

ĐẢM BẢO TÍNH NHẤT QUÁN VÀ HIỆU QUẢ CỦA

MẠNG ĐƠN VỊ TĂNG DẦN TRONG MỘT MẠNG PHẢN TÁN

NGÀNH KIẾN TRÚC

qua

Mir Tahsin Imtiaz



Một luận án

nộp một phần để hoàn thành

của các yêu cầu về mức độ

Thạc sĩ Khoa học Máy tính

Đại học Boise State

Tháng 5 năm 2021

© 2021

Mir Tahsin Imtiaz

MỌI QUYỀN ĐƯỢC BẢO LƯU

TRƯỜNG CAO ĐẲNG ĐẠI HỌC BANG BOISE

ỦY BAN QUỐC PHÒNG VÀ PHÉ DUYỆT ĐỌC CUỐI CÙNG

của luận án được nộp bởi

Mir Tahsin Imtiaz

Tiêu đề luận án: Đảm bảo tính nhất quán và hiệu quả của mạng đơn vị gia tăng
trong Kiến trúc Phân tán

Ngày thi vấn đáp cuối kỳ: 12 tháng 3 năm 2021

Những cá nhân sau đây đã đọc và thảo luận về luận án do sinh viên Mir Tahsin Imtiaz nộp, và họ đã đánh giá bài trình bày và phản hồi của sinh viên đối với các câu hỏi trong kỳ thi vấn đáp cuối cùng. Họ thấy rằng sinh viên đã vượt qua kỳ thi vấn đáp cuối cùng.

Casey Kennington, Tiến sĩ

Chủ tịch, Ủy ban giám sát

Tiến sĩ Bogdan Dit

Thành viên, Ủy ban giám sát

Tiến sĩ Steve Cutchin

Thành viên, Ủy ban giám sát

Sự chấp thuận đọc cuối cùng của luận án đã được Casey Kennington, Tiến sĩ, Chủ tịch Ủy ban giám sát chấp thuận. Luận án đã được chấp thuận bởi Cao đẳng sau đại học.

LỜI CẢM ƠN

Tôi phải cảm ơn cố vấn của tôi, Tiến sĩ Casey Kennington, vì sự ủng hộ to lớn của ông và hướng dẫn, bao gồm cả trợ lý sau đại học hỗ trợ nghiên cứu của tôi và việc theo đuổi bằng Thạc sĩ Khoa học Máy tính của tôi tại Đại học Boise State. Tôi cũng phải thừa nhận rằng ông không chỉ là một cố vấn mà còn là một người bạn trong suốt toàn bộ bằng cấp của tôi. Lời khuyên, sự quan tâm và thời gian liên tục của anh ấy là lý do tôi có thể đã đi được đến đây và hoàn thành luận văn này. Tôi cũng muốn cảm ơn các thành viên trong ủy ban cố vấn Tiến sĩ Bogdan Dit và Tiến sĩ Steve Cutchin đã cung cấp cho tôi phản hồi và lời khuyên cần thiết để hình thành nên luận văn này.

Tôi cũng muốn cảm ơn các giáo sư khác ở Boise State mà tôi đã tương tác với. Tôi phải bắt đầu với Tiến sĩ Michael Ekstrand, người đã cung cấp cho tôi kiến thức phù hợp trong lĩnh vực Khoa học Dữ liệu đã giúp tôi bắt đầu ở đây tại Boise Tiểu bang. Tôi cũng cảm ơn Tiến sĩ Gaby Dagher, Tiến sĩ James Buffenbarger, Tiến sĩ Amit Jain, Tiến sĩ Francesca Spezzano và Tiến sĩ Edoardo Serra đã giúp tôi trở thành một người tốt hơn Sinh viên khoa học máy tính, lập trình viên và nhà nghiên cứu.

Tôi cũng muốn cảm ơn các thành viên hiện tại và trước đây của nhóm nghiên cứu SLIM nhóm đã chào đón tôi và giúp tôi trưởng thành. Mỗi thành viên của nhóm SLIM đều dạy cho tôi một thứ gì đó có giá trị trong thời gian tôi là thành viên của nhóm. Tôi cảm thấy vui mừng và rất vinh dự được gặp gỡ và làm việc với tất cả những người tài năng này. Tôi cũng muốn cảm ơn số ít người tôi gặp ở Boise mà tôi có thể gọi là bạn bè vì đã mời tôi vào cuộc sống của họ và

khiến tôi cảm thấy như ở nhà.

Cuối cùng tôi muốn dành tặng luận văn này cho gia đình tôi, những người đã ủng hộ tôi trong quyết định thực hiện một bước nhảy vọt và đi nửa vòng trái đất để học

ở nước ngoài.

TÓM TẮT

Một hệ thống gia tăng tận dụng dữ liệu sắp tới càng sớm càng tốt. Trong các hệ thống khác nói cách khác, một hệ thống gia tăng xử lý dữ liệu nhận được theo cách gia tăng. Gia tăng hệ thống có thể hữu ích hơn các hệ thống không gia tăng để xây dựng các hệ thống đối thoại nói khi chúng ta đang tìm kiếm hành vi nhanh hơn và giống con người hơn. Ví dụ, con người- các cuộc trò chuyện giữa người với người là gia tăng, vì người nghe không chờ người nói kết thúc việc nói để bắt đầu hiểu. Lấy cảm hứng từ thực tế là Robot-Ready Hệ thống Đối thoại Nói phải gia tăng và cần phải hoạt động phân tán, và IU khuôn khổ “bị phá vỡ” trong kiến trúc phân tán, tôi đã cố gắng sử dụng mạng IU để đáp ứng các yêu cầu gia tăng và có thể mở rộng khuôn khổ IU để hoạt động hoàn hảo trong môi trường phân tán. Công trình này nhằm mục đích trả lời câu hỏi liệu chúng ta có thể tạo ra một mạng IU phân tán hiệu quả và nhất quán hay không. Thêm cụ thể, tôi đã khám phá những cách tối ưu để thiết lập một kho dữ liệu IU phức tạp có thể tạo điều kiện thuận lợi cho việc bảo tồn và tiếp cận toàn bộ mạng dữ liệu IU được tạo ra trong một môi trường phân tán tránh việc “phá vỡ” mạng IU và hoạt động như xương sống cho hệ thống hộp thoại gia tăng “Robot-Ready” cuối cùng và hoàn chỉnh. Chúng tôi đánh giá sự khác biệt về phản ứng của HRI xảy ra cùng với việc triển khai cửa hàng IU sự khác biệt trong một nghiên cứu tương tác trực tiếp với robot và phát hiện ra rằng con người làm nhận thấy những khác biệt nhỏ về hiệu suất và vô thức trở nên phán xét robot đặc điểm nhân hóa liên quan đến hiệu suất của robot.

MỤC LỤC

LỜI CẢM ƠN iv

TÓM TẮT vi

DANH SÁCH HÌNH ẢNH x

DANH SÁCH BẢNGxv

DANH SÁCH CÁC CHỮ VIẾT TẮTxvii

1 GIỚI THIỆU 1

1.0.1 Hệ thống hộp thoại “Robot-Ready”. 2

1.1 Bối cảnh 3

1.1.1 Xử lý gia tăng 3

1.2 Mạng IU trong môi trường phân tán. 6

1.3 Giao tiếp giữa các tiến trình 7

1.4 Nền tảng tình báo vị trí (PSI). 9

1.5 ReTiCo 10

1.6 So sánh tính năng PSI và ReTiCo 10

1.7 Phát biểu luận đề. 12

2 CÔNG TRÌNH LIÊN QUAN 15

3 PHƯƠNG PHÁP	18
3.1 Chuẩn bị nền tảng cho tình báo tình huống (PSI).	19
3.2 Mô-đun PSI trong mạng	20
3.2.1 Kết nối hoặc hợp nhất dữ liệu	20
3.2.2 Nhận dạng giọng nói	21
3.2.3 Phát hiện đối tượng.	21
3.3 Các mô-đun ReTiCo trong mạng.	21
3.3.1 Mô-đun NLU (Hiểu ngôn ngữ tự nhiên) của Rasa	21
3.3.2 Keras Object Feature Extractor và WAC.	22
3.3.3 Mô-đun PyOpenDial.	23
3.3.4 Khả năng tương tác giữa hai khuôn khổ	24
3.3.5 Khả năng tương tác giữa các mô-đun trong mạng IU	25
3.3.6 Bảo tồn và duy trì toàn bộ mạng IU phân tán	26
3.4 Cấu trúc mạng tổng thể cuối cùng.	29
3.5 Tiêu chí đánh giá	29
3.6 Giả thuyết	32
4 ĐÁNH GIÁ	33
4.1 Thí nghiệm 1: Đánh giá có hệ thống về tính nhất quán và hiệu quả. 33	
4.1.1 Đánh giá kho IU.	33
4.1.2 Phân tích sâu hơn về đường ống.	39
4.2 Thí nghiệm 2: Nghiên cứu tương tác trực tiếp với người tham gia.	41
4.2.1 Tuyển dụng người tham gia và bối cảnh nghiên cứu.	42
4.2.2 Kết quả	44

5 KẾT LUẬN 56

5.1 Những hạn chế và công việc trong tương lai 57

TÀI LIỆU THAM KHẢO 58

PHỤ LỤC 65

A TRẢ LỜI CHO CÁC CÂU HỎI TRONG BẢNG CÂU HỎI GODSPEED . 65

BẢNG CÂU HỎI TĂNG CƯỜNG GODSPEED. 81

DANH SÁCH CÁC HÌNH ẢNH

1.1 Ví dụ về SLL và hoạt động Thêm, Thu hồi và Cam kết cho một
máy nhận dạng giọng nói incremental. 5

1,2 IU từ hai mô-đun (nhận dạng giọng nói và gắn thẻ các loại giọng nói)
được kết nối bởi GRIN. 6

1.3 Một mạng lưới phức tạp các IU được tạo ra bởi nhiều mô-đun. 8

3.1 Tổng quan về đa khung, đa phương thức, gia tăng, phân tán
mạng; hai quá trình giao tiếp với nhau bằng cách sử dụng một tin nhắn
xe buýt đi qua cây xô thơm; mục tiêu của luận án này là duy trì dữ liệu được chia sẻ
cấu trúc cần thiết cho quá trình gia tăng mà cả hai quá trình
có thể truy cập. 19

3.2 Cấu trúc chung của một IU nhận dạng giọng nói. 20

3.3 Từ ngữ liên quan đến các phương thức vật lý ảnh hưởng đến hành động của rô-bốt. . . 23

3.4 Cấu trúc lưu trữ mạng IU phân tán. 28

3.5 Robot Cozmo trong quá trình thực hiện nhiệm vụ. 31

4.1 NLU tạo ra các IU phù hợp tương ứng với các IU ASR. 40

4.2 Trực X: Đánh giá của người tham gia từ 1: Không bao giờ đến 5: Luôn luôn.
Trực Y: tỷ lệ phần trăm người tham gia đã chọn những phản hồi đó. 45

4.3 Trục X: Đánh giá của người tham gia từ 1: Không bao giờ đến 5: Luôn luôn.	
Trục Y: tỷ lệ phần trăm người tham gia đã chọn những phản hồi đó.	45
4.4 Trục X: Đánh giá của người tham gia từ 1: Không bao giờ đến 5: Luôn luôn.	
Trục Y: tỷ lệ phần trăm người tham gia đã chọn những phản hồi đó.	46
4.5 Trục X: Đánh giá của người tham gia từ 1: Không bao giờ đến 5: Luôn luôn. Trục Y:	
% người tham gia đã chọn những câu trả lời đó.	46
4.6 Trục X: Đánh giá của người tham gia từ 1: Hoàn toàn không hợp lý đến 5: Hợp lý	
có thể so sánh được. Trục Y: % người tham gia đã chọn những phản hồi đó.	47
4.7 Phản ứng trung bình của người tham gia đối với từng câu hỏi cụ thể cho từng loại nhiệm vụ	
của Cửa hàng IU.	47
4.8 Phản ứng trung bình của người tham gia đối với các câu hỏi liên quan đến nhân học của robot	
sự biến dạng, sự hoạt bát, sự dễ mến và trí thông minh được nhận thức trước đó	
đã đưa ra Kho lưu trữ IU dùng chung (Redis)	49
4.9 Phản ứng trung bình của người tham gia đối với các câu hỏi liên quan đến nhân học của robot	
sự biến dạng, sự hoạt bát, sự dễ mến và trí thông minh được nhận thức trước đó	
đưa ra Cửa hàng IU gốc (C#).	51
A.1 Trục X: Xếp hạng của người tham gia từ 1: Không hề đến 5: Rất. Trục Y:	
% người tham gia đã chọn những câu trả lời đó.	66
A.2 Trục X: Xếp hạng của người tham gia từ 1: Không hề đến 5: Rất. Trục Y:	
% người tham gia đã chọn những câu trả lời đó.	66
A.3 Trục X: Đánh giá của người tham gia từ 1: Không hề đến 5: Rất nhiều. Trục Y:	
% người tham gia đã chọn những câu trả lời đó.	67
A.4 Trục X: Đánh giá của người tham gia từ 1: Giả đến 5: Tự nhiên. Trục Y: %	
của những người tham gia đã chọn những phản hồi đó	67

A.5 Trục X: Đánh giá của người tham gia từ 1: Giống máy đến 5: Giống con người. Y-	
trục: % người tham gia đã chọn những phản hồi đó. 68
A.6 Trục X: Đánh giá của người tham gia từ 1: Vô thức đến 5: Có ý thức. Y-	
trục: % người tham gia đã chọn những phản hồi đó. 68
A.7 Trục X: Đánh giá của người tham gia từ 1: Nhân tạo đến 5: Giống thật. Trục Y:	
% người tham gia đã chọn những câu trả lời đó. 69
A.8 Trục X: Xếp hạng của người tham gia từ 1: Di chuyển cứng nhắc đến 5: Di chuyển các yếu tố	
Gantly. Trục Y: % người tham gia đã chọn những phản hồi đó.	. 69
A.9 Trục X: Xếp hạng của người tham gia từ 1: Đã chết đến 5: Còn sống. Trục Y: %	
của những người tham gia đã chọn những phản hồi đó	70
A.10 Trục X: Xếp hạng của người tham gia từ 1: Tụt hậu đến 5: Sôi động. Trục Y:	
% người tham gia đã chọn những câu trả lời đó.	70
A.11 Trục X: Đánh giá của người tham gia từ 1: Cơ học đến 5: Hữu cơ. Trục Y:	
% người tham gia đã chọn những câu trả lời đó. 71
A.12 Trục X: Xếp hạng của người tham gia từ 1: Trơ đến 5: Tương tác. Trục Y:	
% người tham gia đã chọn những câu trả lời đó.	71
A.13 Trục X: Đánh giá của người tham gia từ 1: Thờ ơ đến 5: Phản hồi. Y-	
trục: % người tham gia đã chọn những phản hồi đó. 72
A.14 Trục X: Xếp hạng của người tham gia từ 1: Không thích đến 5: Thích. Trục Y: %	
của những người tham gia đã chọn những phản hồi đó	72
A.15 Trục X: Đánh giá của người tham gia từ 1: Không thân thiện đến 5: Thân thiện. Trục Y:	
% người tham gia đã chọn những câu trả lời đó. 73
A.16 Trục X: Đánh giá của người tham gia từ 1: Không tử tế đến 5: Tử tế. Trục Y: %	
của những người tham gia đã chọn những phản hồi đó	73

- A.17 Trục X: Đánh giá của người tham gia từ 1: Không hài lòng đến 5: Hài lòng. Trục Y:
% người tham gia đã chọn những câu trả lời đó. 74
- A.18 Trục X: Đánh giá của người tham gia từ 1: Tệ đến 5: Tốt. Trục Y: %
của những người tham gia đã chọn những phản hồi đó 74
- A.19 Trục X: Đánh giá của người tham gia từ 1: Không đủ năng lực đến 5: Có năng lực.
Trục Y: tỷ lệ phần trăm người tham gia đã chọn những phản hồi đó. 75
- A.20 Trục X: Đánh giá của người tham gia từ 1: Không biết gì đến 5: Có hiểu biết.
Trục Y: tỷ lệ phần trăm người tham gia đã chọn những phản hồi đó. 75
- A.21 Trục X: Đánh giá của người tham gia từ 1: Vô trách nhiệm đến 5: Có trách nhiệm.
Trục Y: tỷ lệ phần trăm người tham gia đã chọn những phản hồi đó. 76
- A.22 Trục X: Đánh giá của người tham gia từ 1: Không thông minh đến 5: Thông minh. Y-
trục: % người tham gia đã chọn những phản hồi đó. 76
- A.23 Trục X: Đánh giá của người tham gia từ 1: Ngu ngốc đến 5: Hợp lý. Trục Y:
% người tham gia đã chọn những câu trả lời đó. 77
- A.24 Trục X: Đánh giá của người tham gia từ 1: Lo lắng đến 5: Thoải mái. Trục Y:
% người tham gia đã chọn những câu trả lời đó. 77
- A.25 Trục X: Đánh giá của người tham gia từ 1: Lo lắng đến 5: Thoải mái. Trục Y:
% người tham gia đã chọn những câu trả lời đó. 78
- A.26 Trục X: Xếp hạng của người tham gia từ 1: Kích động đến 5: Bình tĩnh. Trục Y:
% người tham gia đã chọn những câu trả lời đó. 78
- A.27 Trục X: Xếp hạng của người tham gia từ 1: Kích động đến 5: Bình tĩnh. Trục Y:
% người tham gia đã chọn những câu trả lời đó. 79
- A.28 Trục X: Xếp hạng của người tham gia từ 1: Chán đến 5: Quan tâm. Trục Y:
% người tham gia đã chọn những câu trả lời đó. 79

A.29 Trục X: Xếp hạng của người tham gia từ 1: Chán đến 5: Quan tâm. Trục Y:

% người tham gia đã chọn những câu trả lời đó. 80

DANH SÁCH CÁC BẢNG

1.1 So sánh tính năng của PSI và ReTiCo.	11
4.1 Độ trễ và tỷ lệ thành công phân phối IU của từng chức năng cho từng cửa hàng loại cho PSI	35
4.2 Độ trễ và tỷ lệ thành công phân phối IU của từng chức năng cho từng cửa hàng loại cho ReTiCo.	36
4.3 Độ trễ và tỷ lệ thành công phân phối IU của từng Chức năng cho Chia sẻ (Re- dis) IU Lưu trữ khi nó được đặt ở máy thứ ba.	37
4.4 Độ trễ và tỷ lệ thành công phân phối IU của từng chức năng cho Shared (Redis) Lưu trữ IU khi nó được đặt trong cùng một máy với PSI.	38
Độ trễ tạo ra 4,5 IU ở các mô-đun khác nhau.	41
4.6 Giá trị trung bình của phản hồi của người tham gia đối với các câu hỏi cụ thể về nhiệm vụ.	48
4.7 Trung bình phản hồi của người tham gia cho các câu hỏi thích Chia sẻ (Redis) Cửa hàng IU.	50
4.8 Trung bình phản hồi của người tham gia cho các câu hỏi thích Native (C#) Cửa hàng IU.	51
4.9 Mối tương quan giữa các câu hỏi liên quan đến việc thiết lập nhiệm vụ và các Godspeed khác các câu hỏi có mối tương quan cao hơn 0,45 đối với kho IU dùng chung. 53	

4.10 Sự tương quan giữa các câu hỏi liên quan đến việc thiết lập nhiệm vụ và các Godspeed khác

các câu hỏi có mối tương quan cao hơn 0,45 đối với Cửa hàng IU bản địa

Bảng 1. 54

4.11 Sự tương quan giữa các câu hỏi liên quan đến việc thiết lập nhiệm vụ và các Godspeed khác

các câu hỏi có mối tương quan cao hơn 0,45 đối với Cửa hàng IU bản địa

Bảng 2. 55

DANH SÁCH CÁC TỪ VIẾT TẮT

Nhận dạng giọng nói tự động ASR

GRIN Liên kết nền tảng

Đơn vị gia tăng IU

Mặt nạ RCNN Mặt nạ “Mạng nơ-ron tích chập dựa trên vùng”

Hiểu ngôn ngữ tự nhiên của NLU

Từ loại POS

Nền tảng PSI cho trí thông minh định vị

Hệ thống đối thoại nói SDS

Liên kết cùng cấp SLL

Mô hình WAC Words-As-Classifiers của Ngữ nghĩa từ vựng

CHƯƠNG 1:

GIỚI THIỆU

Khi robot trở nên phổ biến hơn, kỳ vọng về con người ngày càng tăng sẽ tương tác với họ. Phương tiện giao tiếp tự nhiên nhất giữa mọi người và robot không có bàn phím hoặc các thiết bị tương tác thông thường khác (ví dụ như cảm ứng màn hình), nhưng có hộp thoại nói vì robot được thiết kế để con người sử dụng với ít hoặc không có kinh nghiệm tính toán [1]. Trong một tình huống mà con người và rô-bốt là đối tác, đối thoại hợp tác có thể tránh được nhiều thao tác tốn kém tài nguyên chỉ bằng cách có sự giao tiếp tự nhiên như con người làm với người khác [2]. Do đó, theo nghĩa mở rộng, tôi cho rằng đối thoại sẽ là cách chung cuối cùng để con người giao tiếp với robot.

Luận án này liên quan đến cơ sở hạ tầng để tạo ra các hệ thống đối thoại nói (SDS) “sẵn sàng” hơn để làm việc với các nền tảng robot khác nhau. Cụ thể, chúng tôi đã khám phá những cách để tạo ra một hệ thống đối thoại đạt được tất cả các khả năng mong muốn trong khi vẫn giữ nguyên cơ sở hạ tầng nhất quán và hiệu quả cũng như giảm thiểu khối lượng công việc xây dựng lại một khuôn khổ mới để tạo điều kiện thuận lợi cho tất cả các yêu cầu. Chúng tôi đã đưa vào hai mẫu khung có một tập hợp con các khả năng mong muốn cho một robot sẵn sàng nói hệ thống đối thoại để làm cho chúng hoạt động cùng nhau xây dựng một robot chung sẵn sàng hệ thống đối thoại nói hoạt động nhất quán bằng cách giữ lại tất cả thông tin mà không phá vỡ toàn bộ mạng lưới. Cụ thể, chúng tôi đã sử dụng PSI [3] là một khuôn khổ từ

Microsoft để xây dựng các hệ thống AI tích hợp đa phương thức và ReTiCo [4])

là một khuôn khổ để xây dựng các hệ thống đối thoại nói tăng dần. Chúng tôi đã khám phá những cách khác nhau để xây dựng và duy trì một mạng lưới gia tăng chung sẵn sàng cho robot sử dụng hai khuôn khổ và đánh giá hiệu suất của chúng dựa trên các bài kiểm tra căng thẳng và thí nghiệm tương tác trực tiếp giữa người và robot.

1.0.1 Hệ thống hộp thoại “Robot-Ready”

Sau đây [5], một hệ thống hộp thoại chuẩn để sử dụng trong một tác nhân được thể hiện (ví dụ như một rô-bốt) phải đáp ứng các yêu cầu sau:

- mô-đun: hệ thống bao gồm nhiều mô-đun và các mô-đun mới có thể được tích hợp với hệ thống
- đa phương thức: hệ thống có thể tiếp nhận và tích hợp dữ liệu đầu vào từ nhiều cảm biến
- phân phối: các mô-đun có thể giao tiếp trong môi trường phân tán hoàn hảo (tức là giao tiếp phải đáng tin cậy)
- gia tăng: các mô-đun trong quá trình hệ thống nhận được đầu vào một cách nhanh chóng và trong liên tục
- căn chỉnh theo thời gian: nhiều đầu vào cảm biến phải được căn chỉnh với nhau (ví dụ căn chỉnh theo thời gian)

Khi nói đến việc tạo điều kiện thuận lợi cho cơ sở hạ tầng của các hệ thống đáp ứng các yêu cầu trên yêu cầu, đã có một số triển khai đa phương thức bao gồm

PSI [3] và Pythia [6], mỗi loại được viết bằng các ngôn ngữ lập trình khác nhau cho các mục đích khác nhau môi trường hoặc cho mục đích cụ thể giúp phát triển và nghiên cứu của các hệ thống AI phức tạp, đa phương thức. Ý nghĩa của xử lý gia tăng là

đầu vào giọng nói được xử lý từng từ một. Cho mục đích nghiên cứu trong lĩnh vực đối thoại hệ thống, vốn có bản chất gia tăng [7], nhiều khuôn khổ gia tăng có đã được phát triển (ví dụ InproTK [8], ReTiCo [4]) hỗ trợ xử lý gia tăng và có thể được sử dụng để xây dựng các giao diện máy tính phức tạp dựa trên giọng nói của con người. Mặc dù các khuôn khổ được phát triển cho đến nay đều hữu ích cho cộng đồng nghiên cứu, không có khuôn khổ nào khuôn khổ hiện có đáp ứng tất cả năm yêu cầu được đề cập ở trên và đảm bảo tính nhất quán của mạng gia tăng cả bên trong và bên ngoài mạng cụ thể đó khuôn khổ. Tuy nhiên, hai triển khai khác nhau được kết hợp lại, PSI và ReTiCo, có thể đáp ứng tất cả năm yêu cầu nếu chúng ta giải quyết được vấn đề bảo tồn mạng dữ liệu được tạo ra bởi một mạng IU nhất định cho một kiến trúc phân tán.

Vấn đề chúng tôi muốn giải quyết trong luận án này là liệu chúng ta có thể mang lại nhiều các quá trình gia tăng đơn lẻ cùng nhau hợp nhất các thuộc tính gia tăng của chúng để chúng giao tiếp với nhau một cách hiệu quả và nhất quán, và bảo toàn sự gia tăng trong quá trình này. Trong phần sau, tôi cung cấp thêm thông tin cơ bản cần thiết để xử lý gia tăng và cách nó có thể bị ảnh hưởng tiêu cực bởi một phân tán môi trường. Sau đó, tôi cung cấp thông tin cơ bản về PSI và ReTiCo và giải thích cách chúng có khả năng đáp ứng được tất cả năm yêu cầu.

1.1 Bối cảnh

1.1.1 Xử lý gia tăng

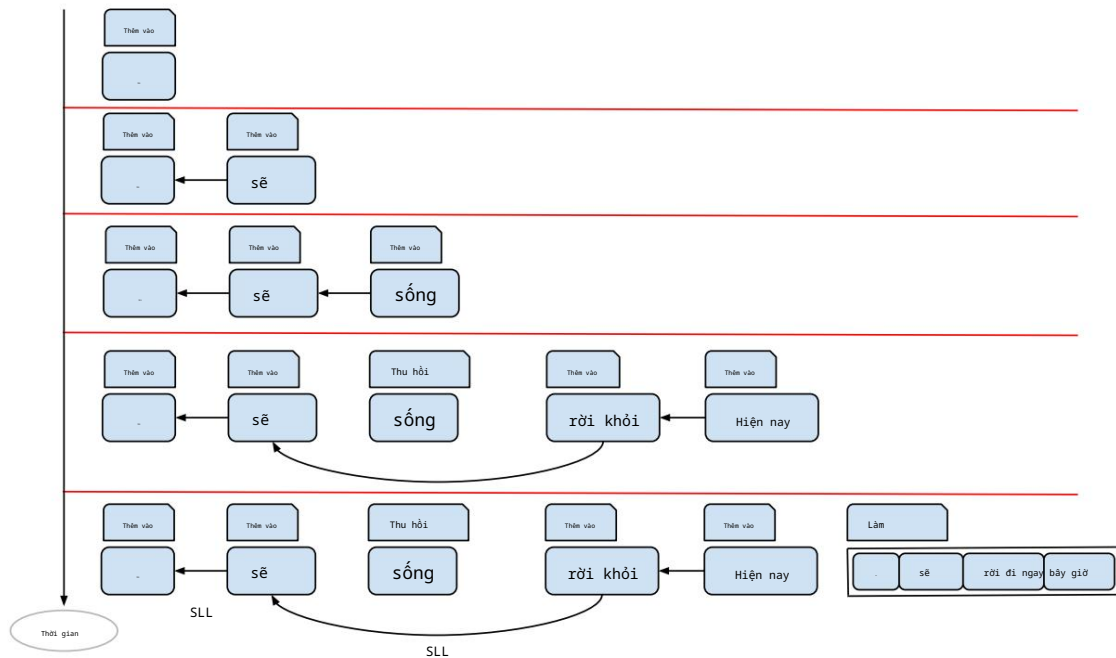
Một hệ thống gia tăng tận dụng dữ liệu sắp tới càng sớm càng tốt. Trong các hệ thống khác nói cách khác, một hệ thống gia tăng xử lý dữ liệu nhận được theo cách gia tăng [9]. Thay vào đó của việc chờ đợi tất cả thông tin được đưa vào, các mô-đun trong một hệ thống gia tăng

bắt đầu xử lý ngay khi họ bắt đầu nhận được lượng đầu vào tối thiểu [10] từ các mô-đun trước đó, tận dụng khả năng xuất ra các khối nhỏ hơn và cập nhật nó khi có thêm thông tin được tiết lộ như là đầu vào sau này. Số lượng tối thiểu của đầu vào hoặc dữ liệu mà một mô-đun gia tăng nhận được được gọi là Đơn vị gia tăng (IU) [11]. Khung IU [7] là một phương pháp tiếp cận có cấu trúc và khái niệm để thực hiện hệ thống gia tăng dựa trên IU.

Một mạng IU điển hình bao gồm nhiều mô-đun IU, mỗi mô-đun có một bên trái và một bộ đệm bên phải. Các mô-đun nhận dữ liệu dưới dạng IU từ mô-đun trước đó bằng cách sử dụng bộ đệm bên trái của nó bộ đệm, thực hiện tính toán trên dữ liệu đã nhận và chuyển các IU vừa được xử lý sử dụng bộ đệm phù hợp [12]. Mỗi IU trong một mô-đun cụ thể được kết nối bằng cách sử dụng Liên kết cùng cấp (SLL), trong khi các IU từ các mô-đun khác nhau được kết nối bằng cách sử dụng Liên kết cơ bản (GRIN) (giải thích chi tiết hơn ở bên dưới).

Vì các mạng gia tăng hoạt động bằng cách xử lý dữ liệu càng sớm càng tốt với lượng dữ liệu trong tay, họ cần một số loại cơ chế để cập nhật IU dựa trên về thông tin được cập nhật theo thời gian. ADD, REVOKE và COMMIT là ba loại hoạt động chính mà chúng ta có thể thực hiện trên IU [8]. ADD là hoạt động của thêm một IU mới khi có thông tin mới. Khi thông tin mới tiếp tục được cập nhật được tiết lộ, các mô-đun có thể xác định thông tin của một số IU cụ thể trước đó được tạo ra có thể không còn liên quan hoặc hữu ích nữa. Trong trường hợp này, một mô-đun HỦY BỎ một IU trước đó cũng như cho các mô-đun khác biết về hành động của nó vì một số mô-đun có thể đã xử lý các IU đó. Cuối cùng, khi nó là xác định rằng một số IU nhất định được thêm vào mạng không có cơ hội bị thay đổi hoặc được cập nhật và là kết quả cuối cùng tại thời điểm đó, các mô-đun đánh dấu IU đó là một CAM KẾT.

Một ví dụ điển hình của mô-đun gia tăng là trình nhận dạng giọng nói gia tăng



Hình 1.1: Ví dụ về SLL và hoạt động Thêm, Thu hồi và Cam kết cho trình nhận dạng giọng nói gia tăng.

mô-đun xử lý đầu ra từng từ một thay vì chờ im lặng hoặc

người dùng hoàn thành toàn bộ câu nói. Hình 1.1 cho thấy cách các IU được kết nối với nhau

trong một mô-đun nhận dạng giọng nói gia tăng sử dụng Liên kết cùng cấp và cách IU được

được thêm vào, thu hồi và được thêm vào nữa khi có thêm thông tin từ

mô-đun trước (mô-đun micro lắng nghe lời nói của người dùng). Bài phát biểu

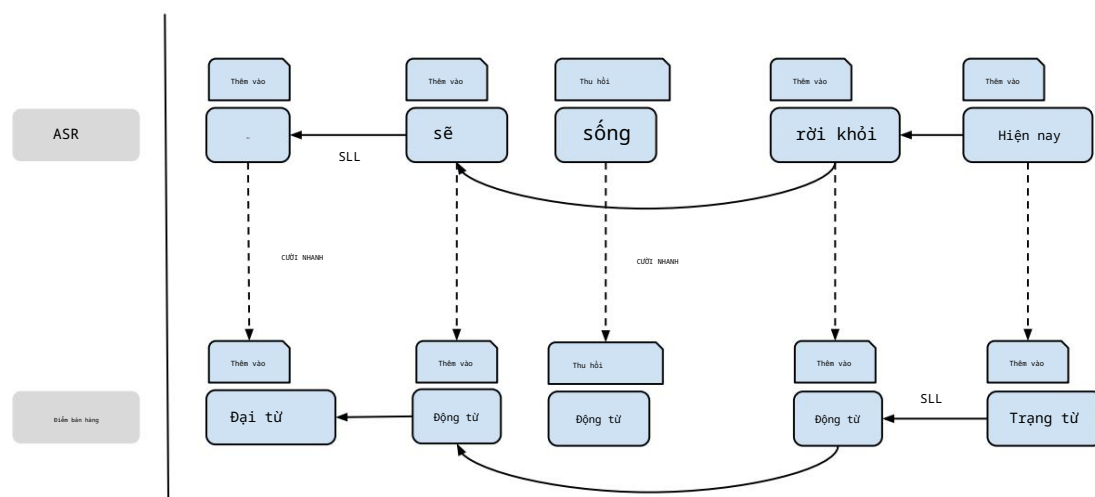
người nhận dạng ban đầu dự đoán kết quả một phần là "Tôi sẽ sống". Tuy nhiên, sau khi nhận được

thông tin thêm, nó nhận ra rằng kết quả thực sự là "Tôi sẽ rời đi ngay bây giờ", thu hồi

IU "live", thêm IUs "leave" và "now", và cuối cùng cam kết kết quả cuối cùng là

"Tôi sẽ rời đi ngay bây giờ". Hình 1.2 cho thấy cách IU của ASR (Nhận dạng giọng nói tự động)

Mô-đun nition) và mô-đun POS (Phần của bài phát biểu) được kết nối bằng cách sử dụng Grounded-in Liên kết.



Hình 1.2: IU từ hai mô-đun (bộ nhận dạng giọng nói và bộ gắn thẻ từ loại) được kết nối bằng GRIN.

Các hệ thống gia tăng có thể hữu ích hơn các hệ thống không gia tăng khi chúng ta tìm kiếm hành vi nhanh hơn và giống con người hơn. Ví dụ, giữa người với người các cuộc trò chuyện và xử lý ngôn ngữ là gia tăng, vì người nghe không chờ đợi người nói kết thúc việc nói để bắt đầu hiểu [13], và các hệ thống gia tăng thực hiện tương đối nhanh hơn các hệ thống không gia tăng vì các thành phần hoặc mô-đun làm việc đồng thời thay vì chờ một mô-đun hoàn tất toàn bộ quá trình xử lý của nó [14]. Vì hệ thống gia tăng hoạt động với lượng đầu vào tối thiểu nên nó mong đợi rằng một hệ thống đối thoại gia tăng sẽ có thể nắm bắt được các hành vi như phản hồi đồng thời, thay phiên nhau nhanh chóng và xây dựng phát ngôn hợp tác, không thể thực hiện được đối với hệ thống không gia tăng [15].

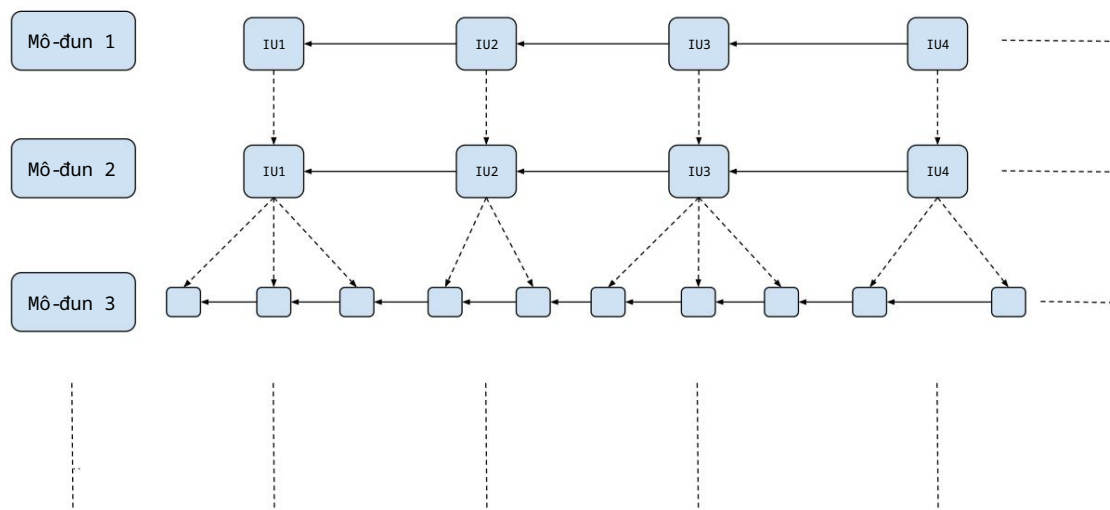
1.2 Mạng IU trong môi trường phân tán

Nếu chúng ta nhìn vào Hình 1.1 và Hình 1.2 và kiểm tra cách thông tin được truyền đi và được chia sẻ thông qua các mô-đun của mạng IU, chúng ta có thể thấy rằng tất cả các mô-đun trong

một mạng lưới IU tạo ra một mạng lưới phức tạp của IUs với nhau. Như đã đề cập ở trên, Các IU trong một mô-đun cụ thể được kết nối bằng Liên kết cùng cấp (SLL) trong khi các mô-đun xung quanh duy trì kết nối với các IU của nhau bằng cách sử dụng Grounded- trong Liên kết (GRIN) dẫn đến mạng lưới phức tạp của IU. Bất kỳ mô-đun nào cũng có thể truy vấn các mô-đun khác bắt đầu từ một đối tượng IU cụ thể để tìm ra lịch sử của nó chỉ bằng cách đi qua mạng lưới này. Mạng lưới IU này có thể được thực hiện nhiều hơn từ Hình 1.3. Nếu các mô-đun này nằm trong các hệ thống khác nhau, chúng tôi sử dụng các cơ chế như liên quy trình giao tiếp để thiết lập mạng và sử dụng định dạng tin nhắn chuẩn như JSON hoặc định dạng nhị phân chuẩn như Thrift [16] để gửi dữ liệu thích hợp. Tuy nhiên, đối với mạng IU, điều này là không đủ vì khi có những kết nối như vậy giữa hai mô-đun nằm trong các hệ thống khác nhau, mô-đun đích sẽ mất khả năng quay trở lại một số điểm nhất định trong mạng IU nằm ở hệ thống khác vì các thuộc tính của đối tượng kết nối nó với mạng còn lại bị lạc. Nói cách khác, một hệ thống gia tăng tổng thể bao gồm các mô-đun nằm trong các quá trình khác nhau phải đối mặt với việc phá vỡ toàn bộ mạng IU mặc dù ở mức tối thiểu kết nối được thực hiện và một số công việc nhất định được hệ thống hoàn thành.

1.3 Giao tiếp giữa các tiến trình

Một trong những vấn đề chúng ta cần giải quyết là làm thế nào để tạo ra hai quy trình gia tăng nói chuyện với nhau? Nói cách khác, hai hệ thống này trao đổi những thứ cần thiết như thế nào thông tin (các IU có liên quan và cách chúng được kết nối với các IU khác) để chúng hoạt động hoàn hảo? Đây là nơi khả năng tương tác phát huy tác dụng, đó là khả năng của hai việc triển khai các hệ thống hoặc thành phần từ các quy trình khác nhau để cùng tồn tại và làm việc cùng nhau bằng cách chỉ dựa vào nhau khi cần thiết [17]. Nếu chúng ta xem xét toàn bộ dự án như một hệ thống đầy đủ, "Khả năng tương tác" là thuật ngữ được sử dụng để chỉ



Hình 1.3: Mạng lưới phức tạp của các IU được tạo ra bởi nhiều mô-đun.

với ý tưởng sử dụng hai quy trình trong hai ngôn ngữ lập trình cùng nhau để cần thiết mức độ cần thiết [18]. Điều quan trọng là chúng ta phải tìm ra một cách chung và thông thường để hoạt động hiệu quả với các hệ thống đang được đề cập. Các lập trình viên thường cho phép sử dụng các thư viện hiện có được viết bằng ngôn ngữ khác bằng cách sử dụng các kỹ thuật tương tác kể từ không phải lúc nào cũng khả thi khi viết lại toàn bộ thư viện bằng ngôn ngữ mới do nó quá lớn và việc thực hiện phức tạp, và giao diện với việc thực hiện hiện có là cách tiếp cận thực tế hơn [19]. Ví dụ, dotnet quản lý điều này bằng C++ bằng cách hỗ trợ khả năng tương tác trực tiếp chỉ với một tập hợp con của C++ (Managed C++ hoặc C++/CLI) [20]. Python cũng cung cấp hỗ trợ giao diện hàm nước ngoài để chạy các mô-đun mở rộng được viết bằng các ngôn ngữ cấp thấp hơn như C, mặc dù hỗ trợ bị hạn chế đối với các triển khai hiện đại của ngôn ngữ động [19]. Chúng ta có thể cũng sử dụng giao tiếp giữa các tiến trình dựa trên tin nhắn thay vì ngôn ngữ soạn thảo triển khai ở cấp độ triển khai của họ. Ví dụ về loại triển khai này hỗ trợ việc truyền tin giữa các hệ thống được viết bằng các ngôn ngữ khác nhau

Bộ đệm giao thức [9] từ Google và Thrift [16] từ Facebook. Trong khi sử dụng những các loại triển khai, kỹ sư và lập trình viên làm việc với ngôn ngữ không phụ thuộc ngôn ngữ định nghĩa giao diện. Loại giao diện này sắp xếp dữ liệu thành một sự biểu diễn có thể được hiểu ở cả hai phía.

Để duy trì toàn bộ mạng IU trong khi hai tiến trình đang chạy và giao tiếp bằng cách sử dụng kỹ thuật giao tiếp giữa các tiến trình, chúng ta cần lưu trữ và cập nhật mạng lưới từ góc nhìn của cả hai quá trình theo một số loại logic cơ sở dữ liệu được chia sẻ. Hiệu suất sẽ phụ thuộc vào cách chúng tôi gửi dữ liệu giữa hai quy trình và cách chúng tôi triển khai cơ sở dữ liệu được chia sẻ này. Trong khi chia sẻ dữ liệu với với nhau, chúng ta có thể tiếp tục gửi tham chiếu đến toàn bộ cấu trúc dữ liệu hoặc chỉ gửi các khối dữ liệu mới. Ngoài ra, chúng ta có thể triển khai cơ sở dữ liệu trong một của các quy trình mà quy trình khác có thể truy vấn dữ liệu có liên quan (ví dụ PSI giữ cơ sở dữ liệu và các truy vấn ReTiCo tới PSI), hoặc chúng ta có thể triển khai một cơ sở dữ liệu được chia sẻ nơi cả hai quy trình sẽ có quyền truy cập vào (chi tiết hơn và giải thích về các chiến lược thực hiện được đưa ra trong phần phụ thí nghiệm). Trong ba phần tiếp theo các phần, chúng tôi sẽ giới thiệu ngắn gọn hai khuôn khổ mục tiêu của chúng tôi, PSI và ReTiCo, và so sánh chúng về mặt tính năng.

1.4 Nền tảng cho tình báo định vị (PSI)

PSI [3] là một khuôn khổ từ Microsoft được viết bằng C# mở ra cánh cửa dễ dàng phát triển và nghiên cứu các hệ thống AI đa phương thức và tích hợp. Mặc dù không được viết bằng một trong những ngôn ngữ lập trình phổ biến nhất trong Học máy [21], PSI hoạt động bằng cách cung cấp một mô hình lập trình song song tập trung vào dữ liệu luồng, cho phép phát triển và kết nối các thành phần dễ dàng trong khi vẫn giữ nguyên các thuộc tính hiệu suất của một hệ thống được viết gốc và bao gồm nhiều AI khác nhau

công nghệ cho phép xây dựng nhanh các ứng dụng AI phức tạp.

Một trong những lý do chính để đưa PSI vào nghiên cứu của chúng tôi là các đặc tính và các tính năng mà nó cung cấp. PSI đã đáp ứng một số yêu cầu cho SDS sẵn sàng cho robot như đã đề cập ở trên. Hơn nữa, khuôn khổ do PSI cung cấp có nhận thức về thời gian và có khả năng hợp nhất luồng có ý nghĩa, đây là một trong những yêu cầu cốt lõi cho một hệ thống gia tăng tùy thuộc vào cấu trúc mạng [9]. Tính năng này có chưa từng được giới thiệu trước đây trong bất kỳ khuôn khổ nào khác hiệu quả như PSI. Ngoài ra với điều này, PSI đưa vào các công cụ và API cho phép trực quan hóa dữ liệu đa phương thức và phân tích theo thời gian thực.

1.5 ReTiCo

ReTiCo [4] là một khuôn khổ gia tăng được viết bằng Python và cho phép xây dựng hệ thống đối thoại nói tăng dần cung cấp nhiều loại tăng dần các mô-đun (ví dụ Rasa NLU, PyOpenDialog Dialog Manager, v.v.). Khung là thân thiện với người dùng và cho phép xây dựng mạng lưới với một vài dòng mã khởi tạo các mô-đun và kết nối chúng theo bộ đệm trái và phải của chúng.

Trong khi ReTiCo là một ví dụ điển hình về một khung gia tăng tiêu chuẩn để sử dụng làm việc dựa trên hệ thống đối thoại nói, nó thiếu một số tính năng chính cần thiết cho nghiên cứu trong các hệ thống gia tăng, mô-đun và đa phương thức. Cụ thể là, nó đang thiếu các cơ chế thích hợp có thể hoạt động với các luồng dữ liệu đồng thời đến từ các mô-đun khác nhau và cơ sở lưu trữ dữ liệu tiêu chuẩn để phân tích dữ liệu.

1.6 So sánh tính năng PSI và ReTiCo

Như đã đề cập, các hệ thống hợp thoại tiêu chuẩn phải đáp ứng một số yêu cầu nhất định mặc dù không cần thiết phải đạt được tất cả chúng để đạt được sự gia tăng hoàn toàn

hệ thống đối thoại. Bảng 1.1 cho thấy sự so sánh giữa PSI và ReTiCo về mặt những yêu cầu này.

Bảng 1.1: So sánh tính năng của PSI và ReTiCo.

Yêu cầu	PSI	ReTiCo
Mô-đun	Đúng	Đúng
Đa phương thức	Đúng	Đúng
Phân phối	Chỉ truyền dữ liệu bằng bên thứ ba Chỉ truyền dữ liệu bằng bên thứ ba	
Tăng dần	Về mặt logic	Về mặt logic và cấu trúc
Sự kết hợp luồng có ý nghĩa được căn chỉnh theo thời gian	KHÔNG	

Như thể hiện trong Bảng 1.1, cả PSI và ReTiCo đều không đáp ứng được tất cả các yêu cầu hoàn toàn. Trong khi PSI giữ khả năng căn chỉnh thời gian, nó không giữ khả năng tăng tâm lý bằng cách tuân theo một cấu trúc chuẩn. Mặt khác, ReTiCo có cấu trúc gia tăng có nghĩa là IU được truyền giữa các mô-đun theo một tiêu chuẩn. Tuy nhiên, nó không thể đạt được sự liên kết thời gian. Ngoài ra, cả PSI và ReTiCo chỉ hỗ trợ các tính năng phân phối bằng cách chỉ cho phép truyền dữ liệu dưới dạng JSON. Không có cấu trúc hoặc cơ chế nào tạo điều kiện cho kết nối cấu trúc giữa các mod- hoặc giữ cho đầu ra mạng IU tổng thể nhất quán. Do đó, trong trường hợp thiết lập phân phối thực tế, cả hai đều mất khả năng xây dựng một mạng IU chia sẻ thực sự, và chúng chỉ có thể được sử dụng cho một số công việc đơn giản nhất định (ví dụ như xây dựng một công trình gia tăng mạng, mạng còn lại lưu trữ kết quả cuối cùng dưới dạng một trình ghi nhật ký).

1.7 Luận đề

Làm thế nào chúng ta có thể sử dụng mạng IU để đáp ứng các yêu cầu gia tăng và có thể mở rộng khuôn khổ IU để hoạt động hoàn hảo trong môi trường phân tán? Công việc của tôi nhằm mục đích kết hợp nhiều quy trình gia tăng phức tạp để chúng hoạt động cùng nhau để duy trì một mạng IU chung trên cả hai quy trình, tận dụng lợi thế của từng quy trình điểm mạnh của người khác (các đặc tính mà chỉ một trong hai quy trình có mà cả hai đều có thể sử dụng) để thực hiện các nhiệm vụ phức tạp mà họ không thể tự mình làm được. Cụ thể hơn, tôi đã khám phá những cách để trả lời câu hỏi liệu chúng ta có thể kết hợp hai khuôn khổ đa phương thức lại với nhau hay không trong khi tận dụng tốt nhất cả hai, thêm các thuộc tính gia tăng vào một, sau đó hợp nhất khả năng gia tăng của họ, tránh phá vỡ mạng IU, đảm bảo khả năng truy cập của toàn bộ mạng lưới cho cả hai khuôn khổ và duy trì hiệu quả và tính nhất quán của mạng lưới gia tăng trong môi trường phân tán trong khi xây dựng một Hệ thống đối thoại "Robot-Ready". Hơn nữa, tôi đã khám phá những cách tối ưu để thiết lập một kho dữ liệu IU phức tạp có thể tạo điều kiện thuận lợi cho việc bảo tồn và truy cập tổng thể tạo ra mạng IU trong môi trường phân tán tránh việc "phá vỡ" IU mạng lưới và hoạt động như xương sống cho quá trình gia tăng "Robot-Ready" cuối cùng và hoàn chỉnh hệ thống đối thoại. Khi nói đến việc xây dựng các hệ thống phần mềm lớn và có thể mở rộng, một cách quản lý độ phức tạp của phần mềm là sử dụng một tập hợp các thành phần đảm bảo khả năng tái sử dụng, tính mô-đun và cô lập lỗi [22]. Trong phát triển phần mềm, nó là phổ biến để các lập trình viên sử dụng ngôn ngữ phù hợp nhất cho một công việc cụ thể, kết hợp các bộ ngôn ngữ khác nhau và tái sử dụng mã nguồn hiện có [23]. Điều này cho phép họ chọn hệ thống tốt nhất cho các nhiệm vụ cụ thể khi hoàn thành một công việc phức tạp tổng thể.

Luận án này tìm cách áp dụng cách tiếp cận tương tự cho các hệ thống gia tăng tiêu chuẩn để thực hiện một nhiệm vụ phức tạp bằng cách kết hợp các bộ công cụ từ nhiều môi trường khác nhau. Trước đó

ngiên cứu của chúng tôi đã cố gắng sử dụng nhiều hệ thống cùng nhau chia sẻ chung kỹ thuật tương tác trong đó một hệ thống đóng góp vào việc xây dựng cổng thông tin chính của mạng lưới, và hệ thống kia thực hiện một phần nhỏ cụ thể của nhiệm vụ chung [5]. Mạng IU đã có tiềm năng lý thuyết để hoạt động trong một môi trường phân tán mà không phá vỡ các nghiên cứu trước đây trong lĩnh vực này [7]. Tôi đưa ra giả thuyết rằng các phương pháp tiếp cận được đề xuất để xây dựng một mạng IU phân tán cùng với việc xây dựng một kho IU tùy chỉnh lưu trữ dữ liệu tổng thể tạo điều kiện thuận lợi cho việc bảo tồn dữ liệu IU được tạo ra bởi mạng sẽ nhất quán và hiệu quả, và Nhìn chung, hệ thống được đề xuất sẽ thúc đẩy nghiên cứu sâu hơn bao gồm phân phối Mạng lưới IU.

Chúng tôi đã thiết kế một thí nghiệm trong đó chúng tôi đưa vào hai khuôn khổ gia tăng với các bộ khả năng khác nhau, cùng nhau thực hiện một nhiệm vụ phức tạp (chi tiết hơn về cách tiếp cận tổng thể nằm trong các chương sau) của một cuộc đối thoại được nói sẵn sàng cho robot hệ thống. Chúng tôi đã sử dụng các phương pháp tiếp cận có hệ thống để đánh giá việc thực hiện cuối cùng và sử dụng robot Cozmo¹ để tương tác với các đối tác là con người để hiểu được nhận thức của con người trong những cách tiếp cận khác nhau của chúng ta. Con người đã đặt câu hỏi và đưa ra bình luận lệnh cho robot, mong đợi câu trả lời hoặc hành động thích hợp như phản hồi. Robot sử dụng hai quy trình của chúng tôi ở chế độ nền cùng nhau để hoàn thành các phần khác nhau của nhiệm vụ tổng thể. Điều này, cùng với đánh giá có hệ thống của chúng tôi, đã đánh giá hiệu quả và tính nhất quán của việc thực hiện.

Vấn đề chúng tôi đề cập trong luận án này rất quan trọng vì nó cung cấp cho chúng tôi cái nhìn sâu sắc về xây dựng các mạng lưới gia tăng nhất quán có thể tạo điều kiện cho sự gia tăng dữ liệu ổn định bên trong các mô-đun gia tăng không chỉ bị cô lập ở một nơi mà còn có thể được phân tách

¹<https://www.digitaldreamlabs.com/pages/cozmo>

được chuyển đến các máy khác nhau mà không làm hỏng mạng, đây là một vấn đề phổ biến trong các hệ thống gia tăng đa phương thức. Hy vọng rằng nghiên cứu này sẽ giúp chúng ta có thể và thúc đẩy để xây dựng các hệ thống gia tăng phức tạp và hiệu quả hơn, cuối cùng sẽ dẫn đến nghiên cứu tiên tiến hơn trong lĩnh vực này. Kết hợp những lợi thế của nhiều khuôn khổ của một khu vực cụ thể và tạo điều kiện dễ dàng hơn để tiếp cận các hệ thống phức tạp có thể đẩy nhanh nghiên cứu và phát triển cũng như làm cho nhiều người quan tâm hơn đến việc làm việc trong những khu vực.

CHƯƠNG 2:

CÔNG VIỆC LIÊN QUAN

Nghiên cứu trước đây trong lĩnh vực AI, robot và hệ thống đối thoại đã chứng minh việc sử dụng và lợi thế của việc làm việc với hoặc xây dựng trên các hệ thống hiện có và kết nối chúng cùng nhau. MultiBot [24] được xây dựng bằng cách sử dụng và tận dụng các thành phần đã có sẵn nents từ ScoutBot [25] bằng cách mở rộng chế độ tương tác cho nhiều người tham gia hội thoại. Trong quá trình phát triển không gian văn phòng thông minh tạo điều kiện cho việc học tập cộng tác, Wang et al. [26] đã sử dụng nhiều hệ thống phần mềm để tích hợp khả năng của họ trong đó bộ công cụ Bazaar [27] đã được sử dụng làm nền tảng cho hỗ trợ dựa trên hội thoại của- được truyền trong không gian và PSI được sử dụng để phối hợp các luồng dữ liệu. Nhìn chung, Chợ chủ yếu được sử dụng như một mô-đun mở rộng và để đảm bảo thông tin chính xác Sage chuyển đổi giữa hai công cụ, họ đã sử dụng một đa phương thức được phát triển nội bộ định dạng tin nhắn bao gồm bất kỳ sự kết hợp nào của vị trí, văn bản lời nói, nội dung vị trí, biểu cảm khuôn mặt và bất kỳ cảm xúc nào được phát hiện.

Kennington et al. [12] đã mở rộng bộ công cụ xử lý gia tăng InproTK [8] thành InproTKS để có thể nhận dữ liệu cảm biến đa phương thức và đạt được vị trí và hội thoại thời gian thực. Là một hệ thống gia tăng, InproTKS là một tiềm năng khác Ứng cử viên cho luận án này tập trung vào việc xây dựng một hệ thống đối thoại hoàn chỉnh và sẵn sàng. Tuy nhiên, hai khuôn khổ được chọn (PSI và ReTiCo) tập hợp tất cả các yếu tố yêu cầu được xác định. Cụ thể, PSI mang lại sự liên kết thời gian hoàn hảo của dữ liệu và

ReTiCo, được viết bằng Python và đã được gia tăng, biểu diễn nhiều các công cụ hiện có liên quan đến hệ thống đối thoại. Kousidis et al. [28] đã tạo ra một đa phương thức Hệ thống đối thoại trong xe bằng cách sử dụng bộ công cụ OpenDS [29] như một trình mô phỏng lái xe và sử dụng InproTKS [12] để xây dựng hệ thống đối thoại cho thí nghiệm của họ. Mặc dù hệ thống đối thoại là gia tăng, họ sử dụng các kỹ thuật truyền tin nhắn khác nhau có sẵn cho họ (Kiến trúc truyền tin nhắn Robotics Service Bus (RSB)¹ [30], và InstantIO/InstantReality²) chỉ để ghi lại kết quả ở định dạng tệp XML để biết thêm phân tích. Carlmeyer et al. [31] đã kết hợp InproTK [8] với PaMini [32] để cho phép các vòng phản hồi khép kín trong HRI để hệ thống tương tác của họ có thể thích ứng với người dùng. Trong thí nghiệm của họ, họ cũng sử dụng Robotics Service Bus (RSB). Để gửi và nhận tin nhắn có cấu trúc giữa InproTK và PaMini, họ đã sửa đổi Trình lắng nghe và cung cấp thông tin InproTK để có thể gửi các hành động đối thoại thích hợp và thông tin các lời nói có thể được chia nhỏ và xử lý thành các cụm từ nhỏ. Hơn nữa, vì PaMini chỉ phản ứng với các đầu vào có trạng thái "COMMIT", chúng đã tạo ra một nguồn đầu vào mới cho PaMini phản ứng với các hành động đối thoại từ InproTK. Nói cách khác, họ đã cố gắng giải quyết vấn đề kết nối một mô-đun bên ngoài (trình quản lý hộp thoại PaMini) với khung gia tăng InproTK. Kennington et al. [5] đã làm việc hướng tới một robot sẵn sàng hệ thống đối thoại bằng giọng nói bằng cách tích hợp nhiều bộ công cụ trong ReTiCo. Họ chỉ sử dụng PSI như một công cụ ghi nhật ký để tận dụng các tiện ích lưu trữ dữ liệu của nó và chưa sử dụng khả năng căn chỉnh thời gian của nó. Mặc dù tất cả công việc trước đây này liên quan đến xây dựng các hệ thống đa khung (tăng dần hoặc không tăng dần), họ đã không xây dựng hệ thống của họ từ góc nhìn mạng IU. Nói cách khác, họ không tập trung vào việc giải quyết các vấn đề liên quan đến tính nhất quán của mạng IU trong phân phối

¹<https://code.cor-lab.de/projects/rsb>

²<https://www.instantreality.org/>

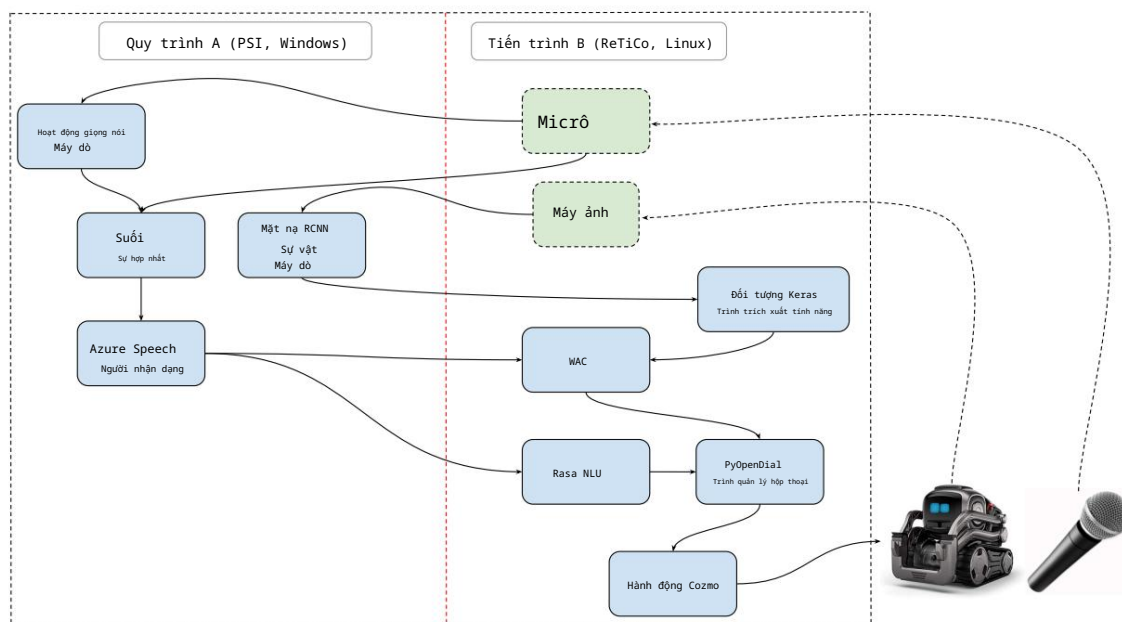
môi trường-mục tiêu của luận văn này.

CHƯƠNG 3:

PHƯƠNG PHÁP

Để chứng minh rằng một mạng lưới gia tăng phức tạp có thể được xây dựng và sử dụng sử dụng hiệu quả hai khuôn khổ gia tăng trong một môi trường phân tán bảo tồn và duy trì mạng IU, chúng tôi đã tiến hành một thí nghiệm tương tự như đã đề cập trong [5] nơi một robot nhận dữ liệu từ micrô và camera và quá trình xử lý tổng thể của toàn bộ đường ống được thực hiện bằng cách sử dụng kết hợp hai khuôn khổ của chúng tôi.

Chúng tôi đã sử dụng PSI và ReTiCo, một trong số đó là một khuôn khổ mở, có thể mở rộng được viết trong C# cho phép phát triển và nghiên cứu các hệ thống AI tích hợp, và cái còn lại là một khuôn khổ Python dựa trên [7] và cho phép xây dựng gia tăng hệ thống đối thoại nói. Trong thí nghiệm của chúng tôi, chúng tôi đã làm cho chúng hoạt động cùng nhau để xây dựng một mạng lưới gia tăng chung như thể hiện trong Hình 3.1. Chi tiết về việc xây dựng nằm trong các phần sau.

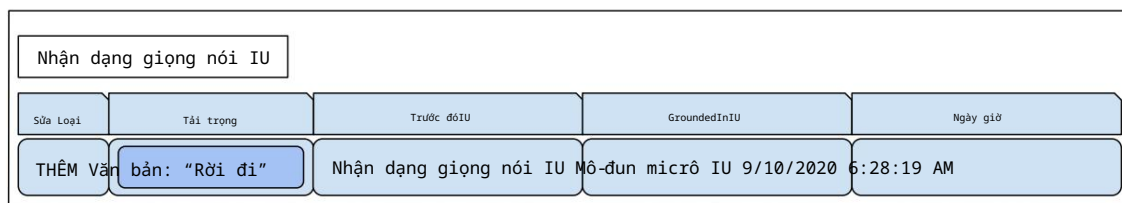


Hình 3.1: Tổng quan về đa khung, đa phương thức, gia tăng, mạng lưới phân tán; hai quá trình giao tiếp với nhau bằng cách sử dụng một xe buýt truyền tin; mục tiêu của luận án này là duy trì dữ liệu được chia sẻ cấu trúc cần thiết cho quá trình gia tăng mà cả hai quá trình có thể truy cập.

3.1 Chuẩn bị nền tảng cho trí thông minh định vị (PSI)

Mặc dù PSI đã hoạt động gia tăng ở chỗ nó có thể xử lý đầu vào liên tục, nhưng nó thiếu một trong những yêu cầu cơ bản nhất của giao tiếp trong mạng gia tăng: một cách tiêu chuẩn để chia nhỏ hoặc phân chia các đơn vị dữ liệu lớn thành các đơn vị dữ liệu nhỏ hơn. Nói cách khác, PSI cần phải được thực hiện để hoạt động trong khuôn khổ IU (một cái gì đó mà ReTiCo thực hiện một cách tự nhiên). Để giải quyết vấn đề này, chúng tôi đã triển khai một deltafier tiếp nhận dữ liệu nhỏ nhất có thể và đóng gói dữ liệu dưới dạng IU. Mỗi IU là bao gồm EditType giải thích liệu dữ liệu đang được Thêm, Thu hồi

hoặc Đã cam kết, TimeStamp là một biến kiểu double lưu trữ thời gian Epoch1 trong quá trình tạo IU (mặc dù phương pháp ToString sẽ chuyển đổi nó thành một Ngày và giờ như thể hiện trong Hình 3.2), tham chiếu đến IU trước đó có liên quan (SLL) trong mô-đun chưa bị thu hồi, một tham chiếu đến IU được nối đất và tải trọng của IU đó. Tải trọng cho một trình nhận dạng giọng nói là một văn bản (ví dụ một từ) chứa kết quả nhận dạng. Toàn bộ dữ liệu được tạo ra trong một mô-đun được lưu trữ và giữ như danh sách các IU và mỗi mô-đun xác định IU và tải trọng riêng của nó. Hình 3.2 cho thấy điều này cấu trúc cho một IU của máy nhận dạng giọng nói.



Hình 3.2: Cấu trúc chung của một IU nhận dạng giọng nói.

3.2 Mô-đun PSI trong Mạng

Như thể hiện trong Hình 3.1, toàn bộ mạng đã được chia thành hai khung, mỗi bên thực hiện một phần công bằng của quá trình xử lý. Trong mạng, PSI chịu trách nhiệm về giọng nói phát hiện hoạt động, hợp nhất luồng dữ liệu gốc và dữ liệu nước ngoài có ý nghĩa, phát hiện đối tượng từ dữ liệu hình ảnh và nhận dạng giọng nói từ dữ liệu âm thanh.

3.2.1 Kết nối hoặc hợp nhất dữ liệu

Có khả năng tham gia hoặc hợp nhất các luồng một cách thích hợp (căn chỉnh dữ liệu theo thời gian với về mặt thời gian) là một tính năng độc đáo của PSI mà ReTiCo không có. Trong ví dụ của chúng tôi thí nghiệm, PSI chịu trách nhiệm hợp nhất luồng micro đến nhận được từ

¹https://en.wikipedia.org/wiki/Unix_time

ReTiCo với luồng phát hiện hoạt động giọng nói được tạo ra trong PSI. Điều này chứng minh rằng sự hợp nhất luồng có ý nghĩa có thể được thực hiện trong môi trường phân tán.

3.2.2 Nhận dạng giọng nói

PSI chịu trách nhiệm triển khai mô-đun nhận dạng giọng nói. Chúng tôi đã sử dụng Azure Dịch vụ Nhận thức Ngôn ngữ để triển khai mô-đun nhận dạng giọng nói hoạt động cùng với deltafier được đề cập ở trên để thực hiện nhận dạng giọng nói gia tăng.

3.2.3 Phát hiện đối tượng

Mô-đun phát hiện đối tượng PSI sử dụng Google MaskRCNN [33] để xác định đối tượng thích hợp kiểm tra các luồng camera nhận được từ ReTiCo để truyền lại cho ReTiCo để trích xuất tính năng.

3.3 Các mô-đun ReTiCo trong mạng

ReTiCo có các triển khai mô-đun hiện có để thực hiện các tác vụ chung liên quan đến hệ thống đối thoại. Do đó, ReTiCo được sử dụng trong mạng để hỗ trợ các phân đoạn đó. Cụ thể, ReTiCo có trách nhiệm đọc các đầu vào cảm biến (micro và camera), hiểu ngôn ngữ tự nhiên, quản lý hộp thoại và gửi tín hiệu cuối cùng đến robot để chứng minh hành động đúng đắn.

3.3.1 Mô-đun Rasa NLU (Hiểu ngôn ngữ tự nhiên)

Mô-đun NLU Rasa trong ReTiCo nhận được IU nhận dạng giọng nói từ PSI và tạo ra các IU hành động đối thoại thích hợp chứa hành động và khái niệm thích hợp. Mô-đun này chủ yếu chịu trách nhiệm kiểm soát mọi hành động của robot chỉ liên quan đến đầu vào ngôn ngữ. Đầu ra từ Rasa NLU được gửi đến PyOpenDialog [34] mô-đun để quản lý hộp thoại thích hợp.

3.3.2 Keras Object Feature Extractor và WAC

Mô-đun Keras Object Feature Extraction lấy IU từ Google MaskRCNN

trong PSI để tạo ra một vectơ được đưa vào mô hình phân loại từ (WAC) [35].

Ngoài các IU từ Keras Feature Extractor, WAC cũng nhận được nhận dạng giọng nói

kết quả phân loại từ PSI. Mô hình WAC hoặc mô hình từ như phân loại là một mô hình có cơ sở

mô hình ngữ nghĩa từ vựng có thể liên kết các từ với thế giới vật lý. Nói cách khác,

WAC là một cách để ánh xạ thế giới trực quan vào lời nói của người dùng. Do đó, nó hữu ích

khi robot phải tương tác với thế giới vật lý để đáp ứng một lệnh

hoặc một câu hỏi. Ví dụ, khi người tham gia nói "Khối màu đỏ phải không?", các vectơ

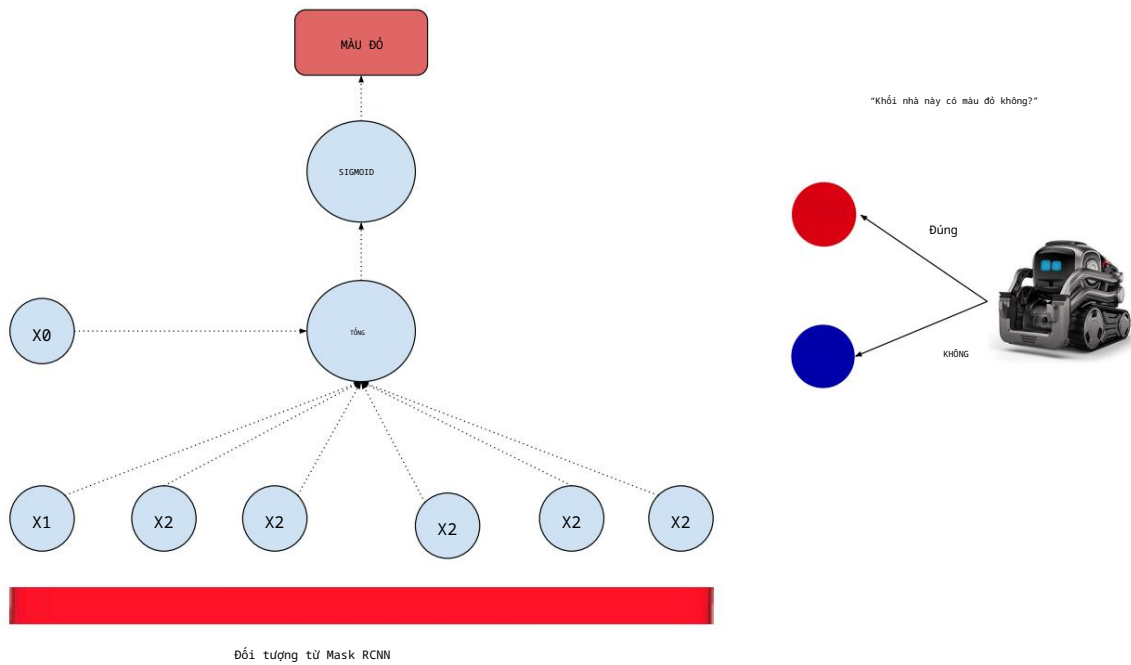
được tạo ra bởi Mask RCNN được đưa vào WAC để quyết định xem "đỏ" có phải là câu trả lời tốt nhất hay không

đối với đối tượng cụ thể và hành động của robot được tạo ra dựa trên thông tin đó

tion. Hình 3.3 cho thấy ý tưởng về việc đưa từ vào các đặc điểm vật lý và cách

quyết định hành động của robot. Đầu ra từ WAC cũng được gửi đến PyOpenDial

Mô-đun quản lý hộp thoại.



Hình 3.3: Các từ ngữ liên quan đến các phương thức vật lý ảnh hưởng đến hành động của rô-bốt.

3.3.3 Mô-đun PyOpenDial

PyOpenDial [34] là trình quản lý hộp thoại kiểm soát hành vi cuối cùng của Cozmo

robot để trả lời một truy vấn. Nó sử dụng IU nhận dạng giọng nói từ PSI ASR, hộp thoại

hành động IU từ ReTiCo RASA NLU và IU khung nối đất từ ReTiCo WAC

trực tiếp hoặc bằng cách nối đất vào mạng IU khi cần thiết. Nói cách khác, trong số

tất cả các mô-đun khác, mô-đun này tận dụng tối đa khả năng di chuyển

mạng phân tán sử dụng kho lưu trữ IU để truy cập các IU có liên quan nhằm đưa ra quyết định.

Mô-đun này báo hiệu cho mô-đun hành động Cozmo để khiến rô-bốt thực hiện một số hành động nhất định

hành động thích hợp.

3.3.4 Khả năng tương tác giữa hai khung

Một trong những yêu cầu cốt lõi để khuôn khổ tổng thể của chúng tôi hoạt động là khả năng tương tác tính của hai hệ thống mục tiêu, hoặc cách chúng giao tiếp với nhau (theo yêu cầu sẵn sàng của robot phân phối). Trong thí nghiệm của chúng tôi, chúng tôi chủ yếu cố gắng hai lựa chọn để đạt được khả năng tương tác. Cụ thể, chúng tôi đã đánh giá các nước ngoài khác nhau các tùy chọn giao diện chức năng có sẵn cho C# và Python, và hàng đợi tin nhắn việc triển khai.

Về mặt giao diện chức năng nước ngoài, sẽ hữu ích và dễ dàng hơn trong điều khoản thực hiện nếu các thư viện này hỗ trợ tất cả các chức năng phức tạp được yêu cầu bởi hai khuôn khổ gia tăng của chúng tôi. Thật không may cho cả hai nền tảng .NET form và Python, các giao diện này chỉ hỗ trợ một tập hợp con giới hạn của một số cấp thấp nhất định ngôn ngữ (ví dụ C++). Ngoài ra còn có các dự án nguồn mở giúp ích cho việc tích hợp sự kết hợp của Python và nền tảng .NET như Python.NET2 và IronPython3 . Trong khi Python.NET cung cấp khả năng tích hợp liền mạch với Ngôn ngữ chung .NET Runtime (CLR) và IronPython là ứng cử viên tốt để triển khai ngôn ngữ lập trình Python nhắm mục tiêu vào .NET Framework, không có ngôn ngữ nào trong số này các dự án có thể truy cập các phương pháp mở rộng chung4 mà PSI sử dụng để thực sự tạo ra đã xử lý và hoàn thiện mạng lưới đường ống vì các tham chiếu đến các chức năng này được giải quyết trong thời gian biên dịch và không thể truy cập trong thời gian chạy.

Tùy chọn hợp lý tiếp theo để đạt được khả năng tương tác là sử dụng hàng đợi tin nhắn vì cả PSI và ReTiCo đều hỗ trợ truyền tin nhắn bằng ZeroMQ, đây là một giải pháp phổ biến

2<https://github.com/pythonnet/pythonnet>

3<https://ironpython.net/>

4<https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/classes-and-structs/phương-thức-mở-rộng>

Thư viện truyền tin nhắn. Nó xây dựng và duy trì các socket mang tin nhắn nguyên tử trên nhiều phương tiện vận chuyển khác nhau. Cả hai khuôn khổ đều sử dụng định dạng JSON để gửi dữ liệu bằng Mã ZeroMQ

3.3.5 Khả năng tương tác giữa các mô-đun trong mạng IU

Từ các nghiên cứu trước đây, chúng tôi đã chứng kiến những nỗ lực giải quyết các vấn đề từ mô-đun này sang mô-đun khác. kết nối. Tuy nhiên, chúng chủ yếu áp dụng cho các kết nối giữa hai mô-đun cụ thể [26; 28] và không chỉ định hoặc giải thích bất kỳ cấu trúc chung nào. Như đã đề cập ở trên, thí nghiệm bao gồm nhiều mô-đun giao tiếp với mô-đun trong phân phối môi trường. Do đó, để thí nghiệm của chúng tôi có hiệu quả thì điều quan trọng là cả hai khung mục tiêu thực hiện đóng gói IU tương tự trong khi gửi IU từ một mô-đun đến khác và tuân theo một cấu trúc nhất định trong quá trình tương tác để đạt được sự hoàn hảo kết nối. Theo mặc định, ReTiCo có bao bì IU tiêu chuẩn cho các mô-đun của mình chứa thông tin tải trọng, IU trước đó được liên kết trong mạng, Grounded-in Thông tin IU và độ tuổi. Mặt khác, PSI ban đầu không có IU tương tự đóng gói. Như đã đề cập ở trên, chúng tôi đã triển khai chức năng đó. Hiện tại PSI giữ IU dưới dạng danh sách trong đó mỗi phần tử trong danh sách là một IU giữ tải trọng, tham chiếu đến IU trước đó và IU Grounded-in, và dấu thời gian, tương tự đến ReTiCo.

Để truyền tải thông tin này, mỗi khi chúng ta tương tác từ một mô-đun này sang cái kia nằm trong một khuôn khổ hoặc quy trình khác, chúng ta cần phải đảm bảo cấu trúc này không bị mất hoặc bị hỏng và tất cả các Liên kết cùng cấp (SLL) đều được bảo tồn. Để đảm bảo điều đó, chúng ta cần đảm bảo rằng chúng ta xử lý các thành phần mạng như các đối tượng [7] và sử dụng một mô-đun phần mềm trung gian trong khuôn khổ đích sẽ thực hiện công việc chuyển đổi dữ liệu thành đối tượng thích hợp và thiết lập lại tất cả cùng một cấp độ

kết nối trong khuôn khổ tiếp nhận trước khi truyền dữ liệu đến phần tiếp theo mô-đun. Tôi đã triển khai phần mềm trung gian này theo hai cách khi xem xét các hiệu ứng về mặt về tính nhất quán và hiệu quả.

Cách tiếp cận đầu tiên yêu cầu gửi một tham chiếu đến toàn bộ mạng SLL mỗi thời gian IU mới được tạo ra. Điều này đảm bảo tính nhất quán tối đa nhưng lại hy sinh hiệu quả hiệu quả. Trong cách tiếp cận này, hiệu quả được cải thiện bằng cách chỉ phân tích các IU mới và bỏ qua những cái cũ trong phần mềm trung gian đích. Cách tiếp cận thứ hai tập trung vào hiệu quả bằng cách chỉ gửi các IU mới được tạo. Trong cách tiếp cận này, phần mềm trung gian đạt được tính nhất quán bằng cách xây dựng lại toàn bộ mạng SLL tại đích theo thứ tự để thiết lập lại các kết nối đối tượng và theo dõi lịch sử.

3.3.6 Bảo tồn và duy trì toàn bộ IU phân phối

Mạng

Các cấu trúc được đề cập trong phần 3.3.4 đảm bảo rằng giao tiếp giữa các mô-đun cation được thiết lập bằng cách chuyển IU đúng cách và bảo quản mức độ giống nhau của chúng Liên kết (SLL). Tuy nhiên, về Liên kết nối đất (GRIN), việc xen kẽ các IU từ một hệ thống này sang hệ thống khác dẫn đến mất tham chiếu đến phần mạng đó nằm trong quy trình hoặc máy khác vì hầu hết các thuộc tính mạng đều nhận được bị mất dẫn đến phá vỡ toàn bộ mạng lưới. Không thể cố gắng vượt qua mạng khi các phần khác nhau của mạng nằm trong các môi trường khác nhau. Để để giải quyết tình trạng tiến thoái lưỡng nan này, chúng tôi đã cố gắng triển khai một kho dữ liệu IU có thể lưu trữ toàn bộ dữ liệu IU được tạo ra bởi các mô-đun trong các quy trình. Bất kể loại nào và chiến lược triển khai, tất cả các triển khai cửa hàng IU của chúng tôi (triển khai (chi tiết như sau) có các chức năng chính sau:

- InsertIU: Hàm này chèn một IU vào kho lưu trữ.

- RetrieveIU: Với ID hoặc khóa của IU, hàm này trả về giá trị cụ thể đó

File: map.1000000

- GetPreviousIU: Với ID hoặc khóa của IU, hàm này trả về giá trị trước

IU viious của IU cụ thể đó (ví dụ: cung cấp ID của IU chứa từ

“now” sẽ trả về IU chứa từ “leave” trong Hình 1.2)

- GetGroundedInIU: Hàm này trả về IU được nối đất của một IU cụ thể

cung cấp ID của IU cụ thể đó (ví dụ cung cấp ID của IU chứa

tải trọng “Trạng từ” sẽ trả về IU chứa từ “bây giờ” trong Hình 1.2)

Theo cách này, các mô-đun trong các quy trình khác nhau có thể sử dụng các hàm này để truy xuất bất kỳ dữ liệu nào cần thiết để chạy đường ống và không cần phải nghĩ đến sự phức tạp thực hiện. Các chức năng này tạo ra một yêu cầu lưu trữ IU (chức năng RetrieveIU yêu cầu hoặc truy vấn cho yêu cầu “Lấy lại”) đến cơ sở dữ liệu ẩn để lấy dữ liệu có liên quan. Nói cách khác, các loại lưu trữ IU khác nhau mà tôi đã triển khai tạo điều kiện thuận lợi cho việc duyệt logic của mạng gia tăng và do đó là mạng IU được bảo tồn và tránh “phá vỡ”. Tôi đã triển khai hai cách chính để tạo điều kiện thuận lợi cho việc này kho dữ liệu IU logic cho hệ thống hợp thoại sẵn sàng cho robot như sau.

- Lưu trữ IU trong PSI: Trong cách tiếp cận đầu tiên của chúng tôi, chúng tôi đã chọn một khuôn khổ (PSI) chịu trách nhiệm lưu trữ toàn bộ mạng IU dưới dạng cơ sở dữ liệu. Cả hai đều được chọn khung (PSI) và khung khác (ReTiCo) gửi mạng IU của họ thông tin vào cơ sở dữ liệu. PSI có thể lưu trữ IU của mình trực tiếp vào cửa hàng trong khi ReTiCo gửi IU của mình đến PSI bằng cách sử dụng yêu cầu lưu trữ IU truyền tin. Theo cách này, PSI nắm giữ và xem toàn bộ mạng IU trực tiếp trong khi ReTiCo truy vấn PSI để lấy thông tin nhất định từ mạng. (Chúng tôi coi hai hệ thống của mình là

máy khách (ReTiCo) và máy chủ (PSI), và tuân theo giao thức ghi từ xa5.

khung phục vụ (PSI) giữ một cơ sở dữ liệu lưu trữ mạng từ cả hai

khuôn khổ. Vì chỉ có một máy chủ trong trường hợp của chúng tôi, cách tiếp cận này cho phép

chúng ta đạt được sự nhất quán tuần tự6 . Trong khi khuôn khổ người hầu có trực tiếp

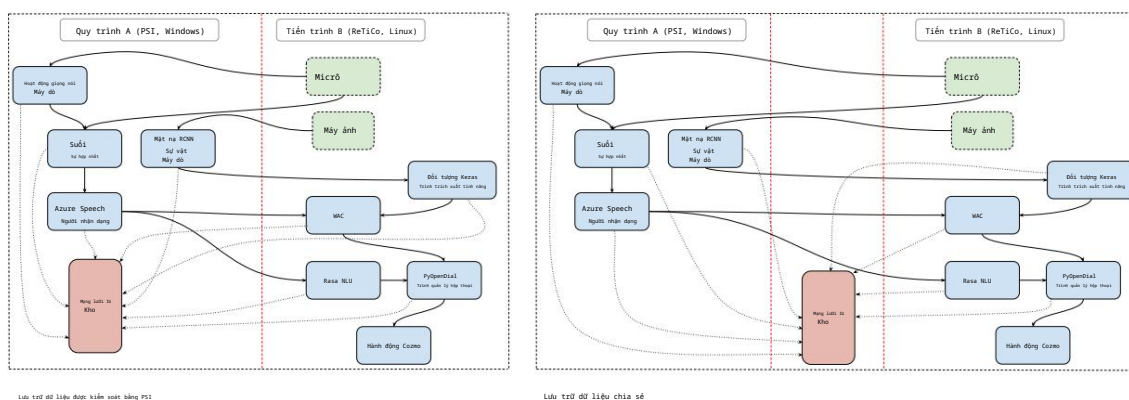
truy cập vào bộ lưu trữ, khung máy khách khác truy vấn khung máy chủ

để biết thông tin có liên quan.) Chúng tôi đã thử hai loại triển khai gốc của

lưu trữ PSI: một bằng C# vì PSI được viết bằng C# và một bằng C++ để

so sánh hiệu quả của việc triển khai ở mức độ tương đối thấp hơn với bản gốc

Triển khai C# và cách tiếp cận thứ hai.



Hình 3.4: Cấu trúc lưu trữ mạng IU phân tán.

- Lưu trữ IU trong Redis: Trong cách tiếp cận thứ hai, chúng tôi đã thử chia sẻ dữ liệu

lưu trữ mà cả hai khuôn khổ của chúng tôi đều có quyền truy cập và có thể lưu trữ tương ứng

IUs được tạo ra. Một trong những vấn đề cơ bản trong bất kỳ hệ thống phân tán nào là

đạt được sự nhất quán tuần tự với các hoạt động đồng thời. Trong bất kỳ IU nào

mạng, không có IU nào được tạo ra trước khi các IU và mô-đun cha của nó có rõ ràng

5https://en.wikipedia.org/wiki/Consistency_model#Remote-write_protocols

6https://en.wikipedia.org/wiki/Sequential_consistency

độ trễ giữa chúng. Do đó, việc triển khai lưu trữ dữ liệu được chia sẻ đối với một mạng IU phân tán đạt được tính nhất quán tuần tự, vì chúng ta chỉ xử lý một máy chủ dữ liệu logic. Đối với cách tiếp cận này, chúng tôi đã sử dụng data-structure-project Redis [36]. Những loại dự án này được biết là có liên quan nhanh chóng, hiệu quả và hỗ trợ bộ nhớ chia sẻ [37]. Hình 3.4 cho thấy so sánh giữa hai cách tiếp cận.

3.4 Cấu trúc mạng tổng thể cuối cùng

Hình 3.1 cho thấy mạng cuối cùng tôi đã sử dụng trong luận án này. Chúng tôi đã sử dụng hai máy để thiết lập hai khuôn khổ của chúng tôi. PSI được đặt trong một máy Windows (Windows 10 máy tính xách tay) trong khi ReTiCo được đặt trong máy Linux (máy tính để bàn Ubuntu). Lưu trữ IU gốc nằm trong máy Windows (trong PSI cho dù đó là C# hay C++) và phiên bản Redis được lưu trữ trên cùng một máy Linux với ReTiCo. Cả hai các máy được kết nối với cùng một mạng riêng. Máy Windows là được kết nối bằng kết nối Wi-Fi trong khi máy tính để bàn Linux được kết nối có dây vào mạng.

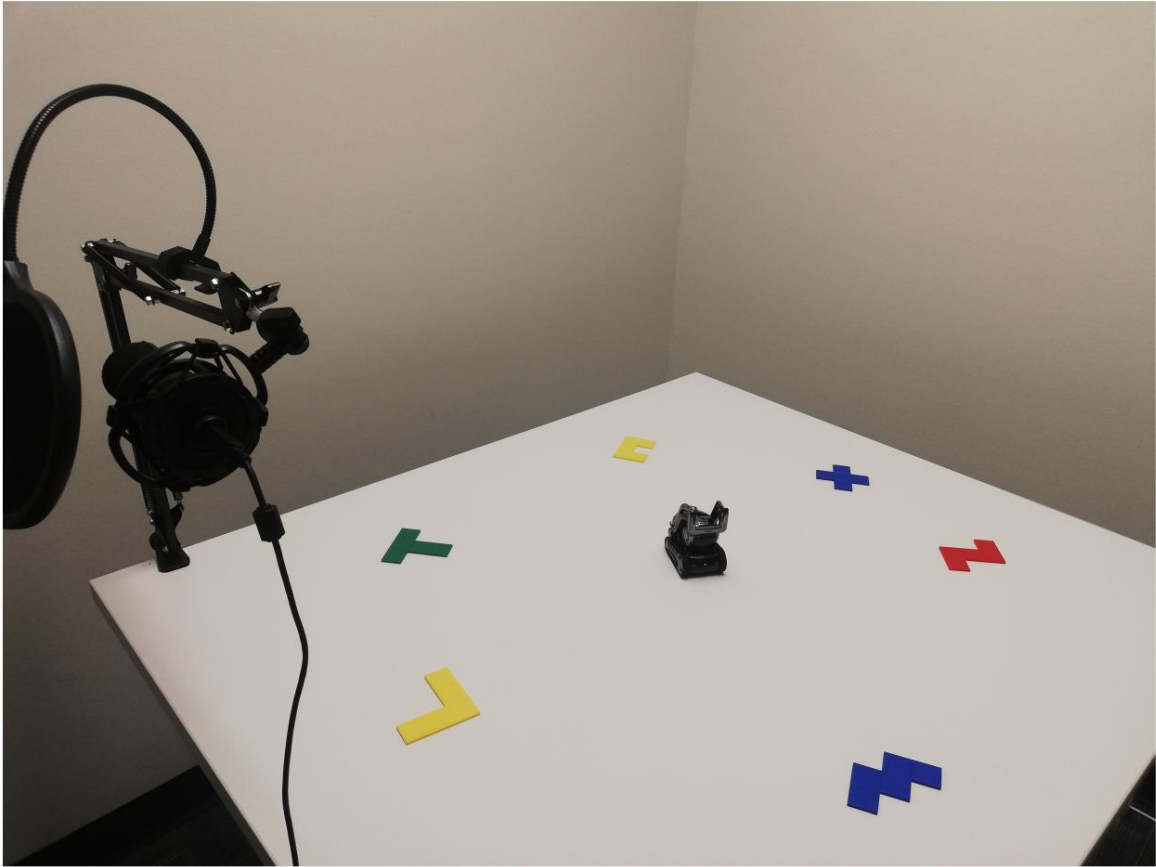
3.5 Tiêu chí đánh giá

Để đánh giá hệ thống của chúng tôi với tất cả các tổ hợp phép được đề cập ở trên, chúng tôi chủ yếu tập trung vào tính nhất quán và hiệu quả của các bộ phận khác nhau của hệ thống và hệ thống nói chung. Chúng tôi chia đánh giá của mình thành hai phần, một trong số đó được đánh giá một cách có hệ thống về tính nhất quán và hiệu quả của mạng lưới, và các thí nghiệm sử dụng đánh giá trực tiếp hệ thống với con người.

Thí nghiệm đầu tiên tập trung vào hai số liệu: số liệu đầu tiên chỉ tập trung vào tính nhất quán độ bền của hệ thống. Chúng tôi đã kiểm tra xem các kết nối có nhất quán trong suốt

toàn bộ mạng. Điều này bao gồm việc kiểm tra xem mỗi mô-đun-tới-mô-đun com-truyền thông là nhất quán (ví dụ đầu ra ASR từ PSI tạo ra IU thích hợp trong ReTiCo NLU mà không mất bất kỳ dữ liệu nào), kết quả cuối cùng của toàn bộ quá trình xây dựng là chính xác, và toàn bộ mạng được lưu trữ đúng cách. Để làm được điều này, chúng tôi đã tạo ra đơn vị kiểm tra cho mỗi giao tiếp giữa các mô-đun khi các mô-đun được đặt trong các hệ thống khác nhau. Các bài kiểm tra đơn vị cũng được tạo ra để kiểm tra đầu ra của tổng thể hệ thống. Kiểm tra đầu ra cuối cùng bằng các bài kiểm tra đơn vị cũng đảm bảo rằng toàn bộ mạng công việc được bảo tồn và nhất quán vì một số mô-đun tìm kiếm lịch sử trong Mạng IU thực hiện đúng (ví dụ mô-đun quản lý hộp thoại PyOpenDial yêu cầu đối với từ nhận dạng giọng nói liên quan đến Rasa NLU IU).

Chỉ số thứ hai dựa trên việc xác định hiệu quả. Mặc dù toàn bộ mạng lưới công việc là nhất quán, chúng tôi cần kiểm tra xem toàn bộ quá trình có hiệu quả và thực sự khả thi cho các mạng lớn hơn. Chúng tôi đã đánh giá mô-đun-to-mô-đun và kết nối tổng thể của chúng tôi của mạng bằng cách tiếp cận tương tự như [38] và đánh giá việc triển khai của chúng tôi dựa trên thời gian chờ và tỷ lệ thành công của việc chuyển dạ.



Hình 3.5: Robot Cozmo trong quá trình thực hiện nhiệm vụ.

Thí nghiệm thứ hai dựa trên sự tương tác của con người. Mặc dù sự khác biệt các số liệu được đề cập ở trên cho thấy việc triển khai của tôi thực sự hoạt động như mong đợi ở cấp độ triển khai, điều quan trọng là phải đánh giá toàn bộ hệ thống đã đề cập ở trên ảnh hưởng đến tương tác với con người khi chúng ta triển khai robot phân tán sẵn sàng hệ thống đối thoại chuẩn. Để kiểm tra điều đó, chúng tôi đã thiết kế một thiết lập nhiệm vụ tương tự như [5] sử dụng robot Cozmo. Hình 3.5 cho thấy thiết lập nhiệm vụ của robot Cozmo cho thiết kế thí nghiệm của chúng tôi. Cozmo được bao quanh bởi các vật thể màu, có thể nhìn thấy nó xung quanh, lắng nghe con người và chờ con người tương tác với nó. Chi tiết giải thích về thiết lập và kết quả của tương tác giữa người và robot được trình bày trong

mục 4.2.

3.6 Giả thuyết

Chúng tôi đưa ra giả thuyết rằng việc thiết lập giao tiếp giữa hai khuôn khổ bằng cách sử dụng hai phương pháp đã đề cập để thiết lập giao tiếp giữa các mô-đun sẽ là nhất quán. Tuy nhiên, sẽ hiệu quả hơn nếu lựa chọn thứ hai chỉ gửi các khối dữ liệu mới và xây dựng lại mạng IU tại đích. Đối với việc giữ toàn bộ mạng IU, chúng tôi đưa ra giả thuyết rằng toàn bộ mạng IU sẽ nhất quán, được bảo quản và có thể truy cập từ cả hai quy trình với cả hai tùy chọn lưu trữ IU, như thế nào-bao giờ hết, mạng sẽ hiệu quả hơn khi IU được lưu trữ gốc trong một trong các quá trình.

CHƯƠNG 4:

ĐÁNH GIÁ

4.1 Thí nghiệm 1: Đánh giá có hệ thống

Sự nhất quán và hiệu quả

Như đã đề cập trong các chương trước, mục tiêu của luận án này là chứng minh một hệ thống đường ống đối thoại robot sẵn sàng hoạt động trong kiến trúc phân tán nơi chúng tôi sử dụng hai quy trình (PSI và ReTiCo) để thực hiện phân công việc hợp lý của họ giúp kết hợp với nhau bằng những đặc điểm độc đáo của chúng. Để đạt được điều đó, chúng tôi đã triển khai hệ thống đường ống đối thoại sẵn sàng cho robot được hiển thị trong Hình 3.1 và chúng tôi đã triển khai ba loại kho dữ liệu IU để đạt được khả năng lưu trữ toàn bộ mạng IU công việc được tạo ra bởi đường ống để cả hai quy trình có thể đi qua mạng khi cần thiết. Chúng tôi đã đánh giá đường ống và các triển khai khác nhau của dữ liệu lưu trữ riêng biệt.

4.1.1 Đánh giá IU Store

Để tạo điều kiện thuận lợi cho việc bảo tồn toàn bộ Mạng lưới IU, chúng tôi đã xây dựng hai các biến thể chính của kho dữ liệu IU như đã trình bày trước đó trong Hình 3.4. Đầu tiên loại kho IU được xây dựng bằng cách chọn một quy trình (PSI) và triển khai nó trong quá trình đó. Quá trình khác (ReTiCo) dựa vào ZeroMQ để truy vấn

thông tin có liên quan (ví dụ yêu cầu IU nối đất của một IU cụ thể) từ quá trình khác. Chúng tôi đã xây dựng hai biến thể của kho IU gốc: một trong C# kể từ PSI bản thân nó dựa trên môi trường .NET và một môi trường khác trong C++ gốc để khám phá hiệu quả của một ngôn ngữ cấp thấp hơn.

Biến thể thứ hai của kho lưu trữ IU được xây dựng như một kho lưu trữ dữ liệu được chia sẻ mà cả hai quy trình đều có quyền truy cập và chúng đều có thể truy vấn kho lưu trữ IU trực tiếp lấy dữ liệu có liên quan. Như thể hiện trong Hình 3.4, kho lưu trữ IU này hoạt động như một trung gian một thực thể độc lập và nắm giữ toàn bộ mạng IU được tạo ra bởi cả hai khuôn khổ. Chúng tôi đã sử dụng kho lưu trữ cấu trúc dữ liệu trong bộ nhớ Redis để triển khai IU này của hàng.

Như đã đề cập trong phần 3.3.6, bất kể loại và chiến lược triển khai, tất cả các triển khai lưu trữ IU của chúng tôi đều có các chức năng chính sau: InsertIU, RetrieveIU, GetPreviousIU và GetGroundedInIU. Tất cả các chức năng này đều đã được thử nghiệm về độ trễ và tỷ lệ thành công của việc chuyển giao IU. Chúng tôi đã thử nghiệm kho IU của mình bằng cách tạo, chen và truy xuất các IU hiện tại, trước đó và được nối đất của 5000 IU và tính trung bình thời gian trễ và tỷ lệ thành công phân phối IU trong 50 lần lặp lại cho mỗi biến thể. Đối với đối tác PSI, chúng tôi đã tạo ra 5000 Nhận dạng giọng nói giả và IU Nhận dạng giọng nói gia tăng, mỗi IU được kết nối với IU trước đó tương ứng của nó và các IU nối đất, trong khi đối với đối tác ReTiCo, cùng một lượng Dialog IU hành động (Rasa) và IU quyết định đối thoại (PyOpenDial) được tạo ra với cùng một yêu cầu được đáp ứng cho mỗi lần lặp lại. Đối với thử nghiệm này, tôi đã sử dụng bản hoàn thành cuối cùng cấu trúc mạng được đề cập trong phần 3.4. Bảng 4.1 và Bảng 4.2 hiển thị kết quả dành cho tất cả các biến thể của cửa hàng IU dành cho PSI và ReTiCo.

Chức năng	Loại cửa hàng	Tỷ lệ thành công của việc chuyển dạ IU trễ
Lấy lạiIU	Bản địa (C#)	0,003ms 100%
	Bản địa (C++)	0,098ms 100%
	Đã chia sẻ (Redis)	6.249ms 100%
LấyIU trước đó	Bản địa (C#)	0,007ms 100%
	Bản địa (C++)	0,203ms 100%
	Chia sẻ (Redis)	12.513ms 100%
Nhận được nền tảng trong IU	Bản địa (C#)	0,005ms 100%
	Bản địa (C++)	0,151ms 100%
	Chia sẻ (Redis)	10.498ms 100%

Bảng 4.1: Độ trễ và tỷ lệ thành công của việc cung cấp IU của từng chức năng cho từng loại cửa hàng cho PSI

Chức năng	Loại cửa hàng	Tỷ lệ thành công của việc chuyển dạ IU trễ
Lấy lạiIU	Bản địa (C#)	15.661ms 100%
	Bản địa (C++)	15.941ms 100%
	Chia sẻ (Redis)	0,098ms 100%
LấyIU trước đó	Bản địa (C#)	16.659ms 100%
	Bản địa (C++)	16.744ms 100%
	Chia sẻ (Redis)	0,331ms 100%
Nhận được nền tảng trong IU	Bản địa (C#)	16.570ms 100%
	Bản địa (C++)	16.605ms 100%
	Chia sẻ (Redis)	0,163ms 100%

Bảng 4.2: Độ trễ và tỷ lệ thành công của việc cung cấp IU của từng chức năng cho từng loại cửa hàng cho ReTiCo

Như chúng ta có thể thấy, một triển khai gốc (C#, C++) ưu tiên quy trình gốc với độ trễ rất thấp, nhưng lại tốn kém cho quá trình khác do phải truyền tin nhắn các ràng buộc về việc thực hiện và biên dịch (ví dụ tuần tự hóa/hủy tuần tự hóa liên tục và đi qua mạng để đến được tiến trình đích). Mặt khác, việc triển khai lưu trữ dữ liệu chia sẻ cho thấy hiệu suất tốt hơn so với triển khai thô phương pháp sử dụng việc truyền tin nhắn như một phương tiện giao tiếp. Redis thực hiện tốt hơn cho ReTiCo so với PSI vì lưu trữ dữ liệu được chia sẻ được tạo điều kiện thuận lợi trong cùng một máy như ReTiCo. Để xác định xem điều đó có đúng không hoặc nếu có các yếu tố khác cần xem xét, tôi đã thực hiện nhiều bài kiểm tra ứng suất hơn cho riêng Redis bằng cách di chuyển Redis vào máy thứ ba và di chuyển Redis đến máy có PSI.

Bảng 4.3 hiển thị kết quả cho Redis được đặt vào máy thứ ba, trong khi

Bảng 4.4 hiển thị kết quả khi Redis được đặt trong cùng một máy với PSI.

Hai bảng này cho thấy hiệu suất của kho IU dùng chung cũng phụ thuộc vào của kết nối. Cụ thể, thực tế là máy PSI đã được kết nối bằng Wi-Fi dẫn đến độ trễ tăng lên cho bất kỳ giao tiếp nào với mục đích cụ thể đó máy. Ngoài ra, chúng ta có thể thấy rằng việc triển khai C++ gốc đã vượt trội hơn được hình thành bởi việc triển khai C# gốc trên mọi khía cạnh do sự phụ thuộc của nó vào tuần tự hóa và hủy tuần tự hóa các đối tượng. Vì lý do đó, chúng tôi chỉ sử dụng bản địa Triển khai C# và triển khai chia sẻ của Redis trong Thí nghiệm 2.

Chức năng	Tỷ lệ thành công của quá trình phân phối IU độ trễ		
Lấy lạiIU	PSI	6.658ms	100%
	ReTiCo	0.528ms	100%
LấyIU trước đó	PSI	13.296ms	100%
	ReTico	1.918ms	100%
Nhận được nền tảng trong IU	PSI	11.063ms	100%
	ReTiCo	1.099ms	100%

Bảng 4.3: Độ trễ và tỷ lệ thành công phân phối IU của mỗi Chức năng cho Kho lưu trữ IU dùng chung (Redis) khi nó nằm trên máy thứ ba.

Chức năng	Tỷ lệ thành công của quá trình phân phối IU độ trễ		
Lấy lạiIU	PSI	0,988ms	100%
	ReTiCo 6.4	72ms	100%
LấyIU trước đó	PSI	1,977ms	100%
	ReTico 22	216ms	100%
Nhận được nền tảng trong IU	PSI	1.643ms	100%
	ReTiCo 11	094ms	100%

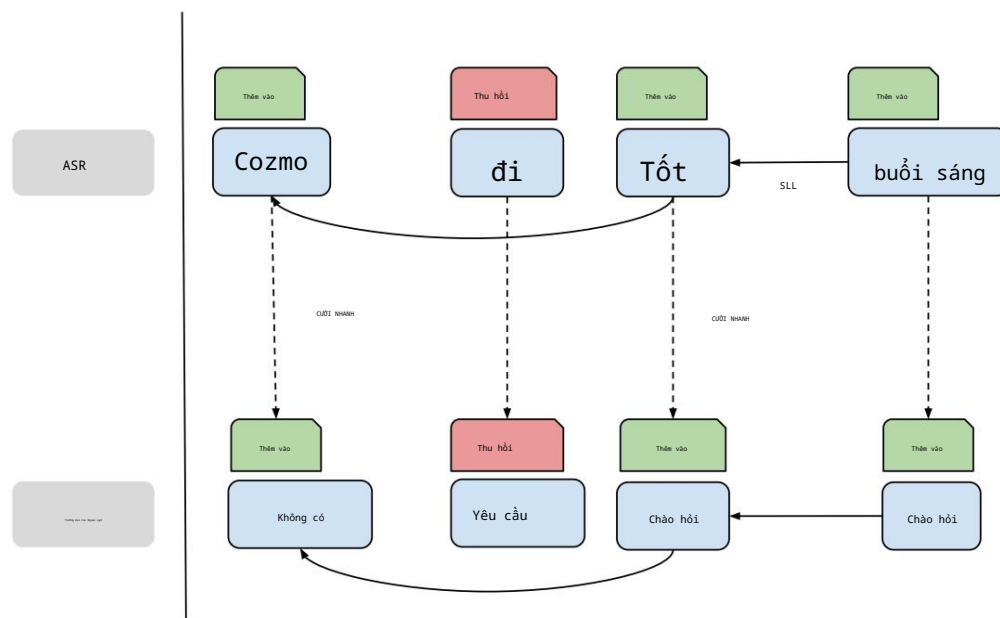
Bảng 4.4: Độ trễ và tỷ lệ thành công phân phối IU của từng chức năng cho Kho lưu trữ IU dùng chung (Redis) khi nó nằm trên cùng một máy với PSI.

Dựa trên dữ liệu được thể hiện trong Bảng 4.1 đến Bảng 4.4, chúng ta có thể đi đến kết luận sau đây:

- Nếu chúng ta xây dựng một đường ống phân tán trong đó một quy trình đi qua rộng rãi hơn mạng để có được thông tin hơn quá trình khác, sau đó thực hiện Lưu trữ IU theo quy trình đó sẽ cho phép quy trình đó truy cập vào lưu trữ IU thông tin với độ trễ rất thấp. Nói cách khác, một triển khai gốc là được ưa chuộng trong trường hợp đó.
- Hiệu suất cũng phụ thuộc rõ ràng vào khả năng kết nối. Một en-kết nối đóng và có dây đảm bảo giao tiếp tốt hơn cho cửa hàng IU hơn kết nối không dây.
- Mặc dù độ trễ cho kho IU được chia sẻ (Redis) không thấp bằng kho gốc lưu trữ cho quy trình gốc, một triển khai được chia sẻ được ưu tiên khi các quy trình phân tán sử dụng kho lưu trữ IU như nhau hoặc gần như nhau.

4.1.2 Phân tích sâu hơn về đường ống

Vì đường ống được phân phối, chúng tôi đã tạo các bài kiểm tra đơn vị cho giao tiếp giữa các mô-đun các phân hệ trong đó các mô-đun được đặt trong các quy trình khác nhau để đảm bảo rằng IU từ các mô-đun nguồn tạo ra các IU phù hợp trong mô-đun đích. Ví dụ, IU nhận dạng giọng nói được tạo ra trong các mô-đun Nhận dạng giọng nói PSI phải tạo ra các IU NLU Rasa chính xác trong ReTiCo như thể hiện trong Hình 4.1. Các bài kiểm tra đơn vị đánh giá đã áp dụng cả hai cách tiếp cận được đề cập trong phần 3.3.5. Cả hai cách tiếp cận gửi tham chiếu đến toàn bộ mạng SLL và chỉ gửi các IU mới được tạo và xây dựng lại mạng trong mô-đun đích ổn định, nhất quán và bảo toàn mạng. Tuy nhiên, trong quá trình thử nghiệm nghiêm ngặt của chúng tôi, chúng tôi phát hiện ra rằng ứng dụng đầu tiên proach không phải lúc nào cũng ổn định đối với các mô-đun xử lý dữ liệu cảm biến (ví dụ: micro hoặc camera). Vì giao tiếp giữa hai quy trình được thực hiện bằng ZeroMQ, gửi tham chiếu tổng thể đến toàn bộ SLL nhiều lần từ một nguồn rất nhanh và đầy đủ (ví dụ như micro) làm cho mạng không ổn định. Do thực tế là các đánh giá còn lại được thực hiện chỉ bằng cách sử dụng phương pháp thứ hai của việc gửi các IU mới được tạo và xây dựng lại chúng trong mô-đun đích.



Hình 4.1: NLU tạo ra các IU phù hợp tương ứng với các IU ASR.

Khi nói đến hiệu quả, chúng tôi đã ghi lại thời gian trung bình cần thiết để một IU được tạo ra bắt đầu từ thời điểm các IU đầu vào cho mô-đun tương ứng đó được đã tạo. Thông tin này được minh họa trong Bảng 4.5. Ví dụ, bảng này hiển thị thời gian cần thiết để tạo ra một Rasa NLU IU trong ReTiCo từ thể hệ thời gian của IU đầu vào của nó (IU Nhận dạng giọng nói trong PSI). Chúng ta có thể thấy rằng các mô-đun nằm trong các quá trình riêng biệt có độ trễ cao hơn về mặt tạo IU trong mô-đun đích do sự phụ thuộc của nó vào việc truyền tin nhắn và các tác vụ tuần tự hóa giữa hai quy trình. Hơn nữa, mô-đun PyOpenDial sử dụng kho dữ liệu rộng rãi nhất bằng cách gọi các hàm `GetGroundedInIU` và `GetPreviousIU`, vì nó cần thông tin về một số loại IU (ví dụ: Nhận dạng giọng nói, Đối tượng Phát hiện, Hiểu ngôn ngữ tự nhiên). Vì chúng tôi đã tiến hành với cấu trúc được đề cập trong phần 3.4, chúng ta có thể thấy lợi thế rõ ràng của việc triển khai một

lưu trữ dữ liệu vì Redis trả lại dữ liệu cần thiết cho PyOpenDial (là một phần của Máy ReTiCo trong quá trình triển khai của chúng tôi) trong 95ms, nơi lưu trữ gốc có trung bình độ trễ 301ms.

Bảng 4.5: Độ trễ tạo IU ở các mô-đun khác nhau.

Loại cửa hàng	Rasa NLU IU Keras Trích xuất tính năng IU PyOpenDial DM IU		
Đỏ lại	0,613 giây	0,851 giây	0,106 giây
Tự nhiên	0,581 giây	0,843 giây	0,286 giây

4.2 Thí nghiệm 2: Học tập tương tác trực tiếp với

Người tham gia là con người

Mặc dù chúng tôi có kết quả về số lượng và kết cấu để chứng minh điều đó phương pháp này được ưa chuộng hơn các phương pháp khác dựa trên việc tạo kho lưu trữ dữ liệu IU và mô-đun IU xây dựng mạng lưới, điều quan trọng là phải biết về tác động của chúng đối với một cuộc đối thoại thực tế thiết lập hệ thống nơi con người tương tác với hệ thống. Để đánh giá điều này hiệu ứng, chúng tôi đã tiến hành một thí nghiệm trực tiếp với người dùng sử dụng robot Cozmo trong một bối cảnh như thể hiện trong Hình 3.5. Hơn nữa, đối với nghiên cứu tương tác trực tiếp với con người, tôi quyết định tiếp tục với thiết lập được đề cập trong phần 3.4 để có được kết quả đa dạng hơn từ hai triển khai lưu trữ IU, một trong số đó được triển khai gốc với một kết nối có dây tránh xa quy trình sẽ sử dụng cửa hàng nhiều nhất và cái kia được thực hiện trong một cấu trúc chia sẻ trong cùng một máy với quy trình sẽ sử dụng cửa hàng nhiều nhất.

4.2.1 Tuyển dụng người tham gia và bối cảnh nghiên cứu

Chúng tôi đã tuyển dụng mười bốn người tham gia nghiên cứu để tương tác với robot Cozmo hai lần, mỗi lần là một khoảng thời gian mười phút trong suốt một phiên duy nhất. Sau

mỗi lần tương tác kéo dài mười phút, những người tham gia được yêu cầu điền vào một bảng câu hỏi chứa tất cả các câu hỏi từ Bản câu hỏi Godspeed (có trong Phụ lục)

[39] cùng với các câu hỏi bổ sung liên quan đến định dạng cụ thể của chúng tôi về việc thiết lập nhiệm vụ.

Các câu hỏi bổ sung như sau:

- Robot phản hồi yêu cầu/lệnh của bạn như thế nào?
- Robot phản hồi chính xác đến mức nào với yêu cầu/lệnh của bạn?
- Robot đã xác định đúng màu của vật thể bao nhiêu lần?
- Bạn đã giúp robot di chuyển đến vật thể mong muốn bao nhiêu lần?
- Bạn có nghĩ rằng robot phản hồi bạn trong khoảng thời gian hợp lý không?

Bản câu hỏi Godspeed là bản câu hỏi được thang điểm Likert với 24 câu hỏi từ đánh giá tiêu cực đến tích cực về tính nhân hóa, sự hoạt bát của robot, khả năng thích ứng, trí thông minh được nhận thức và sự an toàn được nhận thức. Toàn bộ nghiên cứu có áp dụng khoảng một giờ cho mỗi người tham gia. Để đổi lấy thời gian quý báu của họ, những người tham gia được trả tám đô la Mỹ. Những người tham gia nghiên cứu chủ yếu là sinh viên đại học được tuyển dụng từ Khoa Khoa học Máy tính tại Boise State Đại học, mặc dù 28,5% người tham gia đến từ các chuyên ngành khác nhau. Trong khi 57% số người tham gia là người bản ngữ, số người tham gia còn lại là người gần Na-Có năm người tham gia là phụ nữ và chín người là nam giới.

Chúng tôi đã triển khai hai phiên bản của kho lưu trữ IU (Gốc (C#) và Chia sẻ (Redis)) để xây dựng đường ống được hiển thị trong Hình 3.1 để robot sử dụng trong hai mươi các phiên làm việc nhỏ trong đó robot có ba chức năng chính như sau:

- Người dùng có thể chào robot và robot sẽ chào lại (ví dụ: Xin chào, Tốt buổi sáng, tạm biệt).
- Người dùng có thể ra lệnh cho robot và khiến nó di chuyển xung quanh nhiệm vụ được thiết lập (ví dụ: Đi tiếp, Rẽ trái, Lái xe trở lại, Dừng lại, Tiếp tục đi, v.v.) như thể hiện trong Hình 3.5.
- Khi robot ở gần một vật thể, người dùng có thể đặt câu hỏi về màu sắc của các khối và mong đợi phản hồi thích hợp (ví dụ: Khối có màu gì?, Có phải

màu xanh da trời?).

Nhà nghiên cứu vẫn có mặt gần bối cảnh nhiệm vụ để theo dõi tình trạng robot, khắc phục mọi sự cố có thể phát sinh và trả lời mọi câu hỏi hoặc các câu hỏi mà người tham gia có thể có trong suốt quá trình tương tác với robot. Nhà nghiên cứu được phép cung cấp một bộ mẹo huấn luyện hạn chế cho người tham gia trong quá trình tương tác. Một phần của nghiên cứu được quan sát bằng máy ảnh, đã ghi lại âm thanh và video từ tương tác. Sau mỗi tương tác, người dùng di chuyển đến chỗ ngồi của nhà nghiên cứu để hoàn thành bảng câu hỏi liên quan trên máy tính xách tay của nhà nghiên cứu. Sau khi hoàn thành cả hai tương tác và tiếp theo khảo sát, người tham gia được trả tám đô la và ký vào mẫu xác nhận đã nhận của thanh toán.

4.2.2 Kết quả

Hình 4.2 đến Hình 4.6 cho thấy câu trả lời cho câu hỏi cụ thể về “thiết lập nhiệm vụ”

và chúng ta có thể thấy rằng có một mô hình chung về sở thích của cửa hàng

loại. Đối với bốn câu hỏi đầu tiên, mặc dù phần lớn người tham gia nghĩ rằng

robot hoạt động chính xác cho cả hai loại, một số người tham gia báo cáo rằng

phản hồi của robot có thể tốt hơn cho loại gốc (C#). Điều này có nghĩa là những người tham gia

những người tham gia đặc biệt nhận ra sự khác biệt về thời gian phản hồi giữa bộ nhớ chia sẻ

được triển khai trong cùng một máy và bộ lưu trữ gốc được triển khai trong máy kia

quá trình. Ví dụ, đối với câu hỏi đầu tiên (như thể hiện trong Hình 4.2), 35,71%

những người tham gia đồng ý rằng robot phản hồi lệnh mọi lúc cho cả hai

các loại, 64,29% và 50% số người tham gia đồng ý rằng robot đã phản hồi

lệnh hầu hết thời gian cho loại cửa hàng Chia sẻ và Bản địa tương ứng. Tuy nhiên,

14,29% người tham gia báo cáo rằng phản ứng của robot đối với loại Bản địa

có thể tốt hơn. Đối với câu hỏi cụ thể thứ năm về “thiết lập thử nghiệm” về việc liệu robot

đã trả lời người tham gia trong khoảng thời gian hợp lý, 85,72% và 64,28%

của những người tham gia đồng ý với câu trả lời tích cực cho sự tôn trọng kiểu Chia sẻ và Bản địa

tích cực. Tuy nhiên, 14,28% và 35,72% số người tham gia cho loại Chia sẻ và Bản địa

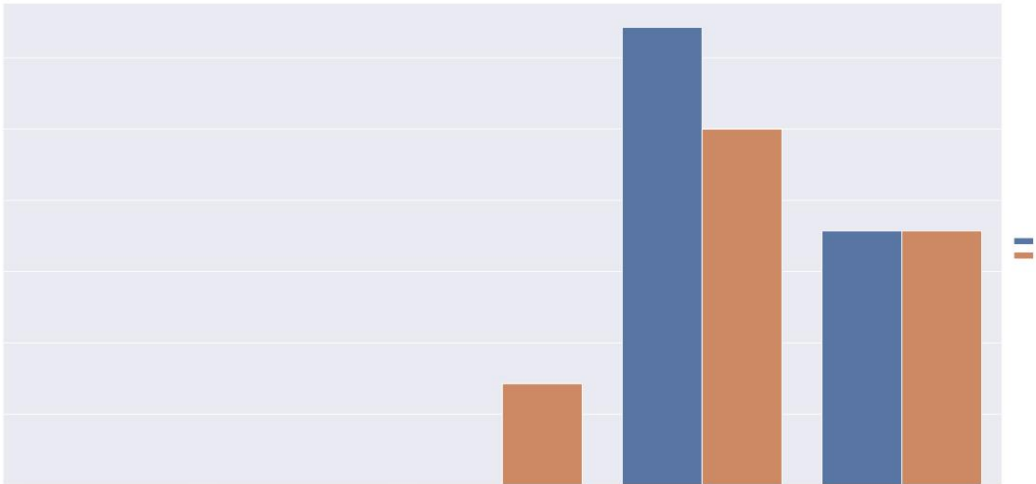
tương ứng đồng ý rằng thời gian phản hồi có thể được cải thiện. Ngoài ra,

ý nghĩa của mỗi câu hỏi “nhiệm vụ cụ thể” cho từng loại IU Store được hiển thị trong

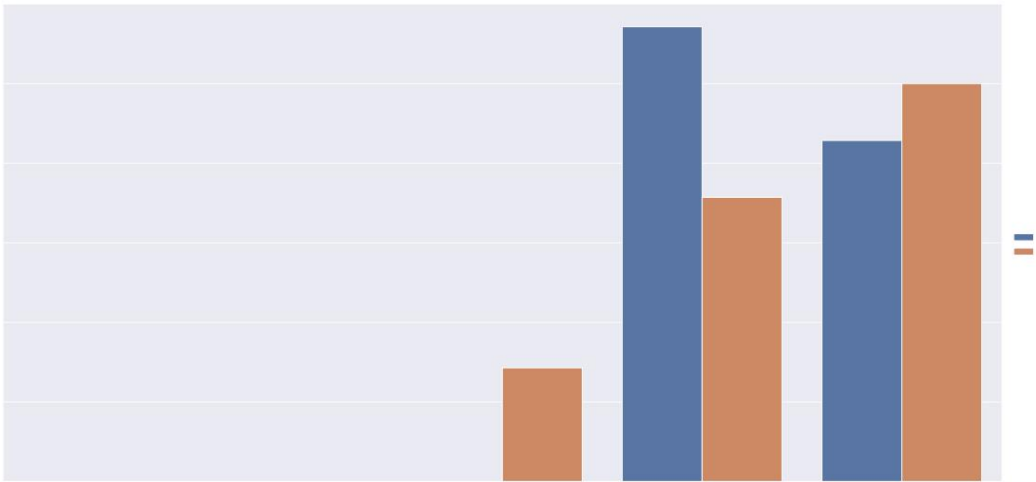
Hình 4.7 và Bảng 4.6 cho thấy rõ ràng rằng mặc dù cả hai loại đều có tác động tích cực

có nghĩa là, kho IU dùng chung được triển khai trong cùng một máy là mong muốn về mặt

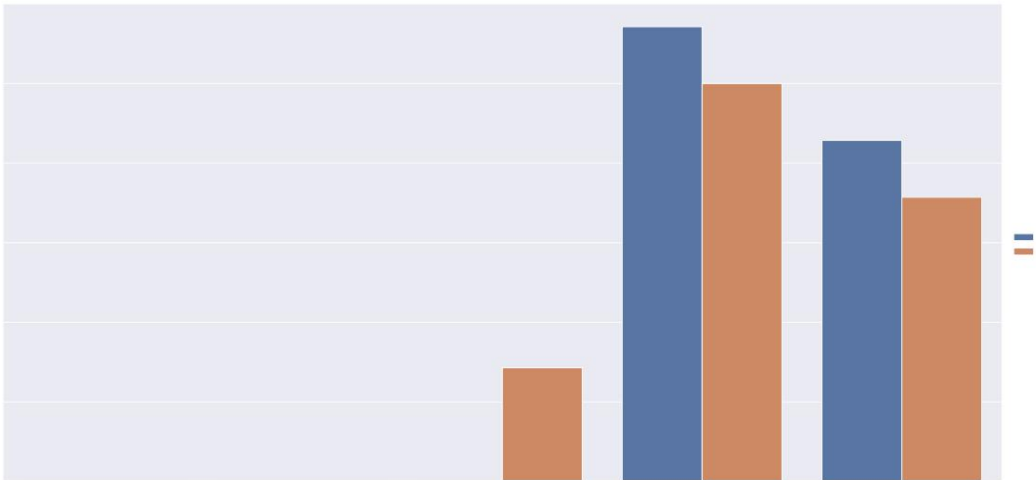
về hiệu suất của robot.



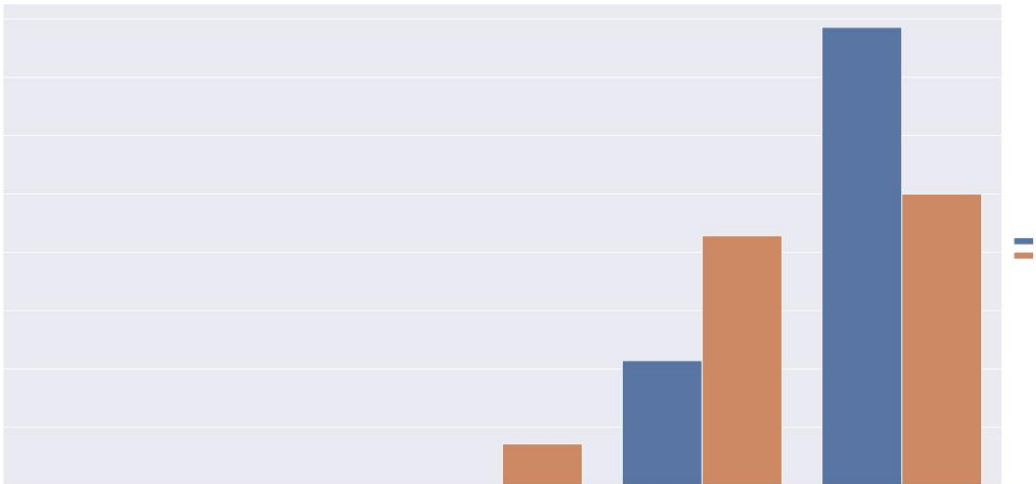
Hình 4.2: Trục X: Xếp hạng của người tham gia từ 1: Không bao giờ đến 5: Luôn luôn.
Trục Y: % người tham gia đã chọn những phản hồi đó



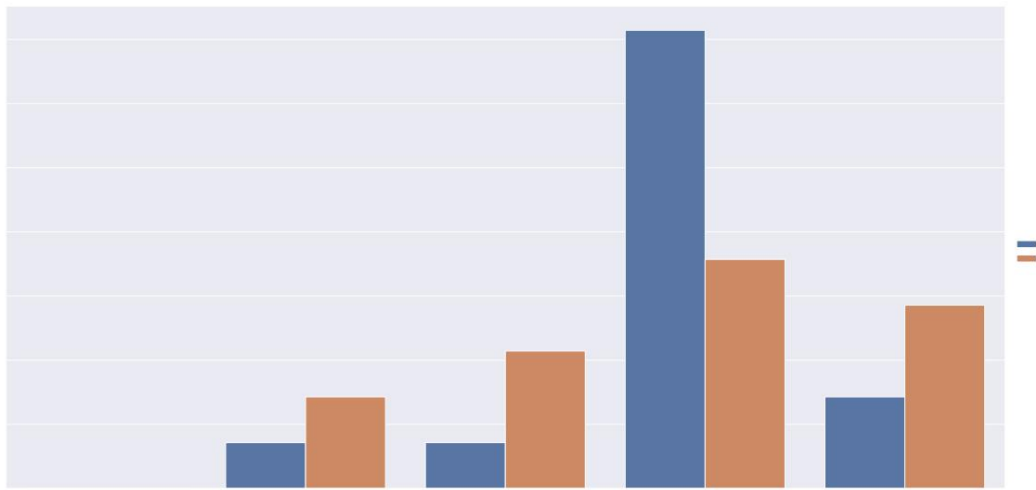
Hình 4.3: Trục X: Xếp hạng của người tham gia từ 1: Không bao giờ đến 5: Luôn luôn.
Trục Y: % người tham gia đã chọn những phản hồi đó



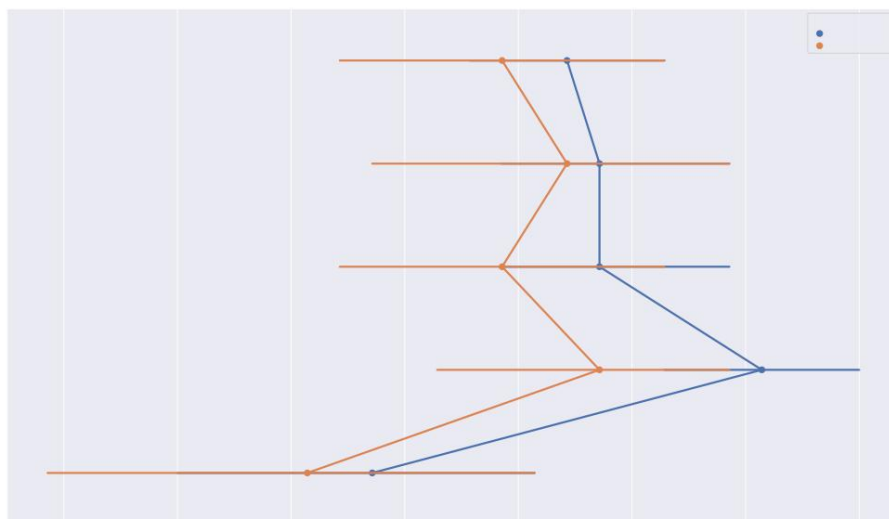
Hình 4.4: Trục X: Xếp hạng của người tham gia từ 1: Không bao giờ đến 5: Luôn luôn.
Trục Y: % người tham gia đã chọn những phản hồi đó



Hình 4.5: Trục X: Đánh giá của người tham gia từ 1: Không bao giờ đến 5: Luôn luôn.
Trục Y: % người tham gia đã chọn những phản hồi đó



Hình 4.6: Trục X: Xếp hạng của người tham gia từ 1: Hoàn toàn không hợp lý đến 5: Hợp lý. Trục Y: % người tham gia đã chọn những phản hồi đó

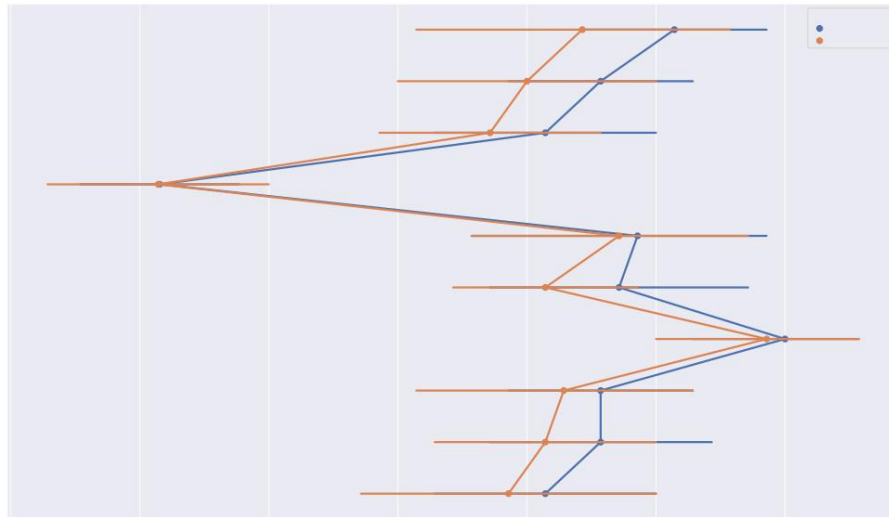


Hình 4.7: Phản hồi trung bình của người tham gia đối với từng câu hỏi cụ thể cho từng loại IU Store

Bảng 4.6: Giá trị trung bình của phản hồi của người tham gia đối với các câu hỏi cụ thể về nhiệm vụ.

Câu hỏi	Đỏ lại	Tự nhiên
Robot phản hồi yêu cầu/lệnh của bạn như thế nào?	4.357143	4.214286
Robot đã phản hồi chính xác như thế nào với yêu cầu/lệnh của bạn?	4.428571	4.357143
Robot đã xác định đúng màu của vật thể bao nhiêu lần?	4.428571	4.214286
Bạn đã giúp robot đến được vật thể mong muốn bao nhiêu lần?	4.785714	4.428571
Bạn có nghĩ là robot phản hồi bạn trong khoảng thời gian hợp lý không?	3.928571	3.785714

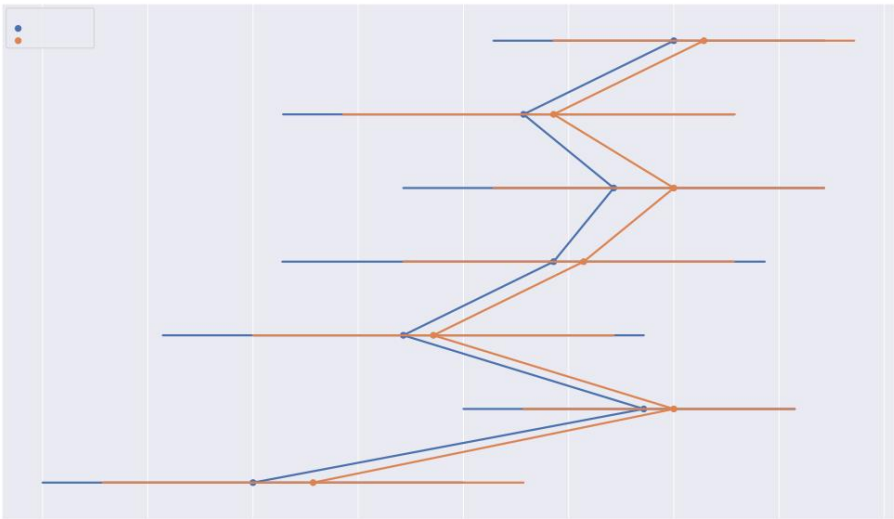
Mặc dù luận án này tập trung nhiều hơn vào việc xây dựng một hệ thống phân tán sẵn sàng cho robot nhất quán hệ thống đối thoại đường ống đáp ứng tất cả các yêu cầu của mạng IU, tôi đã ghi lại những thông tin quan trọng liên quan đến tính nhân hình của robot, sự hoạt bát, khả năng thích nghi và trí thông minh được nhận thức. Có vẻ như mặc dù hai IU việc triển khai lưu trữ khác nhau về mặt hiệu suất, phản ứng với những điều này các câu hỏi trong bảng câu hỏi Godspeed không phải lúc nào cũng tỷ lệ thuận với hiệu suất số liệu quản lý. Hình 4.8 và Bảng 4.7 cho thấy phản hồi trung bình cho các câu hỏi ưa thích Kho IU được chia sẻ (Redis) trong khi Hình 4.9 và Bảng 4.8 hiển thị phản hồi trung bình cho các câu hỏi ưa thích Native (C#) IU Store. Điều này chỉ ra rằng mặc dù chúng tôi đã chọn hai loại kho lưu trữ IU riêng biệt để triển khai với một kho lưu trữ có lợi thế hiệu suất đáng kể so với cái khác, sự khác biệt không phải lúc nào cũng trực tiếp suy ngẫm về câu trả lời cho những câu hỏi này.



Hình 4.8: Phản ứng trung bình của người tham gia đối với các câu hỏi liên quan đến tính nhân hóa, hoạt hình, khả năng thích ứng và trí thông minh được nhận thức của rô-bốt mà họ thích Cửa hàng IU được chia sẻ (Redis)

Bảng 4.7: Trung bình phản hồi của người tham gia đối với các câu hỏi thích Chia sẻ (Redis) Cửa hàng IU.

Câu hỏi	Đỏ lại	Tự nhiên
Bạn cảm thấy gắn bó với robot như thế nào? 4.0	71429 3.714286	
Giả/Bình thường	3.785714 3.500000	
Tồn đọng/Sống động	3.571429 3.357143	
Cơ học/Hữu cơ	2.076923 2.071429	
Trơ/Tương tác	3.928571 3.857143	
Thờ ơ/Phản ứng	3.857143 3.571429	
Không dễ chịu/Dễ chịu	4.500000 4.428571	
Không đủ năng lực/Có năng lực	3.785714 3.642857	
Không biết/Có hiểu biết	3.785714 3.571429	
Ngu ngốc/Thông minh	3.571429 3.428571	



Hình 4.9: Phản hồi trung bình của người tham gia đối với các câu hỏi liên quan đến robot nhân cách hóa, sự sống động, sự dễ mến và trí thông minh được nhận thức rằng ưa thích Native (C#) IU Store

Bảng 4.8: Trung bình phản hồi của người tham gia cho các câu hỏi thích tiếng bản địa (C#) Cửa hàng IU.

Câu hỏi	Đồ lại	Tự nhiên
Tương tác với robot có thú vị không?	4.500000	4.571429
Bạn có muốn dành nhiều thời gian hơn với robot không?	4.142857	4.214286
Không thích/Thích	4.357143	4.500000
Không thân thiện/Thân thiện	4.214286	4.285714
Không tử tế/Tử tế	3.857143	3.928571
Tệ hại/Tốt	4.428571	4.500000
Không thông minh/Thông minh	3.500000	3.642857

Để hiểu được sự phụ thuộc lẫn nhau của các câu hỏi Godspeed với hiệu suất của các cửa hàng IU thậm chí còn xa hơn, tôi cũng đã đánh giá mối tương quan be- giữa các câu hỏi cụ thể về nhiệm vụ và các câu hỏi khác trong bảng câu hỏi Godspeed được thể hiện trong Bảng 4.9, 4.10 và 4.11. Chúng ta có thể thấy rằng kho lưu trữ Shared (Redis) có mối tương quan cao giữa thời gian phản hồi hợp lý và việc liệu robot có có trách nhiệm/vô trách nhiệm, phản ứng/thờ ơ, và tốt bụng/tệ hại. Nói cách khác, một Thời gian phản hồi tốt hơn từ robot được coi là có trách nhiệm, nhạy bén và tốt bụng. Tuy nhiên, chúng ta có thể thấy rằng kho lưu trữ Shared (Redis) không thể hiện sự tương ứng rộng rãi mối quan hệ cho những câu hỏi này như là kho lưu trữ Native (C#). Điều này chứng minh rằng một Cửa hàng IU thậm chí chỉ có một sự khác biệt nhỏ với cửa hàng IU hiệu suất cao một tác động đáng kể đến cách con người cảm nhận tính nhân hóa liên quan đến robot đặc điểm. Nói cách khác, một sự khác biệt nhỏ trong hiệu suất dẫn đến con người những người tham gia quan tâm nhiều hơn đến hào quang tổng thể của robot. Nó ảnh hưởng đến mức độ những người tham gia nhận thấy robot là tự nhiên, nhạy cảm, thân thiện và nó làm cho mọi người bình tĩnh như thế nào họ cảm thấy. Ví dụ, có một mối tương quan rất cao giữa phản ứng của robot đúng và có trách nhiệm/vô trách nhiệm mà không có trong Shared (Redis) lưu trữ. Điều này minh họa rằng ngay cả một sự chậm trễ nhỏ trong hiệu suất cũng thay đổi cách người tham gia đánh giá robot. Khi hiệu suất giảm, những người khác các yếu tố được làm nổi bật hơn trong việc quyết định nhận thức chung. Khi đường ống rất hiệu quả, con người có xu hướng bỏ qua một số đặc điểm đang hài lòng với hiệu suất hiệu quả và có chút thờ ơ về nhân cách hóa- đặc điểm liên quan. Nhưng khi nó thậm chí chỉ chậm một chút, con người vô thức đưa ra đánh giá sâu hơn về đặc điểm của robot liên quan đến hiệu suất. Hơn nữa, những kết quả này xác minh kết quả trong Novikova et al. [40] và Plane et al. [41]

cho thấy chuyển động của robot ảnh hưởng đến nhận thức của con người về những robot đó như thế nào; trong điều này trường hợp, bất kỳ sự chậm trễ xử lý nào trong việc hiểu và phản hồi các phát ngôn đã nói đều là được nhận thức tiêu cực.

Bảng 4.9: Mối tương quan giữa các câu hỏi liên quan đến việc thiết lập nhiệm vụ và các câu hỏi khác
Chúc may mắn với những câu hỏi có hệ số tương quan cao hơn 0,45 cho mục Chia sẻ Cửa hàng IU.

Câu hỏi 1	Câu hỏi 2	Tương quan (Chia sẻ)	Tương quan (Bản địa)
màu phản ứng	sống động trì trệ	0,452267	0,251985
màu phản ứng	dễ chịu khó chịu	0,460566	0,350592
màu phản ứng	có trách nhiệm vô trách nhiệm	0,559793	0,269862
thời gian hợp lý	hấp dẫn	0,485529	0,456488
thời gian hợp lý	dành nhiều thời gian hơn	0,527589	0,498726
thời gian hợp lý	giả tự nhiên	0,570420	0,350211
thời gian hợp lý di chuyển di chuyển thành lịch cứng nhắc		0,563213	0,195482
thời gian hợp lý	đáp ứng thờ ơ	0,561328	0,456488
thời gian hợp lý	thích không thích	0,607751	0,556567
thời gian hợp lý	thân thiện không thân thiện	0,551019	0,696761
thời gian hợp lý	tử tế không tử tế	0,465491	0,222393
thời gian hợp lý	tốt tệ	0,671647	0,281336

Bảng 4.10: Mối tương quan giữa các câu hỏi liên quan đến việc thiết lập nhiệm vụ và các câu hỏi khác
Chúc may mắn với những câu hỏi có hệ số tương quan cao hơn 0,45 đối với người bản xứ
Bảng 1 của IU Store.

Câu hỏi 1	Câu hỏi 2	Tương quan (Chia sẻ)	Tương quan (Bản địa)
phản ứng	có năng lực không có năng lực	0,399667	0,515672
phản hồi đúng	sống động trì trệ	0,276385	0,516888
phản hồi đúng	tương tác trợ	0,199681	0,473950
phản hồi đúng	thân thiện không thân thiện	0,216085	0,696984
phản ứng đúng có năng lực không có năng lực		0,053376	0,647699
phản ứng đúng hiểu biết thiếu hiểu biết		0,047946	0,502320
phản ứng đúng có trách nhiệm vô trách nhiệm		0,353553	0,759972
phản hồi đúng	khôn ngoan ngốc nghếch	0,276385	0,460385
phản hồi đúng	kết thúc thư giãn lo lắng	-0.345582	0,534919
phản hồi đúng	bắt đầu bình tĩnh kích động	-0,172345	0,516888
phản hồi đúng	kết thúc bình tĩnh kích động	0,000000	0,662004
màu phản ứng	bị thu hút	-0.087932	0,469200
màu phản ứng	hấp dẫn	0,000000	0,478130
màu phản ứng	tương tác trợ	0,199681	0,474292
màu phản ứng	đáp ứng thờ ơ	0,389249	0,478130
màu phản ứng	thích không thích	0,138233	0,450376
màu phản ứng	thân thiện không thân thiện	0,342134	0,552106
màu phản ứng	bắt đầu thư giãn lo lắng	0,299504	0,495769
màu phản ứng	bắt đầu bình tĩnh kích động	0,229794	0,774619

Bảng 4.11: Mối tương quan giữa các câu hỏi liên quan đến việc thiết lập nhiệm vụ và các câu hỏi khác
Chúc may mắn với những câu hỏi có hệ số tương quan cao hơn 0,45 đối với người bản xứ
Bảng 2 của IU Store.

Câu hỏi 1	Câu hỏi 2	Tương quan (Chia sẻ) Tương	quan (Bản địa)
màu phản ứng	kết thúc bình tĩnh kích động	0,394405	0,559314
màu phản ứng	bắt đầu- quan tâm chán	0,228218	0,514174
di chuyển đúng	giả tự nhiên	-0,166070	0,632817
di chuyển đúng	có ý thức vô thức	-0,339945	0,597284
di chuyển đúng	nhân tạo giống như thật	0,028239	0,453423
di chuyển đúng	thân thiện không thân thiện	0,097728	0,473849
di chuyển đúng	dễ chịu khó chịu	-0,138866	0,479234
di chuyển đúng	tốt tệ	0,060606	0,549031
di chuyển đúng	có năng lực không có năng lực	-0,144841	0,455860
di chuyển đúng	có trách nhiệm vô trách nhiệm	0,035533	0,817689
di chuyển đúng	khôn ngoan ngốc nghếch	-0,272727	0,593171
thời gian hợp lý	bị thu hút	0,298969	0,587700
thời gian hợp lý giống	như con người máy móc	0,189629	0,523799
thời gian hợp lý	tương tác trợ	0,386281	0,539582
thời gian hợp lý	dễ chịu khó chịu	0,404983	0,454304
thời gian hợp lý	khôn ngoan ngốc nghếch	0,318148	0,489525
thời gian hợp lý	kết thúc thư giãn lo lắng	0,040517	0,490710
thời gian hợp lý	kết thúc bình tĩnh kích động	0,208084	0,647270

CHƯƠNG 5:

PHẦN KẾT LUẬN

Trong luận văn này, tôi đã cố gắng đưa tất cả các đặc điểm chính của một hệ thống đối thoại chuẩn nên có một nơi tạo điều kiện thuận lợi cho việc bảo tồn toàn bộ mạng lưới IU mà không phá vỡ kết nối, và tôi đã thực hiện điều đó theo hai cách chính khác nhau và so sánh điểm mạnh và điểm khác biệt của chúng. Mặc dù tôi đã sử dụng hai phiên bản chính trong quá trình thực hiện của tôi trong thí nghiệm cuối cùng, tôi đã chỉ ra những cách để tối đa hóa hiệu suất của từng phiên bản khác nhau và nơi nào có thể hữu ích hơn cái kia. Về mặt kết nối các mô-đun phân phối giữa các quy trình, chúng tôi có thấy rằng cách tiếp cận thứ hai chỉ gửi các khối dữ liệu mới và xây dựng lại Mạng lưới IU tại điểm đến là lựa chọn tốt hơn. Ngoài mục tiêu chính của việc xây dựng một hệ thống đối thoại hoàn chỉnh sẵn sàng cho robot, tôi cũng đã chứng minh rằng sự hợp nhất luồng có ý nghĩa là khả thi giữa các luồng dữ liệu phân tán và nhiều mô-đun NLU có thể được tích hợp trong cùng một đường ống để hoạt động cùng nhau. Hơn nữa, công việc của tôi chứng minh tác động của những thay đổi trong quá trình thực hiện đối với cách thức con người nhận thức được các đặc điểm liên quan đến nhân cách hóa của robot và điều đó có thể được cải thiện với việc thực hiện hiệu quả hơn. Tôi tin rằng luận án này sẽ truyền cảm hứng việc xây dựng các triển khai lớn hơn và phức tạp hơn, và điều đó sẽ dẫn đến nghiên cứu sâu hơn trong lĩnh vực này.

5.1 Những hạn chế và công việc trong tương lai

Như đã đề cập ở trên, chúng tôi chưa khám phá hết tất cả các kết hợp có thể có tối đa hóa hiệu suất của hai loại IU Store. Thay vào đó, tôi đã thử nghiệm với một IU Store có những ưu điểm rõ ràng hơn những cái khác để nhận ra hiệu quả về phản ứng chung về tính nhân hóa, sự hoạt bát, khả năng thích nghi và trí thông minh được nhận thức. Một nghiên cứu trực tiếp về hai cửa hàng tối đa hóa hiệu suất của cả hai biến thể đều phải tiết lộ thêm thông tin về ứng dụng của chúng. Ngoài ra rằng, so sánh toàn bộ hệ thống đường ống đối thoại sẵn sàng cho robot phân tán với đường ống tương tự không phân tán cũng sẽ tiết lộ thông tin chi tiết về hiệu suất của nó.

Trong thí nghiệm của mình, tôi đã dựa vào các kỹ thuật truyền tin nhắn dựa trên JSON để thiết lập các kết nối giữa các mô-đun nằm trong các quy trình khác nhau. Sử dụng các loại dữ liệu có cấu trúc tuần tự khác (ví dụ: Bộ đệm giao thức từ Google và Thrift từ Facebook) có thời gian tuần tự hóa nhanh hơn JSON nên tăng và hiệu suất nữa.

TÀI LIỆU THAM KHẢO

- [1] D. Spiliotopoulos, I. Androutsopoulos, và CD Spyropoulos, "Con người-robot tương tác dựa trên đối thoại ngôn ngữ tự nhiên được nói," trong Biên bản báo cáo Hội thảo châu Âu về dịch vụ và rô bốt hình người, 2001, trang 25-27.
- [2] T. Fong, C. Thorpe và C. Baur, "Sự hợp tác, đối thoại và con người-rô-bốt tương tác, hội nghị quốc tế lần thứ 10 về nghiên cứu robot (lorne, victoria, Úc)," trong Biên bản báo cáo của Hội nghị chuyên đề quốc tế lần thứ 10 về nghiên cứu robot tìm kiếm, 2001.
- [3] D. Bohus, S. Andrist và M. Jalobeanu, "Sự phát triển nhanh chóng của đa phương thức hệ thống tương tác: một cuộc trình diễn nền tảng cho trí thông minh có vị trí," trong Biên bản Hội nghị quốc tế ACM lần thứ 19 về Tương tác đa phương thức tion, 2017, trang 493-494.
- [4] T. Michael và S. Möller, "Retico: Một khuôn khổ nguồn mở để mô hình hóa thực tế thời gian hội thoại trong hệ thống hội thoại nói," Studentexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2019, trang 134-140, 2019.
- [5] C. Kennington, D. Moro, L. Marchand, J. Carns, và D. McNeill, "rrsds: To-hướng đến một hệ thống đối thoại bằng lời nói sẵn sàng cho robot," trong Biên bản báo cáo thường niên lần thứ 21 Cuộc họp của Nhóm quan tâm đặc biệt về Diễn ngôn và Đối thoại, 2020, trang 132-135.

- [6] A. Singh, V. Goswami, V. Natarajan, Y. Jiang, X. Chen, M. Shah, M. Rohrbach, D. Batra và D. Parikh, "Mmf: Một khuôn khổ đa phương thức cho tầm nhìn và ngôn ngữ nghiên cứu đo lường," <https://github.com/facebookresearch/mmf>, 2020.
- [7] D. Schlangen và G. Skantze, "Một mô hình trừu tượng, tổng quát về sự gia tăng xử lý đối thoại," *Đối thoại và Diễn ngôn*, tập 2, số 1, trang 83-111, 2011. [Trực tuyến]. Có sẵn: <https://journals.linguisticsociety.org/elanguage/dad/bai-viet/tai-xuong/361/361-2892-1-PB.pdf>
- [8] T. Baumann và D. Schlangen, "Bản phát hành inprok 2012," trong NAACL-HLT Hội thảo về các hướng đi và nhu cầu trong tương lai của cộng đồng Spoken Dialog: Công cụ và Dữ liệu (SDCTD 2012), 2012, trang 29-32.
- [9] C. Kennington, T. Han và D. Schlangen, "Căn chỉnh thời gian bằng cách sử dụng in-khung đơn vị cremental," trong *Biên bản báo cáo của Hội nghị quốc tế ACM lần thứ 19 Tài liệu tham khảo về Tương tác đa phương thức*, 2017, trang 297-301.
- [10] LWJM, *Nói*. Cambridge, Hoa Kỳ: MIR Press, 1989.
- [11] C. Kennington, "Giải quyết dần dần các tham chiếu để xác định trực quan trình bày các đối tượng trong bối cảnh đối thoại có tình huống cụ thể," 2016.
- [12] C. Kennington, S. Kousidis và D. Schlangen, "Inprokts: Một bộ công cụ để tăng xử lý tình huống tinh thần," trong *Biên bản cuộc họp thường niên lần thứ 15 của Nhóm quan tâm đặc biệt về diễn ngôn và đối thoại (SIGDIAL)*, 2014, trang 84-88.
- [13] MK Tanenhaus, MJ Spivey-Knowlton, KM Eberhard và JC Sedivy, "Tích hợp thông tin hình ảnh và ngôn ngữ trong việc hiểu ngôn ngữ nói sion," *Khoa học*, tập 268, số 5217, trang 1632-1634, 1995.

- [14] G. Aist, J. Allen, E. Campana và CG Gallo, "Hiểu biết gia tăng trong đối thoại giữa con người và máy tính và bằng chứng thực nghiệm về những lợi thế so với không phương pháp gia tăng," *Decalog* 2007, tr. 149, 2007.
- [15] O. Buß và D. Schlangen, "Mô hình hóa các hiện tượng dưới lời nói trong văn nói hệ thống alogue," trong *Biên bản Hội thảo quốc tế lần thứ 14 về Semantics và Ngữ dụng học của Đối thoại (Pozdial 2010)*, 2010.
- [16] M. Slee, A. Agarwal và M. Kwiatkowski, "Tiết kiệm: Khả năng mở rộng đa ngôn ngữ "Triển khai dịch vụ", Sách trắng của Facebook, tập 5, số 8, năm 2007.
- [17] AS TANENBAUM và M. VAN STEEN, "Quy trình", *Hệ thống phân tán Nguyên tắc và mô hình*, trang 69-114, 2006.
- [18] TF Bissyand'e, F. Thung, D. Lo, L. Jiang, và L. Réveill'ere, "Sự phổ biến, khả năng tương tác và tác động của ngôn ngữ lập trình trong 100.000 mã nguồn mở dự án," trong *hội nghị thường niên lần thứ 37 về phần mềm máy tính và ứng dụng của IEEE năm 2013 IEEE*, 2013, trang 303-312.
- [19] M. Grimmer, R. Schatz, C. Seaton, T. Würrthinger, M. Luján, và H. Mössenböck, "Khả năng tương tác giữa các ngôn ngữ trong thời gian chạy đa ngôn ngữ," *Giao dịch ACM về Ngôn ngữ lập trình và Hệ thống (TOPLAS)*, tập 40, số 2, trang 1-43, 2018.
- [20] D. Chisnall, "Thách thức về khả năng tương tác giữa các ngôn ngữ," *Truyền thông của ACM*, tập 56, số 12, trang 50-56, 2013.
- [21] A. Verma, "Các ngôn ngữ lập trình phổ biến nhất cho ma-

lưng heo học hỏi Và dữ liệu khoa học," <https://fossbytes.com/>

ngôn ngữ lập trình phổ biến hàng đầu - khoa học dữ liệu học máy/, 2016.

- [22] M. Wegiel và C. Krintz, "Đối tượng xuyên ngôn ngữ, an toàn về kiểu và minh bạch chia sẻ cho các thời gian chạy được quản lý đồng vị trí," Thông báo ACM Sigplan, tập 45, số 10, trang 223-240, 2010.
- [23] M. Grimmer, C. Seaton, R. Schatz, T. Würrthinger, và H. Mössenböck, "High-hiệu suất khả năng tương tác giữa các ngôn ngữ trong thời gian chạy đa ngôn ngữ," trong Biên bản Hội nghị chuyên đề lần thứ 11 về Ngôn ngữ động, 2015, trang 78-90.
- [24] M. Marge, S. Nogar, CJ Hayes, SM Lukin, J. Bloecker, E. Holder, và C. Voss, "Nền tảng nghiên cứu cho cuộc đối thoại giữa nhiều rô-bốt với con người," arXiv bản in trước arXiv:1910.05624, 2019.
- [25] SM Lukin, F. Gervits, CJ Hayes, A. Leuski, P. Moolchandani, JG Rogers III, CS Amaro, M. Marge, CR Voss và D. Traum, "Scoutbot: Một hệ thống đối thoại cho điều hướng cộng tác," bản in trước arXiv arXiv:1807.08074, 2018.
- [26] Y. Wang, RC Murray, H. Bao, và C. Rose, "Hợp tác năng động dựa trên tác nhân hỗ trợ trong không gian văn phòng thông minh," trong Biên bản báo cáo của Hội nghị thường niên lần thứ 21 của Nhóm quan tâm đặc biệt về Diễn ngôn và Đối thoại, 2020, trang 257-260.
- [27] D. Adamson, G. Dyke, H. Jang và CP Rosé, "Hướng tới một cách tiếp cận nhanh nhẹn đối với điều chỉnh hỗ trợ cộng tác năng động theo nhu cầu của sinh viên," Tạp chí quốc tế Trí tuệ nhân tạo trong giáo dục, tập 24, số 1, trang 92-124, 2014.

- [28] S. Kousidis, C. Kennington, T. Baumann, H. Buschmeier, S. Kopp, và D. Schlangen, "Một hệ thống đối thoại đa phương thức trong xe theo dõi hành vi của người lái xe chú ý," trong *Biên bản báo cáo của hội nghị quốc tế lần thứ 16 về đa phương thức tương tác*, 2014, trang 26-33.
- [29] R. Math, A. Mahr, MM Moniri, và C. M"uller, "Opens: Một hệ thống nguồn mở mới mô phỏng lái xe cho nghiên cứu," *GMM-Fachbericht-AmE* 2013, tập. 2, 2013.
- [30] J. Wienke và S. Wrede, "Một phần mềm trung gian cho nghiên cứu hợp tác trong thí nghiệm robot học," trong *Hội nghị chuyên đề quốc tế IEEE/SICE năm 2011 về Tích hợp hệ thống hoạt động (SII)*. IEEE, 2011, trang 1183-1190.
- [31] B. Carlmeier, D. Schlangen và B. Wrede, "Hướng tới vòng phản hồi khép kín trong hri: Tích hợp inprotk và pamini," trong *Biên bản Hội thảo năm 2014 về Tương tác giữa người và rô-bốt trong thế giới thực, đa phương thức, đa bên*, 2014, trang 1-6.
- [32] J. Peltason và B. Wrede, "Pamini: Một khuôn khổ để lắp ráp sáng kiến hỗn hợp tương tác giữa người và robot từ các mô hình tương tác chung," trong *Biên bản báo cáo Hội nghị SIGDIAL 2010*, 2010, trang 229-232.
- [33] K. He, G. Gkioxari, P. Doll'ar, và R. Girshick, "Mask r-cnn," trong *IEEE 2017 Hội nghị quốc tế về tầm nhìn máy tính (ICCV)*, 2017, trang 2980-2988.
- [34] Y. Jang, J. Lee, J. Park, K.-H. Lee, P. Lison, và K.-E. Kim, "PyOpenDial: Một bộ công cụ độc lập với miền dựa trên python để phát triển hệ thống đối thoại nói với các quy tắc xác suất," trong *Biên bản Hội nghị năm 2019 về Phương pháp thực nghiệm trong Xử lý ngôn ngữ tự nhiên và Hội nghị chung quốc tế lần thứ 9 về Xử lý ngôn ngữ tự nhiên*

(EMNLP-IJCNLP): Biểu tình hệ thống. Hồng Kông, Trung Quốc: Hiệp hội
cho Ngôn ngữ học tính toán, tháng 11 năm 2019, trang 187-192. [Trực tuyến]. Có sẵn:
<https://www.aclweb.org/anthology/D19-3032>

- [35] C. Kennington và D. Schlangen, "Học tập đơn giản và sáng tác
ứng dụng ý nghĩa của từ ngữ có cơ sở nhận thức để tham khảo gia tăng
nghị quyết," trong Biên bản cuộc họp thường niên lần thứ 53 của Hiệp hội
cho Ngôn ngữ học tính toán và Hội nghị chung quốc tế lần thứ 7 về
Xử lý ngôn ngữ tự nhiên (Tập 1: Bài báo dài). Bắc Kinh, Trung Quốc:
Hiệp hội Ngôn ngữ học tính toán, tháng 7 năm 2015, trang 292-301. [Trực tuyến].
Có sẵn: <https://www.aclweb.org/anthology/P15-1029>
- [36] JL Carlson, Redis trong hành động. Manning Publications Co., 2013.
- [37] S. Sun, J. Gong, AY Zomaya và A. Wu, "Một thông tin gia tăng phân tán
mô hình thu thập thông tin cho dữ liệu văn bản quy mô lớn," Cluster Computing, tập 22,
số 1, trang 2383-2394, 2019.
- [38] A. Venkataraman và KK Jagadeesha, "Đánh giá giao tiếp giữa các quá trình
cơ chế cation," Kiến trúc, tập 86, trang 64, 2015.
- [39] C. Bartneck, D. Kulić, E. Croft, và S. Zoghbi, "Các công cụ đo lường cho
sự nhân cách hóa, sự hoạt động, sự dễ mến, trí thông minh được nhận thức và được nhận thức
an toàn của robot," Tạp chí quốc tế về robot xã hội, tập 1, số 1, trang 71-81,
2009.
- [40] J. Novikova, G. Ren, và L. Watts, "Không phải cách bạn nhìn, mà là cách bạn di chuyển:

xác nhận một chương trình chung cho hành vi tình cảm của robot,” trong Hội nghị IFIP

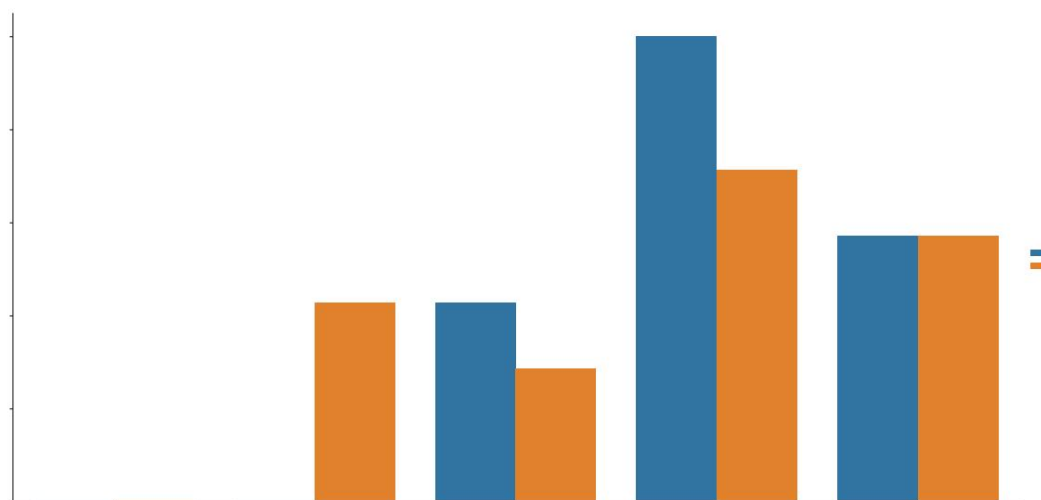
về tương tác giữa người và máy tính. Springer, 2015, trang 239-258.

[41] S. Plane, A. Marvasti, T. Egan và C. Kennington, “Dự đoán tuổi được cảm nhận:

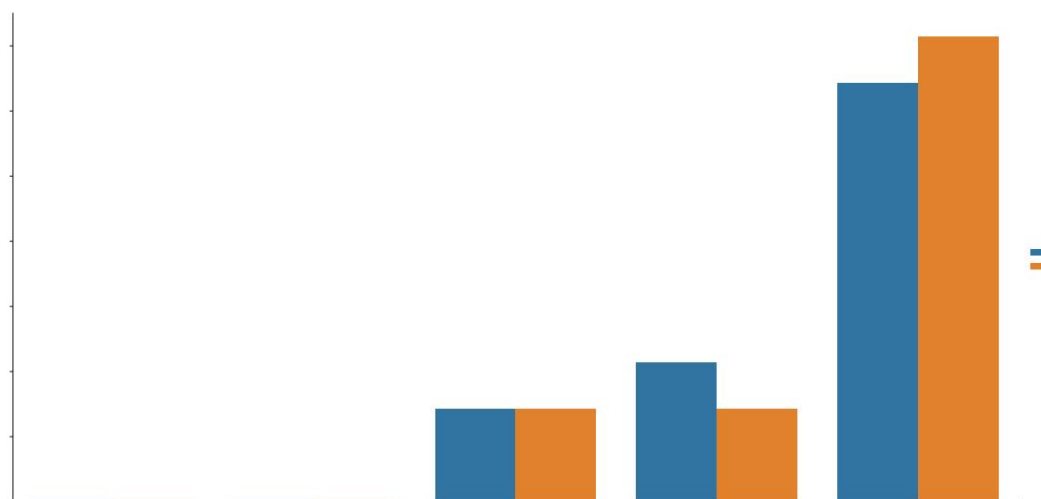
Cả khả năng ngôn ngữ và ngoại hình đều quan trọng,” trong Biên bản báo cáo của Hội nghị lần thứ 19

Cuộc họp thường niên của SIGdial về Diễn ngôn và Đối thoại, 2018, trang 130-139.

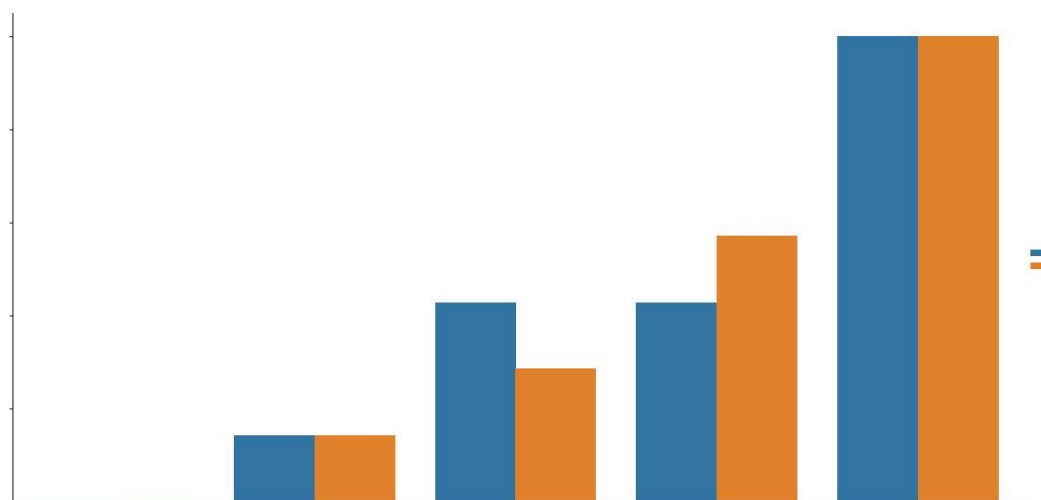
PHỤ LỤC A:
TRẢ LỜI CHO CÁC CÂU HỎI TRONG
BẢNG CÂU HỎI GODSPEED



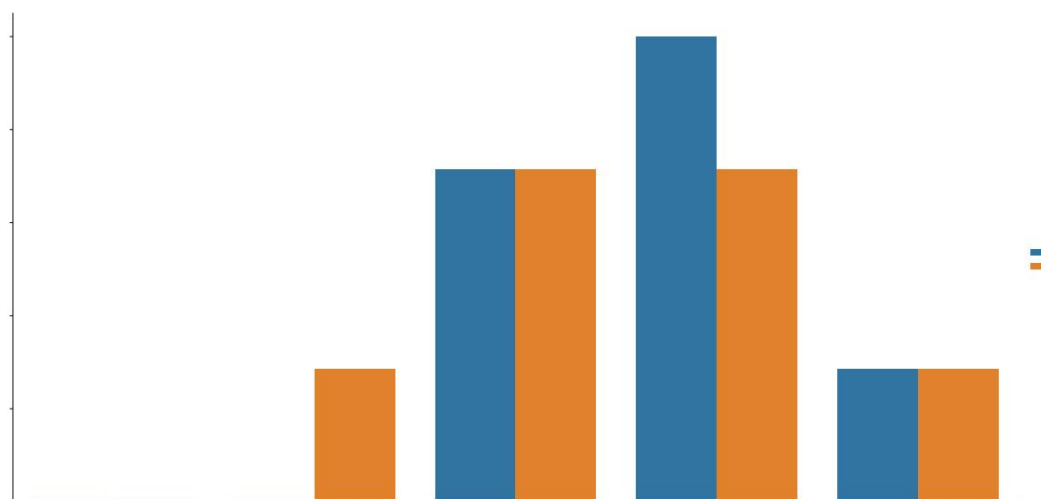
Hình A.1: Trục X: Đánh giá của người tham gia từ 1: Không hề đến 5: Rất.
Trục Y: % người tham gia đã chọn những phản hồi đó



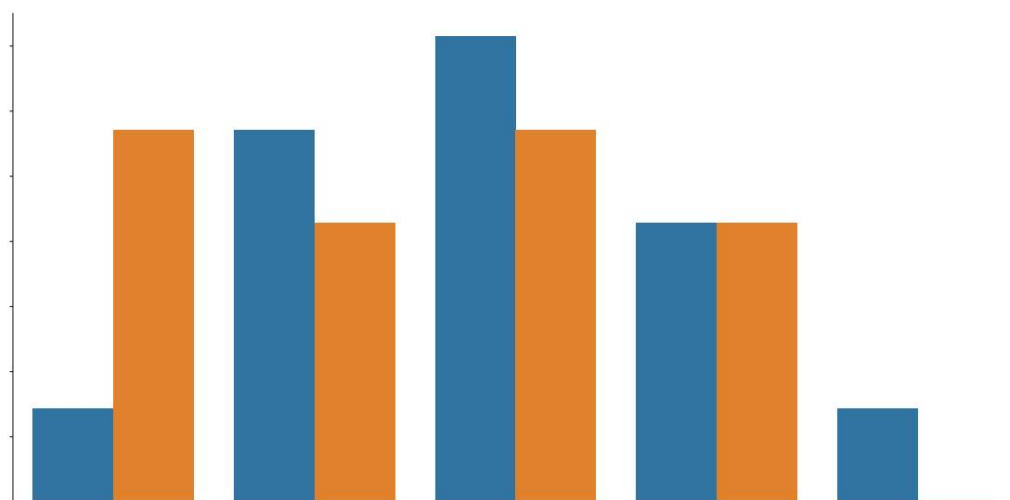
Hình A.2: Trục X: Đánh giá của người tham gia từ 1: Không hề đến 5: Rất.
Trục Y: % người tham gia đã chọn những phản hồi đó



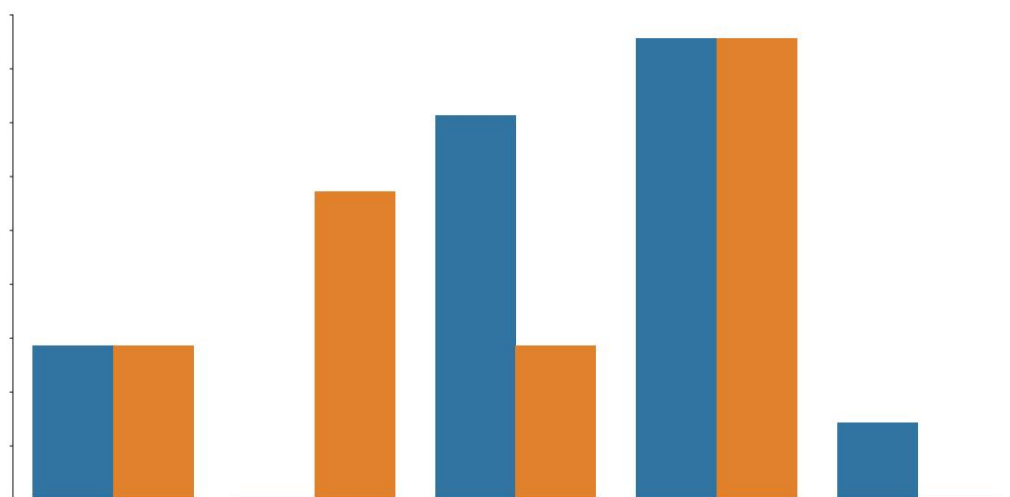
Hình A.3: Trục X: Xếp hạng của người tham gia từ 1: Không hề đến 5: Rất nhiều. Trục Y: % người tham gia đã chọn những phản hồi đó



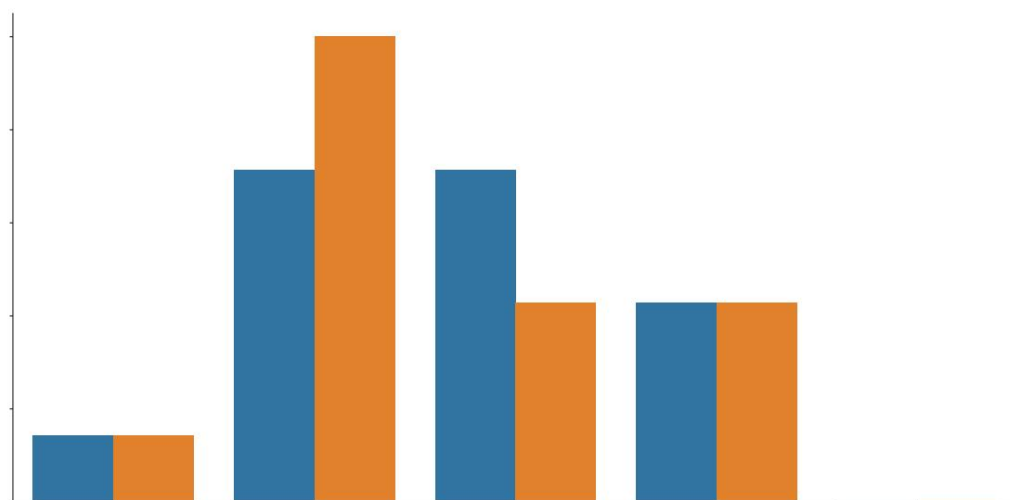
Hình A.4: Trục X: Xếp hạng của người tham gia từ 1: Giả đến 5: Tự nhiên. Trục Y: % người tham gia đã chọn những phản hồi đó



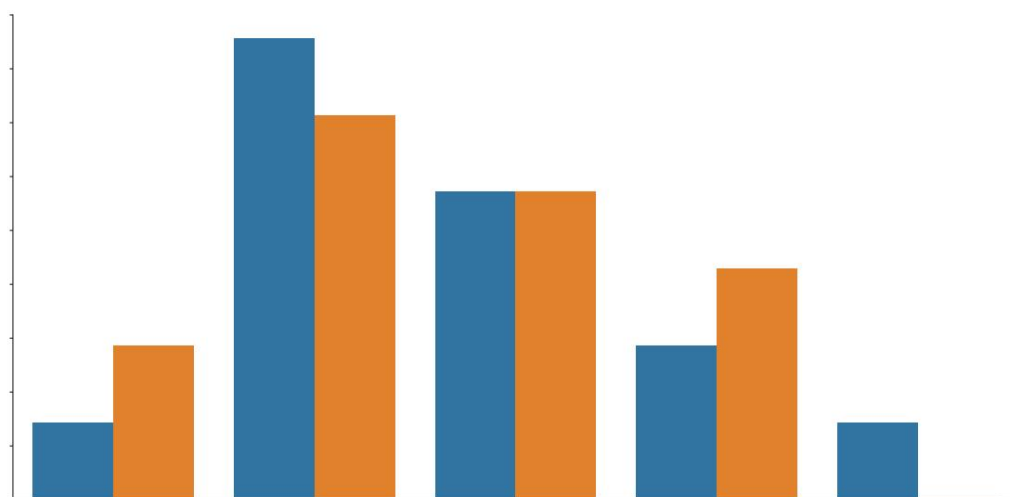
Hình A.5: Trục X: Xếp hạng của người tham gia từ 1: Giống máy đến 5: Giống con người. Trục Y: % người tham gia đã chọn những phản hồi đó



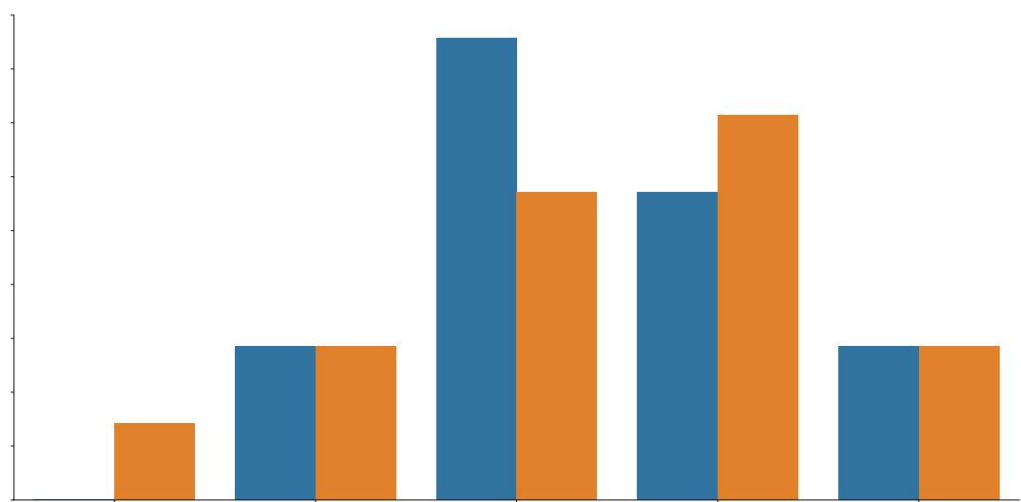
Hình A.6: Trục X: Xếp hạng của người tham gia từ 1: Vô thức đến 5: Có ý thức. Trục Y: % người tham gia đã chọn những phản hồi đó



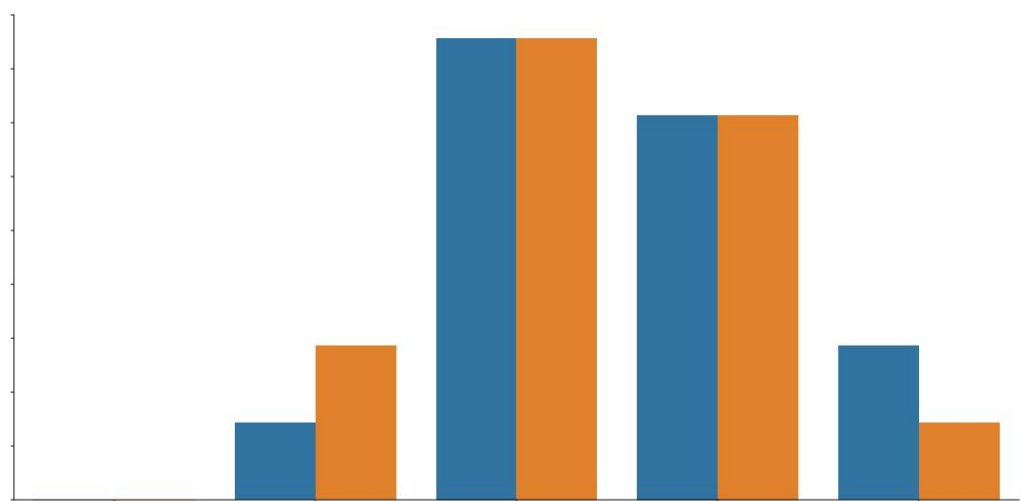
Hình A.7: Trục X: Đánh giá của người tham gia từ 1: Nhân tạo đến 5: Giống thật.
Trục Y: % người tham gia đã chọn những phản hồi đó



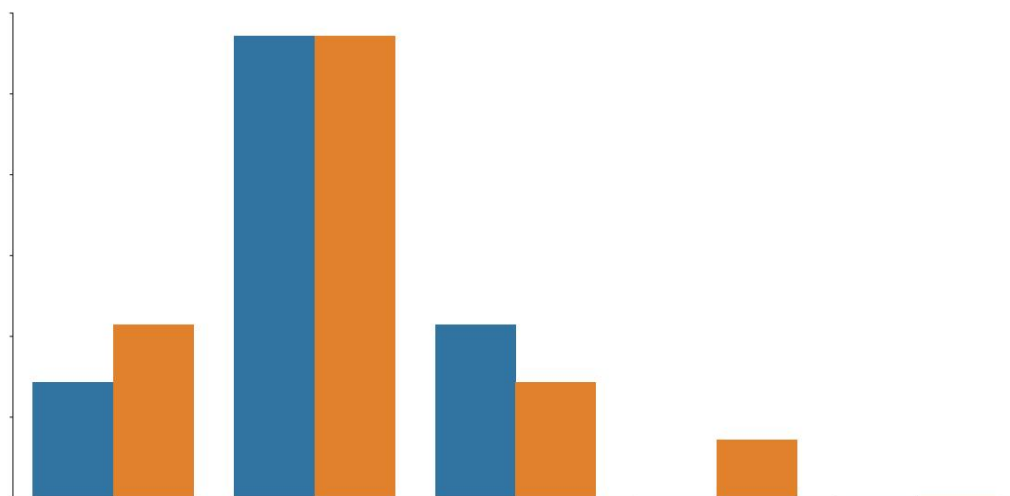
Hình A.8: Trục X: Xếp hạng của người tham gia từ 1: Di chuyển cứng nhắc đến 5: Di chuyển thanh lịch.
Trục Y: % người tham gia đã chọn những phản hồi đó



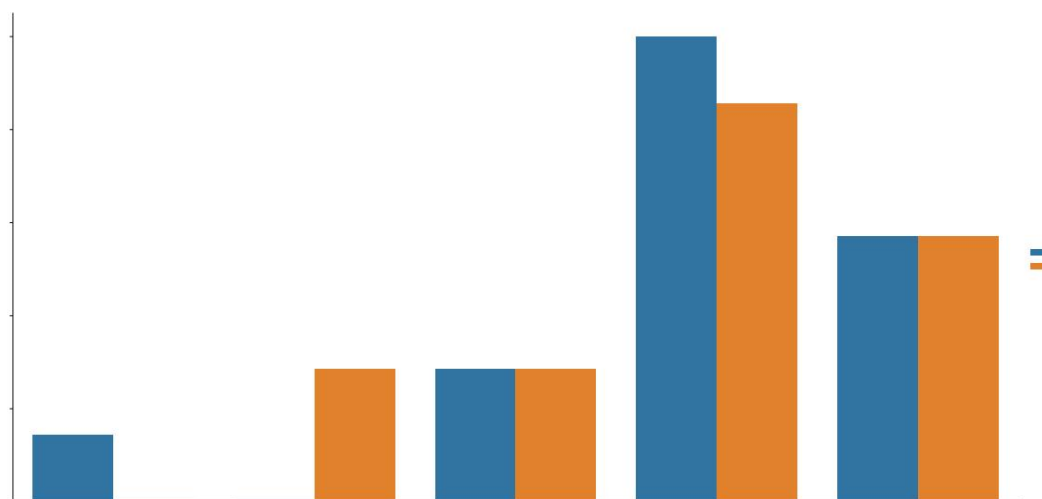
Hình A.9: Trục X: Xếp hạng của người tham gia từ 1: Đã chết đến 5: Còn sống. Trục Y: % người tham gia đã chọn những phản hồi đó



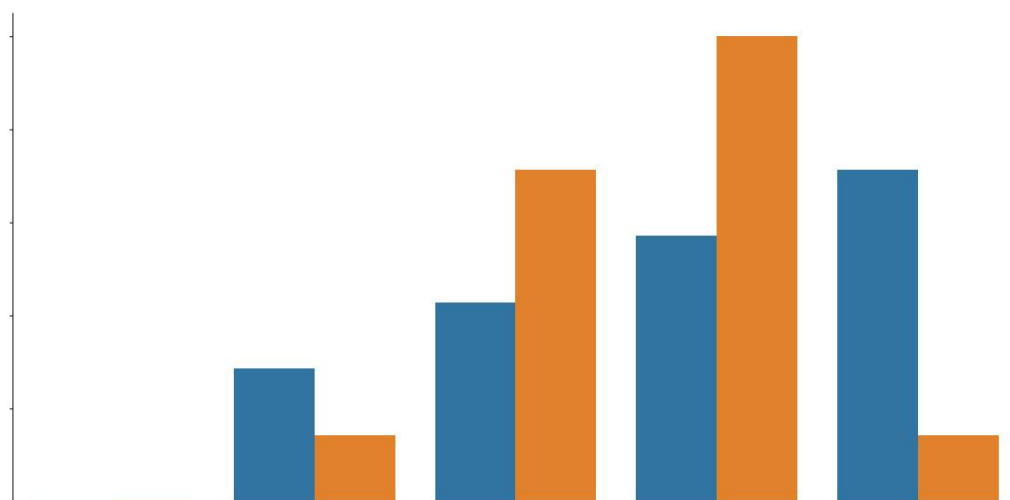
Hình A.10: Trục X: Đánh giá của người tham gia từ 1: Trì trệ đến 5: Sôi động. Trục Y: % người tham gia đã chọn những phản hồi đó



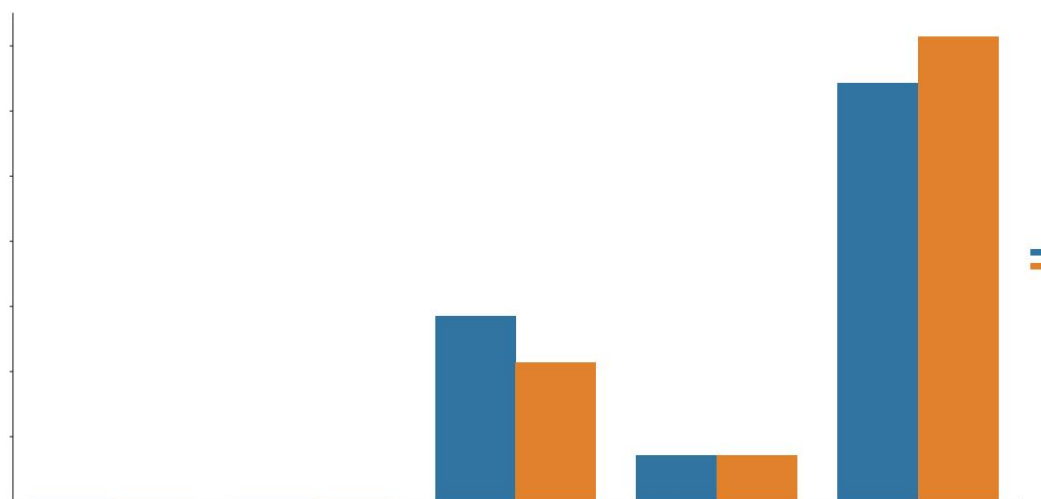
Hình A.11: Trục X: Đánh giá của người tham gia từ 1: Cơ học đến 5: Hữu cơ.
Trục Y: % người tham gia đã chọn những phản hồi đó



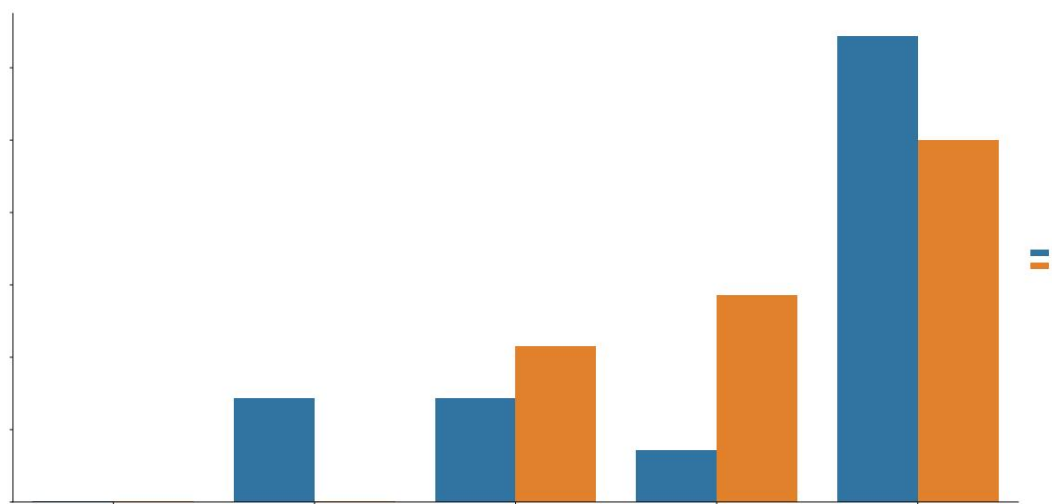
Hình A.12: Trục X: Đánh giá của người tham gia từ 1: Trơ đến 5: Tương tác.
Trục Y: % người tham gia đã chọn những phản hồi đó



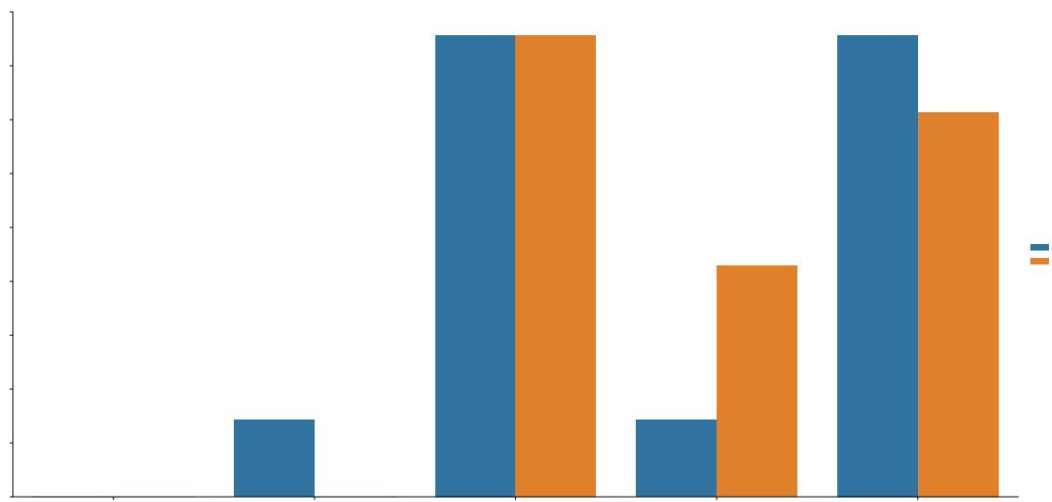
Hình A.13: Trục X: Xếp hạng của người tham gia từ 1: Thờ ơ đến 5: Phản hồi. Trục Y: % người tham gia đã chọn những phản hồi đó



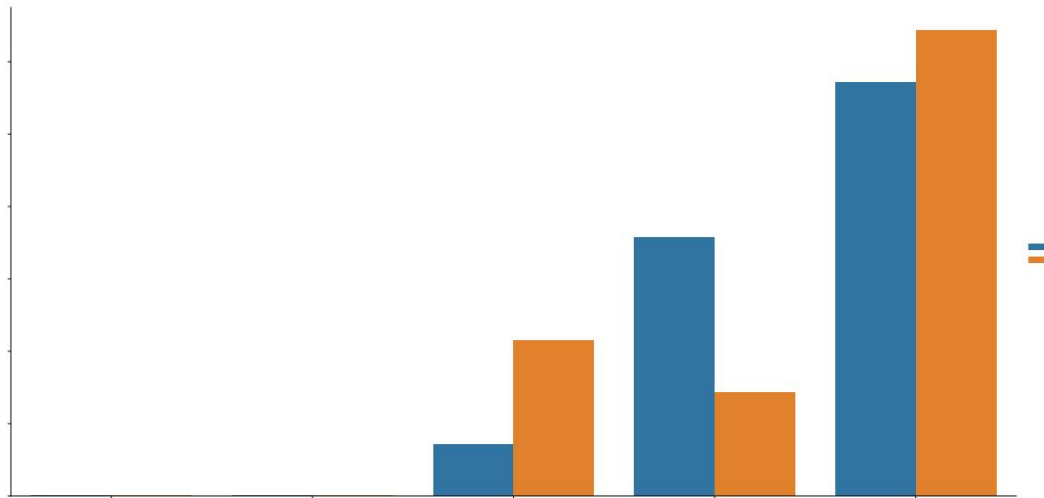
Hình A.14: Trục X: Xếp hạng của người tham gia từ 1: Không thích đến 5: Thích. Trục Y: % người tham gia đã chọn những phản hồi đó



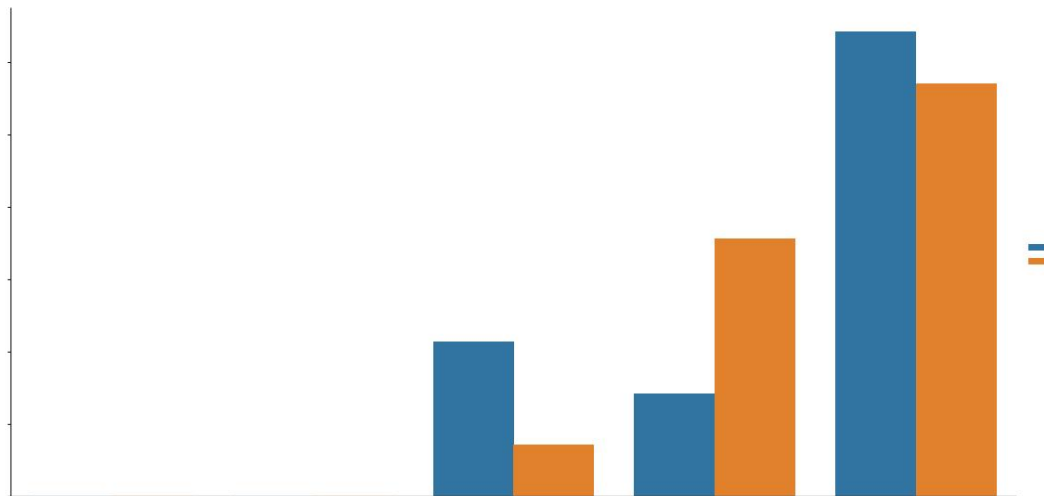
Hình A.15: Trục X: Đánh giá của người tham gia từ 1: Không thân thiện đến 5: Thân thiện.
Trục Y: % người tham gia đã chọn những phản hồi đó



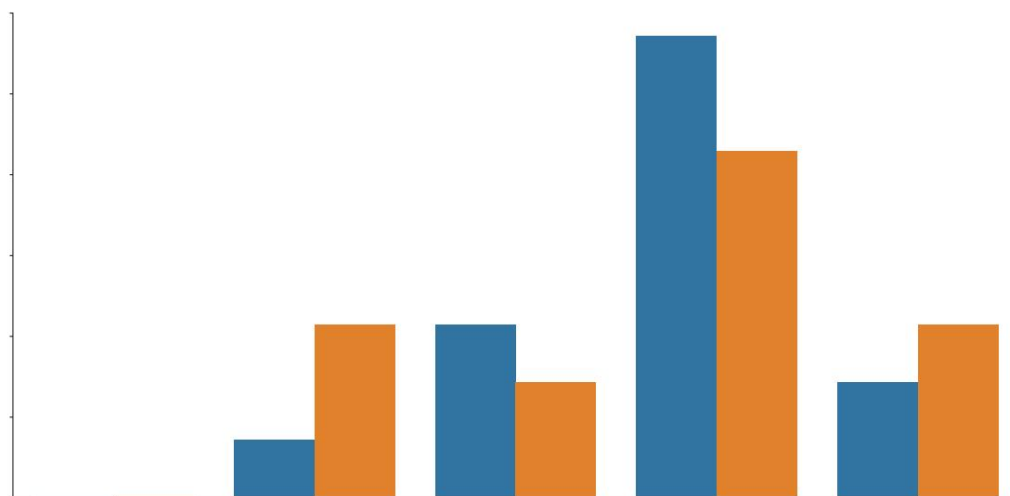
Hình A.16: Trục X: Đánh giá của người tham gia từ 1: Không tử tế đến 5: Tử tế.
Trục Y: % người tham gia đã chọn những phản hồi đó



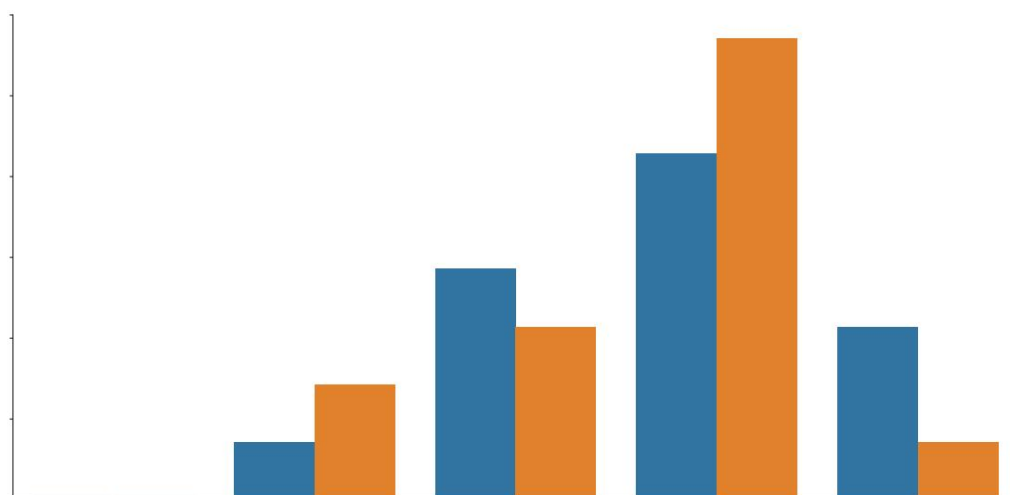
Hình A.17: Trục X: Xếp hạng của người tham gia từ 1: Không hài lòng đến 5: Hài lòng. Trục Y: % người tham gia đã chọn những phản hồi đó



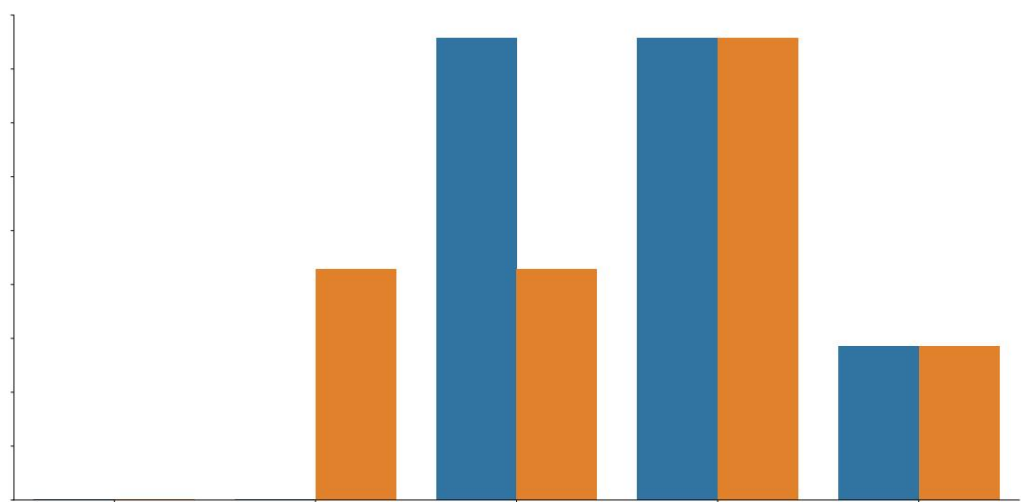
Hình A.18: Trục X: Xếp hạng của người tham gia từ 1: Tệ đến 5: Tốt. Trục Y: % người tham gia đã chọn những phản hồi đó



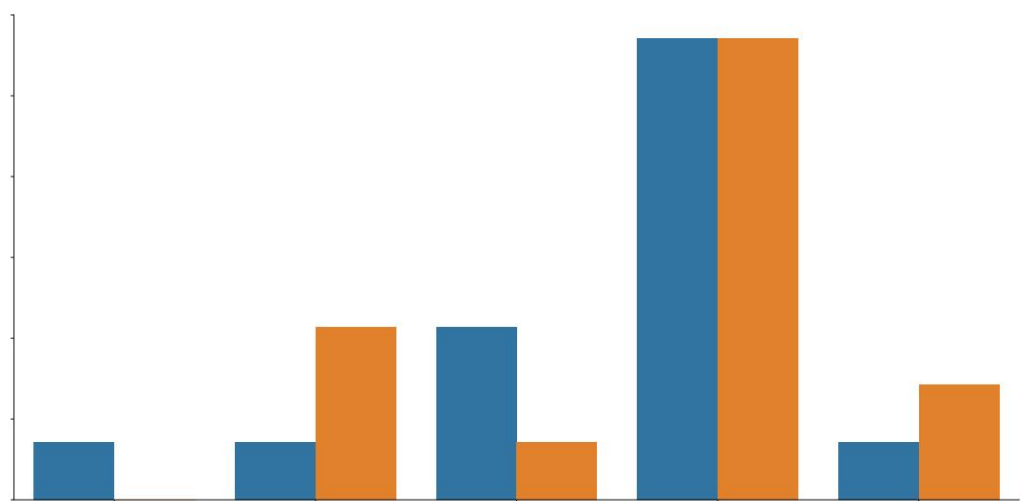
Hình A.19: Trục X: Xếp hạng của người tham gia từ 1: Không đủ năng lực đến 5: Có năng lực. Trục Y: % người tham gia đã chọn những phản hồi đó



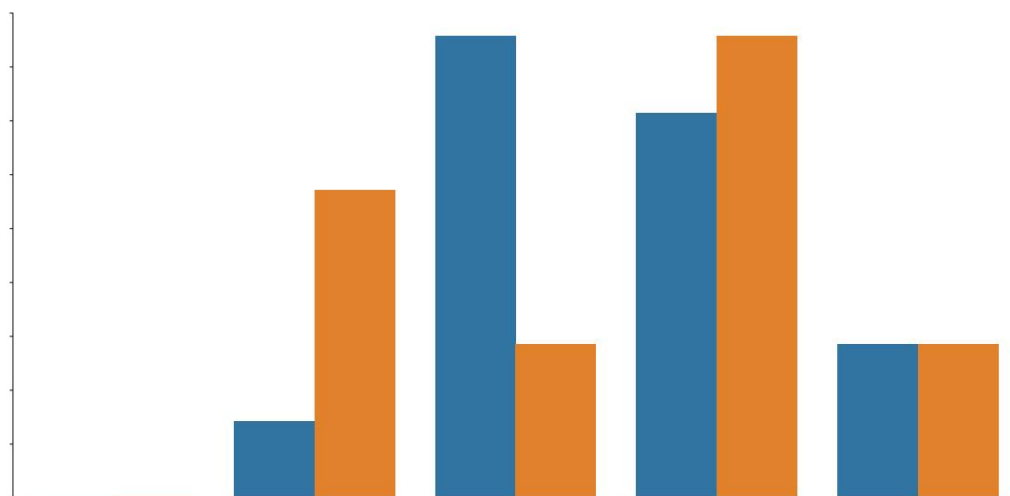
Hình A.20: Trục X: Xếp hạng của người tham gia từ 1: Không biết đến 5: Có thể biết. Trục Y: % người tham gia đã chọn những phản hồi đó



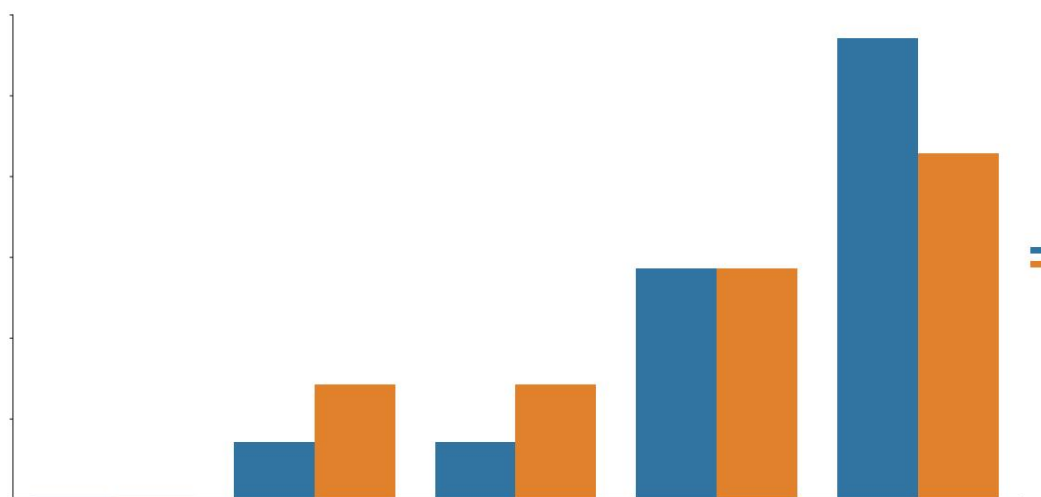
Hình A.21: Trục X: Xếp hạng của người tham gia từ 1: Vô trách nhiệm đến 5: Có trách nhiệm. Trục Y: % người tham gia đã chọn những phản hồi đó



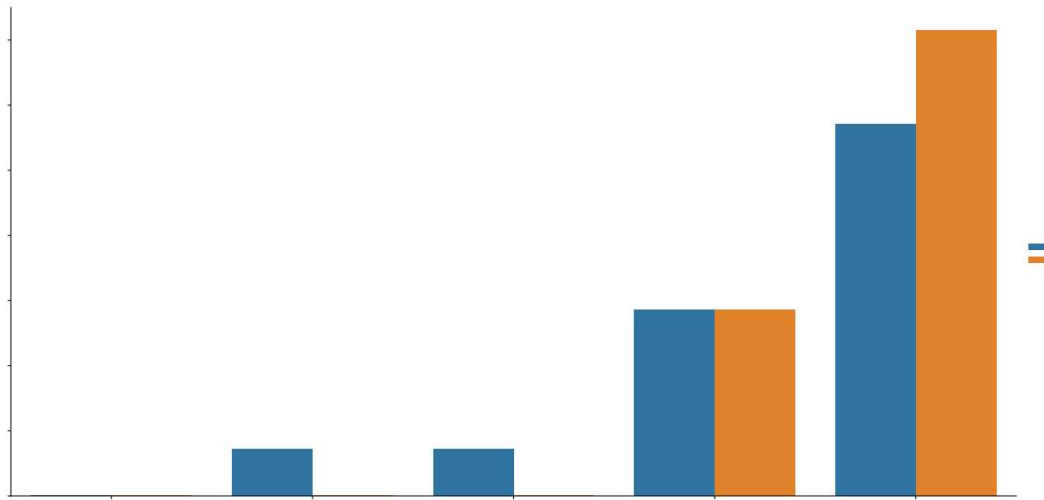
Hình A.22: Trục X: Xếp hạng của người tham gia từ 1: Không thông minh đến 5: Thông minh. Trục Y: % người tham gia đã chọn những phản hồi đó



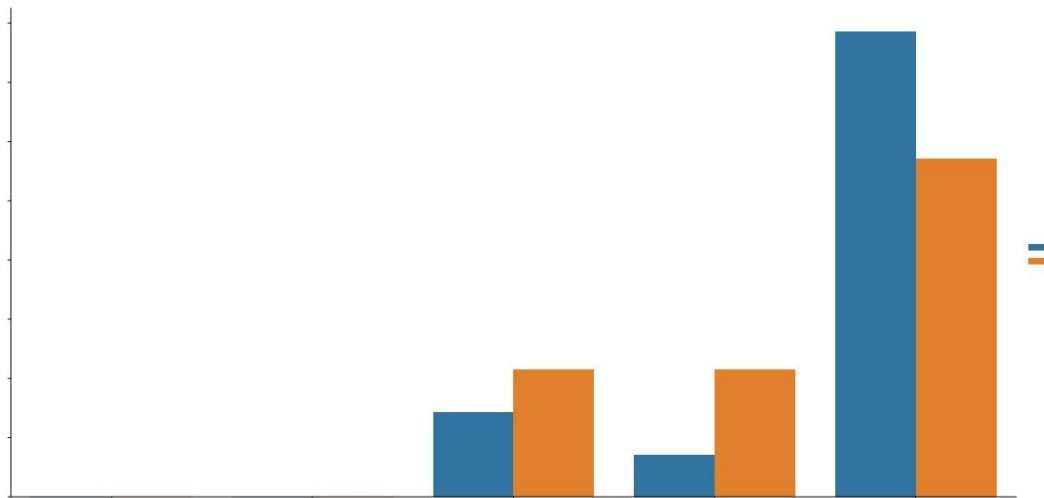
Hình A.23: Trục X: Đánh giá của người tham gia từ 1: Ngủ ngốc đến 5: Hợp lý.
Trục Y: % người tham gia đã chọn những phản hồi đó



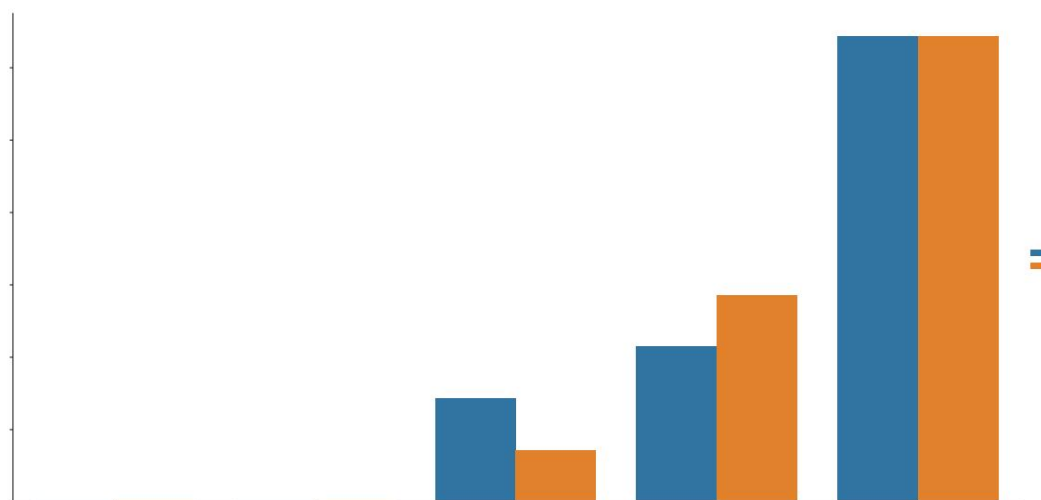
Hình A.24: Trục X: Đánh giá của người tham gia từ 1: Lo lắng đến 5: Thư giãn.
Trục Y: % người tham gia đã chọn những phản hồi đó



Hình A.25: Trục X: Đánh giá của người tham gia từ 1: Lo lắng đến 5: Thư giãn.
Trục Y: % người tham gia đã chọn những phản hồi đó



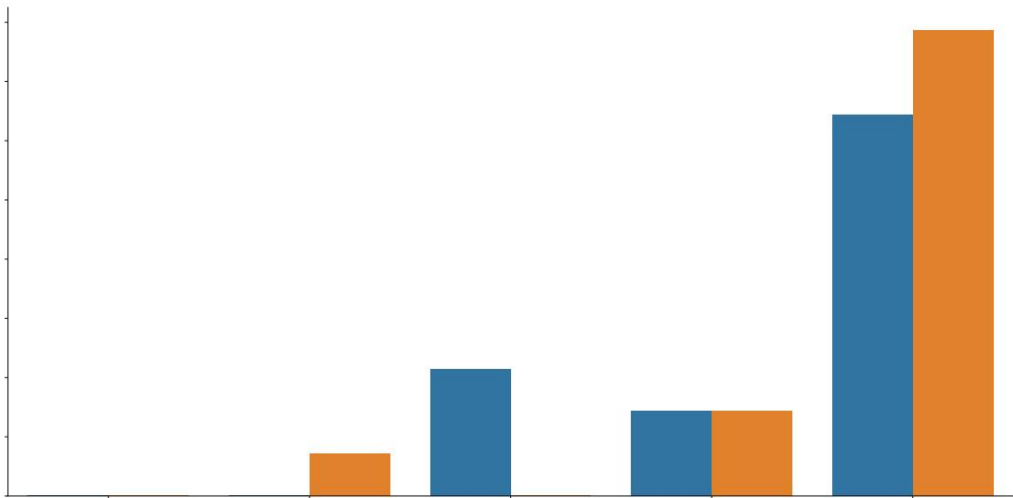
Hình A.26: Trục X: Đánh giá của người tham gia từ 1: Kích động đến 5: Bình tĩnh.
Trục Y: % người tham gia đã chọn những phản hồi đó



Hình A.27: Trục X: Đánh giá của người tham gia từ 1: Kích động đến 5: Bình tĩnh.
Trục Y: % người tham gia đã chọn những phản hồi đó



Hình A.28: Trục X: Đánh giá của người tham gia từ 1: Chán đến 5: Quan tâm.
Trục Y: % người tham gia đã chọn những phản hồi đó



Hình A.29: Trục X: Đánh giá của người tham gia từ 1: Chán đến 5: Quan tâm.
Trục Y: % người tham gia đã chọn những phản hồi đó

PHỤ LỤC B:

BẢNG CÂU HỎI TĂNG CƯỜNG GODSPEED

Bản câu hỏi

1. Bạn cảm thấy gắn bó với robot như thế nào? Chỉ đánh dấu một hình bầu dục.

Không hề 1 2 3 4 5 Rất

2. Tương tác với robot thú vị như thế nào? Chỉ đánh dấu một hình bầu dục.

Không hề 1 2 3 4 5 Rất

3. Bạn có muốn dành nhiều thời gian hơn với robot không? Chỉ đánh dấu một hình bầu dục.

Không hề 1 2 3 4 5 Rất nhiều

4. Đọc câu dưới đây và chọn một trong các tùy chọn được đưa ra: Robot có một

mục tiêu. Chỉ đánh dấu một hình bầu dục.

Có Không

5. Nếu bạn đồng ý, bạn nghĩ mục tiêu của robot là gì? Tại sao bạn nghĩ vậy?

6. Nếu bạn không đồng ý, tại sao bạn lại không đồng ý? Bạn nghĩ robot đã làm gì?

7. Bạn nghĩ con robot này bao nhiêu tuổi (xét về hành vi của nó)?

8. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Giả 1 2 3 4 5 Tự nhiên

9. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Giống máy móc 1 2 3 4 5 Giống con người

10. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Vô thức 1 2 3 4 5 Có ý thức

11. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Nhân tạo 1 2 3 4 5 Giống thật

12. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Di chuyển cứng nhắc 1 2 3 4 5 Di chuyển thanh lịch

13. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Đã chết 1 2 3 4 5 Còn sống

14. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Tù động 1 2 3 4 5 Sôi động

15. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Cơ học 1 2 3 4 5 Hữu cơ

16. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Trơ 1 2 3 4 5 Tương tác

17. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Thờ ơ 1 2 3 4 5 Phản ứng

18. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Không thích 1 2 3 4 5 Thích

19. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Không thân thiện 1 2 3 4 5 Thân thiện

20. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Không tử tế 1 2 3 4 5 Tử tế

21. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Không vui 1 2 3 4 5 Vui

22. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Tệ hại 1 2 3 4 5 Tốt

23. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Không đủ năng lực 1 2 3 4 5 Có năng lực

24. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Không biết 1 2 3 4 5 Có hiểu biết

25. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Vô trách nhiệm 1 2 3 4 5 Có trách nhiệm

26. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Không thông minh 1 2 3 4 5 Thông minh

27. Vui lòng đánh giá ấn tượng của bạn về robot theo thang điểm này: Chỉ đánh dấu một hình bầu dục.

Ngu ngốc 1 2 3 4 5 Khôn ngoan

28. Khi BẮT ĐẦU tương tác, bạn cảm thấy thế nào về thang điểm này: Chỉ đánh dấu

một hình bầu dục.

Lo lắng 1 2 3 4 5 Thư giãn

29. Khi KẾT THÚC tương tác, bạn cảm thấy thế nào về thang điểm này: Chỉ đánh dấu một

hình bầu dục.

Lo lắng 1 2 3 4 5 Thư giãn

30. Khi BẮT ĐẦU tương tác, bạn cảm thấy thế nào về thang điểm này: Chỉ đánh dấu

một hình bầu dục.

Kích động 1 2 3 4 5 Bình tĩnh

31. Khi KẾT THÚC tương tác, bạn cảm thấy thế nào về thang điểm này: Chỉ đánh dấu một

hình bầu dục.

Kích động 1 2 3 4 5 Bình tĩnh

32. Khi BẮT ĐẦU tương tác, bạn cảm thấy thế nào về thang điểm này: Chỉ đánh dấu

một hình bầu dục.

Chán 1 2 3 4 5 Quan tâm

33. Khi KẾT THÚC tương tác, bạn cảm thấy thế nào về thang điểm này: Chỉ đánh dấu một

hình bầu dục.

Chán 1 2 3 4 5 Quan tâm

34. Trong các mối quan hệ sau, bạn cảm thấy mối quan hệ nào mô tả robot tốt nhất? Chỉ đánh dấu

một hình bầu dục.

Anh trai hoặc chị gái

Bạn cùng lớp

Người lạ

Người thân (ví dụ, anh chị em họ hoặc dì)

Bạn bè

Cha mẹ

Giáo viên

Hàng xóm