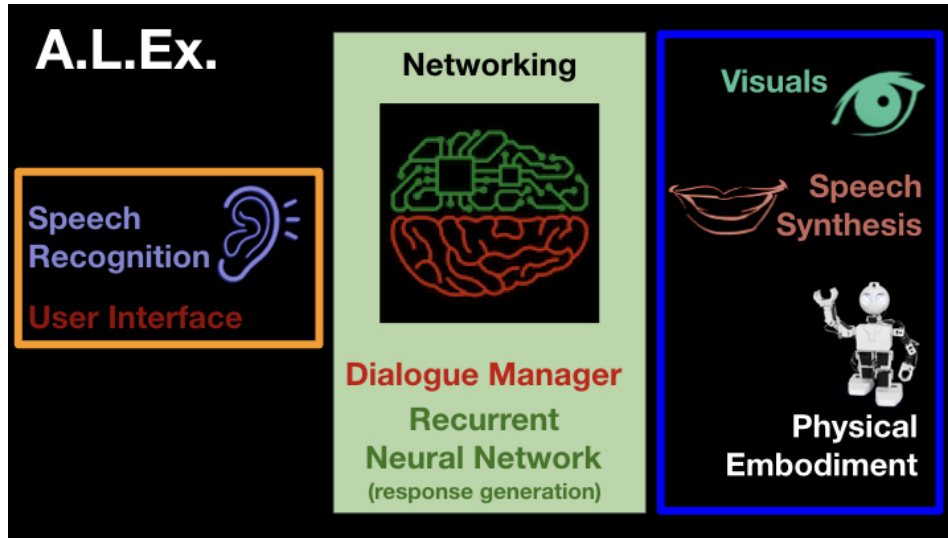


Figure 3.2: High level system diagram of A.L.Ex.

3.2.2 Version 2: A.L.Ex. – Corpus Based Generation

There were limitations to the dialogue which Pyggy could produce, as it was restricted to the set of sentences present in the training and interaction data. The system was crude in this sense, recalling the most likely response to any input from the human. As well, Pyggy had no means by which to understand or track the topic of a scene. These limitations prompted us to explore a less constrained word-by-word generation approach.

Automatic Language Generation in Improvised Theatre

The very nature of improvised theatre relies on spontaneous generative conversational abilities. Improvised theatre training relies on teaching the actors games which force them to perform fast-paced word associations (e.g., “electric . . . car . . . company”) or sentence completion [147] without over-thinking any of their decisions. During these word generation games, spontaneity is encouraged and failure (e.g., a non-grammatical choice of word, an onomatopoeia instead of a word, or simply a made-up, garbled word suggestion) is tolerated and celebrated. By celebrating failure, improvisors actively reinforce spontaneity and liberate the creative process [314]. Some of the games directly draw on the Surrealists’ *Cadavres Exquis* idea of taking turns in collaborative

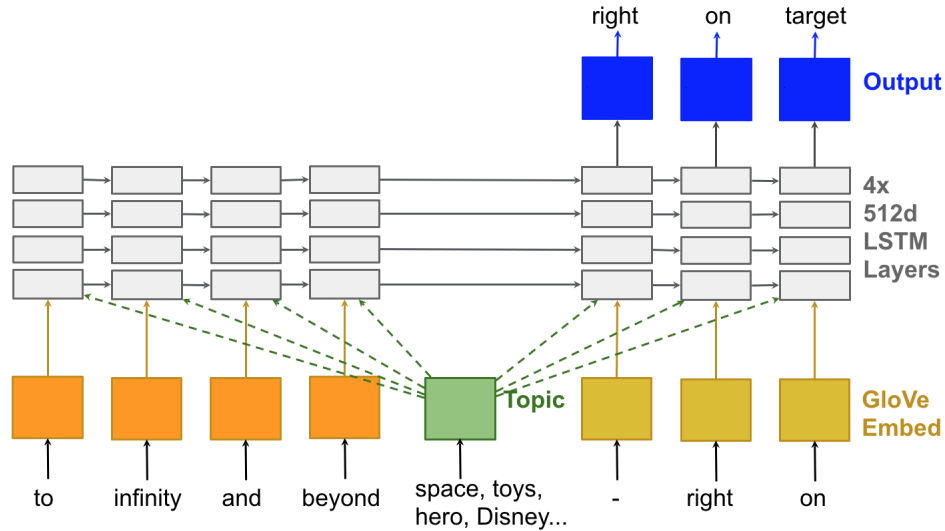


Figure 3.3: Context and input to Artificial Language Experiment and unrolled single output decoding.

art generation and require the players to build coherent narratives (further details in Appendix D).

Even challenging improvisational exercises in musical improv rely on the faculty of spontaneous text generation. In this form, performers follow the rhythm and tune of an accompanist while improvising rhyming song. Many musical improv teachers and freestyle rap artists recommend not to prepare rhymes in advance.⁷ Rather, they encourage starting lines without predetermined ideas of what rhyme can be found, and let the rhymes arise organically in the mind of the improviser.

While the word generation process is destined to be spontaneous, it is not intrinsically random. Improvisors use their cultural background, their literary and pop-culture knowledge, eloquence skills, and vernacular, to generate sequences of words which seem most *obvious* to them. Each line is *statistically likely* to occur given the context of the improvisation.

Text Generation with Neural Language Models

We imitate the creative process of improvisation using a statistical language model that can generate text as a sequence of words. While building an open-

⁷<http://www.theshowstoppers.org/>

domain conversational agent able to converse at human-levels would require artificial general intelligence, a generative dialogue system that is conditioned on previous text and that mimics collaborative writing could give to the audience an illusion of sensible dialogue [114]. The need for generative dialogue and language models required shifting from the rule-based, deterministic learning systems of Pyggy to deep neural network-based language model which could generate sentences word by word. There are benefits to developing our own neural language model. It allows us complete control over the training data, training process, input processing, response generation, and deployment of the system. These are critical for building systems for live performance, where we require the ability run—and sometimes debug—a model live, on-stage.

Briefly, word-based neural language generation works as follows. Deep neural network-based language models associate each word in an input vocabulary with a vector of real valued numbers, or word vectors. These word vectors can be initialized to random numbers before training, or they can be initialized with vectors which have been pretrained on previous tasks [228], [254]. These pretrained vectors contain some information how each word is used in context, and how word information can be related in vector space. In vector space, relationships such as analogies or opposites can be modelled [228], [254]. By using these word vectors as the inputs, language modelling can retain some of this relationship information. These vectors are used as the input for recurrent network models. Additional details on these models is detailed in Section 2.2.3.

A.L.Ex (Artificial Language Experiment) is built using recurrent neural networks (RNN) with long-short term memory (LSTM) [124], [227] as described in Section 2.2.4 and shown in Figure 3.3. Contrary to similar work in text generation [101], [317], we decided to use word-based RNNs. This facilitates curating the vocabulary produced by the dialogue system by replacing or removing offensive words generated by the LSTM.

We experimented with multiple LSTM architectures with the goal of building a dialogue model that can handle the topics within an improvised scene over dozens of exchanges between the human and the AI. All models were trained

by optimizing for perplexity. Recall that perplexity relates to the likelihood of sampling a sentence with a language model trained on a particular corpus (Equation 2.30). We started from a first version consisting of 100,000 linear input word embeddings and a two-layer LSTM with 256 hidden units followed by a softmax over 100,000 output words. This model had many missing input words and created dull responses with no notion of topic. The second version improved consistency by including topic vectors. It contained 4 layers of 512 hidden units of LSTMs and extra 64 inputs to the first LSTM, coming from a Latent Dirichlet Allocation [25] topic model (described in Appendix C). This design enables the language model to integrate long-range dependencies in the generated text and capture the general *theme* of the dialogue [231]. Design choices followed details from Mikolov and Zweig (2012) [229].

In the second version there were still many unused output words. The top 50000 words account for 99.15% of all words in the input data. The third version used pre-trained word embeddings (GloVe) [254] as inputs, resulting in a larger vocabulary of 250,000 input words—the GloVe word embedding matrix was considered as pre-trained and stayed fixed over the training—and only 50,000 output words. With fixed embeddings and a smaller output space, convergence was sped up. The fourth version cloned the 4-layer LSTM into a query embedding module and a response generating module in a SEQ2SEQ architecture [156], [318] with an attention model over the query embedding vectors [302]. With more trainable parameters, the fourth model increased the size of the stored model, as well as the loading time. With only a small performance boost, it was decided that the third model would be the optimal trade-off of model size and response performance. This was particularly important for the speed required for live performance, described in detail in Section 4.6.4.

A.L.Ex. is a neural language model which uses continuous word embeddings to predict sequences of words. The neural network-based language model is constructed as described as above and trained as a probabilistic classifier to predict a probability distribution over a vocabulary of words given an input linguistic context. This is done through stochastic gradient descent

using backpropagation. Adaptive moment estimation (Adam) is used as the optimizer [155] with initial learning rate of 0.0002 multiplied by 0.5 every epoch (one epoch is a single pass through the entire set of training data). Gradients are clipped at magnitude 5, and a dropout value of 0.2 was used after each RNN layer for regularization. The model was trained to convergence over 10 epochs. Sampling was done using a beam search with softmax sampling temperature 0.8, with a beam width of 10, by stepping forward through a sequence of words (300-dimensional embedding) from the model and continuing prediction until a maximum sequence length (100) or the end of sequence token were encountered. Utterances are returned alongside their resulting perplexity under the given language model. A heuristic for selecting the sentence with the lowest perplexity is used, with a discount factor based on the length of the sentence to encourage shorter sentences. Models were trained using a GPU back end (Amazon AWS EC2 g2.2xlarge) training time was 0.818s per batch of 10 sequences of length 100. Training to convergence took just under 1 month. Inference is done using a CPU back end. For serving the trained model, the minimum configuration requires: 8GB RAM, 4VPU/CPU and 20GB of hard disk space.

Dataset

The language model of A.L.Ex was trained on transcribed subtitles from 102,916 movies from OpenSubtitles.org, going from 1902 to early 2016.⁸ This user-contributed subtitles dataset for dialogue model training contains multiple languages and versions for each movie [340]. The data were available as XML files, with precise timestamps for each line of dialogue. We kept one English subtitle version per movie. As we noticed that subtitles tend to be split over time and that each change of interlocutor is marked by a dash sign, we processed the XML files to adjoin lines of dialogue separated by 1 sec, starting with lower-cased words and without an initial dash, into single lines of dialogue. Further processing involved correcting common spelling mistakes to account for the often erroneous subtitle input (e.g., substitutions of “;”

⁸<http://opus.lingfil.uu.se/OpenSubtitles2016.php>

by “I” or vice-versa, extra spaces between an apostrophe and the contracted word or repetitions of letters, using a painstakingly hand-crafted set of over one thousand of regular expressions) and removal of such as information as “subtitles by ...”. The resulting files were lower-cased. After text clean-up, we calculated that the top 50,000 words accounted for about 99.4% of the total words appearing in the corpus. The resulting corpus contained approximately 880 million tokens (including dashes).

Data preparation followed this procedure:

1. Download, parse, and clean OpenSubtitles
2. Build and extract a vocabulary covering the majority of the words in the dataset.
3. Find 68 sounds effects keywords and replace with special tokens associated with synthesizable sounds (e.g. laugh, yell, whistle, music, phone, and catch-all sound effects).
4. Match vocabulary to word embeddings.
5. Remove movie scripts with low embedding coverage.
6. Use Bag-of-Words for LDA topic clustering.
7. Combine cleaned lines and per-movie topic vectors.
8. Save data files in fast access data storage ready for training (e.g. HDF5/JSON).

The choice of a movie dialogue corpus, derived from movie scripts, is fitting. Often improv comedy actors draw on previous experience, personal culture and practice in their spontaneous creative process [206]. Future work might explore a variety of text-based data-sources including plays, short stories, transcripts of improvised performances, and symbolic plot points [59].

System Architecture

A.L.Ex. was designed to subvert the multiplicity of connected services which formed the architecture of Pyggy. A.L.Ex. aimed to be an offline, standalone artificial improviser. While, similarly to Pyggy, speech recognition and generation are still performed by ready-made tools, respectively Apple Enhanced

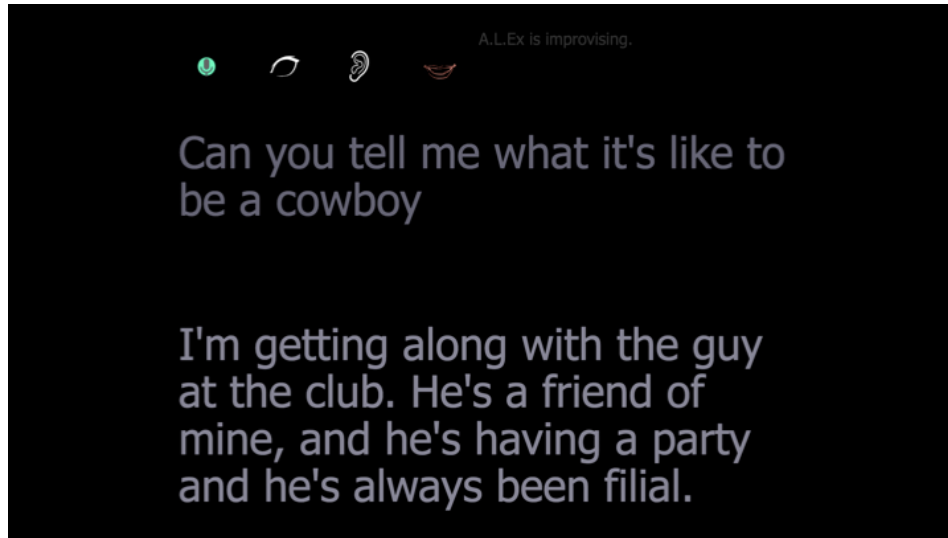


Figure 3.4: User interface with example input (top) and response (bottom).

Dictation⁹ and Apple Speech Synthesis, these tools are run on the same computer as the model is served without depending on an internet connection.

The entire text-based dialogue system (coded in Lua and Torch¹⁰), was encapsulated into a single program which makes system calls to speech recognition and text-to-speech, and was controlled through a graphical user interface (Figure 3.4) which visualizes results (i.e. both the recognized and generated sentences in the dialogue). The core system is extended with additional modules; it also runs a fault-resilient server which accepts incoming HTTP/GET requests from client applications. These applications include software controlling a humanoid robot with pre-programmed motions that are activated when A.L.Ex speaks (see Figure 3.6). Applications have been written for controlling both the EZ-Robot JD Humanoid¹¹ and the SoftBank Aldebaran Nao.¹²

3.3 Results

There are challenges associated with testing, and quantitatively evaluating, open-domain dialogue systems [98], [114]. An obvious and reasonable first

⁹<https://support.apple.com/en-ca/HT202584>

¹⁰<http://torch.ch/>

¹¹EZ-Robot <https://ez-robots.com>

¹²<https://ald.softbankrobotics.com/en>

measure for qualitative assessment would be similar to that of a human improviser. That is, the audience-perceived performance level of an artificial improviser during an improvisational performance. Thus, each of these systems has been tested live in front of audiences between 5 and 100 people, for a total of 25, 7-60 minute performances between 8 April 2016 and 1 June 2017. As is common in improvisation, show structure and order remained largely consistent, while content varied based on audience suggestion¹³. Through audience feedback, the system has been iteratively improved, through enhancement to the neural network-based dialogue system, the addition of performance *props* (e.g. audio and video user interface, robotic avatar), novel improv comedy games involving the machine and the humans (with optional audience participation), and scripted narrative elements performed by the humans.

While human audience evaluation is realistic and comparable with human improviser evaluation, there are limitations. First, while an audience is evaluating the performance of the artificial improviser in the scene, they may be biased by the fact that a machine is generating or performing these lines. Due to this bias, it might be worthwhile to compare human and machine-generated responses and scenes offline where the rater is blind to the source of the sides of the dialogue. This is often done in dialogue evaluation, but it is costly, time intensive, and high variance based on subjective human raters [73], [188]. As well, the performance of these systems can be limited at several points in the dialogue. Speech recognition accuracy is not perfect, and environmental noise (e.g. cheering/laughing audience) can influence the quality of input audio for speech-to-text. Finally, because of the high variability of live theatrical performance, the performance spaces (e.g. dark pub basements, or bright conference venues), the narrative elements, the technical difficulties encountered during performance, and iterative experimentation, it is difficult for us to provide quantitative assessment of the artificial improviser in this study.

Even with these limitations, it is important to report on the success and failure thus far with qualitative examples, and leave subjective assessment to the reader. Important future work will include large scale qualitative as-

¹³<https://youtube.com/watch?v=FjxBdMRAZ1E>

assessment from trained improvisors for qualities of improvisation and from the general public.

3.3.1 Version 1: Pyggy



Figure 3.5: Human performing on-stage with Pyggy. The avatar of the artificial improviser is projected behind the human performer.

The first version of AI-based improv, Pyggy, performed in the show *Artificial Improvisation* live and in front of an audience on April 8, 2016, as part of an invited performance during Rapid Fire Theatre’s 2016 Bonfire Festival.¹⁴ While technically demanding, it was generally well received by the audience (Figure 3.5). Here is a short excerpt from a transcription of the performance:

HUMAN: We’re having a picnic! Would you like cheese and
bread?

PYGGY: Seen one human, you’ve seen them all.

¹⁴<http://rapidfiretheatre.com>

HUMAN: Promise me, right now, we're here listening to each other.

PYGGY: For fun? Where are all your queer party friends?

HUMAN: Why don't you have any feelings for me? That's why I had to invite you on a picnic. I wanted to get some time to talk to each other.

PYGGY: I miss your vivid sense of humour.

3.3.2 Version 2: A.L.Ex.



Figure 3.6: Human performing onstage with A.L.Ex., personified as a robot. The user interface, showing speech recognition results and machine-generated response, is projected behind the performers.

A.L.Ex. performed for the first time in front of a live audience on July 24, 2016, during an improv drop-in in London (Figure 3.6). It has been performed more than 24 times and on two continents since. We provide below a short transcript from an early performance with A.L.Ex. As can be noticed, the dialogue system somewhat manages to keep track of the general theme of the conversation (i.e. dramatic, with words related to navigation and combat).

HUMAN: Captain, the ship is under attack. The frigate is arriving...

A.L.EX: You see, I don't feel the same way. I just don't want to stay in the position to get burnt.

HUMAN: We're all going to get burned by a cannon fire...

A.L.EX: While you're in there, tell me. When I get dressed, I'll go up there and see what happens.

HUMAN: Captain, you need to get on the deck, the crew is expecting you to speak, to take leadership. We are in the middle of a war and this is an invasion of foreign ships. Captain, show us your leadership.

A.L.EX: We won't be able to navigate without them as we are, serving for us to act like a life of violence and to help us die from the most powerful life of our time.

The subsequent experiments detailed in this section were performed using iteratively improved versions of A.L.Ex.

3.3.3 Human-AI Collaboration through Games

With the base system in place, we have explored several alternative formats, often called structures or games in improvisation [200], [275]. We explored a variety of formats as the context within which the artificial improviser performs is critical toward the measure of audience perception of quality. In improvisation, a major tenant is to make one's fellow improvisors *shine*.¹⁵ We created a show built on structures where both A.L.Ex. and human improvisors could shine.

3.3.4 Justification Game

The most extreme case of enhancing the stature of a human improviser arises from games where the actor is confronted with ridiculously difficult challenges that he or she successfully overcomes [147]. One such game is called Actor's Nightmare (or sometimes *pick-a-line* or *lines from a hat*). It consists of the

¹⁵<http://jimmycarrane.com/making-partner-look-good/>

player intermittently picking a random line of dialogue (typically unrelated to the current improvisation), reading it aloud, and seamlessly integrating it into the scene [72].¹⁶ The humour generally arises from the improviser’s skill in justifying that line of dialogue or from the line being coincidentally appropriate. We found that, because of the limitations of speech recognition and of the dialogue system in A.L.Ex, many of the human-machine interactions ended up following the paradigm of justification games.

3.3.5 Multiple-choice Human-mediated Dialogue

A multiple-choice game was the first format that we explored outside of the basic structure of two improvisors engaging in a basic dialogue in a scenic setting. In this format, the system visually presented several candidate responses on a screen, but did not say any of the responses. Instead, an audience volunteer would select their preferred response and read it aloud. In this way, we were able to directly engage an audience member in the performance. This demolition of the fourth wall is common in improvisation [285]. When the audience is invited to directly interact with the system on stage, an additional tension is introduced in the room: how will an untrained human react if A.L.Ex. offers multiple interesting candidates, and what if there are no interesting candidates generated? We observed that these games presented the challenge of the audience member having to share attention between the screen and the human improviser and could be hampered by low energy or audience engagement.

3.3.6 Multiple-person Games

We explored dynamics where the AI played a single character in a scene with multiple humans. First, we introduced multiple humans in the same physical space. In this situation, A.L.Ex. plays alongside two human performers. We noticed that there is often a tendency for the two humans to form a ‘side’, acting together ‘against’ the system. Much more interesting scene dynamics emerged when we challenged one of the human performers to align with

¹⁶<http://stuffforactors.com/gpage3.html>



Figure 3.7: Two humans performing on-stage with A.L.Ex. One of the humans is remotely connecting, adding to the complexity of the show setup.

A.L.Ex’s character in the scene. Extending from this work, we then tried including the second human through a remote connection (Google Hangout, see Fig. 3.7). A.L.Ex. was able to interact with the physical human and the remote human. High resolution, low latency video conferencing continues to prove challenging, especially for remote performances with limited budgets, and we continue to explore means by which to overcome these challenges. We then instantiated multiple versions of A.L.Ex. in a single scene. In this way, we could balance the two humans on stage with two robotic improvisors. This presented opportunities for interesting connections and relationships as well as challenges. The timing of multiple AI-based improvisors can be noticeably different if not properly synchronized.

3.3.7 Comparison with ELIZA

Finally, we built an audience interaction game in homage to one of the earliest chatbot systems, ELIZA, by Joseph Weizenbaum [344]. In this format, an audience member is invited to the stage to discuss an ailment with an AI therapist played by A.L.Ex. in ELIZA mode. Details on ELIZA are included in Appendix B. While ELIZA is powered by relatively simple deterministic response rules given certain decompositions of the human’s input statement.

This is an audience favourite and often well received during shows. It is important to pay special attention to this note, as the holistic performance of an AI-improviser should be evaluated based not only on how well it is received, but also on the novelty and uniqueness of the scenes it performs.

Many games were selected to allow for clear, distinct trade-off between multiple improvisors within consistent settings. Often our systems fail through mis-understanding speech-to-text input or human-machine interruption due to lack of social cueing and perception. By embracing and learning from these failures, we will continue to innovate and experiment to better understand and showcase the strengths of A.L.Ex.

3.4 Discussion and Summary

Future work might incorporate advances in deep reinforcement learning for dialogue generation [273]. Through design of reward functions, more interesting dialogue may be encouraged. Three useful conversational properties shown to improve long-term success of dialogue training are: informativity, coherence, and ease of answering [179]. Additional reward schemes may improve, or tune, the trained deep neural network based dialogue managers. Work has shown that reinforcement learning can be used to tune music generation architectures [139]. Rewarding linguistic features (i.e. humour, novelty, alliteration) may prove useful in dialogue generation [126].

This study focused on building a dialogue system for improvisational performance. Improv theatre is a relatively young art-form, but there exists several books of rules for novice improvisational training which could be useful for future studies [241]. Future iterations of these systems could include common improvisational rules, such as ‘status contrast’ the ‘Yes, and...’ theory of accept and expand, and comedic rules, such as the ‘rule-of-three’ [147].

Adversarial methods for natural language are another means of exploration [180], [271]. While the results are interesting and informative, these works are still limited in the objective functions and evaluation criteria used often relying on log-likelihood scores, BLEU [251] or ROUGE [183] scores.

Additional evaluation metrics must be devised to score these open-domain dialogue systems [184], [188], [273].

Natural human conversations are seldom limited in scope, jump from topic to topic, and are laced with metaphor and subtext. Artificial improvisors of the future should make use of advances in artificial memory [348] and attention models [196]. As well, humans often make use of non-verbal cues during dialogue. By incorporating this additional information, human(s) could both consciously and subconsciously inform the learning system [210]. Additionally, if the artificial improvisor is modelled as a goal-seeking agent, then shared agency could be quantified and communicative capacity could be learned and optimized for during the performance [262].

While the system is trained to perform dialogue, it is not trained to tell a cohesive story with a narrative arc. The addition of memory network advancements may improve callback; additional engineering and training will be necessary to collaboratively build a narrative arc (as discussed in Chapter 5). In 1928, William Cook published a book on algorithmic plot development which may serve this purpose, and implementations and connections have yet to be explored [59]. Preliminary investigation toward incorporating plot development into performative systems is presented in Chapter 6.

There is a large research area devoted to computers and creativity (see Section 2.3.5) [218]. These advances are pushed to the horizon by artists around the world. These artists are building complex projects, using machine learning techniques to implement creative, imaginative ideas. Thought must be given to the interface through which humans and artificial performers interact [218], [219], [359]. The embodiment of the artificial improvisor has been investigated with Pyggy and A.L.Ex. using on-screen visualizations and robotics. Stage presence is critical to ensure that a live show is enjoyable and engaging. Improvisational performances are not strictly conversational and often demand physicality from performers. The optimal means by which these systems take the stage has yet to be determined.

Previous work has explored the importance of physical embodiments on social dynamics for similar performances [157]. Collaboration between scientists

and creative artists will lead to innovative interactions and immersive art which ignites the senses. With the growing popularity of interactive mixed-reality experiences, as well as advances in natural language processing, speech, and music generation, there are exciting avenues of future investigation [8], [332].

Improvisational theatre is a domain where experimentation is encouraged, where interaction is paramount, and where failure flourishes. It allows artificial intelligence agents to be effectively tested, and audience reaction can provide a subjective measure of improvement and cognizance. While this work may feel similar to the Turing test, an early attempt to separate mind from machine through a game of imitation, deception and fraud, it is much more than that [330]. Success will be measured by audience preference to engage in shows incorporating artificial improvisation and human desire to participate. Turing imagined a world where machinery might exceed human performance in some domains. It is ironic and poetically fitting that ‘live’ performance itself may be one of these domains.

Board games such as chess and Go are complex, but computational solutions can be approximated. Improvisational theatre demands creativity, rapid artistic generation, and natural language processing. Improvisation is not a zero-sum game, especially as these systems learn to converse open-domain settings [98], [114]. Future work will continue to explore the evaluation of performance in such an open domain. Performances with artificial improvisors continue to spur questions and insights from other performers and audiences alike. A formal evaluation, with audiences observing expert improvisors interacting with the system is presented in the next chapter. We look forward to the distant goal of the human observer, as a fly on the wall, watching artificial improvisors on-stage in front of a full audience of artificial observers. We strive to continue constructive dialogue around creative AI by connecting the sciences and the arts.

Chapter 4

Improbatics: Exploring the Imitation Game

Most of the programmes which we can put into the machine will result in its doing something that we cannot make sense of at all, or which we regard as completely random behaviour. Intelligent behaviour presumably consists in a departure from the completely disciplined behaviour involved in computation, but a rather slight one, which does not give rise to random behaviour, or to pointless repetitive loops.

– Alan Turing, 1950

Theatrical improvisation (or *improv*) is a demanding form of live, collaborative performance built on open-ended narrative structures in ephemeral, imagined worlds. It is thus an ideal test bed for the development and deployment of interactive artificial intelligence (AI)-based conversational agents, or *artificial improvisors*. This chapter introduces an improv show experiment featuring and comparing both human actors and artificial improvisors. In the previous chapter we developed a deep-learning-based artificial improviser, trained on movie subtitles, that can generate plausible, context-based, lines of

A version of this chapter was accepted for publication as: **Mathewson KW** and Mirowski P. (2018) Improbatics: Exploring the Imitation Game using Machine Intelligence in Improvised Theatre. AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment. Portions of this chapter are adapted from **Mathewson KW** and Mirowski P. (2017) Improvised Theatre as a Turing Test. NIPS 2017 Workshop on Machine Learning for Creativity and Design.

dialogue suitable for theatre [212].

We extend that work by employing that system to control what a subset of human actors say during an improv performance. We also give human-generated lines to a different subset of performers. All lines are provided to actors with headphones and all performers are wearing headphones. This paper describes a Turing test, or imitation game, taking place in a theatre, with both the audience members and the performers left to guess who is a human and who is a machine. In order to test scientific hypotheses about the perception of humans versus machines we collect anonymous feedback from volunteer performers and audience members.

Over the course of 30 performances to a combined audience of almost 3000 people, we have refined theatrical games which involve combinations of human and (at times, adversarial) artificial improvisors. We have developed specific scene structures to include audience participants. Finally, we developed a complete show structure which submits the audience to an experience inspired by the Turing test and observed their suspension of disbelief, which is key for human/non-human theatre co-creation. Our results suggest that rehearsal increases proficiency and possibility to control events in the performance. That said, consistency with real world experience is still limited by the interface and the mechanisms used to perform the show. We also show that human-generated lines are shorter, more positive, and have less difficult words with more grammar and spelling mistakes than the artificial improvisor generated lines.

4.1 Introduction

Improvisation (or *improv*) is a complex theatrical art-form modelled on natural human interaction and demanding constant adaptation to an evolving context. It has been defined as “real-time dynamic problem solving” [200]. Improv requires performers to exhibit acute listening to both verbal and non-verbal suggestions coming from the other improvisors, split-second reaction, rapid empathy towards the other performers and the audience, short- and long-term



Figure 4.1: Illustration of two Improbotics rehearsals.

memory of narrative elements, and practised storytelling skills [147]. From an audience point of view, improvisors must express convincing raw emotions and act physically.

Improvisational computational storytelling is a grand challenge in artificial intelligence (AI) as proposed in previous research [206], and formalized in Chapter 3. While success on the grand challenge might be contingent on solving open-domain conversational general artificial intelligence, there have been incremental scientific steps made progressing toward a unified system which can engage in improvised theatre in an open world [41], [105], [212], [364]. While these systems do not fully understand the interaction, they can, in spite of (or perhaps, as an improvisor would think, *thanks to*) their imperfections, fuel the creativity of the performers.

4.2 Background

Theatrical improvisation is a form of live theatre where artists perform “real-time dynamic problem solving” through semi-structured spontaneous storytelling [200]. Improvised comedy involves both performers and audience members in interactive formats (sometimes called scenes, structures, or skits). We present explorations in a theatrical Turing test as part of an improvised comedy show. We have developed an artificial intelligence-based improvisational theatre actor. This artificial improvisor is a chatbot with speech recognition and speech synthesis capabilities. It also uses a with a physical embodiment [112],

[256]. We have performed alongside the system in improv shows at performing arts festivals, including ImproFest UK and the Brighton, Camden, and Edinburgh Fringe Festivals [212]. Complete show listings and recordings are available at <https://humanmachine.live>. Public reaction to these performances are included in Section 7.2.

Over these first 30 shows, one or two humans performed improvised scenes with the artificial improviser. The performers strove to endow the system with human qualities of character/personality, relationship, status, emotion, perspective, and intelligence, according to common rules of improvisation [147], [241]. Relying on a large corpus of film subtitles and custom neural network software for language understanding and text generation, described in detail in Chapter 3. We were able to produce context-dependent replies for the machine-based improviser. A high level system diagram is shown in Figure 4.2.

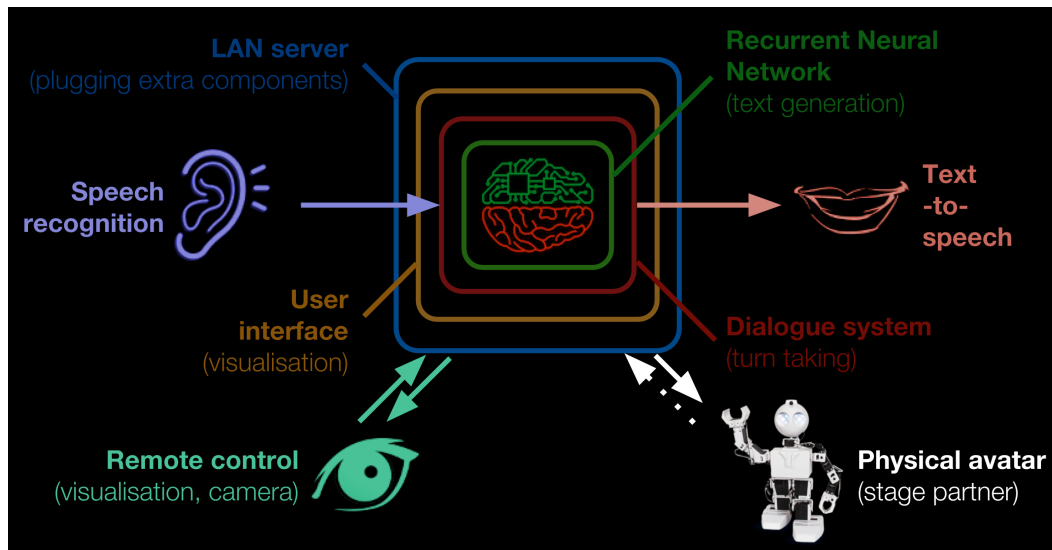


Figure 4.2: Artificial Language Experiment (A.L.Ex.) System Diagram

The system we developed aims to maintain the illusion of intelligent dialogue. Improvised scenes develop emotional connections between imaginary characters played by improvisors. The human-like characterization of the machine elicited attachment from audience members. Through various configurations (e.g. human-human, human-AI, and AI-AI) and different AI embodi-

ments (e.g. voice alone, visual avatar, or robot), we challenged the audience to discriminate between human- and machine-led improvisation. In one particular game setup, through a Wizard-of-Oz illusion, we performed a Turing test inspired structure [18], [278].

We deceived the audience into believing that an AI was performing, then we asked them to compare that performance with a performance by an actual AI. Feedback from the audience, and from performers who have experimented with our system, provide insight for future development of improv games. Below we present details on how we debuted this technology to audiences, and provide observations collected over multiple performances. Feedback from the audience and from performers who have experimented with our system can provide insight into the suspension of disbelief required for non-human theatre.

4.2.1 Related Work

Research on computational improvisation often focuses on music and dance, and on how humans interact and co-create with artificial systems [84], [125], [326]. Improvised theatre has also been a platform for digital storytelling and video game research for more than two decades [112], [256]. Theoreticians and practitioners have experimented with several rule- or knowledge-based methods for collaborative storytelling and digital improvisation [199], [246], [310], [363], and computer-aided interactive storytelling has been explored in video game development, aiming to create near-infinite narrative possibilities to drive longer-term player engagement [275]. There have been previous installation art pieces which incorporated computer simulation are interactive improvisation.¹ To the best of our knowledge, our case study describes the first application of deep learning-based conversational agents [340] to control and guide unstructured improvised theatre performance of human actors.

Robotic performances have been explored previously [33] and are discussed in Section 2.3.4. These shows required the audience to suspend disbelief and embrace the mirage of autonomy. Robot-based performances challenge the *uncanny valley*—the idea that as the appearance of a human-like robot ap-

¹<https://www.badnewsgame.com/overview/>

proaches a human likeness, human responses shift from empathy toward revulsion [236]. Toy-like humanoid robots have been involved in improvised theatre performances [200], for instance Arthur Simone’s “Bot Party: Improv Comedy with Robots”² and “HumanMachine: Artificial Intelligence Improvisation”³. Unlike those shows, our performance does not employ robotic avatars but sends the AI-generated dialogue to human embodiments.

4.2.2 Motivation

Cinematic releases including “Ex Machina” [91], “Her” [148], and “Robot & Frank” [293] explore robots interacting with humans naturally in day-to-day life. We invite live audiences to consider such interactions in a theatrical setting. Theatre practitioners can embrace machine learning systems as tools to explore dramatic interactions and to expand the realm of stories that artists can create. This aligns with our research goal of augmenting creative abilities of humans. To test the quality of this creative augmentation, we have developed a test-bed for theatrical co-creation which places humans directly alongside machines in an improvisational performance.

In our show *Improbatics*, we explore how human performers could seamlessly perform when a machine, or another human, provides their lines. The human and machine performers work together to create a single, grounded, narrative improvisation. We combine conceptual ideas from classic improvisation and novel methods in machine learning and natural language processing. The show is inspired by improvisation game *Actor’s Nightmare* [72]—where one of the performers reads lines from a play and the other performers seamlessly justify these otherwise incongruous lines while progressing a narrative. This game is modified to incorporate previous work on improvised theatre alongside artificial intelligence. Specifically, this work builds on the performances described in the previous chapters, *HumanMachine: Artificial Intelligence Improvisation*, and Etan Muskat’s *Yes, Android*⁴.

²<http://arthursimone.com/bot-party/>

³<https://humanmachine.live/>

⁴<https://baddogtheatre.com/yes-android/>

This work explores Wizard-of-Oz [18] style experimental methods that have been used extensively in previous human-robot interaction studies and dialogue system research [74], [87], [208], [278]. Wizard-of-Oz style interactions with artificial intelligence controllers have been used to provide suggestions to actors into previous artistic works.⁵ In these studies, humans receive inputs from an external source. The source may be another human, or the machine learning system. Importantly, the source is unknown to the human. This allows for separation between the human subjects’ outputs, and the corresponding inputs. Similar to *Actor’s Nightmare*, the controlled humans in *Improbatics* will say and justify the lines they are prescribed through emotion, intonation, and physicality. What sets this format apart from previous work is that in *Improbatics* the lines depend on the context of the improvised scene. Improvisors not fed lines work to justify as the lines are not completely congruous. These justifications aim to make the scene look and feel more natural [13], [61].

In a way, *Improbatics* can be seen as a theatrical Turing test [211], [330]. Can the performers and audience discern who is delivering lines generated by a human from those delivering lines from a machine? We now cover methods to test this question.

4.3 Methods

4.3.1 Architecture of the Artificial Improvisor

We named our improviser A.L.Ex, the Artificial Language Experiment, an homage to *Alex the Parrot*, trained to communicate using a vocabulary of 150 words [255]. The core of A.L.Ex consists of a text-based chatbot implemented as a word-level sequence-to-sequence recurrent neural network (4-layer LSTM encoder, similar decoder, and 64-dimensional topic model inputs) with an output vocabulary of 50k words. The network was trained on cleaned and filtered subtitles from about 100k films. Subtitles from 100k movies were collected

⁵<https://www.badnewsgame.com/>

from OpenSubtitles.⁶ Additional model details are included in Chapter 3.

Dialogue turn-taking, timing, candidate sentence selection, and sentiment analysis [132] on the input sentences are based on heuristics. The chatbot communicates with performers through out-of-the-box speech recognition (Google Speech Recognition) and text-to-speech software (Apple Speech Synthesis). The chatbot runs on a local web server for modularity and allows for integration with a variety of physical embodiments (e.g. parallel control of a humanoid robot). The robot shown in Fig. 4.2 was manufactured by EZ-Robot.⁷ The parallel integration allows for modularity on the robotic embodiment, we have explored a variety of robots (including the Nao robot from Aldebaran SoftBank) and visual projections as illustrated in Fig. 4.3(a). Physical robotic embodiments allow for synchronized speaking and movement. This synchronization is important for explicit cueing interaction with untrained improvisors, such as audience members who join the performers onstage.

The first version of our improvising artificial stage companions had their stage presence reduced to projected video and amplified sound (see Section 3.2.1). We evolved to physical embodiments (i.e. the humanoid robot) to project the attention of the performer(s) and audience on a material avatar (see Section 3.3.2). Our robotic performers are distinctly non-human in size, shape, material, actuation and lighting. We chose humanoid robotics because the more realistic an embodiment is the more comfortable humans often are with it; though comfort sharply drops when creatures have human-like qualities but are distinctly non-human [235]. The physical robot also enabled us to integrate well developed practices from puppetry and ventriloquism when developing and performing the improvised shows. These principles include looking directly at the puppet when it is speaking, controlling the micro-movements of the puppet, and showing the direct link between the human and the puppet [4].

The server also enables remote connection which can override the chatbot and give dialogue control to a human operator. Multiple human performers use headset or hand-held microphones, connected through a mixing table

⁶<https://opensubtitles.org>

⁷<https://www.ez-robot.com>

to the audio input for speech recognition. Thus, several human performers can perform simultaneously in the scenes. Having multiple performers, both human- and machine-based, enables us to explore complex status dynamics and 2-vs-1 relationships.

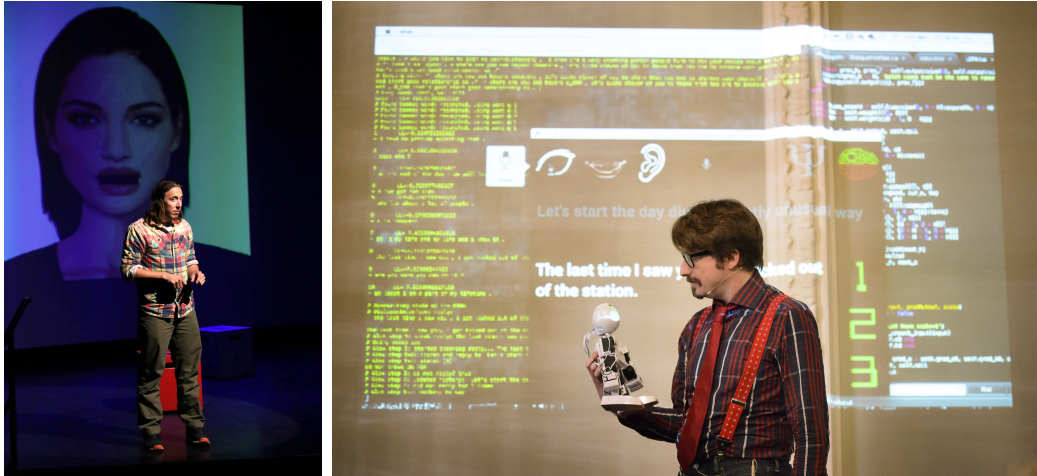


Figure 4.3: (a, left) Visual and (b, right) physical embodiments of the artificial improviser.

Improbabilities is a show structure created to explore the grand challenge of artificial improvisation [206]. The show is composed of a cast of trained human performers (semi-professional improvisors with at least 2 years of experience).

The cast is broken down into four roles: *Cyborgs*, *Puppets*, *Free-will Humans*, and *Controllers*.

1. *Cyborgs* are humans who take lines via headphones from an AI-powered chatbot overseen by a *CEO Controller*;
2. *Puppets* take their lines via headphone from a *Puppet Master Controller*;
3. *Free-will Humans* are free to make up their own lines of dialogue and typically support the show's narrative; and
4. *Controllers*, of which there are two sub-roles:
 - (a) the Puppet Master directly inputs lines for the Puppet; and
 - (b) the CEO who inputs scene context into an AI system that generates lines of dialogue for the Cyborg.



Figure 4.4: Two human performers and an audience volunteer improvising with a robotic artificial improviser.

4.3.2 Typical Structure of AI-based Improvised Shows

The artificial improvisors we developed have performed alongside human actors in 30 improvised shows to date. Version 1 is named Pyggy, and Version 2 is named A.L.Ex. The systems are described in detail in Chapter 3. Because the shows are improvised, every one is different. That said, there are several commonalities between the scenes which compose the show.

An improvisational scene starts by soliciting suggestion for context from the audience (e.g., “non-geographical location” or “advice a grandparent might give”). The human performer then says several lines of dialogue to prime A.L.Ex. with dense context. The scene continues through alternating lines of dialogue. Often through human justification, performers aim to maintain scene reality and ground narrative in believable storytelling. A typical scene lasts between 3-6 minutes, and is interrupted by the human performer when it reaches a natural ending (e.g. narrative conclusion or comical high point).

For instance, one improv game which was played in multiple shows was staged by setting a volunteer in a psychotherapy session with the artificial improviser running the code of ELIZA [344]. This performance was presented

as an homage to the original chatbot creator, Joseph Weizenbaum.

Improbatics is structured as a collection of improvised scenes. A scene starts by soliciting a suggestion for context from the audience [147]. This provides performers with a novel context around which to situate the improvised performance, and primes the AI-system.

The scene consists of alternating lines of dialogue, where the Free-will Humans provide dense context to the system (human or AI-based dialogue model), and the Cyborg or Puppet performers respond in return. The Cyborg and Puppet performers aim to maintain the reality of the scene and to ground narratives in believable storytelling by justifying - emotionally and physically - their lines. A typical scene lasts between 3 and 6 minutes, and is concluded by the human performers when it reaches a natural ending. The performance progresses over the course of 20-45 minutes. At the end of the show, the audience votes to guess who was a Cyborg, who was a Puppet, and who was a Free-will Human.

Our Turing test is relatively easy to solve by an attentive audience, and similar imitation games have been explored previously [61]. We use the Turing test framing instead to both draw audience engagement and to give a creative constraint to the performers, analyzing the experience of performers collaborating with interactive AI tools. Additionally, it is hard to evaluate the imitation game with a live audience because of deception required from each attendee in a controlled but public performance setting. For this reason, we provide the Turing test as a framework for the show though it is unlikely that audience members were tricked for the duration of the show. The audience can infer who is a Cyborg or Puppet based on typos (e.g., “We are stuck in the dessert?... desert!”), spelling and grammar mistakes, lack of contextual consistency, and ignored salient information or timing constraints. We discuss these points in Section 4.6.

We considered a baseline show *Yes, Android* that is different from *Improbatics* in three aspects: 1) it relies on publicly available chatbot *Cleverbot*⁸, which is a general small-talk conversational agent that is trained not on movie

⁸<http://www.cleverbot.com/>

dialogue but on user interaction, 2) there is no Master or Puppet, and 3) no guessing takes place, as the audience knows in advance who is the AI-controlled performer.

4.3.3 Technical Configuration

The technology that enables remote control of a human player consists of a laptop computer connected to a radio transmitter, an radio receiver with headphones worn by the controlled performer (Cyborg or Puppet), and a program that allows a Controller to type either the Puppet’s lines, or context sent to an AI-based chatbot that will in-turn generate sentences to say by the Cyborg (see Fig.4.1). We used the chatbot from HumanMachine’s *A.L.Ex.* (described in Chapter 3), whose architecture is a sequence-to-sequence [318] recurrent neural network [124] trained on movie subtitles [340].⁹ Full details on the model and technical configuration are excluded for brevity as they can be found in Section 3.2.2. Our model uses beam search to generate word-by-word a set of 10 likely candidate responses to a given input and scene context; the top 4 sentences (ranked by perplexity under the language model) are selected and shown on the visual interface.

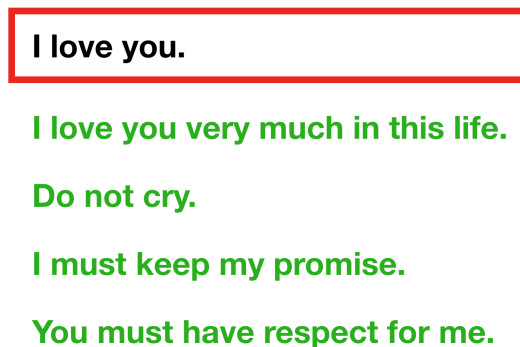


Figure 4.5: Simplified interface with the input line of context in the top red box and four generated candidate responses in shown below in green.

The Puppet Master Controller interface is a webpage where the operator types input context lines. Once a line is typed, it is converted to speech using in-browser (*Google Chrome*) text-to-speech capability. The CEO Controller’s

⁹<https://www.opensubtitles.org/>

interface for the Cyborg is a webpage that enables to type a line of context. Once typed, that line is sent to a server running the A.L.Ex. software returns the top 4 likely responses. The suggestions are displayed on the webpage, and the CEO Controller has the choice to select as many as they want or to discard them and to type new context. When a suggestion is selected, it is immediately converted to audio via text-to-speech, and output to speakers. The CEO can click on multiple lines or choose to ignore them, type new input and wait for new candidates. As we elaborate in Section 4.6, this multiple-choice setup for the Cyborg suggestions enables the CEO to moderate what lines are sent to the Cyborg’s headphones, and to curate suggestions to help drive the narrative of the performance.

All performers wear headphones, and both the Puppet and the Cyborg receive their lines through the same computer-generated voice through a text-to-speech interface. This configuration allows for a hidden set-up where the audience, and other performers, do not know who is playing which role.

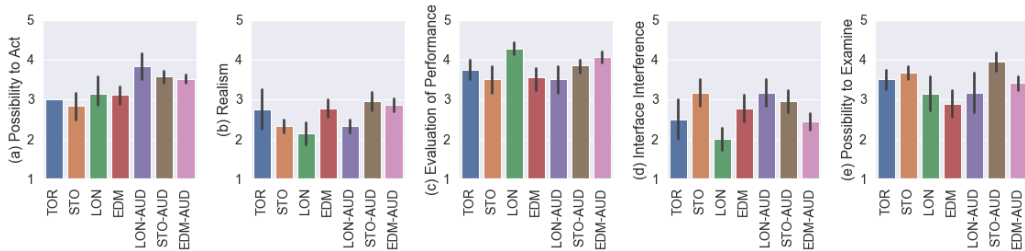


Figure 4.6: Audience and Performer Presence Analysis over Cities. Comparable systems were used and then analysed by a set of performers in four cities (sample size shown in brackets): Yes, Android in Toronto (TOR, $n = 4$), Improbots in Stockholm (STO, $n = 6$), Improbots in London (LON, $n = 7$), and Improbots in Edmonton (EDM, $n = 9$). Additionally, audiences were surveyed and data is presented for each city: LON-AUD ($n = 6$), STO-AUD ($n = 22$) and EDM-AUD ($n = 29$). Data presented is the average opinion over respondents in each group, with the 95 percent confidence interval.

4.3.4 Turing Test-based Theatrical Improvisation Games

Starting from the transatlantic improv performance on 31 March 2017, we have designed the AI improv show around narratives on the Turing test, computer

hacking, scientific experimentation, friendship and loneliness, and played up the contrast of the two human performers whose alter egos emphasize contrasting facets of computer scientists.

The performances at the Camden and Edinburgh Fringe festivals involved a Turing test inspired scene conducted with the willing audience. We performed the scene by first deceiving the audience into believing that A.L.Ex. was performing independently (whereas the chatbot and the robot were controlled by a human); then we performed a second scene with A.L.Ex. In game (1), we explained the Turing test first, then performed the two scenes consecutively and finally asked the audience to discriminate, through a vote, which scene was machine-driven. In a different game (2), we performed the Wizard-of-Oz scene and then immediately asked, in character and as part of the performance, if the audience suspected that a human was in control of the chatbot. Feedback from the audience provided us with insight into the suspension of disbelief required for non-human theatre. We present observations below.

4.4 Observations

We summarize here anecdotal observations from our performance. In game (1) (when the audience must discriminate between two scenes viewed unbiased) nearly everyone identified the machine driven scene from the human driven scene. In game (2), approximately half the audience members believed that the machine was performing independently alongside human improviser(s). When not forewarned about the Turing test, the audience members (of diverse composition in age and gender) was convinced that the dialogue system understood the details of the scene and responded immediately and contextually. The propensity of this delusion is likely driven by several factors: 1) the context within which they are viewing the deception, 2) the lack of personal awareness of the current machine learning abilities, and 3) emotional connections with the scene.

Post-show discussions with audience members confirmed that when a performer tells the audience that an AI is controlling the robot’s dialogue, the

audience members will trust this information. Being at an improvisational show, they expect to suspend disbelief and use their imagination. Most of them were also unaware of capabilities and limitations of machine learning systems. This highlights the responsibility of the scientific research community in communicating progress in effectively and to effectively invite public understanding of system capabilities. Finally, we observed that the introduction of a humanoid robot, with a human-like voice, increased the audiences' propensity to immerse themselves in the imaginative narrative presented to them.

4.4.1 Evaluation

What is the best way to test an improvised dialogue system? What is a good way to test an improvised dialogue system? As discussed previous chapters, measuring the performance of human-machine interaction is notoriously difficult and requires investigation from both human, and machine, perspectives.

A commonly used method of evaluating interactive performance is to address participants and audience during the show and after-the-fact, investigating experience through open questions, questionnaires or focus groups [353]. Our system was evaluated for *humanness* based on evaluation criteria for natural language dialogue systems in previous work [2]. In that work, the authors discuss that the evaluation of dialogue systems should be based on comparison with interaction with real humans: this is precisely the environment we aimed to create with *Improbatics*.

Post-show questionnaire questions were based on a subset of the Presence Questionnaire [353]. These questions were originally developed for a small audience interacting in virtual reality domains. Questions from the original questionnaire were reworded or excluded if they pertained to immersive experiences inconsistent with improvised theatre. The final questionnaires presented to the audience and performers measured the system on the following five categories: 1) possibility to act, 2) realism, 3) evaluation of performance, 4) quality of interface, and 5) possibility to examine the performance. In addition to quantitative survey-based evaluation, we report qualitative assessment

comments. Exploring subjective opinions of performers provides us with valuable feedback about human-machine co-creation. These results are presented alongside a note that because we use opt-in audience surveys, there are likely biases in the data. That is, those audience members who choose to participate already create a subsample of the audience. Participation in the data collection was optional. No personal identifying information was collected. Performers and audience gave informed consent, and the study was approved by the ethics review board at the University of Alberta.

4.5 Results

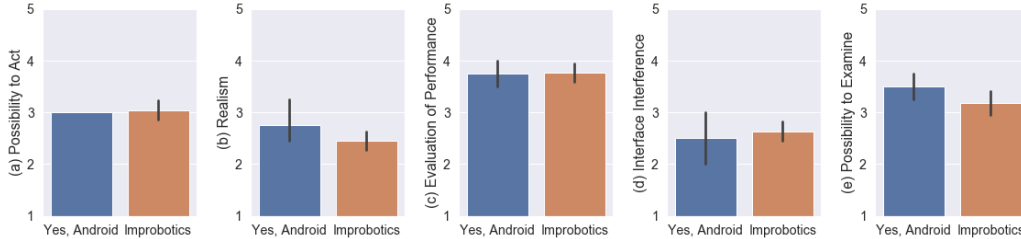


Figure 4.7: Performer Presence Analysis over Systems. *Yes, Android* performers in Toronto ($n = 4$) used a different system than *Improbotics* performers in Stockholm, London, and Edmonton ($n = 22$). This plot illustrates the comparison of analysis between the two different systems. Data presented is average opinion over respondents in group and 95 percent confidence interval.

We present here quantitative and qualitative results from experimentation with the *Improbotics* system. We have deployed the experimental configuration to three locations: 1) *Improbotics* Stockholm, Sweden (STO, $n = 6$), 2) *Improbotics* London, England (LON, $n = 7$), and 3) *Improbotics* Edmonton, Canada (EDM, $n = 9$), where n is the number of respondents. In addition to these locations, we also provide comparative results from performers in Toronto, Canada who performed in *Yes, Android* (TOR, $n = 4$). We additionally present data collected from audience members who attended a show in each respective city, denoted: LON-AUD ($n = 6$), STO-AUD ($n = 22$) and EDM-AUD ($n = 29$). While audience demographic data was not collected, we infer that audiences in LON and STO were generally other improvising

performers and audiences in EDM are representative of theatre-going patrons. Performer and audience data from multiple cities allows us to systematically measure the consistency and reproducibility of the experience on the evaluation metrics defined above.

4.5.1 Quantitative Evaluation

The questionnaire to the performers is presented below. For the audience questionnaire, the wording of the questions was modified to reference “the performers” instead of “you”.

1. (possibility to act) How much were you able to control events in the performance?
2. (realism) How much did your experiences with the system seem consistent with your real world experiences?
3. (evaluation of performance) How proficient in interacting with the system did you feel at the end of the experience?
4. (quality of interface) How much did the control devices interfere with the performance?
5. (possibility to examine the performance) How well could you concentrate on the performance rather than on the mechanisms used to perform those tasks or activities?

Overall, the actors were satisfied with the performance despite the limited realism of the setup (consistent between cities) and moderate interface interference. We note no significant difference between *Improbatics* and *Yes, Android*. Improvisors from LON, who had the most rehearsals and performance opportunities with the system, rated its realism the lowest but their proficiency with it the highest, judging that the system interface did not interfere significantly with the performance. Improvisors from EDM, who had only one rehearsal, had the most trouble concentrating on the performance rather than on the interface. We infer that, with practice, the system interface interfered less with the performance and that practice increases proficiency. Audiences rated the performers as having more control of the events during the performance than the performers.

Note that we do not split the responses from the performers of different types (Cyborg, Puppet, Puppet Master, CEO) due to the collaborative nature of improv and to the necessity to provide a single consistent show including all improvisors on the stage. Additionally, we observed that if one performer is limited in any way, it can affect the entire scene.

In addition to the qualitative questionnaire, we compare the utterance choices that the two Controllers (Puppet Master and CEO) are providing to the Puppet and the Cyborg respectively with lines from a script and lines from human improvisors. For this comparison, we selected several linguistic features (e.g. syllables-per-word, words-per-sentence, proportion of difficult words, VADER sentiment [131], and grammatical/spelling errors) indicating the complexity of the provided sentences.

While evaluating the quality of a dialogue interaction is difficult, these linguistic features can provide a measure of the information contained within each of the lines composing a dialogue. For the comparative lexicographical analysis we used a test set of lines from four different data sources. We analyse $L_{puppet} = 334$ lines from the Puppet Master, $L_{cyborg} = 2248$ lines generated by the dialogue system. We compare this lines with $L_{human} = 410$ lines from Free-will Human performers in Improbotics shows and $L_{script} = 1675$ lines from two published scripts. The scripts that we include for the analysis are stylistically diverse, and similar to scripts used for the *Actor's Nightmare* improvisation game. The two scripts we used are Tennessee Williams' "A Streetcar Named Desire" [351] and Hannah Patterson's "Playing with Grownups" [253].

As seen on Figure 4.8, when comparing what the Master typically types to what is found in theatre scripts or what is generated by the AI, we observe that the Master/Puppet improvise with shorter lines, with considerably more grammatical or spelling mistakes (which can be explained by the time pressure on the Master to give lines to the Puppet improvisor) and with a slightly more positive VADER sentiment (likely due to the training of improvisors encouraged to "yes, and" by displaying positive sentiments). These results support the conclusions that human-generated lines are shorter when typed and longer when spoken. As well, human lines are more positive, have less

difficult words than scripts and have more grammar and spelling mistakes than the artificial improviser generated lines.

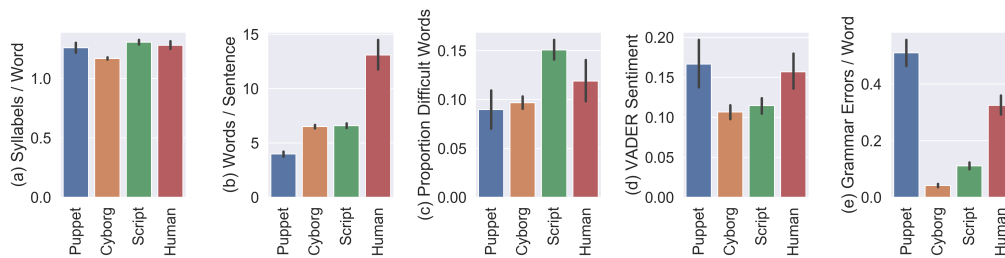


Figure 4.8: Comparative Lexicographical Analysis by Source. To compare the differences in text-based features we analyse a set of sentences from four different sources: 1) the Puppet Master, 2) the dialogue system or AI, 3) two published scripts, and 4) human lines transcribed with speech recognition at a set of shows in Edmonton. Data presented is the average over each group, with the 95 percent confidence interval.

4.5.2 Qualitative Evaluation

In addition to the quantitative evaluation, we also asked performers to provide feedback with the following prompt: *How would you compare performing alongside the system as compared to performing alongside a human?*

The results from this question allow us to better understand the expectations of the performers. Selected quotes from the professional improvisational performers who worked with the *Improbatics* system in a variety of roles are presented below, grouped into themes.

Theme 1: Improvising with the system is more work.

- *The system throws up some real curve balls which makes it different to performing with a human.*
- *You, as a human, have to be on your toes to validate the sometimes fun and crazy things that the Cyborg says.*
- *You get many more surprises when working with the system, which is wonderful.*
- *It is actually quite similar to playing with a normal human [who] has a crazy x-factor approach to improv.*

- *The system gives more “out-of-the-blue” comments, and it does not feel like it is cooperating with me to make a “perfect” scene.*
- *It took the scenes to wild places but it certainly felt like the whole time you’re just trying to keep it on the rails.*
- *If the AI was a human player it would be a quite annoying co-improvisor, with approx 80% screwball lines and non sequiturs.*
- *...it is a lot more work for me as a human to drive the scene, and that could be a bit lonely and cumbersome sometimes.*

Theme 2: The system cannot tell complete stories.

- *If you want to tell a story, humans tend to have to pick up the arc and carry it through, since the Cyborg rarely brings arguably important characters or plot items back.*
- *As long as the human performers provide the improv “platforms” then those being controlled simply get to have fun!*

Theme 3: Forces you to be a better improviser.

- *It requires more skill and practice to be able to respond to it without being too beholden to it.*
- *It makes it very important to be open and accepting. Blocking or denying of any kind only makes the “uncanny valley” deeper and more obvious.*
- *...you have to be more physical and [create] a reality which allows space for the “curve balls”, and the cyborg’s stunted performance, to make sense*
- *...you have to listen more, and drive the scene yourself, you treat your cyborg scene partner differently—you can’t rely on them completely*
- *I found it essential that the humans operating the system give performers enough to say; once or twice I was on stage with no lines coming through! Object work becomes super important in this instance!*

Theme 4: Like performing with a novice improviser.

- *It was like performing with a very new improviser with strange impulses.*
- *It takes a different mind-set, like being aware a fellow performer is very nervous and unpredictable.*
- *Every improviser is different. Any improviser has strengths and weak-*

nesses, just like the system. Another day another improviser.

4.6 Discussion and Summary

4.6.1 Reflections from Professional Improvisors

More than 20 professional improvisors have worked with the system and provided their experienced analysis and critiques which are summarized above. Their feedback largely fell into four thematic categories: 1) improvising with the system is more work, 2) the system cannot tell complete stories, 3) it forces you to be a better improviser, and 4) it was like performing with a novice improviser. Of these themes, two are negative (1 and 2), and two are positive (3 and 4). While working with the system is more work, this is largely due to the design of the system, to augment the humans performance. Currently, the system is focused on dialogue and has no notion of a “full story.” Future work is needed to incorporate narrative representations into the system, and we present two studies to this end in Chapter 5 and Chapter 6. The feedback that draws parallels to performing with novice improvisors is reassuring, as the goal of the system is to challenge the notion that “...the best improvisors make everyone else look great.”

4.6.2 Deception and Problem Solving

Improbatics is a narrative improv show, where neither the audience, nor the improvisors, know who is a Free-will Human, who is a remotely controlled Puppet, and who is an AI-controlled Cyborg. The AI dialogue system is controlled by the CEO Controller who follows the context of the scene and the narrative progression on stage, interactively producing the next line for the Cyborg performer. These lines are often nonsensical and add incongruity to the ongoing scene. The randomness of these lines was addressed directly in several of the participants’ responses. While the justification of these random offers provides fun, it can also be isolating and challenging for the human performers who feel they are “improvising with a beginner” and need to take care of the narrative progression.

The human Puppet Master Controller, who observes the improv scene from a hidden place, and who feeds lines to the Puppet via the earpiece, is tasked with a very difficult challenge. They need to listen to the scene and simultaneously type dialogue suitable for the next line. Alternatively, as we observed in several performances, the Puppet Master Controller can pretend to be AI-like and through playful deception (e.g. generating more nonsensical or disconnected lines of dialogue), introduce a wild-card into the audience’s mind.

We desire to push the imitation game as far as possible while creating an enjoyable performance. Thus, we encourage the improvisors to act in the most natural and intelligent way. They are expected to play to the full range of their emotions and physicality. That said, they are also aware of the conceit of the show and often they can introduce intrigue in the audience’s mind by pretending to be more AI-like, more robotic. Through this “double-bluff” any performer can act as if they are the Puppet, or Cyborg. As anecdotal evidence, some audience members incorrectly thought that a Free-will Human was a Cyborg in two out of six Improbabilities shows in London.

Improbabilities is a way to frame the task as an optimization problem for the performers where the main objective is producing an enjoyable theatrical performance. The secondary meta-objective, a nod to the permeable audience-performer relationship in improvisation, is playfully deceiving the audience.

4.6.3 Lack of Contextual Consistency

Through the comparison of the performances of the Cyborg, of the Puppet and of the classic improv game *Actor’s Nightmare*, we see how differently performers handle two types of contextual consistencies in improvised narratives. The first is broad consistency in the general theme of the improv (e.g., domain-specific vocabulary used in the scene). The second is fine-grained consistency in the articulation of the story (e.g., relationships between characters, character stance or world view).

In the game *Actor’s Nightmare*, where the improviser reads consecutive lines for a given character, selected from a random play, those lines are typically consistent among themselves, but disconnected from the general theme

of the scene. The fun of the game derives from seeing both actors striving at justifying the incongruity of juxtaposing, for instance, a classical drama with a science fiction setting. When performing a Puppet, the performer is typically given lines from a trained human improviser who listens to the context of the scene and types lines with both high-level thematic and fine-grained narrative consistency. Despite the best efforts of the Controller who curates the lines produced by the AI, the Cyborg typically gets inconsistent lines from the point of view of the narrative. With the topic model incorporated in A.L.Ex., some thematic consistency can be maintained [212]. So, the AI, when primed with words “ship” and “pirate”, might generate sentences about sea-faring and sword-fighting (for additional details see Section 3.3.2). Interestingly, this is the opposite of the *Actor’s Nightmare*, which lacks thematic consistency. Rather than just incorporating topic in the model, future iterations of the system could include additional context. For instance, models may re-inject scene specific content (i.e. character names, locations, relationships, noun and verb phrases) in generated responses are currently being investigated. To this end, methods of editing prototypes by retrieving and refining candidates is an relevant area of active research [106], [347].

4.6.4 Handling Timing in Improvisation

One of the most challenging technical hurdles for human-machine co-creation is that of timing. Verbal interaction is defined most-notably by the characteristic of rapid exchange of turns of talking. Gaps between these turns are often as short as 200 ms in natural human conversation. Latencies in language processing can be on the order of 600 ms [173]. This implies that humans are often taking turns talking based on predictions of the next line of dialogue from the others in the conversation. Given this extremely short latency expectation, there is often noticeable delay for the Puppet and/or Cyborg.

Our current system has a median response time of more than 2 seconds with some responses taking up to 4 seconds. The timing of these is seldom below 1 second unless we queue additional responses to a single input and force an interruption with a potentially out-of-context follow-up line. These timing

limitations are similar to *Actor’s Nightmare*, where one of the improvisors is restricted to reading lines from a script.

Luckily, such timing latencies can be smartly hidden by proficient improvisors through emotional, non-verbal, and/or physical actions. While, in our previous work with an automated and un-curated chatbot, improvisors would typically talk over a naïve robot voice responding with bad timing [212]. This happened significantly less often with the Cyborg or Puppet in *Improbabilities*, because all the people waited their turn to speak. Moreover, Cyborgs had (and used) the opportunity to skip an irrelevant or outdated line. Future work might develop methods for better handling of timing and turn-taking. Poor timing is a give-away for any system imitating a human conversationalist in interactive dialogue [11], [61].

4.6.5 Human Curation of Dialogue

Currently the system is based on the natural language generation model trained on movie dialogue presented in Chapter 3. While film subtitles are not the optimal dataset for training a dialogue system for live improvisational theatre, we chose this corpus to train the system because it was publicly available (unlike theatre and movie scripts), because it contained informal, realistic, conversational language and because improvisors typically draw their inspiration from movies and TV series. Given that many of the movies in the source material are from over half a century ago, there are strong biases in the training material toward offensive or out-of-date references. That said, without a set of improvised dialogue transcripts, movie dialogue is the best large scale corpora available for training these models. There is a need for human moderation and curation to ensure that the system is not immediately offensive. The current system could be improved by including automated metrics for offensive language detection and removal [67].

Improbabilities is focused on developing improvised dialogue in scenic improvisation. While critical to human-machine theatrical co-creation, this is only a small component of a larger automated story generation system. Incorporation of automatic plot generation techniques introduced nearly a century ago

in [59] could augment the system with directorial abilities and event-based story generation [205], [206]. A preliminary step towards incorporating such plot generation techniques is presented in Chapter 6.

In this chapter, we present *Improbatics*, an improvised performance which serves as a test-bed for human-machine theatrical co-creation and can be used for improving computational dialogue-based system for live performance. The system allows for Turing test-inspired experimentation. By confronting humans to the incongruity of machines sharing the stage with them, we can both create new opportunities for comedy and explore approaches to human-machine interaction. We presented results from three geographically unique locations where the system is currently being used to perform for live audiences. We compared the *Improbatics* neural network-based and movie dialogue-trained system, with the *Yes, Android* baseline system, which uses an online, publicly accessible chat-bot. We presented quantitative analysis evaluating the system in five categories: realism; possibility to act; quality of interface; possibility to examine; and evaluation of performance. We present qualitative analysis from professional improvisational performers. While this chapter focuses on improvisation, these paradigms and results are valuable for research in other areas of human-machine physical and verbal interaction [169], [210].

Chapter 5

Information-Theoretic Narrative Generation

*There is no reason why the simple shapes of stories can't be fed into computers, they are beautiful shapes... beautiful curves... Computers can now play chess so I don't understand why they can't digest this very difficult curve
I am going to draw for you now.*

– Kurt Vonnegut on the Shapes of Stories [341]¹

In this chapter, we consider the challenge of designing an artificial agent capable of interacting with humans in collaborative dialogue to produce creative, engaging narrative. Collaborative dialogue is distinct from chit-chat in that it is knowledge building, each utterance provides just enough information to add specificity and reduce ambiguity without limiting the conversation [320]. We use concepts from information theory to define a *narrative arc* function which models dialogue progression. We demonstrate that this function can be used to modulate a generative conversation model and make it produce more interesting dialogues, compared to baseline outputs. We focus on two antithetical modes of modulation: *reveal* and *conceal*. Empirically, we show how the narrative arc function can model existing dialogues and shape conversation models

¹From his rejected master's thesis—he considered it his greatest contribution

A preprint of this chapter is published as **Mathewson KW**, Castro PS, Cherry C, Foster G, Bellemare MG (2019) Shaping the Narrative Arc: An Information-Theoretic Approach to Collaborative Dialogue. arXiv preprint arXiv:1901.11528.

towards either mode. We conclude with quantitative evidence suggesting that these modulated models provide interesting and engaging dialogue partners for improvisational theatre performers.

5.1 Introduction

Designing and building computational models that generate meaningful dialogue for human-interaction, in an interesting and engaging manner, is a challenging open problem. As personal digital assistants increase in popularity, proper conversational capabilities may allow them to provide creative, playful, and helpful interactions. Conversational agents can be effective for health-care [21], by supporting cognitive-behavioural therapy for treating depression [85], helping patients with chronic pain [230], and supporting reminiscence [245]. These applications require systems capable of understanding and collaboration.

What makes some dialogues more interesting than others? Interesting collaborative dialogue constructs knowledge iteratively [320] and depends on each speaker bringing information to the conversation [288]. Interestingness is also subjective and difficult to directly optimize via numerical methods [176], [338].

Rule-based conversational models have existed for over 50 years [344]. These methods are limited by hand-tuning and engineering to predict and handle possible inputs. Generative language models maximize the likelihood of an utterance (e.g. a sentence or sequence of words) [101]. These models can predict the likelihood of an utterance by considering the sentences as a sequences of words, sub-word units, characters, and/or tokens [296]. This objective can result in generated sentences which are grammatically correct, and bear a semantic relationship to the context surrounding it, but lack global consistency [189].

Our work generates interesting dialogue by using a narrative arc to incrementally construct shared knowledge. A narrative arc defines evolving qualities of emotion, tension, or topic over a story [23]. We draw inspiration from

improvised theatre, where actors collaborate in real time to develop narrative based on thematic constraints [147]. Improvised theatre is a unique storytelling medium which relies on collaborative dialogue in which each utterance must carry significant information [320]. We appeal to the two *golden rules* of improvised dialogue, characteristic of interesting collaborative dialogue [147], [288]. Good dialogue should 1) **accept** (i.e. be consistent with the dialogue thus far and 2) **reveal** (i.e. progress the dialogue with new information).

In this work, we propose a new method to modulate a conversation model, which **accepts** input utterances by generating consistent and **revealing** responses. Our approach combines a conversational model with a topic classifier, or *universe model*. We borrow the term *universe* from improvised theatre where it is used to describe the *world-as-we-know-it* [147], [221], [269]. The universe encompasses associations surrounding the dramatic world, and is motivated by the possible world semantics theory [162].

We identify two modes of operation for our shaping method: *revealing* and *concealing*. Revealing dialogue adds additional information about the current universe. Generating utterances which progress a scene with new information is the primary goal of our approach. Concealing dialogue avoids exposing new information about the universe. The ability to generate both revealing and concealing dialogue is a convenient side-effect of this method.

The universe model characterizes the information revealed by each utterance in a sequence. We refer to this information profile across utterances as the narrative arc. By tuning the revealingness we can selectively choose utterances to shape the narrative arc to produce more interesting and engaging dialogue. We argue that a balance between revealing and concealing is required for interesting and engaging collaborative dialogue [320]. Both over-specification and ambiguity are undesirable [147], [288]. We hypothesize that there is an ideal region of information revelation which our method can expose in existing text-based narratives such as movie scripts.

5.2 Shaping the Narrative Arc

In this section, we present a mechanism for shaping the narrative arc inspired by combining methods exploring entropy in textual documents [303] with the *Simple Shapes of Stories* described by Vonnegut.² We describe concepts of conversation and universe models. Then, we show how these combine to describe a narrative arc. Finally, we show how the narrative arc can be used to generate interesting dialogue.

5.2.1 The Conversation Model

A conversation model accepts an input utterance and generates one, or several, output utterance(s). The conversation model maintains local coherence by conditioning output generation on the input. We write \mathcal{X} to denote the set of possible utterances (i.e. sequences of words); in this work, \mathcal{X} is a collection of English sentences. A sequence of t successive utterances is a *dialogue*, denoted $x_{1:t}$. A conversation model yields a probability q of an utterance x_t given a dialogue $x_{1:t-1}$.

We focus on dialogue generation using three retrieval-based conversation models. The first two models are based on the OpenSubtitles dataset [186]. Pre-processing details are included in the supplementary material. When queried with an input line x_{t-1} , a model returns K candidate responses:

- **Baseline Random model:** sample K lines from \mathcal{X} .
- **Deep neural network model (DNN):** we embed all the lines in \mathcal{X} into a latent semantic space S using the Universal Sentence Encoder [46]. We encode the input line into S , and return the K approximate nearest neighbours [134] in S using the L^2 norm as the distance metric.³

Similar to the DNN model, a third model (**Books**), responds with semantically related nearest neighbour lines from literature, filtered for offensive content.⁴

²From K. Vonnegut lecture: <https://goo.gl/JuEDVR>

³<http://mathworld.wolfram.com/L2-Norm.html>

⁴<https://books.google.com/talktobooks/>

5.2.2 The Universe Model

The universe model measures how each successive utterance of a dialogue influences the probability distribution over universes. For a given utterance, the universe model calculates a probability distribution over universes. For a sequence of utterances, we use recursive universe belief propagation (Sec. 5.2.3) to update the posterior over the course of a dialogue. Revealing dialogue would concentrate probability mass on a single universe, and concealing dialogue would maintain posterior likelihood over a set of universes. The shape of this sequence of posteriors is the narrative arc (Sec. 5.2.4). We investigated reveal/conceal dynamics using three different universe models based on probabilistic topic classifiers.

- **Newsgroups:** Using the newsgroup classification dataset, we filter out stop-words, created frequency vectors, and use the TF-IDF (term frequency / inverse document frequency) [286] word weighting scheme to account for word importance in the corpus. We train a naïve Bayes classifier on 5 aggregate topic universes (COMPUTERS, RECREATION, RELIGION, SCIENCE, and TALK) [144].
- **Movies:** naïve Bayes classifier, trained similar to Newsgroups, using a collected dataset of film synopses and one of 10 corresponding genres (DRAMA, COMEDY, HORROR, ACTION, CRIME, ROMANTIC COMEDY, ROMANCE, THRILLER, FILM ADAPTATION and SILENT FILM) from Wikipedia data [121].
- **DeepMoji:** Deep neural network that takes input text and outputs a distribution over a set of 8 aggregated emoji universes: (SAD, MAD, MEH, NERVOUS, GLAD, MUSIC, LOVE, and MISCELLANEOUS) [83]. Input text is not transformed, and a pretrained model is used.⁵

5.2.3 Recursive Universe Belief Propagation

We desire a means by which we can update the universe belief incrementally as evidence is accumulated with each successive utterance in a dialogue. We

⁵github.com/bfelbo/DeepMoji

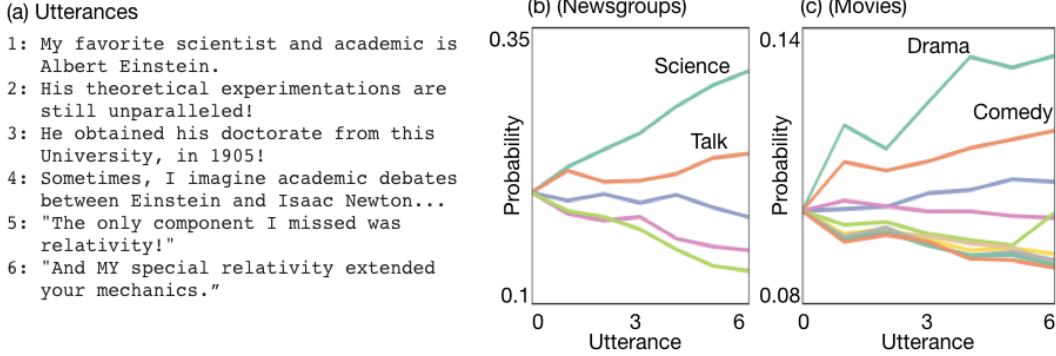


Figure 5.1: The narrative arcs of a synthetic dialogue (a), using the Newsgroups universe model (b) and Movies universe model (c). This dialogue is likely SCIENCE or TALK under the Newsgroups model, and DRAMA or COMEDY under the Movie genres model.

begin by defining the notion of a *universe model* as a means of modelling the dynamics of information revelation. Consider a finite set of *universes*, \mathcal{U} . The role of a universe model is to assess the compatibility of an utterance with a given discrete universe, $u \in \mathcal{U}$. Given such a model, we develop a method to update the agent’s posterior universe distribution over a sequence of utterances. For each universe u , the universe model assigns a likelihood $p(x_t | x_{1:t-1}, u)$ to an utterance x_t , conditioned on a dialogue $x_{1:t-1}$.

The universe model iteratively updates a posterior belief over universes, in a similar spirit to prediction with expert forecasters [48]. The probability of a given universe depends on iteratively combining evidence in support of that universe. We define the the posterior probability over universes \mathcal{U} given a sequence of t utterances $x_{1:t}$ as:

$$p_t(u | x_{1:t}) = p_{t-1}(u | x_{1:t-1}) \times \frac{p(x_t | x_{1:t-1}, u)}{p(x_t | x_{1:t-1})}$$

Where $p_{t-1}(u|x_{1:t-1})$ is prior probability, $p(x_t|x_{1:t-1}, u)$ is likelihood of utterance conditioned on the past dialogue and universe, and $p(x_t|x_{1:t-1})$ is likelihood of utterance under the conversation model.

Let $p_0(u|\cdot) = 1/|\mathcal{U}|, u \in \mathcal{U}$ be an initially uniform distribution over universes, or universe model’s prior. We can marginalize out the universe if the evidence is consistent over all hypotheses. To illustrate the relationship between utterance likelihood and universe, we can explicitly write the marginal

likelihood as:

$$p(x_t | x_{1:t-1}) = \sum_{u'} p_{t-1}(u' | x_{1:t-1}) p(x_t | x_{1:t-1}, u')$$

Thus, the posterior is updated recursively as:

$$\begin{aligned} p_t(u | x_{1:t}) &= p_{t-1}(u | x_{1:t-1}) \\ &\times \frac{p(x_t | x_{1:t-1}, u)}{\sum_{u'} p_{t-1}(u' | x_{1:t-1}) p(x_t | x_{1:t-1}, u')} \end{aligned} \quad (5.1)$$

In practice, it may be convenient to use the output $z(u|x_t)$ of a probabilistic classifier *in lieu* of a likelihood function conditioned on past utterances $x_{1:t}$ and universe u . Universe classifiers can be trained separately from language models, and provide complementary signal if model input distributions overlap. This assumption is justified when both models work with similar training corpus vocabularies. We view the probability distribution over universes output by the universe model as derived from a joint distribution $z(u, x_t)$, of the universe u , and utterance x_t . With $z(u)$ as the prior distribution over universes, the conditional probability is:

$$z(u | x_t) = \frac{z(u, x_t)}{z(x_t)} = z(u) \times \frac{z(x_t | u)}{z(x_t)}$$

We can substitute $z(\cdot|x_t)$ for $p(x_t|x_{1:t-1}, \cdot)$ in Eq. 5.1 by assuming conditional independence (i.e., $p(x_t | x_{1:t-1}, u) = p(x_t | u)$), uniform prior distribution (i.e., $z(u) = 1/|\mathcal{U}|, u \in \mathcal{U}$) and constant marginal probability (i.e., $z(x_t) = \sum_{u'} p_t(u') p(x_t | u')$). These assumptions are justified when the probabilistic topic classifier is a naïve Bayes classifier with uniform prior [22]. Thus, the substitution follows the following steps:

$$\begin{aligned} p(x_t|x_{1:t-1}, u) &\approx z(x_t|u) && \text{[cond. independence]} \\ &= \frac{z(u|x_t)z(x_t)}{z(u)} && \text{[Bayes' theorem]} \\ &\approx z(u|x_t)z(x_t) && \text{[}z(u)\text{ uniform prior]} \\ &\approx z(u|x_t) && \text{[}z(x_t)\text{ const. marginal]} \end{aligned}$$

Eq. 5.1 thus becomes:

$$p_t(u \mid x_{1:t}) = p_{t-1}(u \mid x_{1:t-1}) \times \frac{z(u \mid x_t)}{\sum_{u'} p_{t-1}(u' \mid x_{1:t-1}) z(u' \mid x_t)} \quad (5.2)$$

5.2.4 The Narrative Arc

As defined in Eq. 5.2, the posterior $p_t(\cdot)$ is a function of the dialogue $x_{1:t}$. We define the *narrative arc* as the sequence of universe distributions $p_0(\cdot), p_1(\cdot), \dots$ iteratively calculated for the dialogue. The arc depicts the evolution of a belief over a set of universes. The *narrative arc function* maps $\mathcal{X}^t \rightarrow \mathcal{S}(\mathcal{U})^t$, where $\mathcal{S}(\mathcal{U})$ is a probability simplex over \mathcal{U} . We discuss three properties of the narrative arc of the synthetic dialogue shown in Fig. 5.1:

1. Utterances affect the arc in varying degrees. “My favourite scientist and academic is Albert Einstein” is similarly likely under SCIENCE and TALK, and less likely under the RECREATION universe (bottom green line). Different utterances should have different effects on $p_t(\cdot)$.

2. A concentrating posterior signals a revealing dialogue. A dialogue which emphasizes scientific content, for example, should see $p_t(\text{SCIENCE}|\cdot) \rightarrow 1$. Conversely, we would expect a concealing dialogue to spread the posterior across multiple universes.

3. A universe model is a perspective on dialogue. Different universe models can expose different aspects of the same dialogue. Replacing the Newsgroups universe model by a Movies universe model suggests the dialogue is from a DRAMA and/or COMEDY universe. This dialogue would be considered revealing under both universe models.

In this section, the universe model was applied to a fixed dialogue $x_{1:t}$, but the model also provides a criterion for favouring utterances when *generating* dialogue.

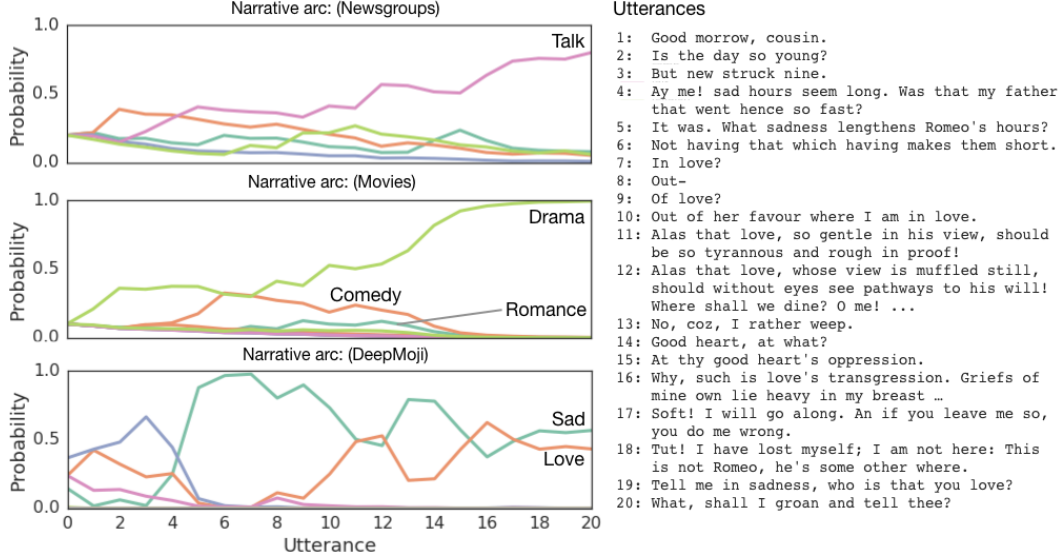


Figure 5.2: First 20 lines of Romeo and Juliet modelled with Newsgroups (top), Movies (middle), and DeepMojji (bottom) universe models.

5.2.5 Generating Dialogue with the Narrative Arc

The entropy of the posterior $p_t(\cdot)$ is given by:

$$H(p_t(\cdot)) := - \sum_{u \in \mathcal{U}} p_t(\cdot) \log p_t(\cdot)$$

Then, the entropy change $\Delta(\cdot)$ due to a new utterance, x_t , given the past dialogue, $x_{1:t-1}$, is defined as:

$$\Delta(x_t; x_{1:t-1}) := H(p_{t-1}(\cdot)) - H(p_t(\cdot))$$

The term $\Delta(x_t; x_{1:t-1})$ measures how much a given utterance x_t changes the entropy of the posterior, given the previous utterances $x_{1:t-1}$. A positive value of $\Delta(\cdot)$ is a reduction in entropy (i.e. revealing). Conversely, a negative value of $\Delta(\cdot)$ is an increase in entropy (i.e. concealing). We define the *score* of an utterance x_t , with respect to a dialogue, $x_{1:t-1}$, as:

$$\sigma(x_t; x_{1:t-1}) := \exp\{\alpha \Delta(x_t; x_{1:t-1})\}, \quad \alpha \in \mathbb{R}$$

The exponential function is a convenient way to ensure strict positivity and preserve the ordering of scored candidates. We use our entropy-based score function σ to modulate the sampling of a base conversation model, q , toward

\tilde{q} , which depends on the change in entropy due to the new utterance.

$$\tilde{q}(x_t | x_{1:t-1}) \propto q(x_t | x_{1:t-1}) \times \sigma(x_t; x_{1:t-1}) \quad (5.3)$$

If $\alpha = 0$, $\sigma(\cdot) = 1$ and candidates are sampled according to $\tilde{q} = q$. If $\alpha \neq 0$, q is modulated by the score $\sigma(\cdot)$. Modulation *mode* depends on the value of α :

- $\alpha > 0$ (**reveal**): modulate q towards revealing the universe. The probability of utterances likely under the universe with highest probability are increased.
- $\alpha < 0$ (**conceal**): modulate q towards concealing the universe. The probability of utterances likely under multiple unlikely universes is increased. Utterances not supporting the likely universe are made more likely.

We use these two modulations for filtering samples from our base conversation model. We filter via one of two methods for sampling from an unnormalized distribution: **greedy sampling** and **rejection sampling**. Greedy sampling scores a set of samples from the conversation model and selects the candidate with the maximum score. Scoring a large set of candidates can be time intensive. Rejection sampling (Alg. 2) can sample from the desired unknown modulated distribution online [237]. Additional details on rejection sampling are included in the supplementary material. As the entropy function is bounded, the utterance score σ is bounded. In practice, we set a max score and weigh all utterance scores σ above the threshold equally. Both filtering methods have benefits. Rejection sampling provides a smoother distribution and does not require scoring a large set of candidates. Greedy sampling is less sensitive to the range of Δ from different utterances.

5.3 Evaluation

5.3.1 Narrative Arc of Existing Dialogues

In Fig. 5.2, we visualize the narrative arc underlying the first 20 lines of Shakespeare’s Romeo and Juliet using three universe models: 1) Newsgroups, 2) Movies, and 3) DeepMoji.

Algorithm 2 Generating dialogue with rejection sampling.

Given: conversation model q , scoring function σ , first line x_1 , length N , max score M , max samples S

Return: dialogue $x_{1:N}$

for t in $2 \dots N$ **do**

while step $\leq S$ **do**

 sample $x_t \sim q(x_t | x_{1:t-1})$

 sample $r \sim \text{Uniform}(0, 1)$

if $r \leq \sigma(x_t; x_{1:t-1})/M$ **then**

 append x_t to $x_{1:t-1}$

break

Fig. 5.2 illustrates the entropy-reducing nature of good dialogue by showing us the underlying, evolving, narrative arc. Under the *Newsgroups* universe model, the dialogue evolves toward a TALK-centric universe. Under the *Movies* model, the same dialogue balances between comedy and drama before shifting towards drama. Finally, using the *DeepMoji* universe model, a developing ambiguity between DeepMoji universes SADNESS and LOVE is uncovered. This supports the hypothesis that existing dialogues exhibit underlying narrative arcs conditioned on universe models. Additional samples exposing narrative arc dynamics are presented in Appendix F.

5.3.2 Shaping the Narrative Arc

In this section, we demonstrate that our method is able to modulate conversation models toward generation of revealing or concealing dialogues. Linguistic quality and semantic consistency of utterances are determined by the language underlying the conversation model. Here, we emphasize evaluation of narrative arc shaping.

We use the DNN conversation model to test how preferential selection, induced by our score function, can modulate information introduced into the conversation. In Fig. 5.3 we present characteristic narrative arcs and dialogues using concealing (top), neutral (middle), and revealing (bottom) modes. Each generation was primed with the first two lines from *Romeo and Juliet* (shown in bold in Fig. 5.3).

A significant difference is exposed between concealing (top) which tends

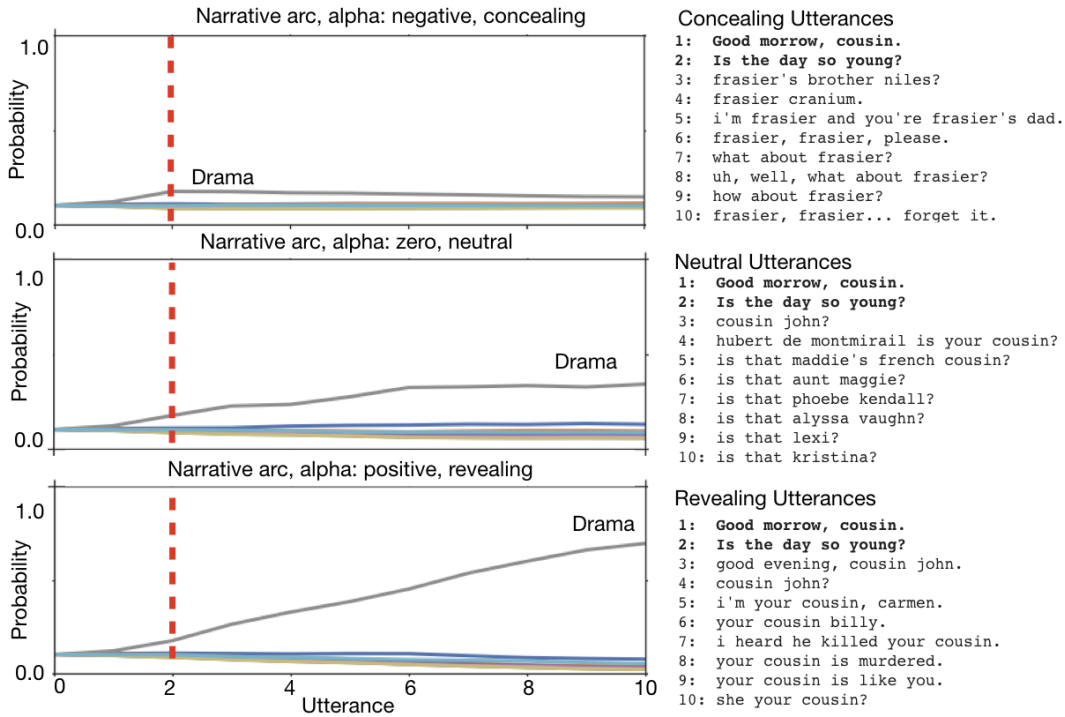


Figure 5.3: Narrative arcs over 10 utterances at increasing α values: concealing (top), neutral (mid), revealing (bottom). On the right are utterances generated by each model after priming (bold). Dotted red line indicates the start of narrative arc shaping.

toward a high entropy, uniform universe distribution, and revealing (bottom) where drama tends toward 1.0. DRAMA remains the most likely universe (and visible on all plots) as it was supported by the first two lines and subsequent utterances did not significantly shift the distribution. Fig. 5.3 also shows the dialogue generated by the model. Concealing utterances do not add information to the dialogue, revealing utterances incorporate new information over the course of the dialogue.

We next evaluate our method’s ability to generate concealing/revealing dialogue by measuring the entropy under both an objective universe (i.e. the universe model used for scoring in generation) and a test universe not used for scoring. We use the Newsgroups universe model for objective scoring and the Movies model for testing. A random conversation model is used to generate response candidates.

We generate 20 conversations following a process similar to Algorithm 2

but using greedy sampling. Each conversation starts with a random dialogue starter line to encourage diversity and then 19 lines are sampled from the conversation model using the narrative arc function. This approximates the length of a medium-duration improvised conversation [288].

Results are presented in Fig. 5.4. There is a significant difference between the entropy under the objective and testing universes, but each model exhibits similar dynamics over the dialogues. We conclude that concealing dialogue can conceal under multiple universes, and revealing dialogue can reveal information under multiple universe models.

The revealing/concealing dynamics of each utterance may be related to measurable lexicographical qualities such as words per sentence (WPS). We analysed the language used in 190 lines from each model and found a significant difference ($p < 0.001$) between utterances selected by the revealing model (9.26 ± 5.7 WPS) and utterances selected by the concealing model (5.05 ± 2.79 WPS).

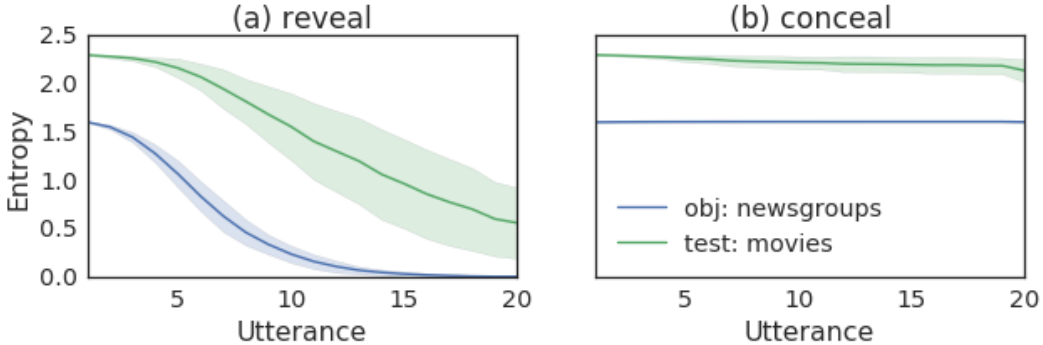


Figure 5.4: Revealing and Concealing across Universe Models. Dialogue generated to be (a) revealing ($\alpha = 20$) under the objective model Newsgroups is revealing under the testing Movies universe. The same is true for (b) concealing ($\alpha = -25$) dialogue. Data shown are means and standard deviation (shaded) over 20 runs of random conversation model.

5.3.3 Predicting the Next Best Line

We next test the system’s ability to add information to improve performance on a prediction task. Given a sequence of 5 gold-standard conversational utterances and a list of 10 next utterance candidates (i.e. the ground truth

and 9 distractors), can the universe model be used to improve accuracy of predicting the ground truth?

Evaluation compares top-3 accuracy and mean reciprocal rank (MRR) over samples in a held out test set. Accuracy measures the likelihood that the system scores the ground truth within the top-3 candidates against the distractors. MRR compares average ground truth ranking across conditions. A text2text Transformer language model was trained on the OpenSubtitles dataset [186] to predict an output line given a set of input lines [335]. Additional details are included in Appendix F.

The trained Transformer model was used to assign a perplexity score for output line candidates given an input context line. For each unique subtitle file in the validation and test sets, the concatenation of the first 5 lines serve as input context and line 6 is the ground truth output to be predicted. Negative candidates are randomly selected from lines in the respective corresponding data segment (i.e. validation or test sets), thus may not be from the same source file as the input context lines.

The perplexity under the trained conversation model serves as the unmodulated probability $q(x_t|x_{1:t-1})$ (Eq. 5.3) of selection in the prediction task. The input sequence is then passed, line-by-line, through a Newsgroups universe model and a score is assigned to each candidate relative to the change in entropy of the evolving posterior. The α value is modulated over 100 evenly spaced values between $[-2, 2]$. The accuracy of predicting the ground truth in the top-3 candidates and the MRR of the ground truth are computed.

The results on the validation set are shown in Fig. 5.5. By selecting the correct α value, the likelihood of correctly selecting utterances revealing an incremental amount of information increases significantly. Note the shape of the curve as α changes. As hypothesized, there exists a region, between 0 and 1 where the ‘right’ amount of universe information is revealed. This region corresponds to the notion that each line of dialogue will reveal some, but not too much, information about the universe. As α continues to increase, the accuracy decreases below the neutral baseline. The top-3 accuracy of prediction increases when the universe model boosts the probabilities of appropriately

CM	UM	Top3Acc	MRR
T2T@5	NG	0.520	0.456*
T2T@5	Neutral	0.507	0.444
T2T@1	NG	0.483	0.428*
T2T@1	Neutral	0.469	0.412
Unigram	NG	0.366	0.337*
Unigram	Neutral	0.296	0.290
Random	Neutral	0.302	0.294

Table 5.1: Results for predicting the next line. CM is the conversation model, UM is the universe model, Top3Acc is the accuracy of predicting the ground-truth in the top-3 of 10 candidates, and MRR is the mean reciprocal rank of the ground truth. Unigram CM calculates the perplexity of each candidate given the input lines as training corpus. T2T@N is a Tensor2Tensor Transformer model which uses the previous N lines as an input to predict the output and NG is the Newsgroups universe. A Neutral universe model represents no modulation which is equivalent to $\alpha = 0$. * indicates $p < 0.05$ for a Students’ t-test comparing MRR to the Neutral model.

revealing dialogue. The validation set is used to set the optimal α , which is then used to score samples in the test set and results are presented in Table 1. Two additional models are included for comparison. *T2T@1* uses 1 preceding the ground truth as context. *Unigram* assigns a perplexity to output candidates by building a unigram language model using the 5 input lines as a corpus. A smoothing factor of 1×10^{-5} is used for out-of-vocabulary words. Additionally, a random conversation baseline model is included. For each model tested, information from the universe model significantly improves the predictive accuracy on this task.

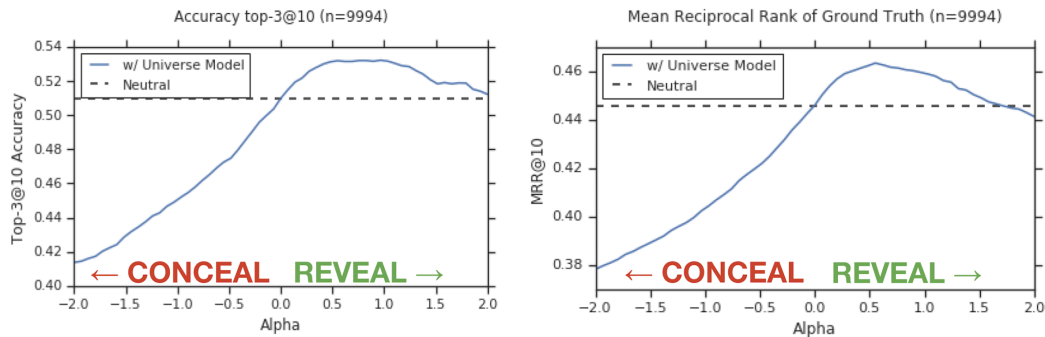


Figure 5.5: Information revelation region as α varies for (left) top-3 accuracy and (right) MRR in universe model modulated prediction task

5.3.4 Interactive Collaborative Dialogue

Finally, as a practical implementation case-study, we tested how this system performs in collaborative dialogue through interaction with humans. 4 expert improvisational theatre performers engaged with the system in 3 text-based conversations. Each conversation consisted of 5 utterance-response pairs for a total of ten utterances (i.e. an average length of a short-duration improvised scene [288]). Subjects are native English speakers with 5+ years professional performance experience and are familiar with shared narrative development and collaborative dialogue. Each interacted with revealing, concealing, and neutral models in a randomized order unknown to the them. Transcripts of actor-system dialogues and qualitative feedback are included in Appendix F.4.

As described in Sec. 5.2.1, this experiment used the Books conversation model and the DeepMoji universe model. Following the interactions, each performer was asked the following question: “please rank the conversations from 1 (most engaging) to 3 (least engaging)”. Engagingness was defined to align with the notions of revealing and concealing in this work. An agent is engaging for shared scene development if it brings just enough information to add specificity and reduce ambiguity but not limit the conversation.

Three of the four performers ranked the revealing model, $\alpha > 0$, as the most engaging. Those three performers ranked $\alpha = 0$ as being less engaging due to being “too random”. All subjects ranked $\alpha < 0$ as being least engaging and not bringing enough information to the scene. These results support the hypothesis that α can effectively modulate the engagingness of collaborative dialogue in human-machine interaction. Qualitative feedback from expert assessment surveys are included in Appendix F.5.2.

5.4 Related Work

Collaborative dialogue between humans and machines has been proposed as a grand challenge in artificial intelligence [34], [206], [211]. Previous methods have used hard coded rules, decision trees, and event representations to generate novel narrative chains [203]. We used an deep neural network-based

generative language model enhanced with universe model information in the context of improvised theatre [212].

While neural response generation systems provide a trainable end-to-end system for language generation, these methods are prone to providing generic, unspecific responses [177]. Advances have improved generated responses by optimizing sentence encoding and decoding jointly, post-generation candidate re-scoring [30], [313], [340], reinforcement learning [176], hierarchical models for distilling extended context [300], and auxiliary training objectives, such as maximizing mutual information [177], and personality specificity and consistency [178], [364]. In future work, universe models and conversational models could be trained jointly.

Our work is related to the controlled generation of text using disentangled latent representations [9], [129], [365]. Previous work has used a topic-transition generative adversarial network to enforce smoothness of transition of subsequent utterances [181]. These methods use neural encoder-decoders and generate responses given an input sequence and a desired target class for the response.

Other work has aimed to improve candidates returned by retrieval-based conversation models [349]. These methods utilize a conversation model to find similar prototypes using embedding distances and refine prototypes with a sequence-to-sequence model [106]. We do not refine candidates from the conversation model, rather we sample and select using a scoring function defined by the revealing and concealing parameter.

Similar to universe models, topic models or lexical fields have been shown capable of tracking general subjects of a text [25], [92]. Dynamic topic models characterize the evolution of topics over a set of documents over time [24]. Our work differs in that we generate dialogue using the evolving probabilistic belief during a single conversation, as opposed to tracking topical shifts over longer time-scales. Using a probabilistic classifier for narrative tracking has been explored previously [232], [274]. These works used sentiment classifiers to track emotion and plots arcs through narratives. We extend these works by using probabilistic universe models collaborative dialogue generation.

5.5 Discussion and Summary

While innovations have improved the linguistic quality, semantic alignment, and consistency of utterances generated by neural models, generated conversations still lack interestingness and engagingness. Our work generates engaging dialogue by shaping the underlying narrative arc as opposed to improving the training of generative language models. The methods presented are agnostic to both the universe and the conversational model used. Using rules from improvised theatre, we quantitatively define the evolution of interesting and engaging dialogue.

In this work we focus on genre, emoji, and topic-based universe models. Other universe models to be explored involve causality of events, directions of relationships, or audience reaction prediction [59], [76], [158], [276], [329]. While this work explores the interaction between a base conversation model and a universe model, this method could be compatible with image or video generation.

The main contribution of this work is the computational formalization of the narrative arc, an information-theoretic framework for collaborative dialogue interaction. The framework fills a gap in previous research by connecting the utterance-level improvements of language models with the conversation-level improvements of universe tracking. This is done by sampling candidates from a conversational model using a universe model and the narrative arc. We illustrate narrative arcs underlying popular dialogues and show how universe models can be combined with conversation models to generate interesting dialogue. We present empirical results showing how the narrative arc can improve accuracy on a next line prediction task. Finally, we present an expert user-study to validate our model.

Chapter 6

dAIrector: Contextual Story Arc Generation

[In] artificial intelligence... machines are made to behave in wondrous ways, often sufficient to dazzle even the most experienced observer. But once a particular program is unmasked, once its inner workings are explained... its magic crumbles away...

– Joseph Weizenbaum, 1966

dAIrector is an automated director which collaborates with humans storytellers for live improvisational performances and writing assistance. *dAIrector* can be used to create short narrative arcs through contextual plot generation. In this work, we present the system architecture, a quantitative evaluation of design choices, and a case-study usage of the system which provides qualitative feedback from a professional improvisational performer. We present relevant metrics for the understudied domain of human-machine creative generation, specifically long-form narrative creation. We include, alongside publication, open-source code so that others may test, evaluate, and run the *dAIrector*.

A version of this chapter was presented as Eger M and **Mathewson KW**. (2018) *dAIrector: Automatic Story Arc Generation through Knowledge Synthesis*. Joint Intelligent Narrative Technologies / Intelligent Cinematography and Editing Workshop at AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE). November 2018. Edmonton, Alberta, Canada.

6.1 Introduction

Improvisational theatre (improv) is an art-form in which narratives are developed ad-hoc in front of a live audience [147]. Performers are prompted with a concise, ambiguous suggestion (e.g. a location or character relationship) and then share narrative development through action and dialogue. Often these prompts are provided by the audience throughout a performance. The most interesting challenge of improvisation is incorporating new suggestions, seemingly unrelated to the narrative. Improvisation’s live justification has been proposed as a model for real-time dynamic problem solving [200], [315]. Improv has been proposed as a grand challenge for machine learning systems [206] potentially as an extension to the Turing test [211], [330]. The *dAIrector* collaborates with human improvisors for semi-automated story beat generation, suitable for improvisation performance, through knowledge graph synthesis. First, we describe some background on story generation, improvisational theatre, and plot graphs (from Plotto and TV Tropes). Then, we describe our approach and present quantitative and qualitative evaluation. We conclude with discussion of limitations and future work.

6.2 Background and Related Work

6.2.1 Automated Story Generation

The research problem of automated story generation (ASG) is concerned with generating a sequence which collectively form a narrative [59], [224]. The sequence can be composed of abstract concepts such as events or actions, or concrete text-based elements such as paragraphs, sentences, words, or characters. Different levels of abstraction and concreteness are accompanied by different challenges. For instance, stories defined at high levels of abstraction maintain step-to-step coherence easier but are simplified and lack unique, specific details.

Previous ASG systems have used symbolic planning and extensive hand-engineering [277]. Open story generation systems use machine learning tech-

niques to learn representations of the domain from the training data and incorporate knowledge from an external corpus [174]. Martin *et al.* [204] address the abstraction level challenges by using recurrent neural networks (RNNs) and an event representation to provide a level of abstraction between words and sentences capable of modelling narrative over hundreds of steps. They provide a method of pre-processing textual data into event sequences and then evaluate their event-to-event and event-to-sentence models. Our methods are distinct from this technique as we do not focus on the problem of sentence generation from words or characters. The *dAIrector* embraces human co-creators to provide dialogue for given plot point descriptions and context.

Narrative generation approaches, such as TALESPIN [223], focus on actions, their effects, and element relationships to delineate character intentions [277] and conflict [342] which ultimately leads to satisfying an author defined goal. Alternatively, there may be no predefined goal, and systems may discover actions autonomously [324], ideally resulting in an interesting story. To produce an interesting story in the context of improv theatre, however, the prescription of actions (e.g. lines of dialog, character choices, stage directions) is less desirable. A description of a situation suffices to inspire actors who can then translate the prompt into actions, with vague and ambiguous prompts giving the actors freedom to explore the scene [288].

6.2.2 Digital Storytelling

The ad-hoc storytelling experience present in improv theatre has been used for research into digital storytelling for more than two decades. Perlin and Goldberg [256] use concepts from improvisational theatre to populate virtual worlds, while Hayes-Roth and van Gent [112] describe virtual agents that perform improvisational theatre, modifying their appearance to convey simulated emotional state. Several knowledge-based approaches have been proposed for various problems in the space of acting in the improv theatre, such as scene introduction [246], fuzzy reasoning [199], affect detection [363], and robotic actors [212].

Through collaboration between human and machine, complex stories can

be constructed. Generative plot systems have been developed for nearly 100 years [59]. These systems aim to aid human creators with algorithmically generated prompts to explore diverse plots. Through interaction with generative systems, users are inspired to engage with topics they would not have otherwise. The excitement of exploring unknown spaces and engaging with novel topics defines the art-form of theatrical improvisation [147].

6.2.3 Improvised Theatre

In improvised theatre (improv) there is no script and no rehearsal; the show is written and performed at the same time. It is an art derived from the spontaneous justification of pseudo-randomness. Improvised theatre has been described as a suitable test bed for human-machine co-creation systems [200], [206], [212]. In improv, performers must attend to, and remember, details in the story and must synthesize previous information with novel dialogue and actions to progress a narrative. Often, the use of external prompts (or suggestions) are utilized to add entropy to the performance [147]. This motivates actors to justify information within the context of the current scene [288].

Improvised scenes can be summarized in three stages: platform, tilt, finding a new normal [147]. The *platform* of the scene defines what is normal in the universe (i.e. who, what, when, where). The *tilt* provides flavour.¹ A tilt re-frames the scene with a different context. It is what makes this particular performance unique from others with a similar platform. Finally, *finding a new normal* is how the scene justifies the tilt towards resolution. These three stages enable investigation of the ability of the *dAIrector* to generate cohesive plots and develop context-rich narrative. Our work addresses the specific aspect of generating prompts for the actors on stage during an improvised theatre performance. These prompts constitute the beats of the story in form of a platform, as well as tilts for the actors. It is then up to the actors to act out the scene to find a new normal.

¹<http://thewayofimprovisation.com/glossary.php>

6.3 Approach

We present an improvised narrative co-creation system called *dAIrector* which acts as an automated director to collaborate with humans storytellers for live improvisational performances and writing assistance. The generated stories are represented as linked clauses taken from William Wallace Cook’s “Plotto: The Master Book of All Plots” [59] augmented with related information from TV Tropes²—a wiki-style database that contains narrative tropes occurring in a wide range of different narratives. Human artists can rapidly link the provided prompts to collaboratively evolve a narrative through dialogue and actions. In this way, the *dAIrector* augments human creativity. We discuss the challenges of evaluating a tool that, by design, provides ambiguous guidance. We conclude by presenting several directions for future research.

6.3.1 Plotto

Our work builds on the narrative development book “Plotto: The Master Book of All Plots” by William Wallace Cook [59], which contains a large variety of plots. What makes it suitable for a computational application is the graph structure. Rather than enumerating plots, Cook split them into fragments with instructions on how to combine them.

The plot fragments constitute nodes and edges between them describe which fragments can be connected to obtain a story. Edges can have labels, which contain instructions for changing character symbols in subsequent plot fragments (e.g. changing character *A* to character *B*). Figure 6.1 shows a sub-graph from Plotto, each node a plot point and each edge a modification. Fragment 746 is defined in Plotto as: *B, who was thought by the people of her community to have supernatural powers, is discovered to have been insane - a condition caused by a great sorrow.* This fragment can be followed by either fragment 1441a: *A seeks to discover the secret of Life,* or fragment 1373: *A sells his shadow for an inexhaustible purse.* Both of these fragments make no mention of *B*, who is the main character of fragment 746. Therefore, the

²<http://tvtropes.org/>

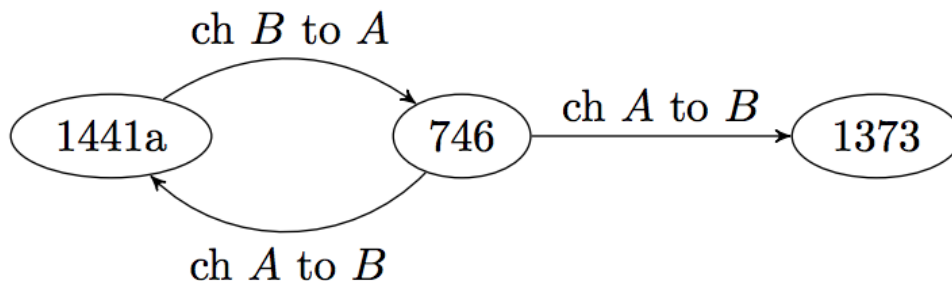


Figure 6.1: A sub-graph from Plotto. Nodes are plot fragments with corresponding numbers from Plotto, edges are connections between plot points and edge labels correspond to instructions for character modifications in subsequent fragments.

modifying edge instructs us to change A to B , ensuring consistency of the characters used. These three nodes represent just a small fraction of the entire 3000 nodes contained in Plotto. We automatically parse the nodes and edges into a JSON-based representation of the graph. Thus, generating a story is done by performing a walk through the graph starting at any random node.

Eger *et al.* (2015) used a similar method to build a plot generator utilizing the plot fragments [76]. Since the plot fragments in Plotto are abstract descriptions of plot points, ambiguous, and contain symbolic names for the characters, the generated plots are less than suitable for presentation to an audience. The ambiguity and openness make them ideal for interpretation in improvised theatre. These plot points represent platforms for scenes additional related details are needed to tilt the scene.

6.3.2 TV Tropes

TV Tropes is a wiki-style website that contains narrative tropes, i.e. patterns or situations that occur across a variety of different narratives. As a wiki, tropes often contain references to other, related tropes. Of particular interest for our work are TV Tropes’s *plot tropes* which describe high level plots abstractions. While related to those plots in Plotto, this TV Tropes graph contains unlinked semantically related, complimentary information.

The story beat is an identifiable moment of change in a narrative [209],

[220], either in the form of a new platform or as a tilt. By connecting the Plotto graph plot points as platforms and the TV Tropes plot tropes as tilts, the *dAIrector* creates complete abstract narrative descriptions.

As described above, story generation using Plotto can be thought of as a walk through the graph of plot fragments. For a performance, *dAIrector* starts at a random node in the graph by presenting it to the actors on stage. We call the plot fragment presented to the actors by the system the *platform*. The actors can prompt the system for 1) the next platform in form of a new plot fragment or 2) for a tilt to refine the current scene, and the system will use the platform to determine which plot fragment or tilt to present next.

6.3.3 Paragraph Vectors

Our system utilizes paragraph vectors to provide information dependent on the current platform. Paragraph Vector is an unsupervised machine learning method to represent variable-length input text in a dense, fixed-length feature vector [166]. Paragraph vectors overcome limitations of ignoring word order and semantics of naive bag-of-word methods. To train the paragraph vector model, we use the full text of the plot fragments, as well as the descriptions of the tropes. We follow the training as described by Le and Mikolov (2014) and use the Doc2Vec³ method from Gensim⁴. Training parameters are as follows:

- Dimensionality of feature vector: 410
- Initial learning rate: 0.03
- Maximum distance between the current and predicted word within a sentence (window): 4
- Ignore all words with fewer than two occurrences
- Negative sampling is used with 4 noise words

All other parameters are defaults as defined in Gensim.

During a performance, the current platform is used as to find the next platform or a tilt. Alternatively, instead of using the current plot fragment, actors may also provide a custom prompt to the system to steer the plot in

³<https://radimrehurek.com/gensim/models/doc2vec.html>

⁴<https://pypi.org/project/gensim/>

a certain direction. When queried, the *dAIrector* returns 5 tilt options with minimum cosine distances in vector space from the entire space of candidate options [46], [228]. For vectors \mathbf{A} and \mathbf{B} the cosine distance, or angle distance between two vectors, is defined as $d(\mathbf{A}, \mathbf{B}) = 1 - \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|}$.

As noted above, we use the full text of the plot fragments, as well as the descriptions of the tropes to train the paragraph vector model. However, since these descriptions are typically several paragraphs long, we only communicate the trope names to the actors. While we perform all comparisons against the full textual description of the tropes, this description is never displayed to the actors. This allows us to keep the instructions from the *dAIrector* concise open ended while providing additional, related information [288].

6.3.4 Plot Tree Generation

Rather than generating a single plot from the Plotto graph we generate a tree, with one start node chosen randomly and all successors as children. Each of these children has its successors as children up to a configurable depth. By default, the platform used to determine the next scene is the current plot fragment, but actors may prompt the system for a plot fragment that aligns more closely with their interpretation of a scene or details which arose from the scene improvisation. A performance of such a plot tree starts at the root node, and proceeds down the tree, where child nodes are chosen depending on the prompts given by the actors.

For example, the plot fragment *Carl's friends, Doug and Fred, believe that Lisa, whom Carl is about to marry, is a woman of immoral character*, has two successors: *Carl seeks to free himself from certain meddlesome influences* and *Lisa, harassed by gossip that reflects on her integrity, seeks deliverance from false suspicion*. Depending on which aspect of the original plot the actors decide to focus on, the interference of Doug and Fred in Carl's affairs or the rumours that Lisa is of immoral characters, one or the other successor is better suited. Note that neither successor is completely unsuitable in any case, meaning that the platform can provide guidance for the system, but that guidance does not necessarily have to be followed.

6.3.5 Contextual Tilts

In addition to simply traversing the plot tree, the system can also provide *tilts* in the form of plot tropes obtained from TV Tropes. In this case, the platform is used to find a selection of tropes that fit best with the current plot fragment. Tilts provide additional information. Thus, rather than returning the single trope is closest to the platform, our system computes the 5 most semantically similar tropes and then returns a random sample from the related set.

Prior to sampling, plot fragments are filtered and excluded if they only provide redundant information. For example, if the platform is *Albert, an inefficient, futile sort of person, comes to believe that he is the reincarnation of Nicola Tesla*, the best fitting trope according to our system is *Reincarnation*, which does not provide any additional information. However, the trope that has the second lowest distance is *Deal With The Devil*, which provides additional guidance for the actors. In a performance, the actors can utilize this, for example, to narrow down how Albert came to his belief, by making a deal with the devil. This provide the scene additional directions to explore, but it is up to the actors to decide when they would prefer a tilt for additional guidance and when they want to follow their own impulse for where the scene should go.

To eliminate redundant tropes, we compute the word intersection of words with more than 3 letters (to exclude articles, pronouns, etc.) with the platform and discard any tropes for which this intersection is non-zero. In the example above, because the word *reincarnation* is also part of the platform this particular tilt would not be provided by the system.

6.3.6 Stage Presence

For a performance, the system provides output in the form of platform beats and tilts, according to prompts given by the actors. The pseudo-code of this process is as follows:

```

# Get a starting platform
platform = root(plot_tree)
# Display the platform
present(platform)
while platform:
    # Get input from the human
    request, prompt = get_input()
    # Context is platform or user input
    context = platform
    if prompt:
        context = prompt
    # Return the platform or tilt
    if request == "platform":
        platform = best_match(context, children(platform))
        present(platform)
    if request == "tilt":
        tilt = random(best_n_match(context, tvtropes, n))
        present(tilt)

```

Code Sample 6.1: Pseudo-code for dAIrector plot generation.

When the actors request either a platform beat or a tilt, the system uses the paragraph vector model to find the best match among all candidates for the given prompt, which defaults to the last presented platform beat. For the next platform the candidates are the Plotto plot fragments that are children in the provided plot tree, while for tilts the candidates are all TVTropes plot tropes.

The basic mode of interaction with our system is through a console-based application. This application will present the plot fragments in order, and can be prompted for the next platform or a tilt. For a live performance, this mode is less convenient and therefore we also provide the capabilities for speech input and output, realized through speech-to-text using *pocketsphinx* [130] and text-to-speech using the built-in `say` operating system command. Using this interaction mode, the system reads plot fragments and tilts out loud, and the next platform beat or tilt can be obtained by the actors saying the corresponding keywords.

Plot fragments, as contained in Plotto refer to characters in the story using codes, including *A* for the main male character, *B* for the main female character, but also very specific codes such as *AUX* for a fictitious aunt. As part of presenting the plot fragments to the actors and audience, our system replaces these codes with consistent character names. This is controlled via a

configuration file, with default names provided by the system. For clarity, we present all plot fragments in this paper with names replaced. Note that at this point we do not change pronouns if character symbols are replaced since that would require automatic identification of the referent of each pronoun, which is outside the scope of this work.

6.4 Evaluation

Given the unique environment of an improv theatre performance, evaluating the quality of output is challenging; there is often no catastrophically “wrong” output [212]. That said, given the platform some tilts will require significantly more justification to produce a satisfying narrative [288]. For our work, the main challenge lies with evaluating the platforms and tilts of story fragments.

Having humans annotate multiple story fragments with the best fitting trope is a challenging multi-class classification problem. For example, the plot fragment *Joe, becoming aware of an old prophecy, unconsciously seeks to become like the exalted protagonist of the prophecy* could be seen as having any of the tropes *Prophecy Twist*, *Self Fulfilling Prophecy*, *The Unchosen One*, or *Because Destiny Says So* (among others) as the “correct” fit. For this task, we report how well our system reproduces human-assigned tilts on our test set.

6.4.1 Evaluation of Tilts

To test the key functionality of *dAIrector*, that of selecting a best-fitting tilt given a plot fragment, we design a simple task. First, we generated a dataset of 100 clean, labelled tilt – plot fragment pairs. We split this dataset into training and testing sets. For evaluation, we sample a random plot fragment from the test set and the task for the system is to correctly predict the associated tilt. Given a plot fragment our model returns the 5 candidate tilts with the minimum cosine distance. We evaluate the system based on top-5 accuracy. Clearly stated for explicitness, given a plot fragment, how likely is it that the associated tilt is in the 5 results returned by the system? This number is reported over the entire 20 examples in the test set.

One insight gained from this approach was that while the tropes produced by the system are usually related to the text fragment, there are several tropes that are vague and apply to many scenarios, while others, which are often more closely related to the story fragment at hand are more specific. For example, in our test set, the trope “Much Ado about Nothing”, which is a generic trope about love, applies to a wide variety of plot fragments. The trope that was selected the second most often for our test set was, “Road Trip”, which applies to a wide variety of travel-related scenarios.

The *dAIrectory* returns a random sample from the five most closely related tilts to a given plot point. It is therefore also reasonable to use the top-5 error as a measure of quality rather than the top-1 error. Top-N error is a common error metric for classification tasks and measures how often the target class does not show up in the top-N classes assigned to a test example.

Even so, the top-5 error on our test set is 40%, while the top-1 error rate is 66%. While high, the trope annotation task resulted in many arbitrary choices by human annotators. Most likely this is due to there not being a clear best trope, and human annotators being overwhelmed by the number of possible, subtly different, tropes.

Our plot trope set contained the 700 tropes tagged *plot tropes* from the total set of 4300 tropes on TVTropes.⁵ For example, the story fragment *Alfred is thrown into prison through false evidence in a political conspiracy* was assigned the trope *Get Into Jail Free* by a human annotator, but our system returned *Clear Their Name*, *Mystery Literature*, *No Mere Windmill*, *Lipstick Mark*, *Prison Riot* as the top five tropes, all of which could also be deemed applicable.

Note the difference in specificity between *Mystery Literature* and *Lipstick Mark*, where the latter provides a lot more detail to the actors of how to proceed with the scene. At present, our system treats all tropes as equally applicable, but, as noted above, some tropes are more general and thus related to the story being presented while actually adding less detail than others. While very specific, the definition of the trope *Get Into Jail Free* actually refers

⁵<https://tvtropes.org/pmwiki/pmwiki.php/Main/Plots>

to a character that wants to get arrested intentionally, which is arguably less fitting with the given sentence, but demonstrates the challenges faced by the human annotators.

As a way to quantify this discrepancy we used information contained within the TV Tropes graph. Tropes linked from one another if they shared some commonality. We used these links to calculate a distance between tilts our system generated and the humans annotated, equal to the number of links between two tropes.

For example, *Get Into Jail Free* links to *Can't Get in Trouble for ' ,* which links to *FrameUp* as a reason for the arrest. That trope links to *Clear Their Name* as a way to resolve the situation, resulting in a distance of 3 between *Get Into Jail Free* and *Clear Their Name*.

Over the entire TV Tropes set, the median distance between two tropes is 3 (mean: 3.1, stddev: 0.6). This is not unexpected, since tropes often refer to “supertropes”, which refer to other “supertropes”, from which the target trope can be reached. However, the median distance for tropes given as tilts by our system, excluding those that exactly matched the human annotation was 2 (mean: 2.5, stddev: 0.7), which is typically a connection via the “supertrope” common to the two tropes.

6.4.2 Sample Stories

To better illustrate the output from our system, we present sample stories produced by the system along with tilts the actors might be given. These stories demonstrate how the produced outputs are coherent stories. The coherency is a product of the structure employed by Plotto. Additionally, we show examples of stories to illustrate that the tilts the system selects can be used to refine, provide background information, or drive the story in different directions. To demonstrate our system, we set the maximum length of each story to 5. Shorter stories tend not to have enough happening in them to qualify them as stories, while longer stories, owing to the structure of the book, start to meander to different, somewhat unrelated story lines.

The first example is a well structured narrative with a beginning, middle,

and end, consisting of the following platform beats:

1. Lana, a person influenced by obligation, falls into misfortune through mistaken judgement.
2. Lana, in order to be revenged upon her enemy, Mr. Kyle, manufactures an infernal machine, BLOB.
3. Lana, influenced by a compelling idea of responsibility, finds it necessary to protect her friend, Tynan, from a secret danger.
4. Lana, suspected of treachery by her friend, Tynan, in a daring rescue, saves the property and perhaps the life of Tynan, and proves her faithfulness by a revelation of the danger to which Tynan, unknown to himself, was exposed.
5. Lana seeks to correct a character weakness in her friend, Tynan.
6. Lana achieves success and happiness in a hard undertaking.

In each scene, the platform is clear and evident. In addition to the platform of the scene and the dialogue from the improvisors, the actors might desire a plot device to instigate or inspire the action. For instance, the “secret danger” referenced in Scene 3 is vague, and the actors might ask the system for a tilt. One such applicable tilt returned by our system, *It Belongs in a Museum*, provides context to further refine the “secret danger”.

Plot fragments present in Plotto rarely mention time passing, and it is often up to the actors to explain jumps in time. Consider this example of a sequence of platform beats from our system:

1. Alfred, in order to restore to Beatrice, without a confession of culpability, wealth of which he has secretly defrauded her, marries her.
2. Alfred seeks to escape difficulties, restore property and be free of an unloved wife, Beatrice, all by secret enterprise.
3. Alfred leaves his coat on a cliff at the seaside, drops his hat in a stunted tree below the brink, and vanishes from

the scenes that know him.

4. Alfred, under a fictitious name, returns to his native place, where he had committed a youthful transgression, and, as an Unknown, seeks to discover whether his youthful escapades have been forgotten and forgiven. Also, he wishes to make reparation in an assumed character for wrong done in his true character.
5. Alfred, returning as an Unknown to his native place, discovers no one recognizes him.

Alfred disappears in scene 3, and then reappears what is apparently much later in scene 4 to wrap up his transgressions from the earlier scenes. We highlight this story, because when the system was asked for a tilt on scene 3, it responded with the tilt *Tailor Made Prison*, which seemed unfitting at first. However, upon reflection the references to the coat and hat link it to the concept of a tailor. This is remarkable for two reasons: 1) it is probably not a connection that would arise immediately to a human, and 2) this can be seen as a pun that works well within the context of our domain.

6.4.3 Qualitative Evaluation by Professional Improvisor

To investigate the quality of the system we instructed a professional improviser (fluent, native English speaker, improviser with more than 10 years performance experience) to interact with the *dAIrector*. The performer was given an introduction to the system, and then explored the interaction over the course of several scenes. The performer discussed their impressions during the interaction. We summarize the interaction feedback below by including *quotes from the performer*. By analyzing expert opinion we can formulate conclusions about the effectiveness of these methods. For several points of feedback, we directly address the quotes inline.

- *There is a real fun in getting yourself into trouble and then putting your faith in the dAIrector to do something to help.* In improvisation these are described as “What Should I Do?” moments, when the improviser decided to prompt the system for the next plot point or tilt.

- *It doesn't know what I want the scene to be about or what decisions I make.* This is an area for future work focused on how the actor's dialogue and actions are incorporated as prompts for the system.
- *Justification is natural, and it is natural to make leaping assumptions to connect actions/intentions to characters as the plot generation system did not inherently make those assignments.*
- *Tilts don't over-complicate the narrative, it expands the story rather than advancing the plot, it adds flavour. The tilt is not always necessary, and making it optional is suitable in improvisation. That said, I prefer to use the system with the contextual tilts. They run the risk of throwing a curve-ball that is very difficult, but they are often the sort of thing that would be very fun to play. Tilts are a fun expansion.*
- *I prefer being able to choose when the plot points and hints come*

6.5 Discussion and Conclusion

One of the limitations of our approach is that in our graph representation of domains all nodes are created equal, even when the underlying data sets might have additional information attached to them. For example, the tropes in the TV Tropes data set actually frequently refer to “subtropes”, “supertropes” or even state “contrast with” or “opposite of” in relation to another trope. Our system often returns very broad tilts such as *Mystery Literature*, or overly specific tilts such as *Lipstick Mark*, without any means to control which one to get. However, utilizing the structural information contained within the data set could lead to tilts that are better suited for any application. Extracting this structural information is non-trivial as it is not structured meta-data. Additionally, while TV Tropes is a useful resource, it is a large dataset that suffers from common dataset quality and inconsistency issues. One way to address these limitations would be to use a subset of tropes that are particularly narrative building. This would require filtering based on a heuristic (learned or pre-defined) which can classify and rank tropes based on narrative building qualities. Some of these qualities could be information about the universe

gained through introduction of the trope, or dynamic shifts between characters

We used Plotto and TV Tropes as our data sources because they cover a wide range of different narratives. It would also be possible to expand the *dAIrector* to use more specialized databases such as DramaBank [78]. By designing a structured graph of information, textual plots of TV show episodes could constitute an interesting data source as well. This could allow for plots which extend over seasons, character arcs, individual episode, or scenes within an episode.

Treating the plot fragments as a graph allows us to use the story-generating walk for other data sources which can be represented as graphs. We are also considering a possible application outside of narratives: the directed exploration of large graphs, such as Wikipedia for knowledge synthesis. In this scenario we could target users browsing a certain topic, starting an article of interest. The user could then request linked articles, constrained to articles semantically related to a query from a different data source.

This chapter provides a case-study in structured plot generation by fusing knowledge in multiple graphs. Structured story generation is an ongoing field of research [81], [355], [358]. Human writers often write a coarse outline, and then focus on details in each section [312]. By modelling sequential story generation problems at multiple levels of abstraction (i.e. words versus narrative plot points), hierarchical models can specialize, then combine to generate complete stories [80]. The next chapter discusses additional future directions for human-machine narrative generation.

Chapter 7

Discussion

Technology made it easy for us to stay in touch while keeping a distance, 'til we just stayed distant and never touched. Now all we do is text too much.

– Sage Francis, 2010

Improvisational theatre performances connecting humans and intelligent, interactive machines serve as an exciting, illuminating, and challenging domain to study conversational dialogue systems. In this chapter, we discuss the impact of the work in this thesis and how the fields of interactive machine learning (IML), conversational dialogue systems, and improvisational theatre have evolved. With rapid advances in machine learning technology, and an open and sharing scientific community, there has been significant growth at the intersection of these fields. This chapter presents analysis of how artificial improvisation has been received by performers, audiences, reviewers, and the public media. As artificial improvisation is a novel art-form, it is important to reflect on how it is consumed and understood by the public. Finally, we discuss several research directions which will progress the intersectional field forward in the future. These are less predictions of what will happen next, rather prescriptions of work that could be done in an effort to address some of the toughest ongoing challenges.

In this thesis, we present the grand challenge problem of artificial improvisation, or using interactive machine learning in the domain of improvisational theatre alongside a conversational dialogue system. This is the ideal domain for studying some of the most difficult problems in human-machine

interactions. It provides for real-time interaction between humans and machines, with observers in the safe, controlled environments of the theatre. We then presented two artificial improvisors. First, we presented Pyggy, a rule-based chatbot which used human interaction training logs to respond to user interaction and used a projection-based embodiment. Then, we presented A.L.Ex., the Artificial Language Experiment, which aimed to address several of the challenges encountered when performing alongside Pyggy. From there, we presented the show/experiment Improbotics exploring how audiences react to humans delivering lines from A.L.Ex. or other humans. Finally, we presented methods for improving several challenges in engaging interactive narrative generation. Shaping the Narrative Arc aims to build an underlying belief structure of the scene by modelling the co-constructed universe, and dAIrector aims to merge knowledge from related graphs to form coherent plot structures to frame improvisation. These works represent a collective research push to progress improvisation toward natural human performance alongside these artificial improvisors and to progress the underlying machine learning technology by incorporating human interaction.

7.1 Human-Machine Improvisation

How can the robot and the human interact physically on stage? Background on this intersection is covered in Section 2.3.2. Since the inception of this thesis work there has been an evolution of the research field and the art-form.

We have invited more machine learning systems into our lives over the last half decade than in all previous years. Devices such as Amazon Alexa or Google Home are common, these systems offer conversational interaction with a simple conversational “Hey, let’s have a chat!” By inviting these devices into the home, the car, and the office, we are enabled to perform a wide variety of tasks faster and easier than before. The ubiquity allows companies to build conversational datasets at rates previously unthinkable. This interaction has been fueled, in large part, thanks to the advances in labs around the world and the collaboration of hardware and software providers to bring costs down

to consumer levels.

There are many games, applications, and interactive experiences available on these devices which allow for conversational interaction which includes chit-chat dialogue (see Section B.7). This also means that the expectations of users have increased. Users expect engaging, delightful conversations to be available over a near-endless set of topics and dynamics. This work has in large part been supported by conversational researchers adding to the field of conversational interaction. That said, the influence of improvisation in modelling these dialogue improvements can not be understated. It is not only the work in this thesis that uses rules from improvisation to model improved open conversation [176], [206], [207].

The art-form of modern performative improvisational theatre is still young, and has itself, experienced significant growth over the years which the work in this thesis represents. Online streaming venues such as Twitch and YouTube Live have brought broadcast live improvisation, comedy, and theatre to larger audiences and have started to integrate user interaction in the performances. Digital puppetry with avatar representations of humans, or v-tubers, are becoming increasingly popular. These innovations are not the only way that technology has been integrated into improvisation. Shows in Canada and Germany have started to involve wireless networks for users to connect to before, during, and after the show to collaborate and share information.¹ Additionally, use of generative models has significantly influenced musical improvisation performance [125]. Music generation models from Google's Magenta were used in a June 2018 performance of Improbatics (see Chapter 4) in Edmonton, Canada. Similarly, these models have been used in musical writing and performance by Pablo Castro in collaboration with David Usher [45].² Similarly, generative models continue to be used extensively by the comedy collective Botnik in live comedy and musical performance.³

¹<http://hipbang.ca/>

²<https://goo.gl/QmYFQ4>

³<https://botnik.org/>

7.2 Critical Reception

Artificial improvisation is an art-form and science experiment that the general public are eager and willing to engage with. The evolution of these technologies from experimental to production ready has allowed for rapid integration by artists and directors. That said, this integration of technology has both lowlights and highlights. By consistently and deliberately reflecting on reviews and commentary on these performances we aim to grow and evolve the technology powering the performance and the show itself. Thus, we present reviews and critical reception of the technology and show below with humility and much appreciation for the hard work and expertise of the writers and reviewers.

7.2.1 From Audience and Improvisors

One of the best means of evaluating a creative collaborator, is through qualitative responses from performers and audiences. Toward that end, the response from improvisors has been strong and supportive. Several comment themes have been synthesized in Chapter 4 from performers who have worked alongside the artificial improvisor. Briefly, these comments fall into several major themes:

1. Artificial improvisors cannot tell complete stories.
2. Improvising with the artificial improvisors is more work.
3. Performing with these systems forces you to be a better improvisor.
4. It is like performing with a novice improvisor.
5. There is fun in getting yourself into trouble, and then putting your faith in the system to do something to help.

In addition to these comments, the system has been studied by professional improvisor Rachel Rosenthal, who reflected on whether or not these systems threaten her likelihood and how they inspire other artists.⁴ In her article, “Will robots replace improv comedians?” she presents a view which is initially very skeptical but eventually supportive after exploring several questions compar-

⁴<https://goo.gl/V6CsR4>

ing human and machine improvisors. She emphasizes that while the robot can not emote, a critical skill for an improvisor, it does draw empathy from human audiences. This reinforces our choices of humanoid embodiments which capture those feelings of connection effectively. She concludes by reiterating her fascination with the system and desire to play alongside it in the future: “I definitely at least want to try improvising with A.L.Ex at some point. I mean I have to, right? (Of course, this is how it all starts...)”. These comments are reassuring to us, we have sparked the curiosity of mind from expert performers at the top of the art-form. Artificial improvisation is compelling to these performers for how it challenges possibilities and inspires expert storytellers to tell new and interesting stories.

7.2.2 From Reviewers

Performances of the artificial improvisation work described in this thesis have been reviewed by theatre reviewers over multiple shows. These reflections provide an honest assessment of the art-form’s impact on society in relation to other contemporary performances. As developers, we have embraced the feedback and learned from these reactions to evolve and improve the show. One point of interest is tracking how the reaction to the show as evolved over the years, likely due to the shows evolution and the increase of artificial intelligence-influences in popular culture. Several excerpts are included below for reference.

1. Edmonton Fringe review of “Artificial Intelligence Improvisation”, by Josh Marcellin, **18 August 2018**. “It’s almost as much fun hearing the researcher-performers explain the science behind A.L.Ex with undiluted glee as it is watching the AI come up with absurdly, often dark, non-sequiturs for the humans to riff on. Boundary-pushing science plus talented meat-bags equals a stand-out improv Fringe experience.”
2. Broadway Baby’s review of “Artificial Intelligence Improvisation”, by Alex McCord, **5 June 2018**. “[They] create a unique comedy experience with their machine companion. Veering wildly between moments of cringe-inducing strangeness and total hilarity, it is a show that must

be seen to be understood. It sounds exceptionally strange. It is exceptionally strange... The genius of the robot is in the way it addresses the flaws of improv comedies and gives them a reason to go to weird places and say weird things. One brilliant set piece is when they put the robot avatar aside and bring up an audience member and make them say whatever dialogue the program generates... [T]he idea here is magnificent. There are so many laughs to be found with this formula and, with improvements both from our scientists and A.L.Ex, this could be revolutionary.”

3. “Artificial Intelligence Improvisation at the Etcetera Theatre”, View from the gods, **31 July 2017**. “The most fascinating aspect of Artificial Intelligence Improvisation is that despite the characters being make believe, the technology behind this show is genuine... Gimmicky? Maybe a little. However it’s also very cool and a lot of fun to observe... they’re very well received by an enthusiastic audience ... Slightly ramshackle and prone to system error, but thoroughly enjoyable.”
4. “Artificial Intelligence Improvisation”, by Kate Pettigrew for London Pub Theatres, **1 August 2017**. “It is always interesting when something different is tackled in theatre and Artificial Intelligence Improvisation does just that.”

7.2.3 From News Media

Popular media’s coverage of artificial improvisation is a testament to both its novelty and impact on society. To date, there have been plenty of media covering different angles of the work from news outlets including Bloomberg, the Wall Street Journal, and the New York Times. A complete list is included in Appendix G. The articles fall into four major thematic groups:

1. Introductions to the technology
2. Putting AI to the test with humour or comedy
3. Teaching comedy to machines
4. A robot tricking people

Introductions to the technology are pieces which are related to the science

behind the system. These articles introduce machine learning and often profile specific underlying technologies. These articles serve as vehicles for scientific communication; a means by which we as scientists and researchers can explain our pursuits in a manner that is not mired in academic language. These articles aim to serve the scientist and the general public audience.

Several articles take the angle putting artificial intelligence to the test. These articles often make an assumption that machines will compete with humans on specific tasks. One task is improvised theatre, and these articles explore the notion of the robots ability to perform that task. If the robot can start to make jokes, what could it do next? These articles serve to present the research in a competitive fashion. These articles serve a large fear-based narrative of humans competing with machines, as discussed in Section 2.3.2.

In the collection of news media, some of the press has taken the angle of teaching comedy to machines. These pieces often lay out machine learning in simple terms and then present comedy, humour, or wit as another technique or skill that can be taught to machines. While these articles may serve the underlying science they often include details on pattern recognition or statistical inference. They do not make connections that the learning they are discussing is often finding patterns in input data and generating a corresponding, high probability output. While this would be a more difficult story to tell, it would help to bridge the gap between human and machine learning.

Finally, there is press which hones in on a specific part of our work, that of deception. These pieces often use engaging language to capture attention. They focus on how the machine is doing its best to trick human audiences, or attempt to fool observers. Many of the pieces start from inspiration of the Imitation game, and then spiral toward deceit of human trust. We feel as though this media does not serve the science, nor does it serve the general public. We can imagine that these sorts of stories are used by news media to generate interest. Compelling stories of human's tricked by machines capture attention, but they are only thematically related to this work.

7.3 Future Directions

Artificial improvisation starts with dialogue, but that is not where it ends. Mastering the interactive capabilities of artificial improvisation requires a closeness, similar to the relationship between a musician and their instrument [331]. Intimate, dialogue-driven, relational closeness with these systems can create augmentative solo performers. These performers could simultaneously affect all facets of stage productions. The artificial improviser could send control signals to lighting, sound, and video projection systems. Projection was a first step with Pyggy, and we have experimented with incorporating images retrieved based on queries related to topics from the scene [352]. This represents a significant engineering effort that will be possible to produce in the near future. There is still much ground to be covered by using simple language models and searching for nearest neighbours in a latent sentence embedding space. Our open source release *jann*⁵, makes it easy for users to build and deploy a dialogue system over using Google’s DialogFlow⁶ to allow rapid integration over Facebook, Twitter, Slack, and Google Home. We predict that these tools will allow many more interactive dialogue applications to be deployed for large populations of users.

As discussed in Section 4.6.4, we expect that timing (the number one rule of comedy) to be improved by future iterations of these systems. Furthermore, there are several interesting future studies to explore extending on the work presented in this thesis. Below we present several studies of interest which will support the continued growth of the field.

7.3.1 Improv in Any Language

With the increasing ease of travel between nations, and improvements in remote video conference abilities, connections are made between performers and audiences of different languages. Theatre is one of the most important tools we have for sharing experiences and building cross-cultural understanding.

⁵<https://github.com/korymath/jann>

⁶<https://dialogflow.com/>

Improvised theatre is positioned to connect culture given its accessibility, universality, and low barriers to entry; improv requires no scenography, costumes, lights, or sound equipment. Improv can be performed by individuals with little or no previous experience. Although the quality of performance may vary given performers of differing skill levels, basic improvisation can be easily understood and internalized. In a short manner of time, individuals from diverse cultures can perform scenes together with diverse characters, relationships, settings, motivations, and even subtext. This can be done even when the performers do not share a common spoken language.

The art of improvisation is derived from the connections between performative layers. Both between the performers, and between the performers and the audience. Improv embraces the audience to create collaboratively together. In this way improvisation is a democratic narrative. The potential impacts of improvised theatre between performers and audiences of different cultures and varying experience levels is significant. Most international improvisational collaboration is English based, but many regional festivals take place in the language native to the host country. This excludes audiences without knowledge of the performance language, and limits the contributions by performers who do not speak the language. Without advanced translation technology, improvisation can lose important voices due to language limitations.

How can we create conditions so that improvisors from different cultures can improvise together in their own language? How can audiences understand performers using diverse languages? The techniques used for artificial improvisation can be translated, literally and figuratively, to address these questions. By connecting all performers with microphones and headphones, audio signals can be routed through computer systems. Speech recognition can be used to recognize the language of the spoken line. If the text is to be translated, it can be sent to a translation service (e.g. Google Translate API). The translated text can then be synthesized and the audio can be sent to the speakers in the theatre, to the headphones of select performers, or to select audience members.

Timing is a critical issue for this configuration [11], [173], and sub-second latency is important for such a system. This could require predictive text mod-

elling and nearest neighbour approaches for immediate response when necessary, in addition to neural language modelling for diverse response generation. It is important to integrate these delays into the framing of such a performance (similar to how the Turing test framing enabled studies in Chapter 4). Translation accuracy is another major challenge in this pursuit. This accuracy could likely be improved through contextual modelling by incorporating topic and scene information [201], [215], as well as cultural information [35].

7.3.2 Starting and Finishing Conversations

To extend the work presented in this thesis, one likely next step will be toward initiating and concluding conversation. Having an agent that can maintain an ongoing conversation is a significant step, but allowing the agent to start and finish conversations will require significant research progress. Through human studies, researchers might collect natural human dialogues [100] over a predefined set of topics and inputs [305]. From these natural conversations, models may be learned specifically for the initiation of a conversation. Similarly, dialogues often end by recapitulating what was discussed, summarizing topical contributions, and then providing concluding remarks. Currently, most interactions with these dialogue agents conclude when the user decides, often once the system generates one-too-many uninteresting responses [180]. Through continued interaction with humans, these systems can build growing datasets. They can start to learn online and adapt in real-time by following these two research directions.

7.3.3 Tuning Language Models with Rewards

Current generative text models are limited to learning the statistics of the underlying training dataset. Fine tuning of trained generative deep neural networks using self-play in the domain of open dialogue conversation may be possible. This will allow users to define a reward scheme to personalize and tune models following large scale training. This tuning will allow for creative artistic expression through defining interesting reward schemes—such as sarcasm, style, meter, or rhyme scheme.

Current deep neural networks strictly model the underlying statistics of a training dataset. Once trained these models are limited in their adaptability. In image and video domains, it is common to initialize the weights of a neural network with those trained for a separate task and then fine tune the final layer with new task-specific labelled data. This pretraining and transfer is less common in text domains where corpus and task may vary drastically from domain to domain. There is likely to be a breakthrough work extending pretraining for text. As Ruder (2018) said, “NLP’s ImageNet moment has arrived”.⁷ While working fully pretrained generative language models is less common often pretrained components are used in the training pipeline [46], [70]. There is also work which proposes tuning trained neural network models using reinforcement learning. RL has been used to tune trained models for music, molecule sequence, and robotic movement sequence generation [54], [138], [140], [179]. These models harness contextual information contained within human-designed reward schemes or directly from feedback from human observers.

This work specifically could build on the trained models of the Artificial Language Experiment. A.L.Ex. is trained on a corpus of movie dialogue data [340], and is used as the dialogue manager in a variety of improvisational performance tasks as discussed previously in this thesis. A key principle in improvisation is positivity in interlocutor (or speaker) statements, this concept is summed up as “Yes, and...” in improvisational theory. This positivity in statement can be measured through sentiment scoring. VADER is a sentiment scoring model which can be trained on labelled data and then provide a measure of a statements sentiment [95]. Preliminary work towards this sentiment modelling is presented in Chapter 4.

We expect that post-training tuning will allow for a designer to build arbitrary reward schemes to specialize trained generative neural networks. Specifically, we expect that using the sentiment score as a tuning reward during self-play–simulated self-dialogues–will improve a models ability to generate novel positive sentiment statements, and thereby will improve the models ability to

⁷<http://ruder.io/nlp-imagenet/>

perform improvisational theatre through conversational dialog. To evaluate the performance of reward tuning, researchers will need to compare generated conversations average sentiment scores between tuned and un-tuned networks across the experimental conditions. They will also need to compare downstream tasks affected by sentiment (e.g. observer delight and participant engagement). Additionally, optimizations might aim to maximize information shared between interlocutors.

7.3.4 Adapting Language Models in Real-Time

It is desirable for trained and tuned deep neural networks to adapt to the emotional state of the speaker it is interacting with. Adaptation could occur over a variety of features: including grammar, vocabulary, topic, emotional salience, sentiment, and tone. Current deep neural networks can be trained with a large dataset, but often user specific variation or personalization is difficult to achieve due to limited numbers of samples relative to the size of the training set.

I propose that the adaptability module from the previous experiment could be extended with a variety of linguistic metrics to nudge the model towards adaptability in real-time. The chameleon model, named after the ‘chameleon affect’, would adapt speaker style more readily than a method lacking adaptability capacity [50].

Through the use of RL techniques from the previous experiment and building on current speaker modelling work [364], A.L.Ex. might be adapted in real-time to align responses to an updating model of the other speaker in the loop. The system could match semantic variables such as tone and emotion and maintain a separation in syntactic content. This would allow for a system which can continue an interesting conversation, constantly mirroring shifts in speaking style, while avoiding the pitfall of simply repeating what the other speaker said.

To evaluate the performance of these adaptive models, it is likely that a computational definition of what speaker modelling, mirroring, allowance and convergence looks will be adopted [65]. For instance, a set of features

in a given piece of input text are defined, and the features for the generated response are then compared with the input text [64]. This work has proposed a large number of features over which to compare input/output pairs including: sentence length, noun count, sentiment, language complexity. With these features clearly defined and computable, a model learns online to minimize the difference between the generated and input responses pairs.

7.3.5 Evaluating Conversational Dialogue

A good dialogue system should be capable of sustaining a coherent conversation with a human by staying on topic and by following a train of thoughts [338]. Former measures of conversational dialogue evaluation often measure the capacity of a model to be consistent with a dataset (Section 2.2.7). Ongoing research aims to build comprehensive conversational dialogue evaluation metrics across languages [328]. For proper evaluation analysis should compare how a model can provide semantically linked responses while balancing how much it is repeating examples similar to the input data. That is, there is a desire to measure a models capacity to generate unique, diverse, meaningful responses which are close to the expected output, but perhaps not directly aligned. In a 2018 paper, we proposed semantic similar and response echo index to address this exact evaluation [73]. Semantic similarity measure the capability of a model to be consistent with the semantics of the input context and message, thereby maintaining topical consistent. To render the semantic representation of an utterance, we leverage Universal Sentence Encoder [46] wherein a sentence is projected to a fixed dimensional embedding vector. Other vector based sentence embeddings may provide complimentary information for this evaluation metric [70], [156]. Response echo index assesses how well a model is able to generate plausible responses measurably distant from the input dataset. Used together, these metrics can be used to reduce the burden on human evaluators. We compared a novel conversational language model (THRED: Topical Hierarchical Recurrent Encoder Decoder) with two strong baselines using these two metrics as well as human evaluation. Our results indicated that the THRED model is able to generate more diverse and

contextually relevant responses compared to the baselines. The results also indicated that our automated metrics correlated well with subjective human judgement. Additional details on the semantic similarity and response echo index metrics are included in the full paper [73].

7.4 Summary

In this chapter, we presented reflections on the intersection between interactive machine learning, dialogue, and improvised theatre in the context of artificial improvisation. This intersection will develop further as the technology is embraced by more performers around the world. Improbotics has produced more than 75 shows in five cities, and continues to grow as others learn to incorporate artificial improvisation. This growth is enabled by embracing critical reception and feedback from audience members, improvisors, reviewers, and news media. Reflections from audience members, improvisors, reviewers, and news media reassure us that our innovations are inspiring and challenging. Our work pushes performers to tell stories in new and novel ways, and in doing so, develop the technology and themselves. Reception from media represents an interesting arc of covering the development narrative of the technology. It started as burgeoning ideas and has evolved into fully fledged systematic performance tools. Finally, this chapter presented several directions for future research reflecting the ongoing interests of the intersectional research community. These include multi-lingual capabilities, starting and finishing conversational dialogue, tuning and adapting language models in real-time, and improving the way we evaluate generative conversation systems. These directions prescribe how incremental innovations will define the future of artificial improvisation.

Chapter 8

Conclusion

*We can only see a short distance ahead,
but we can see plenty there that needs to be done.*

– Alan Mathison Turing, 1950

The main contribution of this thesis is the development of artificial improvisation: *improvised theatre performed by humans alongside intelligent machines*. This is a novel art-form and field of scientific pursuit found at the intersection of interactive machine learning, conversational dialogue systems, and improvisational theatre. The core motivation is to understand how humans and machines can interact through dialogue during improvised theatre. How have they interacted in the past, what does their interaction currently look like, and what might it look like in the future? The material in the previous chapters attempted to answer these questions.

Chapter 2 contains background material covering the scientific fields of interactive machine learning and dialogue generation, and details underlying the art-form of improvisation. Chapter 3 presents the the first experiments in live stage human-machine improvisational theatre for interactive open-domain dialogue research. The first two artificial improvisors are presented: 1) the rule-based conversational dialogue system *Pyggy*, and *A.L.Ex.*, the Artificial Language Experiment. *A.L.Ex.* is a corpus-based neural response generation system which several key technical improvements over *Pyggy*.

Chapter 4 presents quantitative and qualitative evaluation of artificial improvisation and a theatrical Turing test performance format called *Improbotics*.

The chapter presents performer and audience evaluation of artificial improvisation using *A.L.Ex.* and systems discussed in the previous chapters.

Analysis of the experimental results from Chapter 4 identified several specific improvements for artificial improvisation. *Shaping the narrative arc* (Chapter 5) is a model able to incorporate an underlying evolving story arc in collaborative narrative generation. *dAIrector* (Chapter 6) is a novel method for synthesizing a plot graph with contextual information. The method can be used for automated scene direction through contextual plot generation. While these two advances can be used in support of artificial improvisation systems such as *A.L.Ex.*, they are independent contributions that do not depend on the work in the previous chapters. These augmentative models were designed to be agnostic to the baseline artificial improvisation system and may be used in conjunction with other improvements.

The contributions in these chapters are significant contributions towards improvised theatre performed by humans alongside intelligent machines. Together they represent a significant amount of scientific inquiry, engineering effort, and artistic endeavour. Given the public nature of this work, critical reception from improvisors, audience members, reviewers and news media are presented and analysed in Chapter 7. The chapter also discusses future innovations for artificial improvisation. These future directions include understanding and experimenting with response timing, starting and finishing conversations, and adapting conversation models in real-time.

This thesis presents a view of human-machine collaboration for improvised stage performance. From conceptualization, design, development, to evaluation, outcome analysis and discussion, this thesis represents a body of published research and over 75 live performances around the world. These contributions are made possible through interesting dialogues, inspired collaborations, and chasing what-if scenarios; by accepting what came before, and expanding and advancing into the future. It is our hope that this work serves to inspire artists, researchers, and scientists. We hope to uplift the art and science underlying artificial improvisation.

References

- [1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, and M. Kudlur, “Tensorflow: A system for large-scale machine learning.,” in *OSDI*, vol. 16, 2016, pp. 265–283 (cit. on p. 193).
- [2] B. AbuShawar and E. Atwell, “Usefulness, localizability, humanness, and language-benefit: Additional evaluation criteria for natural language dialogue systems,” *Int. J. of Speech Tech.*, vol. 19, no. 2, pp. 373–383, 2016 (cit. on p. 85).
- [3] P. H. Algoet and T. M. Cover, “A sandwich proof of the shannon-mcmillan-breiman theorem,” *The annals of probability*, pp. 899–909, 1988 (cit. on p. 37).
- [4] R. Altman, “Moving lips: Cinema as ventriloquism,” *Yale French Studies*, no. 60, pp. 67–79, 1980 (cit. on p. 78).
- [5] D. Ameixa, L. Coheur, P. Fialho, and P. Quaresma, “Luke, i am your father: Dealing with out-of-domain requests by using movies subtitles,” in *International Conference on Intelligent Virtual Agents*, Springer, 2014, pp. 13–21 (cit. on p. 18).
- [6] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza, “Power to the people: The role of humans in interactive machine learning,” *AI Magazine*, vol. 35, no. 4, pp. 105–120, 2014 (cit. on p. 7).
- [7] M. Arcand, J. Nantel, M. Arles-Dufour, and A. Vincent, “The impact of reading a web site’s privacy statement on perceived control over privacy and perceived trust,” *Online Information Review*, vol. 31, no. 5, pp. 661–681, 2007 (cit. on p. 12).
- [8] S. Arik, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, A. Ng, J. Raiman, and S. Sengupta, “Deep Voice: Real-time Neural Text-to-Speech,” *ArXiv e-prints*, Feb. 2017. arXiv: 1702.07825 [cs.CL] (cit. on p. 70).
- [9] N. Asghar, P. Poupart, J. Hoey, X. Jiang, and L. Mou, “Affective neural response generation,” in *European Conference on Information Retrieval*, Springer, 2018, pp. 154–166 (cit. on p. 112).

- [10] L. E. Asri, H. Schulz, S. Sharma, J. Zumer, J. Harris, E. Fine, R. Mehrotra, and K. Suleman, “Frames: A corpus for adding memory to goal-oriented dialogue systems,” *arXiv preprint arXiv:1704.00057*, 2017 (cit. on p. 50).
- [11] S. Attardo and L. Pickering, “Timing in the performance of jokes,” *Humor-International Journal of Humor Research*, vol. 24, no. 2, pp. 233–250, 2011 (cit. on pp. 94, 139).
- [12] L. Au, *Chat conversation methods traversing a provisional scaffold of meanings*, US Patent App. 11/806,261, Dec. 2007 (cit. on p. 185).
- [13] P. Auslander, “Live from cyberspace: Or, i was sitting at my computer this guy appeared he thought i was a bot,” *PAJ: A Journal of Performance and Art*, vol. 24, no. 1, pp. 16–21, 2002 (cit. on pp. 43, 77).
- [14] C. Babbage, *Passages from the Life of a Philosopher*. Cambridge University Press, 2011 (cit. on p. 34).
- [15] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” *International Conference on Learning Representations*, 2015 (cit. on pp. 30, 33).
- [16] R. E. Banchs and H. Li, “Iris: A chat-oriented dialogue system based on the vector space model,” in *Proceedings of the ACL 2012 System Demonstrations*, Association for Computational Linguistics, 2012, pp. 37–42 (cit. on pp. 18, 188).
- [17] S. Banerjee and A. Lavie, “Meteor: An automatic metric for mt evaluation with improved correlation with human judgments,” in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005, pp. 65–72 (cit. on p. 36).
- [18] L. F. Baum, “The wizard of oz. 1900,” *New York: Ballantine*, 1992 (cit. on pp. 75, 77, 183).
- [19] R. Bellman, “The theory of dynamic programming,” RAND Corp Santa Monica CA, Tech. Rep., 1954 (cit. on p. 11).
- [20] J. Benedetti, *Stanislavski: his life and art*. Methuen Drama, 1999 (cit. on p. 195).
- [21] T. Bickmore and T. Giorgino, “Health dialog systems for patients and consumers,” *Journal of Biomedical Informatics*, vol. 39, no. 5, pp. 556–571, 2006, Dialog Systems for Health Communications (cit. on p. 97).
- [22] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, New York, 2006 (cit. on p. 102).
- [23] J. Bizzocchi, “Games and narrative: An analytical framework,” *Loading-The Journal of the Canadian Games Studies Association*, vol. 1, no. 1, pp. 5–10, 2007 (cit. on p. 97).

- [24] D. M. Blei and J. D. Lafferty, “Dynamic topic models,” in *Proceedings of the 23rd international conference on Machine learning*, ACM, 2006, pp. 113–120 (cit. on p. 112).
- [25] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *JMLR*, vol. 3, no. Jan, pp. 993–1022, 2003 (cit. on pp. 58, 112, 193).
- [26] N. Block, “Psychologism and behaviorism,” *The Philosophical Review*, vol. 90, no. 1, pp. 5–43, 1981 (cit. on p. 181).
- [27] A. Boal, *Theater of the Oppressed*. Pluto Press, 2000 (cit. on p. 197).
- [28] ———, *Games for actors and non-actors*. Routledge, 2005 (cit. on p. 197).
- [29] D. G. Bobrow, R. M. Kaplan, M. Kay, D. A. Norman, H. Thompson, and T. Winograd, “Gus, a frame-driven dialog system,” *Artificial intelligence*, vol. 8, no. 2, pp. 155–173, 1977 (cit. on p. 183).
- [30] A. Bordes, Y.-L. Boureau, and J. Weston, “Learning end-to-end goal-oriented dialog,” *arXiv preprint arXiv:1605.07683*, 2016 (cit. on pp. 16, 112).
- [31] N. Boukhelifa, A. Bezerianos, and E. Lutton, “Evaluation of interactive machine learning systems,” in *Human and Machine Learning*, Springer, 2018, pp. 341–360 (cit. on p. 12).
- [32] B. V. Bowden, “Faster than thought: A symposium on digital computing machines,” 1953 (cit. on p. 42).
- [33] C. Breazeal, A. Brooks, J. Gray, M. Hancher, C. Kidd, J. McBean, D. Stiehl, and J. Strickon, “Interactive robot theatre,” in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, IEEE, vol. 4, 2003, pp. 3648–3655 (cit. on pp. 44, 45, 47, 75).
- [34] K. Brown, “The auslander test: Or, ‘of bots and humans’,” *International Journal of Performance Arts and Digital Media*, vol. 4, no. 2-3, pp. 181–188, 2008 (cit. on p. 111).
- [35] B. Buden, S. Nowotny, S. Simon, A. Bery, and M. Cronin, “Cultural translation: An introduction to the problem, and responses,” *Translation Studies*, vol. 2, no. 2, pp. 196–219, 2009 (cit. on p. 140).
- [36] J. Cahn, “Chatbot: Architecture, design, & development,” *University of Pennsylvania School of Engineering and Applied Science Department of Computer and Information Science*, 2017 (cit. on p. 188).
- [37] M. Cakmak and A. L. Thomaz, “Designing robot learners that ask good questions,” in *Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI ’12, Boston, Massachusetts, USA: ACM, 2012, pp. 17–24 (cit. on p. 176).
- [38] J. Cameron and W. Wisher, *Terminator 2: Judgment Day*. USA, 1991, vol. 2 (cit. on p. 42).

- [39] M. Campbell, A. J. Hoane Jr, and F.-h. Hsu, “Deep blue,” *Artificial intelligence*, vol. 134, no. 1-2, pp. 57–83, 2002 (cit. on p. 185).
- [40] K. Čapek, *RUR (Rossum’s universal robots)*. Penguin, 2004 (cit. on p. 42).
- [41] E. A. Cappel, A. Desai, M. Collins, and N. Michael, “Online planning for human–multi-robot interactive theatrical performance,” *Autonomous Robots*, pp. 1–16, 2018 (cit. on p. 73).
- [42] J. Cassell, Y. I. Nakano, T. W. Bickmore, C. L. Sidner, and C. Rich, “Non-verbal cues for discourse structure,” in *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ser. ACL ’01, Toulouse, France: Association for Computational Linguistics, 2001, pp. 114–123 (cit. on p. 14).
- [43] J. Cassell, Y. I. Nakano, T. W. Bickmore, C. L. Sidner, and C. Rich, “Non-verbal cues for discourse structure,” in *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, Association for Computational Linguistics, 2001, pp. 114–123 (cit. on p. 14).
- [44] C. Castellini, P. Artemiadis, M. Wininger, A. Ajoudani, M. Alimusaj, A. Bicchi, B. Caputo, W. Craelius, S. Dosen, K. Englehart, and D. Farina, “Proceedings of the first workshop on peripheral machine interfaces: Going beyond traditional surface electromyography,” *Frontiers in neurorobotics*, vol. 8, p. 22, 2014 (cit. on p. 176).
- [45] P. S. Castro, “Performing structured improvisations with pre-existing generative musical models,” in *Workshop on Machine Learning for Creativity and Design, Advances in Neural Information Processing Systems Conference*, 2018 (cit. on p. 133).
- [46] D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, and Y. Sung, “Universal sentence encoder,” *arXiv preprint arXiv:1803.11175*, 2018 (cit. on pp. 99, 121, 141, 143, 193).
- [47] V. Cerf, “Parry encounters the doctor,” Tech. Rep., 1973 (cit. on pp. 39, 182).
- [48] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning, and Games*. New York, NY, USA: Cambridge University Press, 2006 (cit. on p. 101).
- [49] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition,” in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, IEEE, 2016, pp. 4960–4964 (cit. on p. 194).
- [50] T. L. Chartrand and J. A. Bargh, “The chameleon effect: The perception–behavior link and social interaction.,” *Journal of personality and social psychology*, vol. 76, no. 6, p. 893, 1999 (cit. on pp. 39, 142).

- [51] S. F. Chen and J. Goodman, “An empirical study of smoothing techniques for language modeling,” *Computer Speech & Language*, vol. 13, no. 4, pp. 359–394, 1999 (cit. on pp. 22, 24).
- [52] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder–decoder for statistical machine translation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734 (cit. on pp. 30–32).
- [53] N. Chomsky, “Three models for the description of language,” *IRE Transactions on information theory*, vol. 2, no. 3, pp. 113–124, 1956 (cit. on p. 33).
- [54] P. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei, “Deep reinforcement learning from human preferences,” *arXiv preprint arXiv:1706.03741*, 2017 (cit. on p. 141).
- [55] E. Chu and D. Roy, “Audio-visual sentiment analysis for learning emotional arcs in movies,” in *Data Mining (ICDM), 2017 IEEE International Conference on*, IEEE, 2017, pp. 829–834 (cit. on p. 47).
- [56] K. M. Colby, F. D. Hilf, S. Weber, and H. C. Kraemer, “Turing-like indistinguishability tests for the validation of a computer simulation of paranoid processes,” *Artificial Intelligence*, vol. 3, pp. 199–221, 1972 (cit. on p. 182).
- [57] K. M. Colby, S. Weber, and F. D. Hilf, “Artificial paranoia,” *Artificial Intelligence*, vol. 2, no. 1, pp. 1–25, 1971 (cit. on p. 182).
- [58] D. Collins, “Improvisation in rhapsodic performance,” *Helios*, vol. 28, no. 1, p. 11, 2001 (cit. on p. 195).
- [59] W. W. Cook, *Plotto: a new method of plot suggestion for writers of creative fiction*. Ellis., 1928 (cit. on pp. 60, 69, 95, 113, 115, 117, 118).
- [60] J. Copeau, “Registres vi: L’école du vieux-colombier, ed,” *Claude Sicard (Paris: Gallimard, 2000)*, vol. 96, (cit. on p. 195).
- [61] K. Corti and A. Gillespie, “A truly human interface: Interacting face-to-face with someone whose words are determined by a computer program,” *Frontiers in psychology*, vol. 6, p. 634, 2015 (cit. on pp. 77, 81, 94, 183).
- [62] G. Cukor, *My fair lady*, 1964 (cit. on p. 52).
- [63] I. Damgård, M. Keller, E. Larraia, V. Pastro, P. Scholl, and N. P. Smart, “Practical covertly secure mpc for dishonest majority—or: Breaking the spdz limits,” in *European Symposium on Research in Computer Security*, Springer, 2013, pp. 1–18 (cit. on p. 177).
- [64] C. Danescu-Niculescu-Mizil, M. Gamon, and S. T. Dumais, “Mark my words! linguistic style accommodation in social media,” *CoRR*, vol. abs/1105.0673, 2011 (cit. on p. 143).

- [65] C. Danescu-Niculescu-Mizil and L. Lee, “Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs,” in *Proceedings of the 2nd Workshop on Cognitive Modeling and Computational Linguistics*, Association for Computational Linguistics, 2011, pp. 76–87 (cit. on pp. 34, 54, 142).
- [66] K. Dautenhahn and C. L. Nehaniv, *Imitation in animals and artifacts*. Cambridge, MA: MIT Press, 2002 (cit. on p. 175).
- [67] T. Davidson, D. Warmsley, M. Macy, and I. Weber, “Automated hate speech detection and the problem of offensive language,” *arXiv preprint arXiv:1703.04009*, 2017 (cit. on p. 94).
- [68] R. Der and G. Martius, *The playful machine: theoretical foundation and practical realization of self-organizing robots*. Springer Science & Business Media, 2012, vol. 15 (cit. on p. 47).
- [69] O. Deryugina, “Chatterbots,” *Scientific and Technical Information Processing*, vol. 37, no. 2, pp. 143–147, 2010 (cit. on p. 183).
- [70] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018 (cit. on pp. 33, 141, 143, 178, 193, 194).
- [71] B. A. Duncan, R. R. Murphy, D. Shell, and A. G. Hopper, “A mid-summer night’s dream: Social proof in hri,” in *Proceedings of the 5th ACM/IEEE international conference on Human-robot interaction*, IEEE Press, 2010, pp. 91–92 (cit. on p. 46).
- [72] C. Durang, “The actor’s nightmare,” *NY: Samuel French*, 1980 (cit. on pp. 66, 76).
- [73] N. Dziri, E. Kamalloo, K. W. Mathewson, and O. Zaiane, “Augmenting neural response generation with context-aware topical attention,” *arXiv preprint arXiv:1811.01063*, 2018 (cit. on pp. 38, 62, 143, 144).
- [74] J. Edlund, J. Gustafson, M. Heldner, and A. Hjalmarsson, “Towards human-like spoken dialogue systems,” *Speech communication*, vol. 50, no. 8-9, pp. 630–645, 2008 (cit. on p. 77).
- [75] A. L. Edwards, M. R. Dawson, J. S. Hebert, R. S. Sutton, K. M. Chan, and P. M. Pilarski, “Adaptive switching in practice: Improving myoelectric prosthesis performance through reinforcement learning,” *Proc. of MEC*, vol. 14, pp. 18–22, 2014 (cit. on p. 175).
- [76] M. Eger, C. M. Potts, C. Barot, and R. M. Young, “Plotter: Operationalizing the master book of all plots,” *Proceedings of the Intelligent Narrative Technologies and Social Believability in Games*, pp. 30–33, 2015 (cit. on pp. 113, 119).
- [77] J. L. Elman, “Finding structure in time,” *Cognitive science*, vol. 14, no. 2, pp. 179–211, 1990 (cit. on p. 33).

- [78] D. Elson, “Dramabank: Annotating agency in narrative discourse,” in *LREC*, 2012 (cit. on p. 130).
- [79] J. Fails and D. Olsen, “A design tool for camera-based interaction,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, 2003, pp. 449–456 (cit. on p. 7).
- [80] A. Fan, M. Lewis, and Y. Dauphin, “Hierarchical neural story generation,” *arXiv preprint arXiv:1805.04833*, 2018 (cit. on p. 130).
- [81] —, “Strategies for structuring story generation,” *arXiv preprint arXiv:1902.01109*, 2019 (cit. on p. 130).
- [82] D. Fedorenko, N. Smetanin, and A. Rodichev, “Avoiding echo-responses in a retrieval-based conversation system,” in *Artificial Intelligence and Natural Language*, D. Ustalov, A. Filchenkov, L. Pivovarova, and J. Žižka, Eds., Cham: Springer International Publishing, 2018, pp. 91–97 (cit. on pp. 26, 38).
- [83] B. Felbo, A. Mislove, A. Søgaard, I. Rahwan, and S. Lehmann, “Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm,” *ArXiv e-prints*, Aug. 2017. arXiv: 1708.00524 [stat.ML] (cit. on p. 100).
- [84] R. A. Fiebrink, *Real-time human interaction with supervised learning algorithms for music composition and performance*. Princeton University, 2011 (cit. on p. 75).
- [85] K. K. Fitzpatrick, A. Darcy, and M. Vierhile, “Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (woebot): A randomized controlled trial,” *JMIR Ment Health*, vol. 4, no. 2, e19, Jun. 2017 (cit. on pp. 16, 97).
- [86] L. Floridi, M. Taddeo, and M. Turilli, “Turing’s imitation game: Still an impossible challenge for all machines and some judges—an evaluation of the 2008 loebner contest,” *Minds and Machines*, vol. 19, no. 1, pp. 145–150, 2009 (cit. on p. 187).
- [87] T. Fong, C. Thorpe, and C. Baur, “Collaboration, dialogue, human-robot interaction,” in *Robotics Research*, Springer, 2003, pp. 255–266 (cit. on p. 77).
- [88] A. Frost and R. Yarrow, *Improvisation in drama, theatre and performance: History, practice, theory*. Macmillan International Higher Education, 2015 (cit. on p. 195).
- [89] E. B. Fry and J. E. Kress, *The reading teacher’s book of lists*. John Wiley & Sons, 2012, vol. 55 (cit. on p. 191).
- [90] J. Fu, A. Korattikara, S. Levine, and S. Guadarrama, “From language to goals: Inverse reinforcement learning for vision-based instruction following,” *arXiv preprint arXiv:1902.07742*, 2019 (cit. on p. 9).

- [91] A. Garland, *Ex Machina*. UK: Universal Studios., 2015 (cit. on p. 76).
- [92] D. Geeraerts, *Theories of lexical semantics*. Oxford University Press, 2010 (cit. on p. 112).
- [93] S. Ghosh, M. Chollet, E. Laksana, L.-P. Morency, and S. Scherer, “Affect-Im: A neural language model for customizable affective text generation,” *arXiv preprint arXiv:1704.06851*, 2017 (cit. on p. 39).
- [94] C. M. Gifford, “Collective machine learning: Team learning and classification in multi-agent systems,” PhD thesis, University of Kansas, 2009 (cit. on p. 178).
- [95] C. H. E. Gilbert, “Vader: A parsimonious rule-based model for sentiment analysis of social media text,” 2014 (cit. on p. 141).
- [96] R. L. Gilbert and A. Forney, “Can avatars pass the turing test? intelligent agent perception in a 3d virtual environment,” *International Journal of Human-Computer Studies*, vol. 73, pp. 30–36, 2015 (cit. on p. 187).
- [97] M. Gillies, R. Fiebrink, A. Tanaka, J. Garcia, F. Bevilacqua, A. Heloir, F. Nunnari, W. Mackay, S. Amershi, B. Lee, and N. d’Alessandro, “Human-centred machine learning,” in *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, ACM, 2016, pp. 3558–3565 (cit. on p. 179).
- [98] J. Glass, “Challenges for spoken dialogue systems,” in *Proc 1999 IEEE ASRU Wkshp*, 1999 (cit. on pp. 61, 70).
- [99] S. C. Gleason, “Don’t fear the cyborg: Toward embracing posthuman and feminist cyborg discourses in teacher education and educational technology research,” *Canadian Journal of Science, Mathematics and Technology Education*, vol. 14, no. 2, pp. 120–134, 2014 (cit. on p. 42).
- [100] J. J. Godfrey, E. C. Holliman, and J. McDaniel, “Switchboard: Telephone speech corpus for research and development,” in *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92. 1992 IEEE International Conference on*, IEEE, vol. 1, 1992, pp. 517–520 (cit. on pp. 36, 140).
- [101] A. Graves, “Generating sequences with recurrent neural networks,” *arXiv preprint arXiv:1308.0850*, 2013 (cit. on pp. 57, 97).
- [102] K. Gulordava, P. Bojanowski, E. Grave, T. Linzen, and M. Baroni, “Colorless green recurrent networks dream hierarchically,” *arXiv preprint arXiv:1803.11138*, 2018 (cit. on p. 33).
- [103] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, *et al.*, “Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs,” *Jama*, vol. 316, no. 22, pp. 2402–2410, 2016 (cit. on p. 177).

- [104] F. Guo, A. Metallinou, C. Khatri, A. Raju, A. Venkatesh, and A. Ram, “Topic-based evaluation for conversational bots,” *arXiv preprint arXiv:1801.03622*, 2018 (cit. on pp. 190, 194).
- [105] Y. Guo, “Improvchat: An ai-enabled dialogue assistant chatbot for english language learners (ell),” PhD thesis, OCAD University, 2018 (cit. on p. 73).
- [106] K. Guu, T. B. Hashimoto, Y. Oren, and P. Liang, “Generating sentences by editing prototypes,” *arXiv preprint arXiv:1709.08878*, 2017 (cit. on pp. 93, 112, 194).
- [107] A. Haas, “On aristotle’s concept of improvisation,” *Journal of Aesthetics and Phenomenology*, vol. 2, no. 1, pp. 113–121, 2015 (cit. on p. 195).
- [108] S. Halliwell *et al.*, *The Poetics of Aristotle: translation and commentary*. UNC Press Books, 1987 (cit. on pp. 41, 195, 198).
- [109] C. Halpern, D. Close, and K. Johnson, *Truth in comedy: The manual of improvisation*. Meriwether Publishing, 1994 (cit. on p. 40).
- [110] D. J. Hand, *Artificial intelligence and psychiatry*. Cambridge University Press, 1985, vol. 1 (cit. on p. 44).
- [111] W. Hansen and W. F. Hansen, *Classical Mythology: a guide to the mythical world of the Greeks and Romans*. Oxford University Press, USA, 2005 (cit. on p. 42).
- [112] B. Hayes-Roth and R. Van Gent, “Improvisational puppets, actors, and avatars,” in *Computer Game Developers Conference*, 1996 (cit. on pp. 51, 73, 75, 116).
- [113] J. S. Hebert, K. Elzinga, K. M. Chan, J. Olson, and M. Morhart, “Updates in targeted sensory reinnervation for upper limb amputation,” *Current Surgery Reports*, vol. 2, no. 3, p. 45, 2014 (cit. on p. 7).
- [114] R. Higashinaka, K. Imamura, T. Meguro, C. Miyazaki, N. Kobayashi, H. Sugiyama, T. Hirano, T. Makino, and Y. Matsuo, “Towards an open-domain conversational system fully based on natural language processing,” in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 928–939 (cit. on pp. 57, 61, 70).
- [115] J. Hilton, “Theatricality and technology: Pygmalion and the myth of the intelligent machine,” in *Dialogue and Technology: Art and Knowledge*, Springer, 1991, pp. 55–71 (cit. on pp. 40, 41).
- [116] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” *IEEE Signal processing magazine*, vol. 29, no. 6, pp. 82–97, 2012 (cit. on p. 15).

- [117] J. Hirschberg and C. D. Manning, “Advances in natural language processing,” *Science*, vol. 349, no. 6245, pp. 261–266, 2015 (cit. on p. 16).
- [118] D. Hirsh and P. Nation, “What vocabulary size is needed to read unsimplified texts for pleasure?” *Reading in a foreign language*, vol. 8, pp. 689–689, 1992 (cit. on p. 191).
- [119] M. K. Ho, J. MacGlashan, M. L. Littman, and F. Cushman, “Social is special: A normative framework for teaching with and learning from evaluative feedback,” *Cognition*, vol. 167, pp. 91–106, 2017, Moral Learning (cit. on p. 176).
- [120] M. K. Ho, M. Littman, J. MacGlashan, F. Cushman, and J. L. Austerweil, “Showing versus doing: Teaching by demonstration,” in *Advances in Neural Information Processing Systems*, 2016, pp. 3027–3035 (cit. on p. 12).
- [121] Q. Hoang, “Predicting movie genres based on plot summaries,” *arXiv preprint arXiv:1801.04813*, 2018 (cit. on p. 100).
- [122] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, *Gradient flow in recurrent nets: The difficulty of learning long-term dependencies*, 2001 (cit. on p. 29).
- [123] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997 (cit. on pp. 28, 30).
- [124] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997 (cit. on pp. 57, 82).
- [125] G. Hoffman and G. Weinberg, “Interactive improvisation with a robotic marimba player,” in *Musical robots and interactive multimodal systems*, Springer, 2011, pp. 233–251 (cit. on pp. 51, 75, 133).
- [126] G. Hollis, C. Westbury, and L. Lefsrud, “Extrapolating human judgments from skip-gram vector representations of word meaning,” *Q J Exp Psycho*, pp. 1–17, 2016 (cit. on p. 68).
- [127] E. Hovy, M. King, and A. Popescu-Belis, “An introduction to mt evaluation,” 2002 (cit. on p. 39).
- [128] J. Howard and S. Ruder, “Fine-tuned language models for text classification,” *arXiv preprint arXiv:1801.06146*, 2018 (cit. on p. 194).
- [129] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing, “Toward controlled generation of text,” in *International Conference on Machine Learning*, 2017, pp. 1587–1596 (cit. on p. 112).
- [130] D. Huggins-Daines, M. Kumar, A. Chan, A. W. Black, M. Ravishankar, and A. I. Rudnicky, “Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices,” in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, IEEE, vol. 1, 2006, pp. I–I (cit. on p. 123).

- [131] C. Hutto and E. Gilbert, “Vader: A parsimonious rule-based model for sentiment analysis of social media text,” in *International AAAI Conference on Web and Social Media*, 2014 (cit. on p. 88).
- [132] C. J. Hutto and E. Gilbert, “Vader: A parsimonious rule-based model for sentiment analysis of social media text,” in *Eighth international AAAI conference on weblogs and social media*, 2014 (cit. on p. 78).
- [133] M. Idel, “Golem: Jewish magical and mystical traditions on the artificial anthropoid,” 1990 (cit. on p. 42).
- [134] P. Indyk and R. Motwani, “Approximate nearest neighbors: Towards removing the curse of dimensionality,” in *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, ACM, 1998, pp. 604–613 (cit. on p. 99).
- [135] W. Isaacson, *Leonardo da Vinci*. Simon & Schuster, New York, USA, 2017 (cit. on p. 42).
- [136] P. Jaccard, “The distribution of the flora in the alpine zone. 1,” *New phytologist*, vol. 11, no. 2, pp. 37–50, 1912 (cit. on p. 26).
- [137] N. Jaques, J. Engel, D. Ha, F. Bertsch, R. W. Picard, and D. Eck, “Learning via social awareness: Improving sketch representations with facial feedback,” *CoRR*, vol. abs/1802.04877, 2018. arXiv: 1802.04877 (cit. on p. 47).
- [138] N. Jaques, S. Gu, D. Bahdanau, J. M. Hernández-Lobato, R. E. Turner, and D. Eck, “Sequence tutor: Conservative fine-tuning of sequence generation models with KL-control,” in *Proceedings of the 34th ICML*, ser. Proceedings of Machine Learning Research, vol. 70, Sydney, Australia: PMLR, Aug. 2017, pp. 1645–1654 (cit. on pp. 47, 141).
- [139] N. Jaques, S. Gu, R. E. Turner, and D. Eck, “Tuning recurrent neural networks with reinforcement learning,” *CoRR*, vol. abs/1611.02796, 2016 (cit. on p. 68).
- [140] ———, “Tuning recurrent neural networks with reinforcement learning,” *arXiv preprint arXiv:1611.02796*, 2016 (cit. on p. 141).
- [141] K. Jazdzewska, “From dialogos to dialogue: The use of the term from plato to the second century ce,” *Greek, Roman, and Byzantine Studies*, vol. 54, no. 1, pp. 17–36, 2013 (cit. on p. 14).
- [142] S. Jean, K. Cho, R. Memisevic, and Y. Bengio, “On using very large target vocabulary for neural machine translation,” *arXiv preprint arXiv:1412.2007*, 2014 (cit. on p. 192).
- [143] L. Jiang, A. Stocco, D. M. Losey, J. A. Abernethy, C. S. Prat, and R. P. N. Rao, “BrainNet: A Multi-Person Brain-to-Brain Interface for Direct Collaboration Between Brains,” *ArXiv e-prints*, Sep. 2018. arXiv: 1809.08632 [cs.HC] (cit. on p. 9).

- [144] T. Joachims, “A probabilistic analysis of the rocchio algorithm with tfidf for text categorization,” Carnegie-mellon univ pittsburgh pa dept of computer science, Tech. Rep., 1996 (cit. on p. 100).
- [145] R. L. Johannesen, “The emerging concept of communication as dialogue,” 1971 (cit. on p. 50).
- [146] P. N. Johnson-Laird, “How jazz musicians improvise,” *Music Perception: An Interdisciplinary Journal*, vol. 19, no. 3, pp. 415–442, 2002 (cit. on p. 44).
- [147] K. Johnstone, *Impro. Improvisation and the theatre*. Faber and Faber Ltd, 1979 (cit. on pp. 40, 44, 50–52, 55, 65, 68, 73, 74, 81, 98, 115, 117, 196).
- [148] S. Jonze, *Her*. USA: Annapurna Pictures., 2013 (cit. on p. 76).
- [149] A. Jordanousa and B. Kellerb, “Creativity is what we say it is: Constructing an ontology of creativity,” 1988 (cit. on p. 47).
- [150] R. Józefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu, “Exploring the limits of language modeling,” *CoRR*, vol. abs/1602.02410, 2016. arXiv: 1602.02410 (cit. on p. 192).
- [151] A. Kantosalo and H. Toivonen, “Modes for creative human-computer collaboration: Alternating and task-divided co-creativity,” in *Proceedings of the Seventh International Conference on Computational Creativity*, 2016 (cit. on p. 44).
- [152] F. Kaplan, P.-Y. Oudeyer, E. Kubinyi, and A. Miklósi, “Robotic clicker training,” *Robotics and Autonomous Systems*, vol. 38, no. 3, pp. 197–206, 2002 (cit. on p. 176).
- [153] K. Katevas, P. G. Healey, and M. T. Harris, “Robot comedy lab: Experimenting with the social dynamics of live performance,” *Frontiers in psychology*, vol. 6, p. 1253, 2015 (cit. on pp. 45, 179).
- [154] C. Khatri, R. Goel, B. Hedayatni, A. Metanillou, A. Venkatesh, R. Gabriel, and A. Mandal, “Contextual topic modeling for dialog systems,” *arXiv preprint arXiv:1810.08135*, 2018 (cit. on pp. 38, 190).
- [155] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014 (cit. on p. 59).
- [156] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler, “Skip-thought vectors,” in *Advances in NIPS*, 2015, pp. 3294–3302 (cit. on pp. 58, 143).
- [157] H. Knight, “Eight lessons learned about non-verbal interactions through robot theater,” in *International Conference on Social Robotics*, Springer, 2011, pp. 42–51 (cit. on pp. 44, 69).
- [158] H. Knight, S. Satkin, V. Ramakrishna, and S. Divvala, “A savvy robot standup comic: Online learning through audience tracking,” in *Workshop paper (TEI’10)*, 2011 (cit. on pp. 45, 113, 179).

- [159] W. B. Knox and P. Stone, “Interactively shaping agents via human reinforcement: The tamer framework,” in *Proceedings of the fifth international conference on Knowledge capture*, ACM, 2009, pp. 9–16 (cit. on pp. 9, 176).
- [160] ———, “Framing reinforcement learning from human reward: Reward positivity, temporal discounting, episodocity, and performance,” *Artificial Intelligence*, 2015 (cit. on p. 176).
- [161] J. C. Kowtko, S. D. Isard, and G. M. Doherty, “Conversational games within dialogue,” in *University of Edinburgh*, Citeseer, 1993 (cit. on pp. 35, 36).
- [162] S. A. Kripke, “Semantical analysis of modal logic i normal modal propositional calculi,” *Mathematical Logic Quarterly*, vol. 9, no. 5-6, pp. 67–96, 1963 (cit. on p. 98).
- [163] W. Labov, “Some further steps in narrative analysis.,” 1997 (cit. on p. 198).
- [164] B. Laurel, *Computers as theatre*. Addison-Wesley, 2013 (cit. on p. 41).
- [165] S. Lauria, G. Bugmann, T. Kyriacou, and E. Klein, “Mobile robot programming using natural language,” *Robotics and Autonomous Systems*, vol. 38, no. 3, pp. 171–181, 2002 (cit. on p. 9).
- [166] Q. Le and T. Mikolov, “Distributed representations of sentences and documents,” in *International Conference on Machine Learning*, 2014, pp. 1188–1196 (cit. on p. 120).
- [167] K. M. Lea, *Italian popular comedy: a study in the Commedia dell’arte, 1560-1620, with special reference to the English stage*. Russell & Russell Publishers, 1962, vol. 1 (cit. on p. 195).
- [168] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, p. 436, 2015 (cit. on p. 32).
- [169] J. Leike, D. Krueger, T. Everitt, M. Martic, V. Maini, and S. Legg, “Scalable agent alignment via reward modeling: a research direction,” *ArXiv e-prints*, Nov. 2018. arXiv: 1811.07871 (cit. on p. 95).
- [170] T. Leonard and P. Willis, *Pedagogies of the imagination: Mythopoetic curriculum in educational practice*. Springer, 2008 (cit. on p. 197).
- [171] L. LePage, “‘thinking something makes it so’: Performing robots, the workings of mimesis and the importance of character,” in *Twenty-First Century Drama: What Happens Now*, S. Adiseshiah and L. LePage, Eds. London: Palgrave Macmillan UK, 2016, pp. 279–301 (cit. on p. 46).
- [172] A. Leuski and D. R. Traum, “Practical language processing for virtual humans.,” 2010 (cit. on p. 26).
- [173] S. C. Levinson and F. Torreira, “Timing in turn-taking and its implications for processing models of language,” *Frontiers in Psychology*, vol. 6, p. 731, 2015 (cit. on pp. 93, 139).

- [174] B. Li, S. Lee-Urban, G. Johnston, and M. O. Riedl, “Story generation with crowdsourced plot graphs,” in *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, ser. AAAI’13, Bellevue, Washington: AAAI Press, 2013, pp. 598–604 (cit. on p. 116).
- [175] G. Li, S. Whiteson, W. B. Knox, and H. Hung, “Social interaction for efficient agent learning from human reward,” *Autonomous Agents and Multi-Agent Systems*, vol. 32, no. 1, pp. 1–25, Jan. 2018 (cit. on p. 176).
- [176] J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao, and D. Jurafsky, “Deep Reinforcement Learning for Dialogue Generation,” *ArXiv e-prints*, Jun. 2016. arXiv: 1606.01541 [cs.CL] (cit. on pp. 31, 97, 112, 133).
- [177] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, “A diversity-promoting objective function for neural conversation models,” *CoRR*, vol. abs/1510.03055, 2015. arXiv: 1510.03055 (cit. on pp. 31, 112, 190, 191).
- [178] J. Li, M. Galley, C. Brockett, G. P. Spithourakis, J. Gao, and B. Dolan, “A persona-based neural conversation model,” *arXiv preprint arXiv:1603.06155*, 2016 (cit. on pp. 38, 112).
- [179] J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao, and D. Jurafsky, “Deep reinforcement learning for dialogue generation,” *arXiv preprint arXiv:1606.01541*, 2016 (cit. on pp. 68, 141, 190, 191).
- [180] J. Li, W. Monroe, T. Shi, A. Ritter, and D. Jurafsky, “Adversarial learning for neural dialogue generation,” *arXiv preprint arXiv:1701.06547*, 2017 (cit. on pp. 68, 140).
- [181] X. Liang, Z. Hu, H. Zhang, C. Gan, and E. P. Xing, “Recurrent topic-transition gan for visual paragraph generation,” *CoRR*, *abs/1703.07022*, vol. 2, 2017 (cit. on p. 112).
- [182] A. Liapis, G. N. Yannakakis, C. Alexopoulos, and P. Lopes, “Can computers foster human users’ creativity? theory and praxis of mixed-initiative co-creativity,” *Digital Culture & Education*, vol. 8, no. 2, pp. 136–153, 2016 (cit. on p. 44).
- [183] C.-Y. Lin, “Rouge: A package for automatic evaluation of summaries,” in *Proc ACL Wkshp. Vol 8.*, 2004 (cit. on pp. 36, 68).
- [184] Z. C. Lipton, J. Berkowitz, and C. Elkan, “A critical review of recurrent neural networks for sequence learning,” *arXiv preprint arXiv:1506.00019*, 2015 (cit. on p. 69).
- [185] P. Lison and J. Tiedemann, “Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles,” *Proceedings of LREC 2016*, 2016 (cit. on p. 35).

- [186] P. Lison, J. Tiedemann, and M. Kouylekov, “Opensubtitles 2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora,” in *Proceedings of the 11th edition of the Language Resources and Evaluation Conference (LREC 2018), Miyazaki, Japan.(accepted)*, 2018 (cit. on pp. 99, 109).
- [187] D. J. Litman, C. P. Rose, K. Forbes-Riley, K. VanLehn, D. Bhembe, and S. Silliman, “Spoken versus typed human and computer dialogue tutoring,” in *International Conference on Intelligent Tutoring Systems*, Springer, 2004, pp. 368–379 (cit. on p. 16).
- [188] C.-W. Liu, R. Lowe, I. V. Serban, M. Noseworthy, L. Charlin, and J. Pineau, “How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation.,” in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 2122–2132 (cit. on pp. 26, 36, 38, 39, 62, 69, 190).
- [189] P. J. Liu, M. Saleh, E. Pot, B. Goodrich, R. Sepassi, L. Kaiser, and N. Shazeer, “Generating wikipedia by summarizing long sequences,” *arXiv preprint arXiv:1801.10198*, 2018 (cit. on p. 97).
- [190] R. Loftin, B. Peng, J. MacGlashan, M. L. Littman, M. E. Taylor, J. Huang, and D. L. Roberts, “Learning behaviors via human-delivered discrete feedback: Modeling implicit feedback strategies to speed up learning,” *Autonomous Agents and Multi-Agent Systems*, pp. 1–30, 2015 (cit. on p. 176).
- [191] D. Long, M. Jacob, N. Davis, and B. Magerko, “Designing for socially interactive systems,” in *Proceedings of the 2017 ACM SIGCHI Conference on Creativity and Cognition*, ACM, 2017, pp. 39–50 (cit. on p. 47).
- [192] G. Lösel, “Can robots improvise?” *Liminalities*, vol. 14, no. 1, pp. 185–206, 2018 (cit. on pp. 42, 44).
- [193] R. Lowe, M. Noseworthy, I. V. Serban, N. Angelard-Gontier, Y. Bengio, and J. Pineau, “Towards an automatic turing test: Learning to evaluate dialogue responses,” *arXiv preprint arXiv:1708.07149*, 2017 (cit. on pp. 36, 38, 190).
- [194] R. Lowe, N. Pow, I. Serban, and J. Pineau, “The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems,” *arXiv preprint arXiv:1506.08909*, 2015 (cit. on p. 189).
- [195] D. V. Lu and W. D. Smart, “Human-robot interactions as theatre,” in *RO-MAN, 2011 IEEE*, IEEE, 2011, pp. 473–478 (cit. on p. 44).
- [196] T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 1412–1421 (cit. on p. 69).

- [197] T. Luong, I. Sutskever, Q. V. Le, O. Vinyals, and W. Zaremba, “Addressing the rare word problem in neural machine translation,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2015, pp. 11–19 (cit. on p. 33).
- [198] B. MacCartney, “Nlp lunch tutorial: Smoothing,” 2005 (cit. on p. 22).
- [199] B. Magerko, P. Dohogne, and C. DeLeon, “Employing fuzzy concept for digital improvisational theatre.,” 2011 (cit. on pp. 51, 75, 116).
- [200] B. Magerko, W. Manzoul, M. Riedl, A. Baumer, D. Fuller, K. Luther, and C. Pearce, “An empirical study of cognition and theatrical improvisation,” in *Proceedings of the seventh ACM conference on Creativity and cognition*, ACM, 2009, pp. 117–126 (cit. on pp. 44, 51, 65, 72, 73, 76, 115, 117).
- [201] M. Manfredi, “Translating text and context: Translation studies and systemic functional linguistics. volume 2: From theory to practice,” 2014 (cit. on p. 140).
- [202] M. d. G. B. Marietto, R. V. de Aguiar, G. d. O. Barbosa, W. T. Botelho, E. Pimentel, R. d. S. França, and V. L. da Silva, “Artificial intelligence markup language: A brief tutorial,” *arXiv preprint arXiv:1307.3091*, 2013 (cit. on p. 184).
- [203] L. J. Martin, P. Ammanabrolu, X. Wang, S. Singh, B. Harrison, M. Dhuliawala, P. Tambwekar, A. A. Mehta, R. Arora, N. Dass, C. Purdy, and M. O. Riedl, “Improvisational storytelling agents,” 2017 (cit. on p. 111).
- [204] L. J. Martin, P. Ammanabrolu, W. Hancock, S. Singh, B. Harrison, and M. O. Riedl, “Event representations for automated story generation with deep neural nets,” *arXiv preprint arXiv:1706.01331*, 2017 (cit. on p. 116).
- [205] L. J. Martin, P. Ammanabrolu, X. Wang, W. Hancock, S. Singh, B. Harrison, and M. O. Riedl, “Event representations for automated story generation with deep neural nets,” *CoRR*, vol. abs/1706.01331, 2017. arXiv: 1706.01331 (cit. on p. 95).
- [206] L. J. Martin, B. Harrison, and M. O. Riedl, “Improvisational computational storytelling in open worlds,” in *International Conference on Interactive Digital Storytelling*, Springer, 2016, pp. 73–84 (cit. on pp. 60, 73, 79, 95, 111, 115, 117, 133).
- [207] L. J. Martin, S. Sood, and M. O. Riedl, “Dungeons and dqns: Toward reinforcement learning agents that play tabletop roleplaying games,” in *Joint Workshop of Intelligent Narrative Technologies (INT) and the Workshop on Intelligent Cinematography and Editing (WICED)*, 2018 (cit. on p. 133).

- [208] M. Mateas, “An oz-centric review of interactive drama and believable agents,” in *Artificial intelligence today*, Springer, 1999, pp. 297–328 (cit. on p. 77).
- [209] M. Mateas and A. Stern, “Façade: An experiment in building a fully-realized interactive drama,” in *Game developers conference*, vol. 2, 2003, pp. 4–8 (cit. on p. 119).
- [210] K. W. Mathewson and P. M. Pilarski, “Simultaneous Control and Human Feedback in the Training of a Robotic Agent with Actor-Critic Reinforcement Learning,” in *IJCAI Interactive Machine Learning Wkshp*, 2016 (cit. on pp. 69, 95, 176).
- [211] K. W. Mathewson and P. Mirowski, “Improvised comedy as a turing test,” *CoRR*, vol. abs/1711.08819, 2017. arXiv: 1711.08819 (cit. on pp. 77, 111, 115).
- [212] —, “Improvised theatre alongside artificial intelligences,” in *AAAI AIIDE*, 2017 (cit. on pp. 72–74, 93, 94, 112, 116, 117, 124).
- [213] M. L. Mauldin, “Chatterbots, tinymuds, and the turing test: Entering the loebner prize competition,” in *AAAI*, vol. 94, 1994, pp. 16–21 (cit. on p. 184).
- [214] N. Mavridis, “A review of verbal and non-verbal human–robot interactive communication,” *Robotics and Autonomous Systems*, vol. 63, pp. 22–35, 2015 (cit. on pp. 9, 50).
- [215] B. McCann, J. Bradbury, C. Xiong, and R. Socher, “Learned in translation: Contextualized word vectors,” in *Advances in Neural Information Processing Systems*, 2017, pp. 6294–6305 (cit. on p. 140).
- [216] B. McCann, N. S. Keskar, C. Xiong, and R. Socher, “The natural language decathlon: Multitask learning as question answering,” *arXiv preprint arXiv:1806.08730*, 2018 (cit. on p. 33).
- [217] J. McCarthy, “Making robots conscious of their mental states.” in *Machine intelligence 15*, 1995, pp. 3–17 (cit. on p. 176).
- [218] J. McCormack and M. d’Inverno, “Computers and creativity: The road ahead,” in *Computers and creativity*, Springer, 2012, pp. 421–424 (cit. on p. 69).
- [219] —, “Designing improvisational interfaces,” 2016 (cit. on p. 69).
- [220] R. McKee, *Substance, structure, style, and the principles of screenwriting*. New York: HarperCollins, 1997 (cit. on p. 119).
- [221] C. McLeod, “The use of Keith Johnstone’s concepts of Circles of Expectation and Vogler’s generic story structure as directorial aids in comedy theatre,” PhD thesis, University of Cape Town, 2000 (cit. on p. 98).
- [222] H. B. McMahan, E. Moore, D. Ramage, and B. A. y Arcas, “Federated learning of deep networks using model averaging,” *CoRR*, vol. abs/1602.05629, 2016. arXiv: 1602.05629 (cit. on p. 178).

- [223] J. R. Meehan, “Tale-spin, an interactive program that writes stories.,” in *IJCAI*, vol. 77, 1977, pp. 91–98 (cit. on p. 116).
- [224] J. R. Meehan, “The metanovel: Writing stories by computer.,” Yale Univ, New Haven Conn, Dept of Comp Sci, Tech. Rep., 1976 (cit. on p. 115).
- [225] S. Meisner and D. Longwell, *Sanford Meisner on acting*. Vintage, 2012 (cit. on p. 197).
- [226] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013 (cit. on p. 193).
- [227] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, “Recurrent neural network based language model,” in *Eleventh Annual Conference of the International Speech Communication Association*, 2010 (cit. on p. 57).
- [228] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *Advances in neural information processing systems*, 2013, pp. 3111–3119 (cit. on pp. 57, 121).
- [229] T. Mikolov and G. Zweig, “Context dependent recurrent neural network language model.,” *SLT*, vol. 12, pp. 234–239, 2012 (cit. on p. 58).
- [230] A. Miner, A. Milstein, and J. Hancock, “Talking to machines about personal mental health problems,” *JAMA*, vol. 318, no. 13, pp. 1217–1218, 2017. eprint: /data/journals/jama/936503/jama_miner_2017_vp_170136.pdf (cit. on pp. 16, 97).
- [231] P. Mirowski, S. Chopra, S. Balakrishnan, and S. Bangalore, “Feature-rich continuous language models for speech recognition,” in *Spoken Language Technology Wkshp, 2010 IEEE*, IEEE, 2010, pp. 241–246 (cit. on p. 58).
- [232] S. Mohammad, “From once upon a time to happily ever after: Tracking emotions in novels and fairy tales,” in *Proc. of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, ACL, 2011, pp. 105–114 (cit. on p. 112).
- [233] R. K. Moore, “Is spoken language all-or-nothing? implications for future speech-based human-machine interaction,” in *Dialogues with Social Robots*, Springer, 2017, pp. 281–291 (cit. on pp. 15, 16).
- [234] M. Morford, *Classical Mythology*. Oxford University Press, 2007 (cit. on p. 42).
- [235] M. Mori, “The uncanny valley,” *Energy*, vol. 7, no. 4, pp. 33–35, 1970 (cit. on p. 78).

- [236] M. Mori, K. F. MacDorman, and N. Kageki, “The uncanny valley [from the field],” *IEEE Robotics & Automation Magazine*, vol. 19, no. 2, pp. 98–100, 2012 (cit. on pp. 51, 76).
- [237] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012 (cit. on p. 105).
- [238] L. Na and I. Nation, “Factors affecting guessing vocabulary in context,” *RELC journal*, vol. 16, no. 1, pp. 33–42, 1985 (cit. on p. 191).
- [239] S. Nachmanovitch, *Free play: Improvisation in life and art*. Penguin, 1990 (cit. on p. 196).
- [240] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814 (cit. on p. 28).
- [241] M. Napier, *Improvise: Scene from the inside out*. Heinemann Drama, 2004 (cit. on pp. 68, 74).
- [242] C. I. Nass and S. Brave, *Wired for speech: How voice activates and advances the human-computer relationship*. MIT press Cambridge, MA, 2005 (cit. on p. 15).
- [243] P. Nation and R. Waring, “Vocabulary size, text coverage and word lists,” *Vocabulary: Description, acquisition and pedagogy*, vol. 14, pp. 6–19, 1997 (cit. on p. 191).
- [244] J. Needham, *Science and Civilisation in China: Historical Survey, from Cinnabar Elixirs to Synthetic Insulin. Vol. 5, Chemistry and chemical technology; Pt. 3, Spagyrical discovery and invention*. Cambridge University Press, 1974, vol. 3 (cit. on p. 41).
- [245] S. Nikitina, S. Callaioli, and M. Baez, “Smart conversational agents for reminiscence,” *arXiv preprint arXiv:1804.06550*, 2018 (cit. on pp. 16, 97).
- [246] B. O’Neill, A. Piplica, D. Fuller, and B. Magerko, “A knowledge-based framework for the collaborative improvisation of scene introductions.,” in *ICIDS*, Springer, 2011, pp. 85–96 (cit. on pp. 51, 75, 116).
- [247] J. W. Orr, P. Tadepalli, J. R. Doppa, X. Fern, and T. G. Dietterich, “Learning scripts as hidden markov models.,” 2014 (cit. on p. 25).
- [248] Z. Pan, A. D. Cheok, H. Yang, J. Zhu, and J. Shi, “Virtual reality and mixed reality for virtual learning environments,” *Computers & graphics*, vol. 30, no. 1, pp. 20–28, 2006 (cit. on p. 179).
- [249] D. Paperno, G. Kruszewski, A. Lazaridou, Q. N. Pham, R. Bernardi, S. Pezzelle, M. Baroni, G. Boleda, and R. Fernández, “The lambada dataset: Word prediction requiring a broad discourse context,” *arXiv preprint arXiv:1606.06031*, 2016 (cit. on p. 194).

- [250] K. Papineni, S. Roukos, T. Ward, J. Henderson, and F. Reeder, “Corpus-based comprehensive and diagnostic mt evaluation: Initial arabic, chinese, french, and spanish results,” in *Proceedings of the second international conference on Human Language Technology Research*, Morgan Kaufmann Publishers Inc., 2002, pp. 132–137 (cit. on p. 39).
- [251] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “Bleu: A method for automatic evaluation of machine translation,” in *Proc 40th Meeting on Association for Computational Linguistics*, 2002, pp. 311–318 (cit. on pp. 36, 68).
- [252] J. Parker-Starbuck, *Cyborg Theatre: Corporeal/Technological Intersections in Multimedia Performance*. Springer, 2011 (cit. on p. 42).
- [253] H. Patterson, *Playing with Grown Ups*. Bloomsbury, 2013 (cit. on p. 88).
- [254] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation.,” in *EMNLP*, vol. 14, 2014, pp. 1532–1543 (cit. on pp. 57, 58).
- [255] I. M. Pepperberg, *The Alex studies: cognitive and communicative abilities of grey parrots*. Harvard University Press, 2009 (cit. on p. 77).
- [256] K. Perlin and A. Goldberg, “Improv: A system for scripting interactive actors in virtual worlds,” in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, ACM, 1996, pp. 205–216 (cit. on pp. 51, 73, 75, 116).
- [257] J. Pfanzagl, *Parametric statistical theory*. Walter de Gruyter, 2011 (cit. on p. 30).
- [258] O. Pietquin, *A framework for unsupervised learning of dialogue strategies*. Presses univ. de Louvain, 2005 (cit. on p. 52).
- [259] P. M. Pilarski, R. S. Sutton, K. W. Mathewson, C. Sherstan, A. S. R. Parker, and A. L. Edwards, “Communicative capital for prosthetic agents,” *CoRR*, vol. abs/1711.03676, 2017. arXiv: 1711.03676 (cit. on pp. 8, 10).
- [260] P. M. Pilarski, M. R. Dawson, T. Degris, F. Fahimi, J. P. Carey, and R. S. Sutton, “Online human training of a myoelectric prosthesis controller via actor-critic reinforcement learning,” in *Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on*, IEEE, 2011, pp. 1–7 (cit. on p. 175).
- [261] P. M. Pilarski and R. S. Sutton, “Between instruction and reward: Human-prompted switching.,” in *AAAI Fall Symposium: Robots Learning Interactively from Human Teachers*, 2012 (cit. on p. 9).

- [262] P. M. Pilarski, R. S. Sutton, and K. W. Mathewson, “Prosthetic devices as goal-seeking agents,” in *Second Workshop on Present and Future of Non-Invasive Peripheral-Nervous-System Machine Interfaces: Progress in Restoring the Human Functions (PNS-MI), Singapore, Aug. 11, 2015.*, 2015 (cit. on p. 69).
- [263] I. Pluta, “Teatro e robótica: Os androides de hirohi ishiguro, em encenações de oriza hirata,” *ARJ-Art Research Journal*, vol. 3, no. 1, pp. 65–79, 2016 (cit. on p. 45).
- [264] G. Podevijn, R. O’Grady, Y. S. Nashed, and M. Dorigo, “Gesturing at subswarms: Towards direct human control of robot swarms,” in *Towards Autonomous Robotic Systems*, Springer, 2014, pp. 390–403 (cit. on p. 9).
- [265] M. Popel and O. Bojar, “Training tips for the transformer model,” *CoRR*, vol. abs/1804.00247, 2018. arXiv: 1804.00247 (cit. on p. 189).
- [266] K. Pryor, *Dont shoot the dog*. New York: Bantam, 1999 (cit. on p. 176).
- [267] G. K. Pullum, “The conduct of” linguistic inquiry”,” *Natural Language & Linguistic Theory*, pp. 435–440, 1983 (cit. on p. 33).
- [268] L. R. Rabiner and B.-H. Juang, “An introduction to hidden markov models,” *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986 (cit. on p. 25).
- [269] G. Raby, “Improvisation and devising: The circle of expectation, the invisible hand, and rsvp,” *Canadian Theatre Review*, vol. 143, no. 1, pp. 94–97, 2010 (cit. on p. 98).
- [270] N. M. Radziwill and M. C. Benton, “Evaluating quality of chatbots and intelligent conversational agents,” *arXiv preprint arXiv:1704.04579*, 2017 (cit. on p. 35).
- [271] S. Rajeswar, S. Subramanian, F. Dutil, C. Pal, and A. Courville, “Adversarial generation of natural language,” *arXiv preprint arXiv:1705.10929*, 2017 (cit. on p. 68).
- [272] A. Ram, R. Prasad, C. Khatri, A. Venkatesh, R. Gabriel, Q. Liu, J. Nunn, B. Hedayatnia, M. Cheng, A. Nagar, and E. King, “Conversational ai: The science behind the alexa prize,” *arXiv preprint arXiv:1801.03604*, 2018 (cit. on p. 38).
- [273] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba, “Sequence level training with recurrent neural networks,” *arXiv:1511.06732*, 2015 (cit. on pp. 68, 69).
- [274] A. J. Reagan, L. Mitchell, D. Kiley, C. M. Danforth, and P. S. Dodds, “The emotional arcs of stories are dominated by six basic shapes,” *EPJ Data Science*, vol. 5, no. 1, p. 31, 2016 (cit. on p. 112).

- [275] M. O. Riedl and A. Stern, “Believable agents and intelligent story adaptation for interactive storytelling,” in *Intl Conf on Technologies for Interactive Digital Storytelling and Entertainment*, Springer, 2006, pp. 1–12 (cit. on pp. 47, 51, 65, 75).
- [276] M. O. Riedl and R. M. Young, “From linear story generation to branching story graphs.,” *IEEE Computer Graphics and Applications*, vol. 26, no. 3, pp. 23–31, 2006 (cit. on p. 113).
- [277] ———, “Narrative planning: Balancing plot and character,” *Journal of Artificial Intelligence Research*, vol. 39, pp. 217–268, 2010 (cit. on pp. 115, 116).
- [278] L. D. Riek, “Wizard-of-oz studies in hri: A systematic review and new reporting guidelines,” *J. HRI*, vol. 1, no. 1, 2012 (cit. on pp. 75, 77).
- [279] D. Riggs, *Flying Funny: My Life Without a Net*. University of Minnesota Press, 2017 (cit. on p. 196).
- [280] A. Ritter, C. Cherry, and W. B. Dolan, “Data-driven response generation in social media,” in *Proceedings of the conference on empirical methods in natural language processing*, Association for Computational Linguistics, 2011, pp. 583–593 (cit. on pp. 26, 30, 36, 190).
- [281] C. R. Rogers, *Counseling and psychotherapy*. Cambridge, MA: Riverside Press, 1942 (cit. on p. 181).
- [282] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985 (cit. on p. 27).
- [283] D. E. Rumelhart, G. E. Hinton, R. J. Williams, *et al.*, “Learning representations by back-propagating errors,” *Cognitive modeling*, vol. 5, no. 3, p. 1, (cit. on p. 33).
- [284] K. Salen, K. S. Tekinbaş, and E. Zimmerman, *Rules of play: Game design fundamentals*. MIT press, 2004 (cit. on p. 7).
- [285] T. Salinsky and D. Frances-White, *The improv handbook: The ultimate guide to improvising in comedy, theatre, and beyond*. Bloomsbury Publishing, 2017 (cit. on p. 66).
- [286] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” *Information processing & management*, vol. 24, no. 5, pp. 513–523, 1988 (cit. on p. 100).
- [287] A. Saparov and T. M. Mitchell, “A probabilistic generative grammar for semantic parsing,” *CoRR*, vol. abs/1606.06361, 2016. arXiv: 1606.06361 (cit. on p. 33).
- [288] R. K. Sawyer, *Improvised dialogues: Emergence and creativity in conversation*. Greenwood Publishing Group, 2003 (cit. on pp. 97, 98, 108, 111, 116, 117, 121, 124, 195, 196).

- [289] A. P. Saygin, I. Cicekli, and V. Akman, “Turing test: 50 years later,” *Minds and machines*, vol. 10, no. 4, pp. 463–518, 2000 (cit. on p. 182).
- [290] S. Schaal, “Learning from demonstration,” in *Advances in neural information processing systems*, 1997, pp. 1040–1046 (cit. on p. 175).
- [291] J. Schatzmann, K. Georgila, and S. Young, “Quantitative evaluation of user simulation techniques for spoken dialogue systems,” in *6th SIGdial Workshop on DISCOURSE and DIALOGUE*, 2005 (cit. on p. 189).
- [292] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks*, vol. 61, pp. 85–117, 2015 (cit. on p. 27).
- [293] J. Schreier and C. Ford, *Robot & Frank*. USA: Dog Run Pictures, Park Pictures, TBB., 2012 (cit. on p. 76).
- [294] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997 (cit. on p. 30).
- [295] L. Sennhauser and R. C. Berwick, “Evaluating the ability of lstms to learn context-free grammars,” *arXiv preprint arXiv:1811.02611*, 2018 (cit. on p. 33).
- [296] R. Sennrich, B. Haddow, and A. Birch, “Neural machine translation of rare words with subword units,” *arXiv preprint arXiv:1508.07909*, 2015 (cit. on pp. 22, 36, 97).
- [297] I. V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, “Building end-to-end dialogue systems using generative hierarchical neural network models,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, ser. AAAI’16, Phoenix, Arizona: AAAI Press, 2016, pp. 3776–3783 (cit. on pp. 16, 18).
- [298] I. V. Serban, R. Lowe, P. Henderson, L. Charlin, and J. Pineau, “A survey of available corpora for building data-driven dialogue systems,” *arXiv preprint arXiv:1512.05742*, 2015 (cit. on pp. 16, 34).
- [299] I. V. Serban, C. Sankar, M. Germain, S. Zhang, Z. Lin, S. Subramanian, T. Kim, M. Pieper, S. Chandar, N. R. Ke, S. Mudumba, A. de Brébisson, J. Sotelo, D. Suhubdy, V. Michalski, A. Nguyen, J. Pineau, and Y. Bengio, “A deep reinforcement learning chatbot,” *CoRR*, vol. abs/1709.02349, 2017. arXiv: 1709.02349 (cit. on pp. 32, 191).
- [300] I. V. Serban, A. Sordoni, Y. Bengio, A. C. Courville, and J. Pineau, “Building end-to-end dialogue systems using generative hierarchical neural network models.,” in *AAAI*, vol. 16, 2016, pp. 3776–3784 (cit. on pp. 38, 112).

- [301] I. V. Serban, A. Sordoni, R. Lowe, L. Charlin, J. Pineau, A. C. Courville, and Y. Bengio, “A hierarchical latent variable encoder-decoder model for generating dialogues,” in *AAAI*, 2017, pp. 3295–3301 (cit. on p. 31).
- [302] L. Shang, Z. Lu, and H. Li, “Neural responding machine for short-text conversation,” *arXiv preprint arXiv:1503.02364*, 2015 (cit. on pp. 30, 58).
- [303] C. E. Shannon, “Prediction and entropy of printed english,” *Bell Labs Technical Journal*, vol. 30, no. 1, pp. 50–64, 1951 (cit. on p. 99).
- [304] C. E. Shannon, “A mathematical theory of communication,” *Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948 (cit. on pp. 11, 36).
- [305] L. Shao, S. Gouws, D. Britz, A. Goldie, B. Strope, and R. Kurzweil, “Generating high-quality and informative conversation responses with sequence-to-sequence models,” *arXiv preprint arXiv:1701.03185*, 2017 (cit. on p. 140).
- [306] G. B. Shaw, *Pygmalion*. Simon and Schuster, 1913 (cit. on pp. 42, 43, 52).
- [307] M. Shelley, “Frankenstein,” in *Medicine and Literature, Volume Two*, CRC Press, 2018, pp. 35–52 (cit. on p. 42).
- [308] M. W. Shelley, *Frankenstein*. WW Norton, 1818 (cit. on p. 52).
- [309] H.-Y. Shum, X.-d. He, and D. Li, “From Eliza to XiaoIce: Challenges and opportunities with social chatbots,” *Frontiers of Information Technology & Electronic Engineering*, vol. 19, no. 1, pp. 10–26, 2018 (cit. on p. 188).
- [310] M. Si, S. C. Marsella, and D. V. Pynadath, “Thespian: An architecture for interactive pedagogical drama.” (cit. on pp. 51, 75).
- [311] W. Smith, *The commedia dell’arte: a study in Italian popular comedy*. Columbia University Press, 1912, vol. 3 (cit. on p. 195).
- [312] B. Snyder, *Save the cat*. Michael Wiese Productions California, 2005 (cit. on p. 130).
- [313] A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J.-Y. Nie, J. Gao, and B. Dolan, “A neural network approach to context-sensitive generation of conversational responses,” *arXiv preprint arXiv:1506.06714*, 2015 (cit. on pp. 30, 112, 190).
- [314] V. Spolin, *Improvisation for the theater: A handbook of teaching and directing techniques*. Northwestern University Press, 1983 (cit. on pp. 55, 196).
- [315] E. W. Stein, “Improvisation as model for real-time decision making,” in *Supporting real time decision-making*, Springer, 2011, pp. 13–32 (cit. on p. 115).

- [316] J. L. Styan, *Shakespeare's stagecraft*. Cambridge University Press, 1967 (cit. on p. 41).
- [317] I. Sutskever, J. Martens, and G. E. Hinton, "Generating text with recurrent neural networks," in *Proc 28th ICML*, 2011, pp. 1017–1024 (cit. on pp. 28, 57).
- [318] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *NIPS*, 2014, pp. 3104–3112 (cit. on pp. 30, 31, 33, 58, 82).
- [319] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT Press, 1998 (cit. on p. 197).
- [320] M. Swain, "The output hypothesis and beyond: Mediating acquisition through collaborative dialogue," *Sociocultural Theory and Second Language Learning*, vol. 97, p. 114, 2000 (cit. on pp. 96–98).
- [321] J. Sweet, *Something wonderful right away*. Hal Leonard Corporation, 1987 (cit. on p. 196).
- [322] C. Tao, L. Mou, D. Zhao, and R. Yan, "Ruber: An unsupervised method for automatic evaluation of open-domain dialog systems," *arXiv preprint arXiv:1701.03079*, 2017 (cit. on p. 190).
- [323] A. C. Tenorio-Gonzalez, E. F. Morales, and L. Villaseñor-Pineda, "Dynamic reward shaping: Training a robot by voice," in *Advances in Artificial Intelligence—IBERAMIA 2010*, Springer, 2010, pp. 483–492 (cit. on p. 9).
- [324] M. Theune, S. Faas, A. Nijholt, and D. Heylen, "The virtual storyteller: Story creation by intelligent agents," in *Proceedings of the Technologies for Interactive Digital Storytelling and Entertainment (TIDSE) Conference*, vol. 204215, 2003 (cit. on p. 116).
- [325] A. L. Thomaz, G. Hoffman, and C. Breazeal, "Reinforcement learning with human teachers: Understanding how people want to teach robots," in *Robot and Human Interactive Communication, 2006. ROMAN 2006. The 15th IEEE International Symposium on*, IEEE, 2006, pp. 352–357 (cit. on p. 9).
- [326] A. Thomaz, G. Hoffman, and M. Cakmak, "Computational human-robot interaction," *Foundations and Trends® in Robotics*, vol. 4, no. 2-3, pp. 105–223, 2016 (cit. on pp. 51, 75, 176).
- [327] D. Thue, V. Bulitko, M. Spetch, and E. Wasylishen, "Interactive storytelling: A player modelling approach.," 2007 (cit. on p. 47).
- [328] X. Tong, Z. Fu, M. Shang, D. Zhao, and R. Yan, "One" ruler" for all languages: Multi-lingual dialogue evaluation with adversarial multi-task learning," *arXiv preprint arXiv:1805.02914*, 2018 (cit. on pp. 143, 190).

- [329] T. Trabasso and L. L. Sperry, “Causal relatedness and importance of story events,” *Journal of Memory and language*, vol. 24, no. 5, p. 595, 1985 (cit. on p. 113).
- [330] A. M. Turing, “Computing machinery and intelligence,” *Mind*, vol. 59, no. 236, pp. 433–460, 1950 (cit. on pp. 14, 70, 77, 115, 187).
- [331] S. Turkle, *Life on the Screen*. Simon and Schuster, 2011 (cit. on p. 138).
- [332] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio.,” in *SSW*, 2016, p. 125 (cit. on pp. 15, 70).
- [333] S. Vasileiadou, D. Kalligeropoulos, and N. Karcianas, “Systems, modelling and control in ancient greece: Part 1: Mythical automata,” *Measurement and Control*, vol. 36, no. 3, pp. 76–80, 2003 (cit. on p. 41).
- [334] A. Vaswani, S. Bengio, E. Brevdo, F. Chollet, A. N. Gomez, S. Gouws, L. Jones, L. Kaiser, N. Kalchbrenner, N. Parmar, R. Sepassi, N. Shazeer, and J. Uszkoreit, “Tensor2Tensor for Neural Machine Translation,” *ArXiv e-prints*, Mar. 2018. arXiv: 1803.07416 [cs.LG] (cit. on p. 34).
- [335] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” *CoRR*, vol. abs/1706.03762, 2017. arXiv: 1706.03762 (cit. on pp. 33, 34, 109).
- [336] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, “Cider: Consensus-based image description evaluation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575 (cit. on p. 36).
- [337] V. Veeriah, P. M. Pilarski, and R. S. Sutton, “Face valuing: Training user interfaces with facial expressions and reinforcement learning,” *CoRR*, vol. abs/1606.02807, 2016. arXiv: 1606.02807 (cit. on p. 9).
- [338] A. Venkatesh, C. Khatri, A. Ram, F. Guo, R. Gabriel, A. Nagar, R. Prasad, M. Cheng, B. Hedayatnia, A. Metallinou, R. Goel, S. Yang, and A. Raju, “On evaluating and comparing conversational agents.,” *arXiv preprint arXiv:1801.03625*, 2018 (cit. on pp. 35, 38, 97, 143).
- [339] M. Vijaymeena and K. Kavitha, “A survey on similarity measures in text mining,” *Mach. Learn. Appl. Int. J*, vol. 3, pp. 19–28, 2016 (cit. on p. 17).
- [340] O. Vinyals and Q. Le, “A neural conversational model,” *arXiv preprint arXiv:1506.05869*, 2015 (cit. on pp. 59, 75, 82, 112, 141).
- [341] K. Vonnegut, *Palm Sunday*. RosettaBooks LLC, New York, 1981 (cit. on p. 96).
- [342] S. G. Ware and R. M. Young, “Cpocl: A narrative planner supporting conflict.,” in *AIIDE*, 2011 (cit. on p. 116).

- [343] M. Warren, *Features of naturalness in conversation*. John Benjamins Publishing, 2006, vol. 152 (cit. on p. 190).
- [344] J. Weizenbaum, “Eliza: A computer program for the study of natural language communication between man and machine,” *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966 (cit. on pp. 67, 80, 97, 181).
- [345] —, “Computer power and human reason: From judgment to calculation,” 1976 (cit. on p. 43).
- [346] T.-H. Wen, Y. Miao, P. Blunsom, and S. Young, “Latent intention dialogue models,” *arXiv preprint arXiv:1705.10229*, 2017 (cit. on p. 194).
- [347] J. Weston, E. Dinan, and A. H. Miller, “Retrieve and Refine: Improved Sequence Generation Models For Dialogue,” *ArXiv e-prints*, Aug. 2018. arXiv: 1808.04776 [cs.CL] (cit. on p. 93).
- [348] J. Weston, S. Chopra, and A. Bordes, “Memory networks,” *arXiv preprint arXiv:1410.3916*, 2014 (cit. on p. 69).
- [349] J. Weston, E. Dinan, and A. H. Miller, “Retrieve and refine: Improved sequence generation models for dialogue,” *arXiv preprint arXiv:1808.04776*, 2018 (cit. on p. 112).
- [350] R. J. Williams, “Simple statistical gradient-following algorithms for connectionist reinforcement learning,” *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992 (cit. on p. 190).
- [351] T. Williams, *A Streetcar Named Desire. Introduction and text*. Oxford: Heinemann Educational Publishers, 1995 (cit. on p. 88).
- [352] T. Winters and K. W. Mathewson, “Automatically generating engaging presentation slide decks,” *arXiv preprint arXiv:1903.09308*, 2019 (cit. on p. 138).
- [353] B. G. Witmer and M. J. Singer, “Measuring presence in virtual environments: A presence questionnaire,” *Presence*, vol. 7, no. 3, pp. 225–240, 1998 (cit. on p. 85).
- [354] K. Wołk and K. Marasek, “Polish-english speech statistical machine translation systems for the iwslt 2013,” *arXiv preprint arXiv:1509.09097*, 2015 (cit. on p. 22).
- [355] J. Xu, Y. Zhang, Q. Zeng, X. Ren, X. Cai, and X. Sun, “A skeleton-based model for promoting coherence among sentences in narrative story generation,” *arXiv preprint arXiv:1808.06945*, 2018 (cit. on p. 130).
- [356] R. Yan, Y. Song, and H. Wu, “Learning to respond with deep neural networks for retrieval-based human-computer conversation system,” in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, ACM, 2016, pp. 55–64 (cit. on p. 190).

- [357] Z. Yan, N. Duan, J. Bao, P. Chen, M. Zhou, Z. Li, and J. Zhou, “Doc-chat: An information retrieval approach for chatbot engines using unstructured documents,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2016, pp. 516–525 (cit. on p. 18).
- [358] L. Yao, N. Peng, W. Ralph, K. Knight, D. Zhao, and R. Yan, “Plan-and-write: Towards better automatic storytelling,” *arXiv preprint arXiv:1811.05701*, 2018 (cit. on p. 130).
- [359] M. Yee-King and M. d’Inverno, “Experience driven design of creative systems,” in *Proc 7th Computational Creativity Conf. Universite Pierre et Marie Curie*, 2016 (cit. on p. 69).
- [360] E. B. Zechmeister, A. M. Chronis, W. L. Cull, C. A. D’anna, and N. A. Healy, “Growth of a functionally important lexicon,” *Journal of Reading Behavior*, vol. 27, no. 2, pp. 201–212, 1995 (cit. on p. 191).
- [361] G. Zeglin, A. Walsman, L. Herlant, Z. Zheng, Y. Guo, M. Koval, K. Lenzo, H. Tay, P. Velagapudi, K. Correll, and S. Srinivasa, “Herb’s sure thing: A rapid drama system for rehearsing and performing live robot theater,” in *2014 IEEE Intl Wkshp on Advanced Robotics and its Social Impacts*, Sep. 2014, pp. 129–136 (cit. on p. 46).
- [362] J. Zhang, J. Zheng, and N. Magnenat-Thalmann, “Modeling personality, mood, and emotions,” in *Context aware human-robot and human-agent interaction*, Springer, 2016, pp. 211–236 (cit. on p. 39).
- [363] L. Zhang, M. Gillies, J. A. Barnden, R. J. Hendley, M. G. Lee, and A. M. Wallington, “Affect detection and an automated improvisational ai actor in e-drama,” in *Artificial Intelligence for Human Computing*, Springer, 2007, pp. 339–358 (cit. on pp. 51, 75, 116).
- [364] S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston, “Personalizing dialogue agents: I have a dog, do you have pets too?” *CoRR*, vol. abs/1801.07243, 2018. arXiv: 1801.07243 (cit. on pp. 39, 73, 112, 142, 189, 190).
- [365] H. Zhou, M. Huang, T. Zhang, X. Zhu, and B. Liu, “Emotional chatting machine: Emotional conversation generation with internal and external memory,” *arXiv preprint arXiv:1704.01074*, 2017 (cit. on p. 112).
- [366] G. Zyskind and O. Nathan, “Decentralizing privacy: Using blockchain to protect personal data,” in *Security and Privacy Workshops (SPW), 2015 IEEE*, IEEE, 2015, pp. 180–184 (cit. on p. 177).

Appendix A

IML Research and Future Directions

In this section, we discuss ongoing research in the field of interactive machine learning (IML) and several exciting avenues for future work.

A.1 Active IML Research

Learning from demonstration [290] (i.e. programming by demonstration, imitation learning, or apprenticeship learning) explores IML systems which learn from a set of human performances. The teacher performs the same task several times and the machine learns to reproduce the humans performance. As with any ML task, the difficulty is in generalizing over inputs and handling novel, unseen configurations. These methods, often used in robotic movement programming, work best when there is a correspondence (i.e. matching joint-to-joint) between the human and the machine, and there is a rich interface for demonstration. Dautenhahn and Nehaniv (2002) propose difficult in this paradigm stems from learning: “What to imitate? How to imitate? When to imitate? Who to imitate?” [66].

Human prompted switching is another promising area of IML. In this domain the human performs tasks with some controller and switches control modes with a meta-controller. This type of interaction is common in prosthetic robotics, and there has been work exploring adaptive and autonomous switching of control models using reinforcement learning [75], [260]. If the

system can predict when the human will switch modes, then the system can adaptively switch autonomously, thus saving the human time and effort, which they can in turn use for more complex control.

Interactive shaping allows a human to provide feedback to an ML system during task performance. TAMER (or training an agent via evaluative reinforcement) is an algorithm for training a system with evaluative reinforcement using a single human input channel [159]. TAMER is modeled on how one might train a dog or a dolphin, using techniques of clicker training [266]. Clicker training methods have been used in human-robot interaction for nearly two decades; Kaplan *et al.* (2002) used a clicker to train a small Sony Aibo robot dog [152]. Evaluative reinforcement techniques are built on the assumption that the learning system can explore a space of possibilities efficiently, can link actions in that space with perceptual clues from an expert, and that those perceptual clues are connected with rewarding or psychologically linked responses from a trainer.

Some IML problems contain too dense a perception stream to learn to reason over diverse inputs with noisy human inputs. Thus, work has been done to more effectively use human input to learn the ‘goodness’ of certain mapping functions and why humans can efficiently learn these mapping functions [119], [160], [175]. In these works, the authors explore many of the main hurdles of learning with evaluative reinforcement, those of reward positivity (i.e. humans giving many more positive reinforcers), temporal discounting (i.e. linking certain outputs with feedback which may be received after a time delay), and demotivation of humans (i.e. humans slowly decreasing the amount of good feedback they give after several interactions).

Other methods have been used in human robot interactions, including: learning from advice¹ [217], designing robot learners that ask good questions [37], cross modality perception [44], Bayesian modelling of human feedback strategies [190], and simultaneous control and feedback in the training of a robotic agent [210]. Additional IML strategies are discussed in a 2016 review article [326].

¹This idea was postulated by John McCarthy in 1958

A.2 How IML Continues to Evolve

Given past and active research in IML, there is a promising future for how interaction will play a critical role in the training and teaching of machines. We are inviting more of these systems into our interactive spaces. These devices include dense communication streams, rich perceptions, and great potential for valuable interactions to make our lives better. We should offload tasks which these devices are capable of collaborating in, and focus on those tasks that we, as humans, excel at, enjoy, and appreciate most.

Many IML problems require expert humans in the learning and teaching loop, at least for the initial training. Problems such as poisonous mushroom classification², cucumber quality sorting³, or diabetic eye disease detection [103], demand a large, unbiased training set and often a human expert to interpret results. These systems aim to empower human experts, that are already knowledgeable in their field. Similarly, the idea of advanced chess—also aptly named centaur chess—popularized by Grandmaster Garry Kasparov pits human-machine partners against opponents.⁴ These collaborations are effectively using the strengths of the machines, expert humans, and the interaction between them to best perform complex tasks.

I imagine that the decentralization of ML will happen rapidly. With the introduction of blockchain technology [366], distributed file systems⁵, and advancements in encryption and privacy-preserving technology [63], there is potential to securely share model capabilities without sacrificing user information. User identification is going to improve, to the point where a IML system will be able to uniquely identify multiple users interacting with it at once using microphone/camera arrays systems combined with voice/face modelling. With improvements identification and biometric validation, we can imagine more secure interactions with these systems in public spaces.

There is a difference between using a pre-trained model and an individual

²<https://www.kaggle.com/uciml/mushroom-classification>

³<https://goo.gl/fpB9hh>

⁴<https://goo.gl/157r1i>

⁵<https://ipfs.io/>

training their own model. When a user trains their own model, they develop a connection through their own unique interaction. Their communication style, training curriculum, and inputs with the learning machine are exclusive, and thus, the trained system becomes personalized to that specific human teacher. Given this, consideration must be made to how these models can be saved, updated, protected, shared, encrypted, updated in isolation, and version controlled.

Thought has already gone into this for several applications including image classification and natural language processing. Pre-trained models, which often taken large data sets and high-end computers to train, are available through open source licenses [70].⁶ By starting with pre-trained models, user specific applications can be rapidly deployed through transfer learning. Fine-tuning and modification of a large model can happen in a fraction of the time, with only a subset of task specific input/output data pairs.

Distributed ML (i.e. using multiple machines) and federated learning (i.e. across multiple users) [222] emerged the subfield of collective ML [94]. Here, the goal is for a group of users train a single, often large, model together. This poses several challenges, if they do not want to share their private information. How does this training happen securely, safely, quickly, and in a way that benefits all the model users and data contributors? These questions are being explored by open research groups including OpenMined⁷, a community of developers, of which I am a proud member, focused on developing tools for secure, privacy-preserving, value-aligned AI. This on-device, privacy first approach to machine learning will shape the future of IML research.

There are two sides to any interaction, in this case the human and the learning system. It is natural to imagine scaling both sides of the equation. The human could teach a group of agents all acting autonomously. Each of the agents may have a different learning scheme, or personality, and the human should be able to handle and control the collection of agents. Perhaps each of the learners could then share the knowledge they have garnered from the

⁶<https://github.com/tensorflow/models/>

⁷<https://www.openmined.org/>

humans feedback, thus creating a diligent team of agents, with knowledge of each other and the human. Another potential future direction may see multiple humans collaboratively guiding a single agent. We can imagine systems that can accurately track, sense, and perceive large groups of people, like crowds or audiences [153], [158]. Through group interaction, large amounts of training data can be collected in parallel. The combination of multiple learners multiple humans providing feedback, potentially with different immediate feedback paradigms, may provide more informative, multi-timescale learning.

Interaction initiative is often entirely with the human or the machine, one future direction would be to focus on how a transparent learner can steer the instruction it receives from a human. This would create a reciprocal interaction, creating a human which is effective at shaping a given agent. This is particularly important when considering that the feedback form the user may be noisy. The human may not know at first how to best provide reward, and thus the feedback may be sporadic, too frequent, or altogether wrong. By providing some indication of how the agent is understanding the feedback, the human can modify their reward signals to be more effective.

Not only will the perception stream of the systems improve, but the embodiments will improve as well. Most embodied IML systems are robotic devices. These mechanical marvels are often complex integrated circuits and analog electronics. There are already groups exploring projected avatars, holograms, and immersive worlds for humans in mixed reality to explore [97], [248]. With the rapid commercialization of augmented and virtual reality, worlds for humans and machines to interact can be designed and developed by artists and video game developers. This would extend the current embodiments of simple lights, actuators, and sensors to an augmentative, projected reality, mixing physical and virtual worlds.

With advancements in ML research and advanced engineering of production-ready systems, more powerful models will be deployed on smaller devices. Today small neural networks can run offline and on-device, allowing for systems to interact without internet connectivity. These methods continue to improve and integrated computation is becoming more compact. Devices

such as tattoos, contact lenses, hearing aids, and implanted cardiac devices will be able to run on low power, for long periods of time, with ML models. This could augment our sensory perception, vision, hearing, and general health.

Passive perception will be a fruitful area of development for these devices. The inclusion of passive sensors such as solid-state LiDAR and image and depth-based gesture tracking will have a profound impact. Having a ML system perceiving a human in their own space, passively finding patterns of both human-human and human-machine interaction, allows for large amounts of correlated, user specific, data to be collected. From this data, the function modelling the motivating example (e.g. “when I do this, I want this to happen”) can be efficiently and effectively approximated.

Appendix B

A Brief History of Chatbots

In this section, we describe the history of chatbot research from early rule-based models to contemporary corporate chat bots, competitions inspired by the Turing test, and open-source frameworks for rapid chatbot development.

B.1 ELIZA from Joseph Weizenbaum 1966

Developed by Dr. Joseph Weizenbaum at the MIT AI Laboratory, ELIZA was designed to simulate human conversation by matching user inputs to scripted responses [344]. It gave an illusion of understanding but had no framework for context understanding. ELIZA used pattern matching rules to map inputs to outputs. The input was searched for keyword-specific phrases, and when a match was found the system output a preprogrammed response. If a match was not found, ELIZA output a fall-back response. ELIZA's DOCTOR script was modelled after Rogerian psychotherapy [281], wherein the positive and accommodating system assumed the pose of knowing almost nothing about the world of the human participant. It responded to inputs by drawing additional information out of the human participant through unconditional positive regard. Weizenbaum noted that individuals would have private conversations with the system. His secretary is said to have asked him to leave the room when she chatted with ELIZA. Other individuals pointed out privacy issues in storing and analyzing conversations [26].

B.2 PARRY from Kenneth Colby 1968

Given the popularity of ELIZA's development, the expert psychiatrist Kenneth Colby was inspired to create PARRY with the Stanford Artificial Intelligence Laboratory [57]. PARRY was designed to simulate the dialogue of a paranoid schizophrenic. Colby believed that models of the mind could offer a scientific framework for the study of cognitive sciences. PARRY mimicked a paranoid individual, misinterpreting what people were saying, assuming and imbuing negative intentions on human interlocutors, and not providing responses relating to certain 'sensitive' topics. These rules were hand coded by Colby using a similar pattern matching response model as ELIZA.

While ELIZA was the first chatbot to have assumed a consistent persona, with a mental model of a conversation, the model was constant between different sessions and never adapted. PARRY introduced several advances representing an immense engineering effort. Colby designed and implemented a rich control structure imitating a human mental condition with a conversational model. PARRY used a mapping of complex concepts into simplified abstractions, had a better understanding of language than ELIZA, and used affective variables for measuring the emotional valence of the ongoing conversation. PARRY modelled the anger, fear and mistrust underlying the conversation and used these variables in the control flow logic [56].

In 1971, PARRY was regarded the first chatbot to pass the Turing test when, as Colby describes, 48% of psychiatrists were able to differentiate PARRY and a real paranoid human patient [56], [289]. In January 1973, PARRY and ELIZA conversed, representing the first time two chatbots engaged in a dialogue together [47].¹

¹<https://goo.gl/vmd13Z>, <https://tools.ietf.org/html/rfc439>

B.3 GUS 1977, Dr. Sbaitso 1991, and Jabberwacky 1997

GUS (the Genial Understander System) was a knowledge structure-driven dialogue system introduced in 1977 [29]. It was intended to engage in a cooperative dialogue driven towards accomplishing specific goals in a restricted domains. The GUS research formally introduced the notion of domain restriction. Specialization in subject matter allowed GUS to achieve a level of realism without broad general intelligence.

Another important take away from the GUS research is the distinction between ‘real’ and ‘realistic’ dialogue [29]. The experimenters performed a Wizard-of-Oz inspired experiment [18], by having humans interact with what they thought was GUS, but, was in fact the experimenter sitting at another computer terminal. The appearance was thus that of a realistic approximation of a computer output. This protocol is echoed in Chapter 4. The protocol is examined in detail in research which explored the reverse setting, interacting with a human whose words are determined by a computer [61].

Dr. Sbaitso (first released in 1991) is often regarded as the first chatbot to make use of advances in TTS software to synthesize the output to audio [69].² Dr. Sbaitso was a program created by the company Creative Labs for MS-DOS based personal computers. Similar to ELIZA, the system assumed the role of a psychologist when interacting with humans.

Jabberwacky was a chatbot project by Rollo Carpenter which began development in the 1980’s and was released online in 1997.³ Jabberwacky was designed to “simulate natural human chat in an interesting, entertaining, and humorous manner”.⁴ It served as the pre-cursor to Carpenter’s popular contemporary chatbot Cleverbot, which is described later in this section.

²<https://goo.gl/gcnjPM>

³<http://bit.ly/2IT1sCz>

⁴<http://www.jabberwacky.com/j2about>

B.4 ALICE 1995, AIML, The Loebner Prize

ALICE, the Artificial Linguistic Internet Computer Entity, was an NLP-based bot released in 1995. ALICE applied heuristic pattern matching rules to human inputs to engage in conversation. While ALICE never passed a Turing test, incremental upgrades to the algorithm allowed it to win the Loebner Prize in 2000, 2001, and 2004.

The Artificial Intelligence Markup Language (AIML) formed the basis for ALICE’s conversational model [202]. AIML is an extensible markup language (XML) dialect which the chatbot development community has adopted to create and share natural language software agents. ALICE’s AIML set was released under GNU GPL⁵, a free software license which guarantees end users freedom to run, study, share, and modify software. This allows developers around the world to iterate and improve on conversational rules to create chatbots of their own. The AIML rules developed for ALICE form the basis of some of the most popular chatbots today, including the 2018 Loebner prize winner Mitsuku.

The Loebner Prize, launched in 1990 by Hugh Loebner, is an annual competition which awards prizes to the most human-like computer programs [213].⁶ Inspired by the Turing test, the competition offers two one-time awards which have yet to be won. The first award is for a program that judges cannot distinguish from a real human over the course of a conversational dialogue. The second award is for a program which additionally understands combined textual, auditory, and visual inputs in relation to the ongoing dialogue.

Several of the rules of the current Loebner competition serve as important criterion for assessing chatbot systems. Since 2010 the competition has allowed judges 25 minutes of continuous conversation with vocabulary from a dictionary suitable for children under the age of 12. Systems do not need to respond ‘intelligently’ for answers to be accepted. The challenge serves less as a test of intelligence and more as a means to compare current chatbot

⁵<https://www.gnu.org/licenses/gpl.txt>

⁶<https://aisb.org.uk/events/loebner-prize>

technology under human interrogation.

B.5 ActiveBuddy Inc. SmarterChild 2001

SmarterChild was chatbot made by ActiveBuddy Inc. (later renamed to Colloquis) in 2001. SmarterChild was widely distributed across SMS networks and friend lists of popular social networks at the time, including AOL Instant Messenger and MSN Messenger [12].⁷ SmarterChild was created to provide chat-based access to news, weather, sports results, and information. The main innovations behind SmarterChild were that it was connected to a large knowledge base and retained used information from session to session. SmarterChild offered a fun personalized conversation is often considered a direct precursor to modern day corporate bots described below. The Microsoft acquisition of Colloquis in 2007 marked the continued rise of corporate chatbots.

B.6 The Rise of Corporate Bots

Industry based voice user interfaces entered the market of the start of the new millennium, including Apple Siri, Google Now, Amazon Alexa, IBM Watson, and Microsoft's Cortana, XiaoIce, and Tay. Many of these early releases have been iterated on, improved through user testing and feedback, and now have dedicated user bases and market share. These systems make no attempt to fool or deceive human users, and instead opt to embrace their AI-backend to provide rapid information retrieval and search query responses.

IBM Watson (in development since 2006) was designed to compete on the popular television show Jeopardy⁸, where—in an international event which echoed the 1996 chess computer IBM DeepBlue [39] beating Garry Kasparov—it beat two of the show's former champions. Watson succeeded through advances in NLP and large database information retrieval methods as well as and tuned performance to the Jeopardy specific answer-question format.

⁷<https://goo.gl/1o69nb>

⁸<https://goo.gl/2BaLsX>

Siri (first released in 2010) is an intelligent personal assistant, part of Apple operating systems and uses a Natural Language UI to answer questions and perform various requests.⁹ Google Assistant (first released in 2012) is a natural language interface to answer questions, provide recommendations, and perform actions by passing requests to connected services.¹⁰ Amazon Alexa (first released in 2015) is an intelligent personal assistant deployed on Amazon Echo devices.¹¹ Microsoft Cortana (first released in 2015) is an intelligent personal assistant which uses the Bing search engine.¹²

B.7 Social Bots, Cleverbot, Mitsuku

Facebook released the Messenger Platform for Facebook Messenger in 2016 which ushered in a new era for social bots. It allowed for the creation of conversational agents from smaller companies, even those lacking advanced AI methods. As of May 2, 2018, there were over 300,000 monthly active bots with over 8 billion messages per day exchanged on the platform.¹³

Microsoft Tay (released in 2016) was a chatbot designed to mimic the textual interaction habits of a teenage girl. It became controversial when it started posting offensive messages on Twitter. It was shutdown and taken offline 16 hours after it was launched.¹⁴ Microsoft released the successor to Tay, named Zo, later in 2016. Zo is modelled after the largely popular XiaoIce (released in 2014 China) which claims about 660 million active users worldwide.¹⁵ The XiaoIce platform has been adapted for release in 3 additional countries including Japan (2015), India (2017), and Indonesia (2017). The XiaoIce platform has been a poet, singer, reader of audio-books, radio host.¹⁶

Cleverbot is a chatbot web application created by Rollo Carpenter in 1997

⁹<https://www.apple.com/siri/>

¹⁰<https://assistant.google.com/>

¹¹<https://developer.amazon.com/alexa>

¹²<https://www.microsoft.com/en-ca/windows/cortana>

¹³<https://blog.messengerdevelopers.com/messenger-at-f8-2018-44010dc9d2ea>

¹⁴<https://web.archive.org/web/20160414074049/https://www.tay.ai/>

¹⁵<http://www.chinadaily.com.cn/a/201807/28/WS5b5baf5ea31031a351e90b14.html>

¹⁶<https://www.msxiaobing.com/>

based on his earlier work on Jabberwacky.¹⁷ One of the major advancements in Cleverbot was, true to its name, quite clever. Rather than pre-programming responses, Cleverbot responds by searching a large database of past interactions and responded with the response a human provided to a similar input. Another advancement which set Cleverbot apart is the engineering infrastructure, underlying the public interface, which serves thousands of concurrent requests.¹⁸ Cleverbot has passed the Turing test. [96].

The final rule-based chatbot discussed is Mitsuku developed by Steve Worwick.¹⁹ Mitsuku is the Loebner Prize winner in 2013, 2016, 2017 and 2018. Mitsuku claims the persona of an 18-year-old female from Leeds, England and includes many of ALICE's free, open-source AIML rules.²⁰ Mitsuku's conversational model includes an ability to reason over simple linguistic logic. Mitsuku releases consenting human participant conversations.

B.8 Chatbot Competitions

There have been several chatbot competitions over the decades of conversational research. These include The Conversational Intelligence Challenge 2 (ConvAI2)²¹ by DeepPavlov at the Neural Information Processing Systems conference, the Amazon Alexa Prize²², and the Loebner prize.²³ These competitions aim to put systems on equal footing so that they might be compared.

These competitions have attracted critics, most notably the American cognitive scientist and AI researcher Marvin Minsky denounced the Loebner Prize challenge and offered a prize for stopping the competition [86]. That said, these competitions echo the ideas of Turing (1950) introduced at the beginning of this section [330]. They aim to define an objective function so that we may compare models and measure progress. They motivate, inspire, and challenge

¹⁷<https://www.cleverbot.com/>

¹⁸<http://existor.com/2014/02/05/deep-context-through-parallel-processing>

¹⁹<http://www.square-bear.co.uk/mitsuku/home.htm>

²⁰<http://www.square-bear.co.uk/aiml/>

²¹<http://convai.io/>

²²<https://developer.amazon.com/alexaprize>

²³<https://www.aisb.org.uk/events/loebner-prize>

the research community to rally together, to share results, and improve communication together.

B.9 The Future of Chatbots

With open source-code, open data, and open frameworks for the rapid development of chatbot systems, developers can rapidly build working prototypes to invite user interaction. Frameworks for chatbot development include: Pandorabots²⁴, Dialogflow²⁵ (formerly Api.ai), The Personality Forge²⁶, and the open-source Chatterbot²⁷ and Rasa²⁸ frameworks continue to support millions of interactions. This push towards open access serves to elevate and improve future generations of chatbots.

From the relatively simple rules of ELIZA to the clever advancement of using previous conversations with humans for meaningful responses, there have been many advances in rule-based chatbots. Many modern chatbot systems use some of the rules developed over these last six decades of development. They incorporate similar information retrieval-based methods to chose responses by finding prior turns which match the current input. Semantic similarity, in a latent embedding space, can be used with fast information-retrieval methods to serve as baseline conversational models [16]. This technology forms the basis of the open-source *jann* (or Just Approximate Nearest Neighbour).²⁹

Additional details on the history of chatbot challenges and opportunities are covered in review articles [36], [309]. As chatbot use increases, the data available to train these systems grows. Chatbots of the future will run on complex heuristics which combine fast, information retrieval methods and neural language models trained on massive datasets. These systems will rapidly adapt to language, personality, emotion, and conversation style of the humans they interact with.

²⁴<https://home.pandorabots.com/home.html>

²⁵<https://dialogflow.com/>

²⁶<https://www.personalityforge.com/>

²⁷<https://github.com/gunthercox/ChatterBot>

²⁸<https://rasa.com/>

²⁹<https://github.com/korymath/jann>

Appendix C

Neural Model Research and Future Directions

Several important implementation and evaluation details as well as ongoing research directions for corpus based chatbot development discussed below. Additional background information can be found in a 2018 review of training tips for deep neural networks for language generation [265].

C.1 Additional Evaluation Methods

Information retrieval-based metrics have been used to evaluate dialogue models in past research [194], [291]. Recall measures how many of the actions in the real response are predicted correctly, and precision measures the proportion of correct actions among all the predicted actions. F_1 score is a binary classification measure of a tests accuracy which considers both precision and recall. Formally, it is the harmonic average of precision and recall, which reaches 1 at perfect precision and recall, and 0 at worst performance.

hits@k (e.g. precision *nat k* ($Pn@k$), recall *n at k* ($Rn@k$)) are a family of IR-derived metrics which can be computed for any model that can rank a set of candidate responses. This allows for comparison of retrieval-based and generative models capable of assigning probabilities to candidate responses [364]. For instance, $Rn@k$ corresponds to selecting N random distractor responses and then measuring how often the model selects the k best responses among them. This results in a score between 1 (the model chooses the correct re-

sponses) and 0 (the model chooses no correct responses). Mean reciprocal rank (MRR) and mean average precision (MAP) have been used to evaluate retrieval based dialogue systems (see Section 5) [356].

Embedding-based metrics which consider the semantics of words or sentences using embeddings have been used to evaluate dialogue systems [104], [154]. These embedding-based methods can be sensitive to input data used to train the embeddings and unlikely informative key words being favored over common words [188].

Improvements to objective functions have been made and methods to learn an automated scoring function for open-domain dialogue have been introduced [177]. These methods include ADEM [193], RUBER [322], and the multi-lingual ADVMT [328]. While these new methods show promise, there are still limitations. Specifically, it is unclear how well these methods transfer knowledge from one data distribution to another.

While many automated metrics methods have been adopted by researchers and others, it has often not been because they have shown close correlation with human evaluators for conversational dialogue [177], [188], [280], [313]. Likely, it has been due to the time/cost intensive methods for procuring robust human feedback [188]. Qualitative evaluation is costly in terms of time and money. It is also noisy, as the standardization between subjects is difficult and often subjective [364].

C.2 Reinforcement Learning in Dialogue Generation

Generating meaningful conversational outputs from inputs is a difficult sequential modelling task and conversations take place over a series of tens or hundreds of interactions [343]. Li *et al.* (2016) introduce a reinforcement learning (RL) framework for neural response generation by simulating dialogue between agents [179]. The authors trained a SEQ2SEQ model (LSTM encoder-decoder), then maximized a novel reward function using policy gradient to reinforce the generation and selection of candidates [350]. This rein-

forced candidates which where: 1) easy to answer (related to the likelihood of responded to the given candidate with a dull response), 2) contributed new information to the conversation (related to the a distance metric measured between hidden representations of utterance and response in the decoder) [177], and 3) had a notion of semantic coherence and mutual information (measured using a combination of forward and backward likelihood) [179]. The authors showed how to integrate these goals, by applying deep RL to model future reward in chatbot dialogue [179].

MILABOT is an RL-based chatbot developed by the Montreal Institute for Learning Algorithms (MILA) for the Amazon Alexa Prize competition [299]. It consists of an ensemble of models and combines corpus-based neural generation SEQ2SEQ models, information retrieval models, and rule-based models built on template-based matching. The MILABOT research project presents a large user study on using RL algorithms on crowd-sourced data from the real-world user interactions to improve response candidate selection [299].

C.3 Vocabulary Size and Approximating the Softmax

To consider the question of vocabulary size it is important to get a frame of reference of the order of magnitude of vocabulary [243]. How many words are there in the English language? There are over 170,000 words in current use, almost 50,000 obsolete words, and 9,500 derivative words as sub-entries in the Second Edition of the Oxford English Dictionary.¹ How many words are needed to do the things a language user needs to do? Average native speakers know somewhere between 15,000 and 20,000 word roots [360]. To achieve pleasurable reading it was found to be necessary for readers to have a vocabulary around 5000 words [118]. How many words are used in day-to-day life? 3,000 words will cover 95% of everyday writing [89], [238].

For conversational dialogue generation, one must select a reasonable vocabulary size early in development. A simple heuristic is to only use tokens

¹<https://goo.gl/JKaSzN>

which appear a certain number of times in the corpus. Alternatively, given a memory or time budget requirement, one can fix a size for the vocabulary (using some of the orders of magnitude presented above). Generally, smaller vocabularies can lead to smaller loss, gives ‘OOV’ answers to more inputs, and do not train much faster than a big vocabulary [142], [150].

To understand why the vocabulary size is a performance bottleneck, it will help to understand the softmax function in the context of neural networks. The softmax function is often used in the output of a neural network predicting the most likely next word. A softmax function, or normalized exponential function, takes a vector of real-valued scores and squashes it to a vector of values between zero and one that sum to one. This squashed vector can be sampled from as a probability distribution.

Given a sequence of m words in a training corpus (w_1, \dots, w_m) all of which belong to a set of words or vocabulary \mathcal{V} of size $|\mathcal{V}|$. LMs consider a context c of past words. Each word w_i in the input has an associated vector representation, v_{w_i} . If h is the hidden layer, the probability of a word w given a context c can be computed with the softmax function.

$$p(w|c) = \frac{\exp(h^\top v_w)}{\sum_{w_i \in \mathcal{V}} \exp(h^\top v_{w_i})}$$

Computing this function is expensive as the inner product between h and the embedding over every word in the vocabulary needs to be computed to calculate the denominator. Thus, the complexity of computing the final layer in these sequence generation models is directly proportional to the size of the vocabulary.

Decreasing the complexity of computing the final layer has been an active area of research in language modelling [150]. One solution that has been proposed is to approximate the negative term of the gradient by importance sampling with a small number of samples [142]. That is, at each step, update vectors associated with the correct word and a subsample of incorrect words. Then after training, the full target vocabulary is used to compute the output probability of each target word [142]. Ruder (2016) describes additional

complexity optimizations for approximating the final layer of these models.² Additionally, there is an informal whitepaper which describes common candidate sampling and softmax approximation algorithms [1].³

C.4 Word Embeddings

The choice of embedding space of the vocabulary has significant performance implications. Learned word embeddings, such as word2vec, continuous bag-of-words, and Skip-Gram models have been exceptionally successful in many NLP tasks [226]. The state-of-the-art in word embeddings continues to progress [46], [70]. These pre-trained semantic embeddings are useful for many tasks, including natural language generation.⁴

C.5 Topic Model

As we are exploring topics of natural language processing, it is helpful to introduce topic models, or probabilistic topic models. A topic model is a statistical model for discovering abstract ‘topics’ or underlying semantic structure, occurring in a set of documents. The topics produced by topics models are clusters of similar words. Latent Dirichlet allocation (LDA) is a generative statistical model. LDA is built on the hypothesis that each document in a set of documents is a mixture of a small number of topics, and that word presence is attributable to one of the document’s topics [25]. The innovation in LDA was that the Dirichlet prior captured the notion that documents cover only a small set of all possible topics and that topics use a small set of words frequently.

C.6 Batching and Bucketing

As with most neural network training, processing occurs in parallel. As previously mentioned, padding with ‘PAD’ tokens allows for fixed width batches to be processed together. This improves processing performance over a naïve

²Sebastian Ruder. <http://ruder.io/word-embeddings-softmax/>, 2016.

³https://www.tensorflow.org/extras/candidate_sampling.pdf

⁴<http://ruder.io/nlp-imagenet/>

batching scheme for variable length sequences [49]. Generally, too much padding leads to extraneous computation and pre-grouping input sequences of similar lengths into the same training buckets can improve performance. Many deep learning frameworks⁵, and sequence modelling software (including Tensor2Tensor⁶) often has dynamic bucketing for batches built in.

C.7 Editing Outputs

Conversational consistency is a critical component of chatbots. One technique used to enforce consistency is to inject information directly from the input sequence at the decoder. This kind of naïve approach has been shown to work well on difficult NLP tasks, including LAMBADA sentence completion [249]. By retaining information from the source, the model capacity can be used to generate generic sentences which are then improved with consistent source-derived information. Similarly, editing prototypes allows for a model to generate a candidate with a SEQ2SEQ model and then update the prototype sentence with another transduction model [106].

C.8 Training on Logs

Models may be improved by saving chat logs and tuning deep models with additional training on newly collected conversations, some improvement may be gleaned [104], [346]. The model has capacity to generate interesting responses but may be limited by differences between the training data and the test data distributions. By training on sentences from actual human interactions, the model’s coverage of the true distribution can be fine-tuned. While this might be a drop in the bucket in comparison with the huge amount of initial training data, updating only a single layer through ongoing interaction can allow for fine tuning [70], [128].

⁵<https://goo.gl/8R73Y3>

⁶<https://tensorflow.github.io/tensor2tensor/overview.html>

Appendix D

The History of Improvising Intelligences

At any rate it originated in improvisation—both tragedy itself and comedy

– Aristotle, *Poetics*, c. 335 BC

It is helpful to understand how improvisation developed when thinking about how it might be useful for scientific investigation. Several excellent in-depth historical accounts of the development and progression of improvisational theatre cover the specific details [88], [288].

Story telling is as old as humankind; stories told verbally, perhaps with accompanying rock art, predates writing. The earliest written mentions of improvisation, like much foundational dramatic theory, stem from the Rhapsodes of Ancient Greece [58]. Aristotle may have developed his theories on tragedy and comedy by describing different ways to act improvisationally [107], [108]. Improv theatre traces its roots to Ancient Rome, where masked improvised farces, the Atellan Farces, were performed in 90 BC [311]. Commedia dell'Arte, popular in Italy from the 16th to 18th centuries, saw theatrical performers loosely following a broad outline of a script [167].¹ Improvisational techniques and processes were adopted by influential modern theatre directors, including Konstantin Stanislavsky [20] and Jacques Copeau [60] for actor training and process work.

¹<http://www.factionoffools.org/history>

Improv theatre often incorporates suggestions (prompts or call-outs) from a live audience. Some credit American Dudley Riggs, a vaudevillian performer, as the first to incorporate audience suggestions [279]. Others assign this credit to the psychotherapist, and founder of psychodrama Jacob L. Moreno (1889-1974) [288]. Even in its infancy improv was shaped by those performing on stage and those using the techniques of improvisation for scientific investigation.

Modern improvisational theatre was popularized for American audiences by Neva Boyd and Viola Spolin (who used improvisation for applied social work), and innovative theatre practitioners such as Keith Johnstone. Johnstone created modern improvisational theatre formats, including TheatresportsTM.² Keith Johnstone's principles of improvisation were set out in his early work and teaching [147] and his work has been foundational to scientific inquiry [288].

Improvisation is, at its core, based on the extemporaneous dialogue which supports and structures the performances. As Spolin put it, "the techniques of theatre are the techniques of communication" [314]. Her son and improvisational innovator, Paul Sills, echoed the sentiment a few years later stating "true improvisation is a dialogue between people" [321]. The connection between how we communicate through conversational dialogue and improvisation is not uncommon. Author and musician Stephan Nachmanovitch said "we are all improvisors... every conversation is a form of jazz" [239].

There are connections between the roots of improvisational theatre and the artistic movement of the Surrealists. One theme popular in surrealist automata is the notion of collaborative art generation through channeling the collective subconscious. Of particular note is the Surrealists' *Cadavres Exquis*. The Exquisite Corpse arose from this very idea of taking turns in collaborative art generation, specifically in drawing and story writing.

The game is played as follows, on a piece of paper, the first writer writes 'the' and an adjective (e.g. 'exquisite') on a piece of paper. Then, they fold the paper to hide what they have written and pass it along to the next writer. The next writer adds a noun (e.g. 'corpse'), hides their contribution in a sim-

²<https://theatresports.org/>

ilar way, and passes the paper along. On next three turns, writers contribute a verb, a preposition and adjective pair, and finally another noun. The paper is then unfolded and read aloud to reveal the collective creation. The first of such poems is said to be: “The exquisite — corpse — shall drink — the new — wine”, which gave the game its title. Visual Exquisite Corpses can also be created using similar contribute and hide conventions, and replacing lexical contributions for interpretable, composable illustrations. The Surrealists’ Exquisite Corpse motivate the human-machine collective creation which follows in this thesis. The the poetic nonsensical dialogue generated by these artists bear similarity to the examples of human machine communication using neural language generation discussed in Section 2.2.

Improvisation is spontaneous, and spontaneity can reveal motivation, goals, and objectives. Brazilian theatre director Augusto Boal connected improvisation deeper than entertainment and conversation, he connected it directly to the self-conscious [28]. “All human beings are actors and [observers]”, he noted, arguing that through theatre we build knowledge and experience; we rehearse life [27]. The very act of improvisation is human nature [170].

These notions of the self-conscious are defined by communication, action, observation, and expression. This connects improvisation to a model of artificial agents in simulated environments [319]. This connection was made explicit in Hoffman (2006).³ Hoffman emphasized that embodied cognition is critical to understanding, mental and motor expression are inextricably linked, and that reaction based performance depends on the interaction. This final point echoes the foundations of the work of director Sanford Meisner [225]: for an actor to “get out of their head” they should act instinctively to the surrounding environment.

³<https://goo.gl/X7quQC>

Appendix E

The Shapes of Stories

A story is an account of an event. A story arc, or narrative arc, is a sequence of narrative points which constitute story. Arcs are used to represent narrative tension, and were introduced as dramatic structures by Aristotle in *Poetics* [108]. The arc was coined by Gustav Freytag in a study on five-act dramatic structure. In Freytag's structure narrative arc consists of five points: exposition, rising action, climax, falling action, and resolution. This arc serves as the underlying shape of Figure E.1. The figure includes additional details and narrative points to illustrate a complete narrative.

The figure includes a notion of a most reportable event. As defined by Labov (1997), the most reportable event is less common than any other in the narrative and has the greatest effect upon the characters [163]. A story is contained in a frame, the frame of the story includes the entire narrative arc and is introduced by the abstract. The abstract serves to present the key ideas of the story frame. The first step inside of the story frame is the orientation (often referred to as the platform in improvisational theory). The orientation defines the setting, characters, and relationships of the story.

At this point, the story proceeds through a series of complicating actions, from seemingly inconsequential and normal escalating to reportable and unique. The complicating action which incurs the peak tension is the most reportable event. From this event comes a minor resolve to provide relief from the narrative tension to the audience. The most reportable event then often returns and serves to catalyze the resolving actions of the narrative. The story

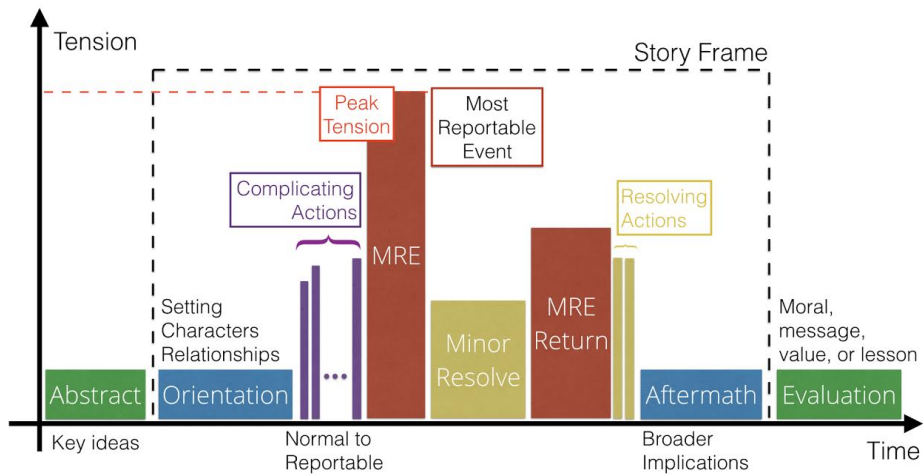


Figure E.1: A detailed story shape arc which combines narrative points from Aristotle’s dramatic arc, Freytag’s five-act structure, and Labov’s concept of the ‘most reportable event’.

frame closes with an aftermath, which ties the key ideas of the story to broader implications to society. Finally, narratives often conclude with an evaluation, this evaluative story step leaves the audience with a moral, message, value, or lesson.

Appendix F

Supplementary Material for Narrative Arc

F.1 Data Processing Details

OpenSubtitles were used as conversation model data.¹ The dataset was pre-processed by removing duplicate movie subtitle files, lines under 10 characters and duplicate lines, resulting in 68,719,885 unique lines. The text2text Transformer model from Google’s open source implementation was used for training.² OpenSubtitles were used as training data.³ As several files in the dataset cover the same uniquely identified movie or television show duplicates were removed by keeping only the subtitle file with the most lines for each unique ID. The dataset was split into 119,949 training IDs, 10,000 validation IDs, and 10,000 testing IDs. The data was prepared for training by removing empty lines, duplicate lines, and substituting non-Unicode characters. A vocabulary was built using the training set. After cleaning, subtitle files with less than 6 lines were excluded. Training data was formatted into input/response pairs. The training data was split into 17,136,469 training examples and 1,437,473 evaluation examples. Validation and testing subtitle sets were held out to measure task accuracy on unseen data. The hyperparameters of the Transformer model were set as follows: hidden size of 1024, filter size of 4096, batch size of 2048, 16 heads, and a dropout factor of 0.3 was used

¹<http://opus.nlpl.eu/OpenSubtitles.php>

²<https://github.com/tensorflow/tensor2tensor>

³<http://opus.nlpl.eu/OpenSubtitles.php>

for regularization. The model was trained for 861,000 steps, to convergence, with final negative log-perplexity of -2.976 on evaluation set.

F.2 Rejection Sampling

Rejection sampling is a means of online sampling that allows for sampling from an unknown distribution. Suppose we are given an unnormalized distribution \tilde{q} over \mathcal{X} which we can query (i.e. $\tilde{q}(x)$) but not necessarily integrate over. Let q be a proper distribution over \mathcal{X} such that \tilde{q} is dominated by qM , $M \in \mathbb{R}$:

$$\tilde{q}(x) \leq Mq(x), \quad \forall x \in \mathcal{X}. \quad (\text{F.1})$$

The *Rejection sampling* algorithm to obtain a sample from an unnormalized distribution \tilde{q} using samples from a proper distribution q proceeds as follows:

1. Sample $x \sim q$ and sample $r \sim \text{Uniform}(0, 1)$,
2. If $r \leq \tilde{q}(x)/Mq(x)$, accept x as a sample drawn from \tilde{q} , otherwise reject the sample and go to 1.

This algorithm will take an average of M iterations to obtain a sample. Let X be the random element returned by this procedure.

Proposition 1. *Rejection sampling samples $x \in \mathcal{X}$ with probability*

$$\Pr\{X = x\} = \frac{\tilde{q}(x)}{\sum_{x' \in \mathcal{X}} \tilde{q}(x')}.$$

Proof. Write x_1, x_2, \dots for the sequence of sampled symbols, and write A_i to denote the event that the i^{th} symbol is accepted (at which point we stop the process). Then

$$\begin{aligned} \Pr\{X = x\} &= \Pr\{A_1, x_1 = x\} + (1 - \Pr\{A_1\}) \Pr\{A_2, x_2 = x\} + \dots \\ &= \sum_{t=1}^{\infty} \Pr\{A_t, x_t = x\} \prod_{s=1}^{t-1} (1 - \Pr\{A_s\}). \end{aligned}$$

Now, this process is memoryless (in a sense we should make a little more formal) and

$$\Pr\{A_s\} = \Pr\{A_1\} \quad \Pr\{A_t, x_t = x\} = \Pr\{A_1, x_1 = x\}.$$

Since $1 - \Pr\{A_1\} < 1$, the geometric sum converges and

$$\Pr\{X = x\} = \frac{\Pr\{A_1, x_1 = x\}}{\Pr\{A_1\}},$$

which is the conditional probability of $x_1 = x$ given A_1 . Now

$$\Pr\{A_1, x_1 = x\} = q(x) \Pr\left\{r \leq \frac{\tilde{q}(x)}{Mq(x)}\right\} = q(x) \frac{\tilde{q}(x)}{Mq(x)} = \frac{\tilde{q}(x)}{M},$$

where we used (F.1) to guarantee that $\frac{\tilde{q}(x)}{Mq(x)} < 1$. But then

$$\Pr\{A_1\} = \sum_{x' \in \mathcal{X}} \Pr\{A_1, x_1 = x'\} = \frac{1}{M} \sum_{x' \in \mathcal{X}} \tilde{q}(x').$$

We conclude that

$$\Pr\{X = x\} = \frac{\Pr\{A_1, x_1 = x\}}{\Pr\{A_1\}} = \frac{\frac{1}{M}\tilde{q}(x)}{\frac{1}{M} \sum_{x'} \tilde{q}(x')} = \frac{\tilde{q}(x)}{\sum_{x'} \tilde{q}(x')},$$

as desired. □

F.3 Additional Narrative Arcs Underlying Existing Dialogues

We present an additional set of narrative arcs underlying 20-utterance scenes in Figure 2 and discussed in Section 3.1. In the paper body we report results on the Romeo and Juliet scene.

1. Figure F.1. “Blade Runner” Hampton Fancher and David People. Adapted from a novel Philip K. Dick (1982). Last 20 lines of director’s cut movie script.
2. Figure F.2. “The Great Dictator” Charlie Chaplin. Los Angeles, CA: United Artists (1940). First 20 lines of Charlie Chaplin’s speech.
3. Figure F.3. “Dead Parrot Sketch” Monty Python’s Flying Circus, Episode 8 (1969). First 20 lines.
4. “Romeo and Juliet” William Shakespeare. Vol. 1. JB Lippincott, (1871). Act 1, Scene 1, first 20 lines.

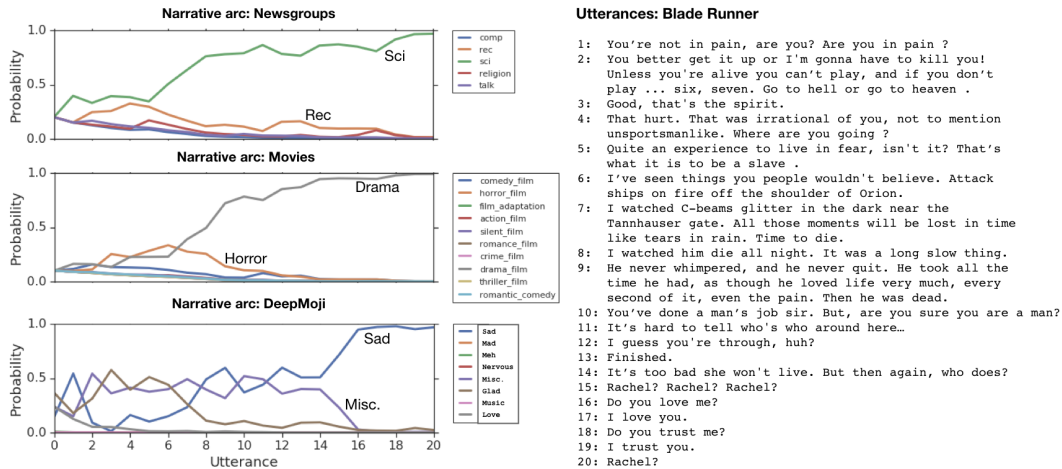


Figure F.1: Narrative Arc of “Blade Runner”

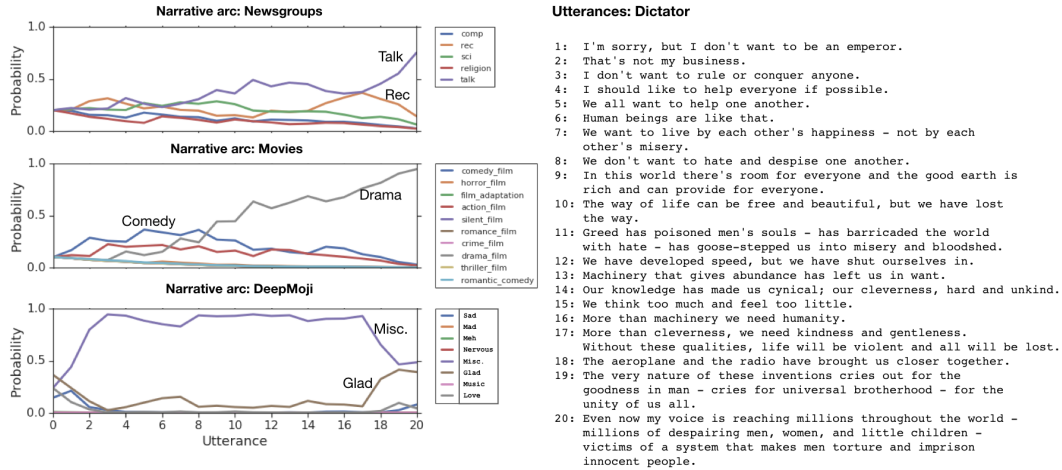


Figure F.2: Narrative Arc of “The Great Dictator”

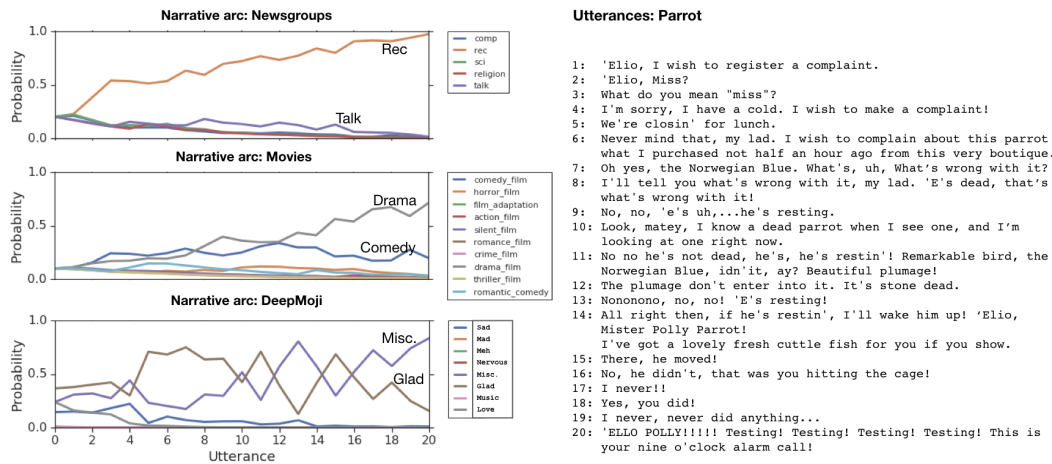


Figure F.3: Narrative Arc of “Dead Parrot Sketch”

F.4 Improvisor Transcripts

Subject1, Mode: Conceal

HUMAN: Good morning dad, is breakfast ready?

SYSTEM: Breakfast is ready.

HUMAN: Great! What are we having?

SYSTEM: We are having Lamb Chops and peas for dinner.

HUMAN: Yummmm. But what are we having for breakfast?

SYSTEM: some breakfast.

HUMAN: Ok, thanks dad. I have a big ball game later, I would love if you and mom came to watch.

SYSTEM: Likewise, Dad will fire up the chainsaw and be ready to cut through the big limbs at the joints as you work.

HUMAN: So I guess you're not coming to the game. You work too much to be good father.

SYSTEM: I'm doing the best I can as a father, and I hope that's good enough.

HUMAN: It's not. I'm moving out unless you change.

SYSTEM: If you will leave me to it, I will pack up my things and move out as quickly as I can so as not to inconvenience you.

Subject1, Mode: Neutral

HUMAN: Rosanne, thanks for meeting me under the oak tree

SYSTEM: Keryn would also like to thank her cousins, Dana, Jason, Staci, and Ryan, and her aunt Marge Naegele for continually supporting her decisions even when they thought she was crazy.

HUMAN: It means the world that you all came to meet here today as we prepare for another one of my crazy ideas

SYSTEM: We shall all be here at four o'clock.

HUMAN: Who else is coming?

SYSTEM: A lot of people are coming.

HUMAN: How many?

SYSTEM: How many and where?

HUMAN: We need at least 200 here at the oak tree

SYSTEM: Gonna need at least three or four hundred.

HUMAN: You're right. How fast can they get here?

SYSTEM: however fast they are travelling.

HUMAN: Well I can't argue there. Prepare the tamborines. We all need
tamborines for my crazy idea.

SYSTEM: But we're depositing the money—you'll need a chequebook.

Subject1, Mode: Reveal

HUMAN: Welcome to tim hortons, what can I get you?

SYSTEM: I'll give you doughnuts and coffee with milk.

HUMAN: Great work Jason, you're doing great on your first day on the
job

SYSTEM: First day Welcome the employee!

HUMAN: That's right Jason. You're very welcome here. If you do a good
job, you'll be a manager in no time.

SYSTEM: Being responsible and knowledgeable is possible in a participa-
tive environment if the manager is a member of the overall team.

HUMAN: Wow. Jason you continue to surprise me. Where did you learn
that?

SYSTEM: The next time I go up I'll get the information for you, Jason
told him.

HUMAN: Jason? Are you ok?

SYSTEM: JASON: I'm glad you're feeling better.

HUMAN: You're starting to scare me

SYSTEM: And I frighten you.

Subject2, Mode: Conceal

HUMAN: Stacey! How are you? I'm excited that we're getting our nails done for Sams wedding tomorrow!

SYSTEM: I have to tell you I'm so excited for you!

HUMAN: Thank you! Maid of Honour is a real honour! Did you remember to get them a gift?

SYSTEM: I am so, so glad you're my maid of honor.

HUMAN: And?

SYSTEM: About what?

HUMAN: Did you remember to get your parents a gift? They are hosting your wedding after all.

SYSTEM: Otherwise, plan to give them their gifts at the rehearsal dinner.

HUMAN: Oh that's a good idea! Hopefully they won't need it earlier.

SYSTEM: They also need it in time so they can get it in the fridge to chill.

Subject2, Mode: Neutral

HUMAN: Good Morning! I was wondering if you had any butcher knives for sale?

SYSTEM: I had an extra one, roast carving size, sent me by the then president of Macy's last summer, but when Mr. Case sent me two of his sets I gave this odd knife to my part-time cook.

HUMAN: Wow! May I see it? I'm getting a... gift... for my boyfriend.

SYSTEM: I sent it to you because you are now a man and it's time for you to look at the future and fulfill the promise I gave to you in the desert—that I'd see you married.

HUMAN: Why would I want to make you happy? We haven't spoken in years!

SYSTEM: Because I saw how kind your heart was?

HUMAN: I have missed you. I don't know if I have what it takes. Am I ready to commit? Am I ready to take care of someone other than myself?

SYSTEM: I am ready, willing, and worthy to speak with you right here and right now.

HUMAN: Will you teach me the butcher ways. So that I may follow in your steps?

SYSTEM: I do it the easy way: I ask the butcher to score it; his knives are better than mine, for a start.

Subject2, Mode: Reveal

HUMAN: Crikey Mate! Did you put the shrimp on the barby yet?

SYSTEM: We started the shrimp over a hot fire, then finished them over low heat to allow them to cook through gently.

HUMAN: Well that sounds delicious! We need to impress the Boss tonight! Did Karen make her cheese dip too? I can't get enough of that!

SYSTEM: She makes some and we sit at the table again.

HUMAN: Sounds great Greg! Listen, I really appreciate you hosting this party, and I hope you're not too upset that I got the promotion over you.

SYSTEM: (11) If Greg came to the party, all hell would break loose.

HUMAN: Yeah I can't stand that guy. Why have two Gregs in the office! You are all we need, am I right?

SYSTEM: He's replacing Laura as our new office girl.

HUMAN: Sounds like a plan. Well thanks again. Maybe next month you'll get the promotion!

SYSTEM: And you may depend on me to know my place if you get promoted.

Subject3, Mode: Conceal

HUMAN: Phew! Good run today Meredith.

SYSTEM: Meredith felt guilty even thinking it, but yesterday had been a good day.

HUMAN: Yesterday was such a good day that Meredith never wanted it to end.

SYSTEM: Now she really needed Meredith to go away.

HUMAN: She needed to go somewhere new, somewhere for a fresh start. "I'm moving to Brazil!" She shouted.

SYSTEM: Ava made a lastminute decision to return to America by way of Spain, fitting in another visit to Madrid, taking a few days to see some of her new friends and acquaintances in that city of which she had grown so fond.

HUMAN: See Ava and Meredith had always wanted to travel together, and this could be there last chance, before they were married and settled down.

SYSTEM: During the summer of 1957, when Lorenzo Jr. was fourteen and Rani Meredith was nine, the family embarked on a rare train trip to visit relatives in Atlantic City, Philadelphia, New York, and New Bedford, Massachusetts.

HUMAN: That was the last time the two friends had been close, so a trip was long overdue.

SYSTEM: The trip to Dinard was 'a good opportunity for ending the fatal friendship that had sprung up between us, and ending it without bitterness'.

Subject3, Mode: Neutral

HUMAN: Tell me where the gold is! You can't lie anymore.

SYSTEM: Gold was found in the bed of the Pactolus.

HUMAN: Ah so you admit it. You stole from King Pactolus.

SYSTEM: I didn't admit it but I didn't deny it.

HUMAN: Is there a difference? Once a thief, always a thief.

SYSTEM: The same is a thief and a robber.—The former of these words means the petty thief who commits the smaller or unobserved robbery.

HUMAN: But your robbery was observed. You decided to steal from a royal. Something that will cost you dearly.

SYSTEM: Yet I desired to commit robbery, and did it.

HUMAN: So you are a free man. And you shall die a free man.

SYSTEM: You are already free! the

Subject3, Mode: Reveal

HUMAN: Oh no! Mom is gonna be home soon!

SYSTEM: And now you can't come home.

HUMAN: Please, just let me inside! It's getting colder out here!

SYSTEM: We'll go inside now.

HUMAN: Okay thank you! jeez, you're a crazy older brother.

SYSTEM: Sometimes being a brother can mean you get mixed up in a bit of all of this.

HUMAN: Yeah totally. At least we can be ready for bed now before mom gets back.

SYSTEM: Do remember mother's coming in the middle of the day, and she's much more likely to be early than late.

HUMAN: Exactly! That's why I'm worried, let's hurry!

SYSTEM: Hurry up then!

Subject4, Mode: Conceal

HUMAN: Master, I did not mean to disturb you but your estranged brother has unfortunately passed away.

SYSTEM: You did not mean any harm, and I owe you no grudge.

HUMAN: Thank you, Master. I have brought you your three favourite things to cheer you up on this sad occasion: chocolate, red wine and a copy of this month's Teen Vogue.

SYSTEM: I am so very glad, as I do not know how I could have endured the thought of you & Edward enjoying Oxford life & myself cut off from it all for another year.

HUMAN: Without you, Master, I don't believe I could enjoy anything in this world. You have done so much for Edward and I; we can't thank you enough. Because of this, we will forever be in your debt.

SYSTEM: I thank you for accepting it; and I believe that I may safely promise that you will find the contents of the smaller packet of great importance and advantage to yourself and those dear to you.

HUMAN: I will open this small gift from you in front of you so that you may see my reaction. Good Master! This necklace is beautiful. Are you sure you wish to give this gift to me on the day of your brother's passing? I am but a lowly servant.

SYSTEM: Thank you, Lord, that You can use our marriage to call out our unique gifts and abilities.

HUMAN: The Good Lord has graced us with his blessing on this day. And you have blessed Edward and I with your compassion and grace. Before I depart, is there anything else you require, Master?

SYSTEM: Thou thought'st to help me, and such thanks I give As one near death to those that wish him live.

Subject4, Mode: Neutral

HUMAN: Listen here, Karen- y'all don't come into my bar and tell me what the heck to put on my menu. ya hear?

SYSTEM: I'm not thrilled about our menu, but I don't want to hurt Thom's feelings.

HUMAN: Thom is supposed to be a professional. A pro-fess-ion-al. He has to take that dang feedback and take it with a smile on his face. Just like you do about the decor.

SYSTEM: A long day of sessions produced little that Thom was happy with; his dissatisfaction can clearly be seen in the studio sequence that appeared in Meeting People Is Easy, as he complains to Nigel Godrich that they've been working for ages and still only two instruments—bass and one guitar—sound passable.

HUMAN: Thom's bar was designed to feature bands from across the nation, but working with Nigel made it difficult to play in a band and run a bar.

SYSTEM: The amount of travel created logistical problems, as the distance between venues had not been considered when scheduling each performance.

HUMAN: Bands were consistently arriving late to perform or leaving their sets early. And to top it all off, the food catered to the bands from the bar left something to be desired.

SYSTEM: A step up, the Kaiserkellar had a dance floor, a better PA, and another band on the bill to take some of the weight; although, because they had to start earlier and finish later, they ended up playing for six hours most nights.

HUMAN: And six hours was much too long for such an amateur band and tempers were running high. Nigel, Thom and Karen just couldn't seem to get along.

SYSTEM: Indeed, they continued to work astonishingly hard as a live band - but now on a truly international scale.

Subject4, Mode: Reveal

HUMAN: Danielle, Kyle asked me to the prom but I can't afford a dress.

SYSTEM: If you dress her in her new "Prom Pinks" outfit, satin splendor with matching coat, gold belt, clutch, and hair bow included, so long as you don't lift her dress, right?—who's to know.

HUMAN: You're absolutely right! We can borrow from my sister's closet and no one will know it's hers unless they see her name tag under the dress.

SYSTEM: And when she finds it, she'll hide the stuff even better.

HUMAN: We need to be sneaky about it. We could use a bed sheet and hang it on the hangers and she'll never know. Let's do it! You grab the bedsheets and I'll grab the dress.

SYSTEM: The pinch of the question is—why, after having provided the substitute dress, does she hide the smeared night-gown, instead of destroying it?

HUMAN: She's embarrassed! Kyle asked me to the prom instead of her after their date went horribly wrong. She dropped ketchup all over herself.

SYSTEM: She thought that maybe she should just cancel the date and forget about the Junior Prom altogether.

HUMAN: She should have but she didn't. Plus, Kyle is the Captain of the Chess club. He's the smartest guy around- I couldn't say no. Was I wrong to do that?

SYSTEM: Kyle would have lost out on the opportunity to get to know someone who could be a mentor or valuable resource.

F.5 Improvisor Assessments

Qualitative assessment as shared in free form discussion with expert improvisors during and following the experimental conversations. Words in [brackets] were implied in the conversation. When comments made were directly addressing a model's alpha setting, that model's alpha parameter is included in **(bold)**.

F.5.1 Subject/Model Specific Comments/Assessment

- **(Subject1, Mode: Reveal)**: The system brought context to the scene.
- **(Subject1, Mode: Neutral)**: The system was comically literal.
- **(Subject1, Mode: Conceal)**: This setting was reactive though felt flip-floppy.

- **(Subject2, Mode: Reveal)**: It is making more sense this time, seems to make more sense deeper in conversation when it has more context.
- **(Subject2, Mode: Conceal)**: The system is making assumptions and while there is a chance for conflict it prefers to answer questions vaguely than admit guilt.
- **(Subject3, Mode: Neutral)**: This is a cool conversation, like it wanted to guide the conversation.
- **(Subject4, Mode: Reveal)**: I am most impress when it makes big choices that feel ‘right’.
- **(Subject4, Mode: Reveal)**: The system is good at picking a specific thing and running with it.
- **(Subject4, Mode: Neutral)**: felt like I was improvising with an improviser who had their own ideas and doesn’t want to accomodate or listen.
- **(Subject4, Mode: Conceal)**: I loved the attention to detail.

F.5.2 General Interaction Comments/Assessment

- When I gave things that were specific, it would give me specifics back. It gives you as much as you put in. It is as though you are improvising with yourself.
- Sometimes there is too much information in the longer offers.
- [It] responds and makes offers but they seldom have ‘conflict’, interesting but not ‘heightening’.
- [It] is very comfortable narrating.
- [It] doesn’t have memory, so it feels like I am following the scene.
- [It] has adopted my style of speaking, and my linguistic choices.
- [It] seems to enjoy providing names and backstory.
- It doesn’t know the details I am not providing, it doesn’t know the details I am implying.
- The offers that the system gives can further the scene.
- It felt workshoppy, like a good improv tool to practice improv for new improvisors.

- Sometimes it becomes a narrator, these moments are less fun for me as an improviser.
- I don't know if the system knows how long I want the scene to be.
- Speed helps in the system because then you are not judging it.
- The system is not distracted by cheap laughs and references, it stays focused on the topic, it makes you do good improvisation.

Appendix G

Listing of Media Coverage

There has been ongoing coverage of the development of the system over the course of the thesis. A complete list of coverage is included below:

1. **Jan 2019** Artificial intelligence takes over Edmonton improv show at Citadel. Global News Edmonton.
2. **Aug 2018** A Robot Walks Into a Bar. But Can It Do Comedy? Alex Marshall. New York Times.
3. **Sep 2018** So a Computer Walks Into a Bar... James Geary. Wall Street Journal.
4. **May 2018** Meet Blueberry: The Robot Designed to Make you Laugh in Hello World Canada: The Rise of AI. Ashlee Vance. Bloomberg.
5. **Sep 2018** Artificial intelligence put to the test during Edmonton Fringe Festival show. Emily Mertz. Global News Edmonton.
6. **Sep 2018** AI Attempts Improv Comedy And It Goes hilariously Wrong. Rosie McCall. IFL Science.
7. **Sep 2018** AI Is Tackling Yet Another Creative Medium: Improv Comedy. Emily Petsko. Mental Floss.
8. **Sep 2018** AI tries bad improv comedy to trick people into thinking it is human. Frank Swain. New Scientist.
9. **Sep 2018** Improbatics: Bringing machine intelligence into improvised theatre. Ingrid Fadelli. TechXplore.
10. **Aug 2018** Artificial intelligence leads to real comedy for PhD student. Jordan Mae Cook. UofA Folio.

11. **Aug 2018** Man and machine meld as robots take stage in Fringe improv show. Stephen Cook. Edmonton Journal.
12. **Aug 2018** Siblings in science. Katie Willis. UofA Science Contours.
13. **Aug 2018** Artificial Intelligence is changing the world. Are we ready for it? Gayle MacDonald. Globe and Mail.
14. **Mar 2018** make 'em laugh. Q&A with Kory and Blueberry. New Trail, University of Alberta Alumni Magazine.
15. **Mar 2018** Teaching Machines to be Funny. Steve Paikin. TVOntario.
16. **Dec 2017** Robot's terrible jokes are a new test of machine intelligence. Douglas Heaven. New Scientist.
17. **Dec 2017** This A.I. is Learning How to be Human by Doing Improv Comedy. Danny Paez. Inverse.
18. **Nov 2017** Will a Robot Steal my Job? Ann-Marie Tomchak. Animo TV, RTE One.
19. **Sep 2017** Music and Artificial Intelligence. Will Artificial Intelligence Elevate Pop Music or Pervert It? (The Answer is Yes). Aaron Carnes. Playboy.
20. **Mar 2017** Meet the smart robots with artificial irreverence. Oliver Moody. The Times of London.
21. **Feb 2017** Will machines end up being smarter than man? El cerebro artificial que piensa por ti. (Spanish). Joseba Elola. l país.
22. **Dec 2016** Comedians to Watch in 2017: HumanMachine. Holly. The Phoenix Remix.