Scientific Research Publishing

Outfit Helper: A Dialogue-Based System for Solving the Problem of Outfit Matching

Boyu Zhang[1*], Qin Wang[2]

[1]Mary Star of the Sea High School, San Pedro, USA
[2]Nanjing University of Posts and Telecommunications, Nanjing, China
Email: *zhangboyu190828@outlook.com

Abstract

With the development of artificial intelligence in recent decades, intelligent algorithms such as visual recognition and natural language processing have affected and will continue to affect our lives. A smart device designed to help people make wise decisions on dress through dialogue would be a very convenient tool. By using a color extraction algorithm and natural language processing methods, we have built an outfit planning chatbot, named Outfit Helper, that can help male users decide what to wear in order to dress well. According to our study, most users think that the Outfit Helper is a good choice for solving the clothing matching problem. This study shows that our lives can become much easier by developing and using artificial intelligence to help us choose our clothing.

Keywords

Artificial Intelligence, Intelligent Systems, Conversational Agents, Outfit Planning System, Conversational User Interfaces

1. Introduction

A great number of applications of artificial intelligence (AI) have helped people live better. A majority of people use intelligent customer service, machine workers, chatbots, and so on to make their life more convenient. With the development of AI technologies, natural language processing (NLP) has become increasingly popular. Researchers have successfully developed many chatbots based on NLP[1], such as ELIZA [1], XiaoIce [2], PARRY, and Racter, among others. The

---

[1]Natural language processing (NLP) is a subfield of linguistics, computer science, information engineering, and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language data.

wide utilization of NLP has led to more intellectual and natural interactions through conversations between people and machines in therapy, customer service, virtual assistance, and many other areas [3] [4] [5].

NLP's successful application in many areas has undoubtedly made our daily lives easier. However, AI's revolutionary impact on society has only just begun. There are occasions where people are required to dress properly, and it often takes a lot of time to make the right outfit decision. In order to help people dress well without too much effort, a smart device, such as a chatbot, would achieve the goal of outfit planning via conversation.

Although many types of chatbots have been developed, there are still limited cases that accept the NLP method in outfit planning situations. In the present study, by combining a color extraction algorithm with NLP methods by Dialogflow[2], we have built a dialogue-based outfit planning system, called Outfit Helper, which aims to give appropriate dressing advice. According to the evaluation criteria derived from previous research, our Outfit Helper performed well after user testing.

This work has three primary contributions. First, we successfully created a dialogue-based application, Outfit Helper, that can solve the problem of outfit matching. Second, we evaluated the effectiveness, satisfaction, and efficiency of Outfit Helper; the results demonstrate that Outfit Helper is more effective and satisfying though less efficient than non-dialogue-based (NDB) applications. Finally, our research set an example for the further penetration of AI, especially NLP, which may lead to the next wave of research.

In this research—based on conversational outfit planning systems, computer vision, and color match analysis—we developed a chatbot named Outfit Helper that consists of four key parts: a dialogue system, a connection with third-party Internet services, a color wheel algorithm, and an intelligent response system. We conducted an experiment to evaluate our chatbot alongside an NDB app from the perspective of effectiveness, satisfaction, and efficiency. In the discussion section, we answer what the effectiveness, satisfaction, and efficiency of Outfit Helper are as compared to an NDB app based on experimental results and offer further suggestions.

2. Data and Methods

Our work has built upon previous studies on conversational agents in outfit planning, computer vision, and color match analysis.

2.1. Conversational Outfit Planning Systems

When a chatbot is used, the NLP handles incoming messages and then outputs personalized responses to a user via conversational agents. In our work, we designed chatbots and a conversational agent, Outfit Helper, to solve the problem

---

[2]Dialogflow (formerly Api.ai, Speaktoit) is a Google-owned developer of human-computer interaction technologies based on natural language conversations https://dialogflow.com/.

of deciding what clothing to wear (and how to wear it) in different scenarios. We used ManyChat[3] to design the conversation structure and Dialogflow to handle the user's input.

Our prior research analysis confirmed our decision to use a conversational agent to ensure that Outfit Helper is more competitive and advanced than some outfit planning applications already on the market, all of which have their advantages and disadvantages.

Based on our investigation on the number of dialogue-based application in the application market, we noticed that existing applications for outfit planning are not dialogue-based systems, which means a user is not able to interact with the application naturally—in this case, by talking. We call this kind of application anon-dialogue-based (NDB) application. All NDB apps have a common disadvantage, which is that the suggestions they provide are not widely accepted because of the lack of effectiveness and satisfaction. Because of this shortage of dialogue-based outfit planning applications, we decided to develop our Outfit Helper by using a conversational agent, which provides a more unique and creative approach to solve this problem.

Additionally, we linked the conversational agent (ManyChat) with Dialogflow, which handles NLP for the user's input by using Integromat[4]. Integromat allows the conversational agent (ManyChat) to interact with third-party Internet services, such as Dialogflow, Google Sheets[5], Azure Computer Vision[6], and so on.

2.2. Computer Vision

For more precise color, we guide users to upload pictures of their outfit top and bottom, because no one can describe a color with enough precision to be used for color match analysis. For example, most people regard "navy blue" and "royal blue" as simply "blue", so our chatbot is not able to give the most suitable advice due to imprecise information. Comparatively, an image shows more accurate colors than words do. In our work, we discovered that it is necessary to extract the main color of the pictures of a user's top and bottom. According to previous work, we found that we needed to put a computer vision process in our chatbot [6] [7].

We decided to choose Azure Computer Vision because it can detect the main color of an image, while other computer vision systems can only detect the mixed color. This means that Azure Computer Vision is less likely to be affected by background color, and thus is more precise than other computer vision services.

---

[3]ManyChat is a service that allows people to create chatbots for Facebook Messenger https://manychat.com/.

[4]Integromat will automate processes that you currently handle manually. It is not only capable of connecting apps but can also transfer and transform data https://www.integromat.com/en.

[5]Google Sheets is a spreadsheet program included as part of a free, web-based software office suite offered by Google within its Google drive service.

[6]Azure Computer Vision can extract rich information from images to categorize and process visual data—and perform machine-assisted moderation of images to help curate your services https://azure.microsoft.com/en-us/services/cognitive-services/computer-vision/.

2.3. Color Match Analysis

One of the most acceptable and authoritative ways to analyze a color match is by utilizing color theory, which provides visual effects of a color combination by contrasting three relative attributes: value, chroma, and hue.

Researchers in the field of color theory have divided colors into six schemes: complementary colors, analogous colors, triadic colors, split complementary colors, tetradic colors, and neutral colors. Technically, every scheme utilizes a unique color match method to make it easy to systematically combine colors.

Principles of color theory have been widely accepted for over two hundred years. Outfit Helper analyzes a user's outfit color match based on these principles so that it can provide more persuasive, rational, and reasonable suggestions.

3. Design of Chatbot

Outfit Helper comprises four key components: a dialogue system, a connection with third-party Internet services, a color wheel algorithm, and an intelligent response system.

3.1. Outfit Helper Dialogue System

Outfit Helper's dialogue system consists of three parts: information collection, judgment, and intelligent response. Generally, our chatbot collects a user's information via conversation. Based on this information, the chatbot judges if the user's outfit style and color match are appropriate for the occasion. If not, Outfit Helper gives a comprehensive response, which includes text and images.

There are three steps in the process. First, Outfit Helper determines where a user is going. Next, our chatbot asks a user's outfit style, color, or both and collects this information. Finally, it automatically analyzes this information and replies to the user. Each of these steps is carried out through the dialogue between a user and Outfit Helper.

Figure 1 shows one round of conversation between a user and Outfit Helper. In this interaction, Outfit Helper asks a user where he is going. A user then types in the place. If this scenario is not included in our database, our chatbot will not only ask the user to input a similar place for suggestions but also record this input scenario for database expansion in the future. If this scenario is in our database, our chatbot will ask the user's current outfit and give four options: style, color, both, and no idea.

3.2. Style

As shown in Figure 2, if a user chooses "style", Outfit Helper will ask him to upload his outfit's top style and bottom style, such as a T-shirt and jeans, jacket and trousers, and so on. Next, our chatbot searches this data in our pre-configured database. If this clothing match is in our database, Outfit Helper will respond to the user that his clothing style is suitable for the place he is going. If not, our
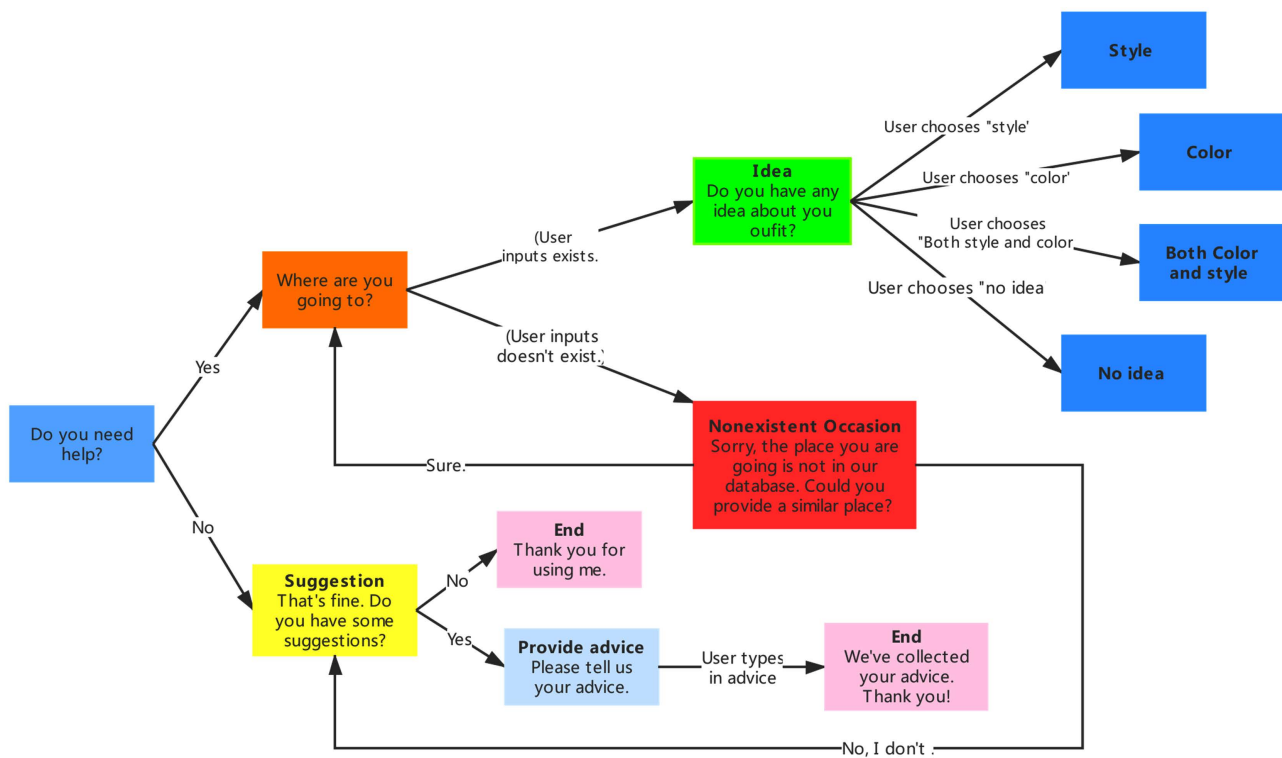
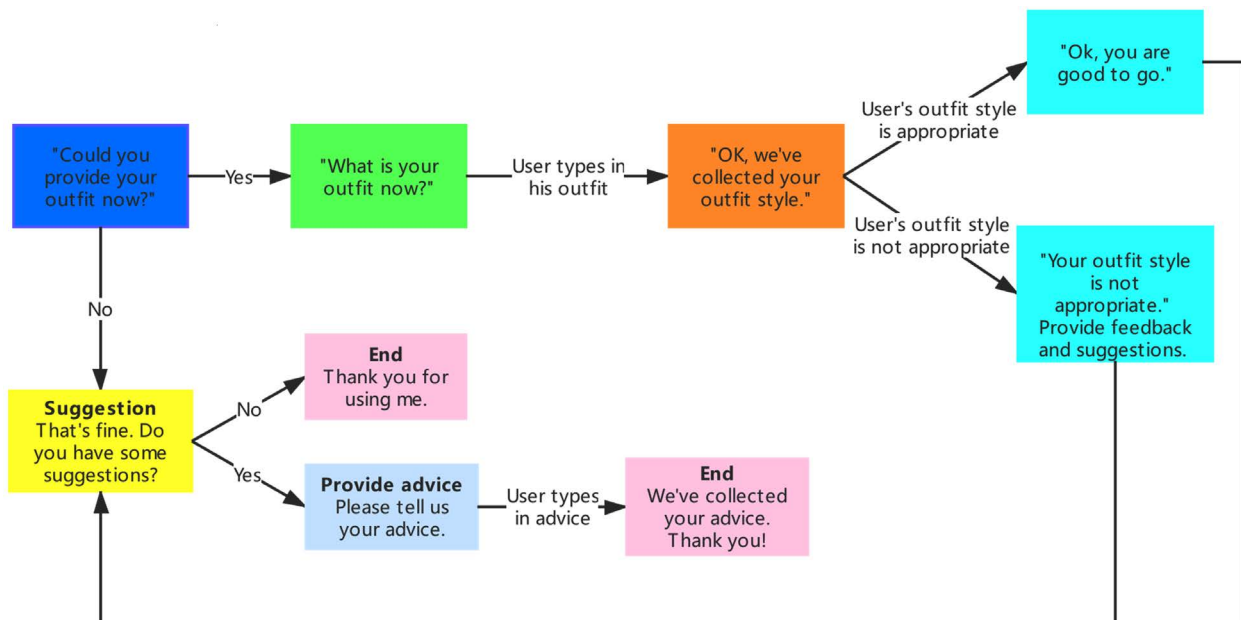**Figure 1.** The entire conversational process of Outfit Helper.

Figure 2. The conversational process of the "style" aspect of Outfit Helper.

chatbot will explain to the user why his clothing style is not suitable for the occasion and provide suggestions via text and images.

3.3. Color

As shown in **Figure 3**, if a user taps on the "color" button, our chatbot will ask
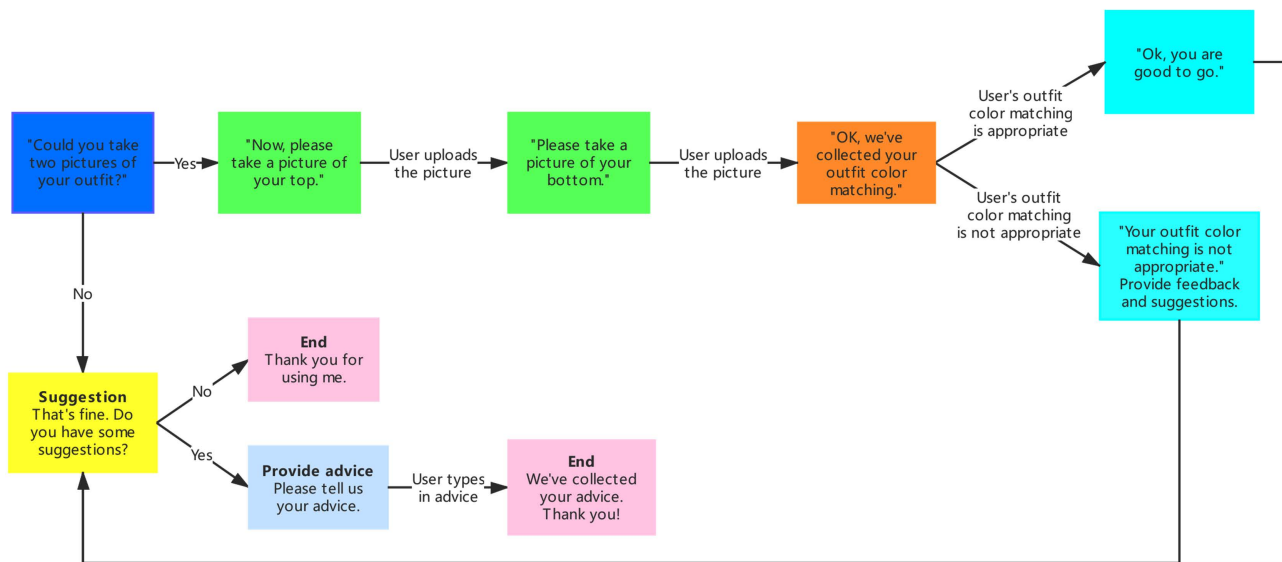
Figure 3. The conversational process of the "color" aspect of Outfit Helper.

him to upload a picture of his top and a picture of his bottom. After receiving the pictures, Outfit Helper extracts their main tone from and searches this color match in our pre-configured database. If this match is in our database, Outfit Helper will respond to the user that his outfit color match is perfect. If not, Outfit Helper will tell the user why this color match is not good and then provide some ideas.

3.4. Both Style and Color

As shown in Figure 4, if a user chooses "both style and color", Outfit Helper will ask him to type in his top style and bottom style. If his outfit style match is suitable for the occasion, our chatbot will then ask the user to upload a picture of his top and a picture of his bottom to judge if his outfit color match is good or not and provide feedback. If the user's outfit color combination is perfect, Outfit Helper will reply to the user that his outfit style and color are good to go. If his outfit color match is not good, our chatbot will tell him the reason this color match is not perfect and respond with some suggestions about color match. If the user's clothing style match is not suitable for the occasion, Outfit Helper will give advice and then ask the user whether he is going to change his clothes or not. If the user is going to change his clothes, our chatbot will ask him to upload two images of his new and original clothes (top first), and Outfit Helper will judge this color match and provide feedback. If the user does not want to change his clothes, our chatbot will ask him to upload a picture of his top and a picture of his bottom, then judge this color match and provide feedback.

3.5. No Idea

As shown in Figure 5, if a user chooses "I don't have any idea", Outfit Helper will ask him what kind of clothes he likes. Afterward, our chatbot gives the user
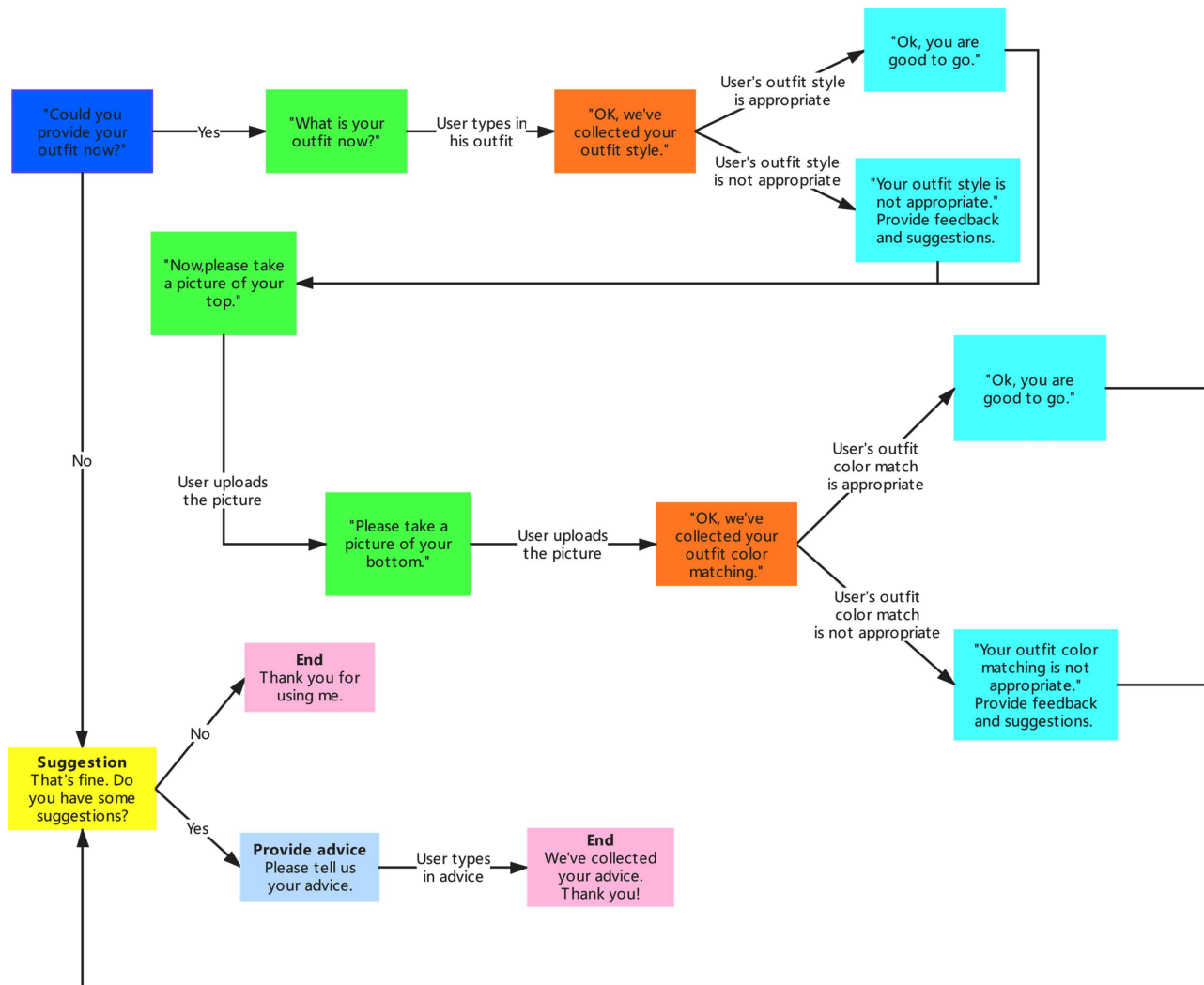
Figure 4. The conversational process of the "both style and color" aspect of Outfit Helper.

Figure 5. The conversational process of the "no idea" aspect of Outfit Helper.

some suggestions about outfit style. After that, Outfit Helper asks the user what color he likes, then responds to advice.

After each final feedback or a user tapping the "I don't need help" button, Outfit Helper asks the user whether he has any suggestions or not. If the user does, he will type in his advice. Then, our chatbot says "See you!" and provides a restart button. If not, our chatbot will say "Goodbye" and provide a restart button.

3.6. A Connection with Third-Party Internet Services

We have created a large database for data collection and analysis in Google Sheets and trained NLP in Dialogflow. We have also applied Azure Computer Vision in Outfit Helper. To develop Outfit Helper become more intelligent, we must connect a third-party Internet service, such as Google Sheets, Dialogflow, Azure Computer Vision, and so on, with a conversational agent (ManyChat). Each of these services has an Application Programming Interface (API), which is regarded as a bridge between itself and our chatbot.

We have used Integromat to integrate these APIs and connect them with the conversational agent ManyChat, because Integromat is not only capable of connecting apps but can also transfer and transform data. Figure 6 shows one possible scenario (occasion determination) using Integromat.

3.7. A Color Extraction Algorithm

In our work, we have used Azure Computer Vision to extract the main color from an image. However, we found that we could only extract a hex number of the main color instead of a name. As a result, we had to transfer this hex number to a color name.

When a user photo is acquired by the Outfit Helper, the photo is sent to Azure Computer Vision service immediately and a hex number representing the main tone or color of the image is returned. An API running on our server is then called to round the hex number to the nearest named color in our database, the name of which is then returned. The API calculates the quadratic sum of the differences of RGB values between the main tone and all named colors.

Let r, g, and b be the RGB value of the main color and r_i, g_i, and b_i be the RGB value of each color in the database.
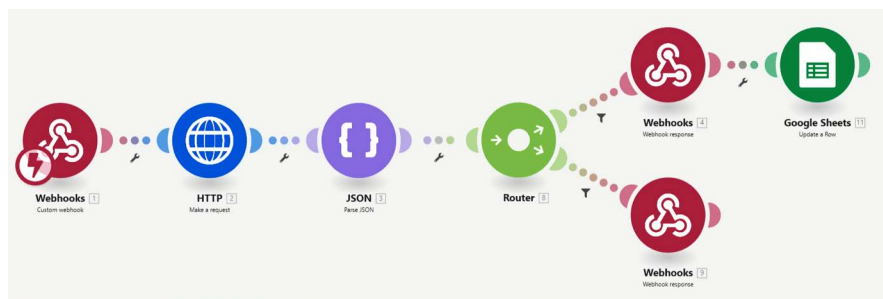

Figure 6. One of the scenarios of Integromat.

$$diffi = \left(r - r_i\right)^2 + \left(g - g^i\right)^2 + \left(b - b^i\right)^2$$

Then we choose the color minimize the differenceias the nearest named color.

3.8. An Intelligent Response System

Originally, when the user's outfit was not appropriate, Outfit Helper told him that his outfit style or color match was not good. However, we thought it was unconvincing to only say no. To solve this, we created a new database that records many reasons why a user's outfit is not right. Now, if a user's clothing style or color match is not suitable, Outfit Helper will tell him why his outfit style or color combination is not good and provide some suggestions.

Here is an example. A user is going to attend a wedding and he wants to know whether his outfit is suitable for the occasion. He wears a shirt and jeans. Outfit Helper tells him that his outfit style is not appropriate at a wedding because jeans are not formal, and then provides some suggestions.

Additionally, the advice outputted by Outfit Helper is based on the user's input. For example, a user who wears a suit and jeans is going to attend school. Outfit Helper tells him that a suit jacket is too formal and provides some suggestions: a T-shirt and jeans, a crewneck and jeans, a hoodie and jeans. All these suggestions are based on jeans. Our chatbot will not provide suggestions unrelated to the user's current outfit, except if his top and bottom are both inappropriate.

4. Experiment Setup

In this section, we describe the experimental setup for our two studies. Based on a review of previous work [8], effectiveness, satisfaction, and efficiency were determined to be the three most important differences between conversational systems and NDB systems.

4.1. Participants

While building Outfit Helper, we adjusted the design using 26 random men and their data to train the NLP, expand the database, and improve the conversational design of our chatbot. Next, we launched Outfit Helper and recruited 39 male students from a high school in San Pedro, CA. Based on alphabetical order of their name, 19 of them participated in the first within-subject study that evaluated the effectiveness and satisfaction of Outfit Helper in comparison to the most popular NDB app, and 20 of them participated in the second within-subject study that compared the efficiencies of the two apps. These participants were compensated $80 for their participation. Each experiment lasted for one week.

4.2. Apparatus

All participants installed Facebook Messenger on their mobile phones, which

provides a platform for the access of Outfit Helper. To begin, they used Messenger to scan a QR code on our website. Participants also downloaded the NDB app on their phones; 34 were iPhones and five were Android phones. Before the experiments started, we evaluated their mobile phones one by one to guarantee that their phones would support these two applications.

5. Evaluation and Results

Effectiveness, satisfaction, and efficiency were the three most important targets in our evaluation of Outfit Helper. Because we evaluated the effectiveness and satisfaction of Outfit Helper in study 1 and evaluated the efficiency of Outfit Helper in study 2, we decided to use two different methodologies. In study 1, we proposed three questions in the form of a questionnaire provided to each participant before starting the experiment:

1) How many times did you use Outfit Helper this week? And how many times did you accept its advice?

2) How many times did you use the NDB app this week? And how many times did you accept its advice?

3) Which application do you feel more satisfied with? Why?

In study 2, we gave participants two sheets, one for Outfit Helper and another one for the NDB app, as shown in Figure 7, and asked them to record the time they spent on each app from the second day to the seventh day. The reason the sheet does not include the first day was to give the user time to become familiar with the two apps. After one week, we collected the questionnaires and analyzed the data.

5.1. Study 1: Evaluation of Effectiveness and Satisfaction

The following section describes our evaluation of the effectiveness and satisfaction of Outfit Helper and the NDB app.

The first experiment we ran was a within-subject study designed to evaluate the effectiveness and satisfaction of Outfit Helper. We recruited 19 male high school students. Their average age was 16.3 years old.

Every user was required to use Outfit Helper and the NDB app every day over a period of one week. Users chose the use frequency of each app of their own

Frequency usage time(s) Day	2	3	4	5	6	7
1						
2						
3						
4						
5						
6						
7						
8						

Figure 7. The sheet for study 2.

volition, though they were advised to use each of the two apps at least once per day to provide more precise information.

The first question of the study relates to frequency of use. The 19 participants used Outfit Helper a total of 196 times in a week, and accepted the advice provided by Outfit Helper a total of 159 times during the same period. We used the following formula to calculate the recommended adoption rate:

$$\text{The recommended adoption rate} = \frac{\text{total number of recommendations adopted}}{\text{total use times}} \times 100\%$$

Therefore, the total recommended adoption rate is 81.1% (159/196 × 100% ≈ 81.122449%).

Next we addressed the second question, regarding frequency of use of the NDB app. The 19 participants used NDB app a total of 142 times over the one-week period, and accepted the advice provided by NDB app 96 times. Therefore, the total recommended adoption rate is 67.6% (96/142 × 100% ≈ 67.605634%).

After analyzing the answers to question number three, regarding satisfaction, we discovered that Outfit Helper is more satisfying than NDB application in two dimensions. First, Outfit Helper is able to build a good relationship with a user due to NLP, so the user will trust and depend on the app.

"*Every time I used Outfit Helper, I felt like I was talking to my assistant who knows what I should wear today and never laugh at me. I was inclined to use it to get some ideas instead of asking other people for their advice.*"

"*It* [*Outfit Helper*] *is not just an app; it is more like my good friend. I felt very relaxed when I was using it, so I was glad to share my ideas with it.*"

In addition, Outfit Helper can give more reasonable suggestions to make the user experience more satisfying.

"*When I wanted to wear a T-shirt and jeans at a wedding, Outfit Helper told me that I couldn't wear that because a T-shirt is not formal enough. It is persuasive and I'd like to believe it.*"

"*...In my life, I will try my best to figure everything that disagrees with me out ... Outfit Helper tells me the reason why my outfit is not suitable for a particular occasion, while the NDB app can't.*"

5.2. Study 2: Evaluation of Efficiency

The second within-subject study was designed to evaluate the efficiency of the two apps. We recruited 20 male high school students. Their average age was 16.7 years old.

Every user was required to use Outfit Helper and the NDB app every day over the period of one week. Participants chose the use frequency of each app of their own volition, though they were advised to use each app at least twice per day to provide more precise information.

After one week, we collected the sheets and analyzed the data to evaluate the effectiveness of Outfit Helper.

Outfit Helper was used a total of 288 times, and users spent a total of 35286.37

seconds on the app. We use the following formula to calculate the average time per user spent using Outfit Helper:

$$\text{Average time} = \frac{\text{total time}}{\text{total frequency}}$$

The average time per user spent using Outfit Helper per time is 122.46 seconds ($35286.37/288 \approx 122.459618$).

According to the questionnaires regarding the NDB app, NDB app was used a total of 263 times, and users spent a total of 26328.64 seconds on the app. Therefore, the average time per user spent using the NDB app is 100.11 seconds ($26328.64/263 \approx 100.108897$).

6. Discussion

In this section, we discuss and answer our research questions based on the evaluation results and offer suggestions for further improving an expanded database and designing a more convenient interface.

6.1. Effectiveness and Satisfaction

As hypothesized, Outfit Helper is more effective and satisfying in comparison to the NDB app. After collecting and calculating all data, we discovered that the recommended adoption rate of Outfit Helper (81.1%) is higher than the recommended adoption rate of the NDB app (67.6%). This demonstrates that the suggestions given by Outfit Helper are more widely acceptable, illustrating that Outfit Helper is more effective than the NDB app. We found that the total number of times participants used Outfit Helper (196) is higher than the number of times participants used the NDB app (142). This demonstrates that users are more inclined to use Outfit Helper than the NDB app. Notably, Outfit Helper is also more satisfying than the NDB app. When analyzing participants' answers to question 3 in study 1, we observed that there are two main contributors to this higher satisfaction. First, Outfit Helper is able to build a good relationship via the use of NLP, which leads to trustworthiness. Second, Outfit Helper can give the user explanations and more reasonable suggestions, which in turn improves satisfaction and a greater likelihood that he will adopt these suggestions.

6.2. Factors Contributing to Effectiveness

Our evaluation demonstrated that Outfit Helper is more effective in teaching people how to dress appropriately. After analyzing participants' comments, we believe three main factors contribute to Outfit Helper's effectiveness.

First, Outfit Helper tells the user not only how to dress but why, which plays an important role in the process of learning. This process is like solving a mathematical problem. If a teacher just tells students the answer to the question, students will not be able to solve similar problems. However, if a teacher tells students not only the answer but also how to solve it, most of them will be able

to solve a similar problem next time. It is important to tell the user why he should dress like Outfit Helper suggests to make sure the user is learning.

Second, recognizing and correcting mistakes is an important part of learning. By using Outfit Helper, the user can recognize his clothing mistakes in terms of style or color match and correct them using the targeted suggestions provided by Outfit Helper. Without recognizing his mistakes, the user will be unable to solve similar problems encountered in the future. For example, if a user is going to attend a funeral wearing a T-shirt and jeans, Outfit Helper will tell the user that this outfit is unsuitable because a funeral is a solemn occasion, while a T-shirt and jeans are very casual. Next time, the user will not ask if it is appropriate to wear a T-shirt and jeans to attend a funeral because he already knows that he should not wear casual clothes on a solemn occasion.

Third, providing professional suggestions about clothing color match is a unique characteristic of Outfit Helper. The user will learn how color theory works by using Outfit Helper because we designed criteria of color matching primarily based on the principles of color theory. Color theory has a wide variety of applications, helping the user to not only when color matching clothing but also in other related questions of color matching such as paintings, decorations, designs and so on. Through the long-term use of Outfit Helper, users will become more and more professional about color matching and will gradually learn to apply these skills to other areas of their life.

6.3. Trade-Offs between Effectiveness, Satisfaction, and Efficiency

As mentioned, effectiveness and satisfaction are the two most impressive advantages of Outfit Helper in comparison to the NDB app. Conversation has been shown to be a good way to improve the effectiveness and satisfaction of Outfit Helper and results in higher usage.

However, a highlevel of effectiveness and satisfaction also causes inefficiency. Because we designed Outfit Helper based on a conversational system, a user has to talk with the app to achieve the purpose of use instead of getting results directly. After analyzing the participants' conversational timestamps, we broke down the interaction time with Outfit Helper, shown in Figure 8. As can be seen, 15.7% of the total time was spent on actions other than planning an outfit with Outfit Helper. For example, we deliberately added delays for Outfit Helper's sequential conversations to make it feel that a real person was typing, which caused a 4.3% reply delay. Users also spent 48.1% of the total time on the app typing in their replies to Outfit Helper. Had we removed these casual aspects from Outfit Helper, users' interest or motivation in using the might have been negatively affected. Therefore, with increased effectiveness and satisfaction, here comes a certain amount of inefficiency.

6.4. Limitations and Future Work

There are three obvious disadvantages of Outfit Helper: inefficiency, lack of a robust database, and lack of personalization.
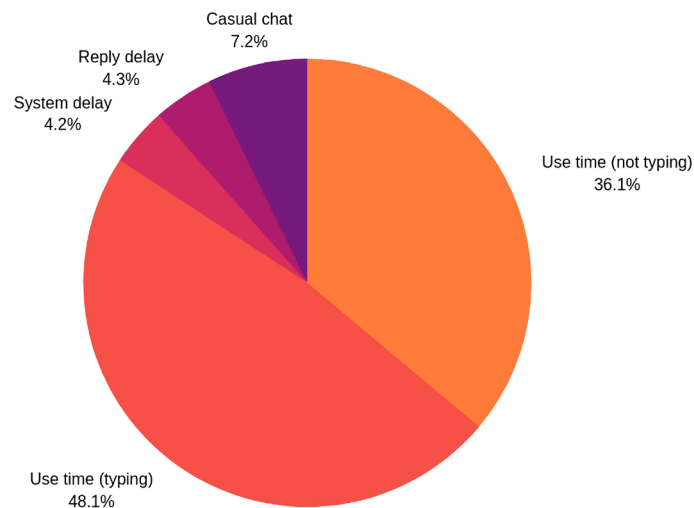
Figure 8. Proportion of time spent on user and system activities in Outfit Helper from our analysis of participants' timestamps.

First, Outfit Helper is inefficient due to its high level of effectiveness and satisfaction. We analyzed the proportion of time spent on different activities, as shown in Figure 8. Some delays can be improved. System delay can be shortened with faster algorithms and better Internet connections. Users also spend a significant amount of time on interactions, especially typing. Previous research shows that speech-based input methods are about three times faster than keyboard-based input methods [9]. To improve users' input speed, we should consider supporting speech in addition to keyboard-based conversation.

Second, Outfit Helper can only provide universal clothing matches and can only be used by males because of the insufficient database. To provide various styles, we could ask for artists' help and add more data to our database. For example, a user could choose which kind of style he likes, such as hip hop, street, athletic, and so on. Then, Outfit Helper would provide some suggestions based on the style he chose. Importantly, different countries have different clothing styles. We should also consider expanding this aspect in our database. Because there are significantly more clothing styles for women than for men, we did not have enough people and time to address women's needs with Outfit Helper. However, we could provide this feature in the future.

Third, because our image database is not large enough, and the image provided is random, once users use Outfit Helper for a longer period of time, they will inevitably receive the same image more than once. This will cause users to lose interest; therefore, we need to add more pictures to our image database. At the same time, because the pictures received by users are all random, Outfit Helper lacks personalized settings. We imagine that we can divide all pictures into different types and add "like" or "dislike" buttons to each picture and collect the likes and dislikes of each user. In this way, Outfit Helper can push similar pictures to the user according to his expressed style and reduce the push of the pictures he does not like.

Additionally, many participants reflected that it is inconvenient to scan the QR code on the website to start using Outfit Helper for first time. Because we could not directly provide an interface on Google Site, we provided a QR code. In the future, we would create a new website to provide an interface for users directly.

7. Conclusion

In this research, we designed, built, and evaluated Outfit Helper, a conversation-based system for outfit planning that incorporates the conversational agent ManyChat with third-party Internet services such as Google Sheets, Dialogflow, and Azure Computer Vision by applying Integromat. We compared Outfit Helper to a traditional outfit planning tool in two user studies. We discovered that Outfit Helper has two primary advantages: effectiveness and satisfaction, mainly because of more practical suggestions and humanization. The trade-off, however, is that the efficiency of Outfit Helper is comparatively low. Our work also offers design suggestions to further improve dialogue-based outfit planning systems. Considering these drawbacks, we will improve our chatbot in the future by increasing its efficiency and expanding the database. With AI becoming increasingly powerful, we hope our work will attract more researchers to develop effective and engaging natural language outfit planning systems.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Weizenbaum, J. (1966) ELIZA—A Computer Program for the Study of Natural Language Communication between Man and Machine, Mass. *Communications of the ACM*, **9**, 36-35. https://doi.org/10.1145/365153.365168

[2] Shum, H.-Y., He, X.-D. and Li, D. (2018) From Eliza to XiaoIce: Challenges and Opportunities with Social Chatbots. *Frontiers of Information Technology & Electronic Engineering*, **19**, 10-26. https://doi.org/10.1631/FITEE.1700826

[3] Alarifi, A., Alsaleh, M. and Al-Salman, A. (2016) Twitter Turing Test: Identifying Social Machines. *Information Sciences*, **372**, 332-346. https://doi.org/10.1016/j.ins.2016.08.036

[4] Metz, R. (2017) Three Weeks with a Chatbot and I've Made a New Friend. *MIT Technology Review*. https://www.technologyreview.com/s/603936/three-weeks-with-a-chatbot-and-ive-made-anew-friend

[5] Ferrara, E., Varol, O., Davis, C., Menczer, F. and Flammini, A. (2016) The Rise of Social Bots. *Communications of the ACM*, **59**, 96-104. https://doi.org/10.1145/2818717

[6] Luca, D., Eleonora, I., Giulio, M. and Andrea, P. (2019) Fashion Product Classification through Deep Learning and Computer Vision. *Applied Sciences*, **9**, 1385. https://doi.org/10.3390/app9071385

[7] Tomohiro, N. and Yuichil, T. (2017) Proceedings of IEEE International Geoscience & Remote Sensing Symposium.23-28 July 2017, 9-12.
http://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=GeneralSearch&qid=42&SID=D6PzQsWhblqGOfvUzng&page=2&doc=15

[8] Radziwill, N.M. and Benton, M.C. (2017) Evaluating Quality of Chatbots and Intelligent Conversational Agents. https://arxiv.org/abs/1704.04579

[9] Ruan, S., Wobbrock, J.O., Liou, K., Ng, A. and Landay, J.A. (2018) Comparing Speech and Keyboard Text Entry for Short Messages in Two Languages on Touchscreen Phones. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, **1**, 23 p.
https://dl.acm.org/citation.cfm?doid=3178157.3161187
https://doi.org/10.1145/3161187