FACULTY OF ENGINEERING AND ARCHITECTURE

Adaptive second language tutoring through generative AI and social robots

Eva Verhelst
Student number: 01704986

Supervisors: Prof. dr. Tony Belpaeme, Prof. dr. ir. Thomas Demeester
Counsellor: Ir. Ruben Janssens

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Computer Science Engineering

Academic year 2022-2023

GHENT UNIVERSITY

Acknowledgements

First of all, I would like to thank my supervisors, prof. Tony Belpaeme and prof. Thomas Demeester and my counsellor, Ruben Janssens for the support, guidance and the many inspiring discussions during our meetings. I wish to express my gratitude to Niels and Elke for being companions throughout this journey. I would also like to thank Maria Jose Pinto Bernal for the repeated feedback on everything related to the Spanish language, as I do not speak it at all.

Finally, I would like to thank those closest to me. First, I would of course like to express my gratitude to my parents for constantly being there throughout all the years of this degree, from the first excited choice through all the stress, until this final stage. Finally, I am grateful for Lucas, for sharing this journey with me, while always providing support, love and interesting discussions, Anna, for listening to all of my concerns and always being there, and all the other friends and family that provided support, distraction and fun.

Eva Verhelst

This master's dissertation is part of an exam. Any comments formulated by the assessment committee during the oral presentation of the master's dissertation are not included in this text.

Adaptive second language tutoring through generative AI and social robots

Eva Verhelst

Supervisors: Prof. dr. Tony Belpaeme, Prof. dr. ir. Thomas Demeester
Counsellor: Ir. Ruben Janssens

Master of Science in Computer Science Engineering
2022-2023

Abstract

This master's dissertation proposes a way for social robots to tutor students learning a second language. It does so through the development of a visually grounded game, where the words are represented by images. Due to the recent strong increase in the quality of large language models and related technology, a novel way to generate real-time educational content through generative AI is proposed, allowing adaptation to the skills of the student while providing new content. The aim of this tutoring system is to provide one-on-one practice for students in the classroom, where this is often not possible due to a shortage of teachers and limited funding. Social robots are especially suited for language education, as language learning is inherently social and the robot's embodiment allows for natural social interactions. Both the learning effect of the proposed game on the vocabulary acquisition and the influence of the robot's presence were tested during a user study with Belgian high school students. From this study, it was concluded that there was a significant learning effect, while the presence of the robot did not have a significant impact on the students' vocabulary acquisition. Further development of the proposed game shows great potential in providing second language practice when played with a social robot tutor, where content is continuously adapted to the needs of the student.

Keywords: Social Robots, Generative AI, Robot-Assisted Language Learning, Natural Language Processing

Adaptive second language tutoring through generative AI and social robots

Eva Verhelst

Supervisors: Prof. dr. Tony Belpaeme and Prof. dr. ir. Thomas Demeester

Counsellor: Ir. Ruben Janssens

Abstract— **This master's dissertation proposes a way for social robots to tutor students learning a second language. It does so through the development of a visually grounded game, where the words are represented by images. Due to the recent strong increase in the quality of large language models and related technology, a novel way to generate real-time educational content through generative AI is proposed, allowing adaptation to the skills of the student while providing new content. The aim of this tutoring system is to provide one-on-one practice for students in the classroom, where this is often not possible due to a shortage of teachers and limited funding. Social robots are especially suited for language education, as language learning is inherently social and the robot's embodiment allows for natural social interactions. Both the learning effect of the proposed game on the vocabulary acquisition and the influence of the robot's presence were tested during a user study with Belgian high school students. From this study, it was concluded that there was a significant learning effect, while the presence of the robot did not have a significant impact on the students' vocabulary acquisition. Further development of the proposed game shows great potential in providing second language practice when played with a social robot tutor, where content is continuously adapted to the needs of the student.**

Keywords— **Social Robots, Generative AI, Robot-Assisted Language Learning, Natural Language Processing**

I. Introduction

Learning a second language in school often starts with a group of students being taught grammar and vocabulary by a teacher in front of the class. The more natural way to learn a language is through social interactions and conversation, but there is not enough funding and too few teachers for this kind of extensive one-on-one practice of the language.

As generative AI such as large language models gained impressive momentum since late 2022 [1], this master's dissertation can propose a novel solution to this problem through the use of social robots as second language tutor, powered by generative AI. These social robots could bring support to teachers and offer one-on-one practice to the students. The tutor will never get tired of practice and will always be in a good mood, which can not be expected of teachers. As language learning is a social experience, the social appearance of the robots and their embodiment make them well suited as tutors.

In this work, a tutoring system is proposed, where the content is fully generated by AI: large language models are used for the text while a generative text-to-image model provides the visual content. As this content can be generated in real time, this system allows for constant adaptation of the difficulty as well as the content to the needs of the student.

For the application to adapt itself to the needs of the student, it must be able to estimate these needs. To do this, a Bayesian student modeling technique is used. This enables the robot tutor to present the student with content of the right difficulty, offering easier practice when the student needs it and increasing it when the student is ready for a challenge.

Combining the social abilities of their embodiment with the generative AI that is becoming increasingly better into an adaptive system that provides the student with what they need, might allow robot tutors to be the perfect support for teachers in the classroom.

The proposed tutoring system is tested in a user study, in order to answer two research questions. The first is on the learning effect of the game on the students' vocabulary acquisition in a second language:

Research question 1: What are the learning outcomes when learning a second language together with a social robot driven by generative AI?

The second research question is on the effect that is caused by the presence of the social robot tutor:

Research question 2: Does the presence of a social robot affect the vocabulary acquisition of students?

In this paper, first a background on the relevant fields will be given (Section II), starting with human-robot interaction and education, after which the field of natural language processing follows. Then, the shape of the game will be discussed, together with all the requirements for implementing it (Section III). Then, in Section IV the technical implementation will be discussed. In Section V, an overview of the results will be given. Finally, Section VI concludes this paper.

II. Background

A. Human-Robot Interaction and Education

As schools often cope with limited funding, shortages of teachers and growing classrooms, technology can offer relief by supporting teachers and providing one-on-one tutoring and adaptive exercises. This is often in the form of Intelligent Tutoring Systems (ITS). These are computer systems that guide learners through exercises, personalized to the needs of the student. To achieve this, the ITS must have an estimation of the skill of each student. How this can be achieved, is researched in the field of student modeling or learner modeling. Here, the goal is to use a model that can estimate the skill and knowledge of a student. A well known model often used to achieve this is Bayesian Knowledge Tracing (BKT), which was used in this master's dissertation.

The advantages of social robots over ITS are partially due to their embodied nature, which enables them to interact with the

physical world. The embodied nature of the technology leads to interactions being perceived as social, which can also be beneficial for learning, as research shows that there are increased learning gains when students are interacting with embodied social robots over virtual agents [2]. The social appearance of the robot also leads to specific expectations that the robot will understand speech and social signal without issues. Both of these have undergone strong improvements over the last years, but these technologies are still imperfect. [3]. Especially the performance of speech recognition when working with children is still insufficient [4, 5]. A lot of research on the effect of social robots has been done, with many promising results, while also uncovering many technical challenges. Up until now, most studies used the robot tutors in restricted scenarios. Here, the affective and cognitive outcomes are generally positive [2].

Learning a language is an inherently social act [6]. Children learn their first language from their parents and the people around them, and learning a second language opens the door to communication with a larger, more varied group of people. Research suggests that social robots are able to help students with vocabulary acquisition, while more research is needed to compare social robots with other technologies in teaching other aspects of language. It has also been demonstrated that robots aid learning when used next to a human teacher. Lastly, research shows that robots have a positive effect on the learners' affective state [4]. Next to this, social robots interact with students the way people interact with each other and they can be customized to the specific needs of the student. Additionally, learning a language involves a lot of repetition. This may lead to fatigue and boredom in human tutors or teachers, while a robot tutor does not suffer from this. [7]

When using robots in applications where social interaction is key such as language education, there is a need for a human face that looks and moves realistically. In social interactions, human lips carry much information on speech and intonation, while eyes and their gaze show much about the attention and affect of a person. [8].

In 2011, Furhat Robotics introduced the Furhat: a robot with a back-projected face. This is an interesting combination of a digital animated face, projected on a physical, three dimensional robot. The Furhat consists of a head with a neck, and can perform some basic natural movements such as nodding, shaking its head and raising its eyebrows. An important advantage of the back-projected face, is that it is very customizable: the gender, skin color, size of its features and amount of makeup can all easily be adjusted, allowing use in various circumstances [8]. The Furhat was the robot that took on the role of social robot tutor within this master's dissertation.

B. Natural Language Processing

Word embeddings When using a computer to process natural language, the words of this language must be represented in a way that computer can understand them: numerically. In the earlier days of the field, this was done by representing each word as its index in the used vocabulary list. The problem with this method is that the representations have no notion of similarity between them. Because of this, the idea of transforming words to an embedding space arose. Here, words are represented by a vector in a multidimensional space, with the assumption that vectors that are closer together, also have a more similar meaning and words with a similar difference vector also have a similar relation [9]. This transformation is usually done by a neural network. Important examples are Word2Vec by Google [9, 10], GloVe [11] and FastText [12, 13].

Large language models One of the most fundamental tasks in NLP is that of language modeling: predicting whether a given sequence of words is likely or not, or, in a more probabilistic view, finding the joint probability function over sequences of words. This task is very difficult to solve due to the so-called curse of dimensionality: the options grow exponentially with the length of the sentence, where the base is the size of the vocabulary, usually a very large number. This task was often attempted using statistical models, but is quickly infeasible due to the aforementioned dimensionality problem. The first well known attempt to tackle this problem using neural networks was by Bengio et al [14]. Here, the model jointly learned distributed representations of each word, or word embeddings, as well as the probability function for word sequences. Then, in 2017, Vaswani et al introduced the transformer [15]. The existence of the transformer suddenly made it possible to train neural networks on larger amounts of data, enabling the emergence of Large Language Models (LMM). These often have billions or more parameters and are trained on large amounts of unsupervised data. They are often trained using a certain language modeling task, after which they can be used for a variety of other NLP tasks, with little to no extra training.

In 2018, OpenAI published GPT: a family of large language models called Generative Pre-trained Transformers. These generative models have large amounts of parameters, e.g., GPT-3 has 175B parameters. These models trained on large corpora opened the door for completing tasks with very little or no training on the specific task. Few-shot learning arose, where it is possible to provide the LLM with few examples of the task at hand, after which the model can complete further instances of this task. Even more extreme is zero-shot learning, where the LLM is able to complete unseen NLP tasks, due to the natural language understanding gained in its general training [16–19]. A second influential family of transformer models was introduced by Google researchers in 2018: Bidirectional Encoder Representations from Transformers (BERT) [20].

Text-to-image A text-to-image model is a form of multimodal NLP model: it takes language as an input, but the output is visual. More specifically, text-to-image models take as input a natural language sequence, e.g. a description, and return an image with similar semantic meaning to the sequence [21].

As text-to-image models take a natural language sequence as input, part of their recent success can be attributed to the progress in large language models. The text input is first processed by a LLM, after which it is used in image generation. These models are trained on immense amounts of data found on the internet, often with the caption of the image as label.

Some of the popular text-to-image models are diffusion models. As input, these models take images consisting of random noise. Then, guided by the processed natural language se-

quence, they iterate over the image many times. In each iteration, some noise is removed to bring the image semantically closer to the text input, until a detailed, sometimes even photo realistic image remains. An example of these diffusion text-to-image models is Imagen by Google [22]. Other well known text-to-image models are Stable Diffusion, an open source diffusion model [23] and Midjourney, published in beta by the research lab Midjourney [24].

Translation The core task of translation models is machine translation: the model must find the most probable sentence in the target language, given a sentence in the original language. In the beginning of machine translation, statistical models were used. These were built over many years using domain knowledge. Later, neural models were used, strongly improving the performance. The first occurrence of neural machine translation was by Sutskever et al. at Google [25]. Since the arrival of the transformer, these have dominated the field of machine translation [26].

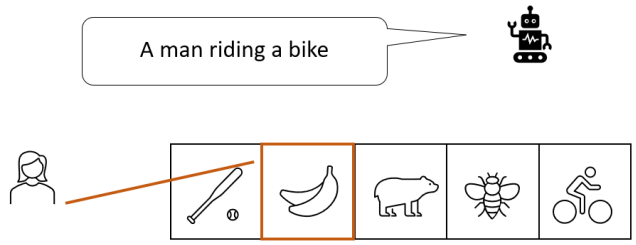
III. Visual Game Playing

The basis of the game is a vocabulary list, that contains all of the words that the student is practicing. The game consists of multiple rounds, in each of which one of the vocabulary words is practiced. The social robot presents the student with a sentence in the goal language containing the practice word. Then, a set of five images is shown. One of these images corresponds to the practice sentence. The other four images are considered distractors. They depict scenes that contain other words from the vocabulary list. Then, the student has to indicate which image corresponds best to the description spoken by the robot. If the student chooses the correct image, a green border appears around it. If the student chose the wrong image, a red border appears around it, while a green border appears around the correct image. Above both images, the corresponding vocabulary word is shown in the goal language. This way, the student gets a chance to learn both words.

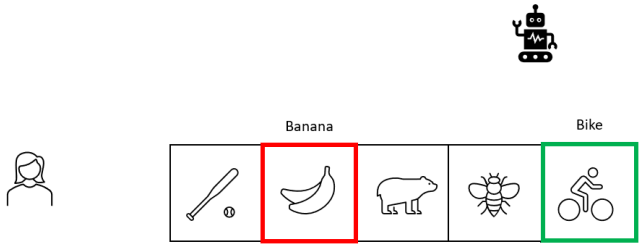
Figure 1 shows a schematic representation of the game, with English as the goal language. The robot speaks a description, "A man riding a bike", which corresponds to the fifth image. The user wrongly indicates the second image. Then, the right and wrong image are shown using a respectively green and red border, while the words corresponding to these images are shown above in English, the goal language.

In order for the game to stay challenging, the difficulty should match the student's knowledge. For this to be possible, student modelling is used. What is required for adjusting the difficulty of the game, is an estimate of how good the student knows every one of the vocabulary words. This estimate should then be adjusted after each exercise. When a student answers an exercise correctly, the student model is updated positively for this one word. When a student gets an exercise wrong, it can be assumed the students did not know the word that was being practiced, as well as the word that was wrongly indicated. To clarify, using the example of Figure 1, the student is here assumed to not know the words "banana" and "bike". Then, the model is updated negatively for these two words.

As the student model provides estimates of the student's skill, the game should be adjusted according to this. This happens



(a) A description of the image is spoken by the robot, the user indicates which one they think corresponds best



(b) The correct and incorrect answers are indicated by green and red respectively, while their corresponding vocabulary word is shown

Fig. 1: Schematic representation of the form of the game.

with the choice of distractors. When a word is not well known, very different distractors from the word that is being practiced are used. As the estimate of the student's knowledge of this word increases, the distractors become more similar to the word that is being practiced.

The proposed game has some requirements. First, a vocabulary list of words that the student should learn should be provided, for example by the teacher. The vocabulary list used here was taken from an English course book [27]. Then, descriptions containing these words as well as images fitting to these descriptions are needed. Both of these are obtained through generative AI models. As the goal language for the students in the study was Spanish, the descriptions should be translated from Spanish to English, as they were generated in Spanish, but the used text-to-image model is intended for usage in English, as indicated on their Huggingface page [28, 29]. In order to adjust the difficulty levels of the game, a student model is needed. Here, because of its intuitive parameters and simple implementation, a Bayesian model is used. Lastly, to choose which distractors are used, a way to model word and sentence similarity is necessary. This is done using a transformation of the words and sentences to an embedding space. All of the models mentioned above are described in detail in Section IV.

IV. Implementation

A. *Choice of words*

When choosing which words appear as practice word and as distractors, an estimate of the students' knowledge as well as a metric for the similarity of the words is needed.

The students' knowledge is estimated using a Bayesian Knowledge Tracing (BKT) model [30], with its parameters chosen from literature [31] or based on the shape of the game.

Equations 1 and 2 show the update equations for a respectively correctly and incorrectly answered exercise, while Equation 3 shows the prediction equation of this model.

$$\text{if } c = 1 : \theta_i' = \frac{\theta_i(1 - P_s)}{\theta_i(1 - P_s) + (1 - \theta_i)P_g} \tag{1}$$

$$\text{if } c = 0 : \theta_i' = \frac{\theta_i P_s}{\theta_i P_s + (1 - \theta_i)(1 - P_g)} \tag{2}$$

$$P_{correct,i} = P_g(1 - \theta_i) + (1 - P_s)\theta_i \tag{3}$$

The similarity between words and sentences was calculated using the Euclidean distance metric on a word or sentence embedding. The embeddings were calculated using the *all-MiniLM-L6-v2* model as published on Huggingface [32].

The word that was chosen to practice next is always the least well known word from the list, based on the BKT-estimates. Then, based on the estimate of student knowledge of the practice word, the level of this round is chosen as easy, medium or hard. The distractors used are based on the level of the round. The idea is that the difficulty increases when the distractors are more similar to the practice word. An example from the vocabulary list: if the practice word is *shirt*, similar distractors would be *t-shirt*, *blouse*, while less similar distractors could be *shoes*, *belt*. So, to choose distractors, the possibilities are divided into three categories according to their similarity to the practice word, and from the list of words in the correct difficulty level, the least well known words are chosen. The idea behind this is that using lesser known words as distractor leads to a smaller chance that the student gets the exercise correct by exclusion. The idea of basing the difficulty of the game on the BKT model was inspired by work done by Schodde et al. [33]

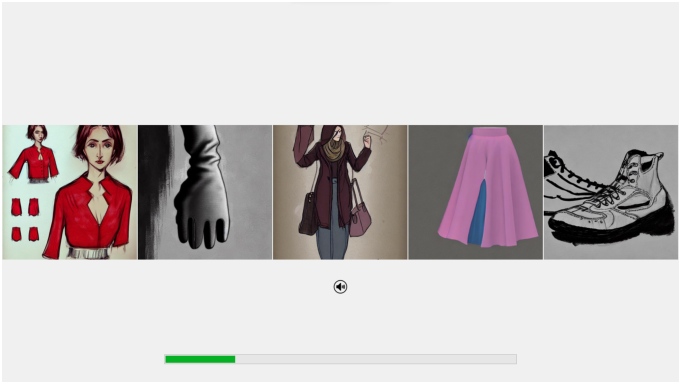
B. Description generation

As the words of the round have been chosen, the descriptions using these words can now be generated. This is done using the *gpt-3.5-turbo* model, which takes two prompts as input: a system prompt for initial instructions and a task prompt for the task at hand and possibly some examples.

As the descriptions presented to the students must be Spanish, and the text-to-image models requires English text, there is a need for translations. After some initial tests, evaluated by a native Spanish speaker, it was concluded that the best Spanish sentences were obtained by generating in Spanish directly using GPT-3.5, as this was trained on a dataset containing several languages [34]. Then, before being passed to the text-to-image model, they were translated to English by the Microsoft Azure translator.

When designing and testing the prompt, these criteria were found necessary: the sentences must be simple and short enough, they must fit within the theme, only the specified word should be used within the theme to avoid confusion, the specified word must appear in the sentence literally and the sentence must be in Spanish. Combining all of these criteria led to the following prompts:

- System prompt: *You generate the descriptions in textbooks for learning words. The descriptions are short, simple sentences with easy words for new students. The book has a theme, and within this theme, only the specified word is used. You speak only Spanish.*
- Task prompt: *Generate a short one sentence description in Spanish of a picture that contains a* {**word**}*. It is in a book for learning the vocabulary for* {**theme**}*. The sentence must contain the word* {**word**}*. Only provide the sentence itself.*



(a) The beginning of a round, after the robot has spoken the description.

(b) The second part of the round, after the user has tapped the fourth image

Fig. 2: Screenshots of a round of the game.

C. Image generation

The model used for image generation was Stable diffusion [28], an open source model that can be run locally. This model takes a prompt as input, which could be the description as generated before. The prompt can also be used to modify the style of the generated images. When no style is added, the result is often photorealistic, but not always completely correct, which can lead to strange and even creepy results. After some testing for different styles, the result was to add the style modifier *concept art* to the generated descriptions, as suggested in many prompt engineering guides and blogs (e.g., [35]). The resulting prompt is shown below.

- Image prompt: {**description**}, *concept art*

D. User interface

All the content as described before was presented to the students in the user interface shown in Figure 2. First, the description was said by the robot, after which the images appeared. After the user has tapped an image, the correct and possibly incorrect image and their corresponding words are shown.

V. Results

In this sections, the results of the user study will be discussed. The game was designed to be used with data that is generated in real time, but for the study, the data was generated beforehand. Because of this, it is possible to investigate the data quality, which also gives us some insight in the data that could be generated in real time. This is done in the first part of the section, then, the results of the user study are discussed.

A. Generated data

As the data can be generated in real time, the possible delay that it introduces for two types of GPU of the IDLab GPU-Lab is summarized in Table I, split up into the delay caused by word choice and description generation, image generation and display.

GPU	NVIDIA GeForce GTX 1080 Ti	Tesla V100-SXM3-32GB
Descriptions	5.45s	5.20s
Images	68.25s	15.10s
Display	5.38s	2.73s
Total	79.62s	23.35s

TABLE I: Time needed to generate the parts of a round of the game.

The generated Spanish sentences and their translations were evaluated by a native Spanish speaker. These were divided into three categories: correct, partially correct and incorrect, where the partially correct translations were grammatically correct but unusually phrased. Of the 100 generated sentences, 95 were evaluated as correct, 2 as partially correct and 3 as incorrect.

The translated generated sentences were evaluated by five people with a sufficient knowledge of the English language. They were evaluated on five criteria: were they appropriate within the theme, did they contain other words from within the theme, were they not unnecessarily long and were they simple enough, all answered with a score out of five. The resulting average scores and their standard deviation are given in Table II.

Criterion	Average	Standard deviation
1	4.99	0.1
2	4.748	0.713
3	4.612	0.654
4	4.428	0.808

TABLE II: Average score and standard deviation of the sentences on the four criteria.

The generated images were evaluated by categorizing them as portraying the description correctly, partially correct and incorrect. In the partially correct images, most of the sentence is correctly portrayed, but some details are incorrect or not portrayed clearly. Of the 100 generated images, 83 were evaluated to be correct, 7 were partially correct and 10 were incorrect.

B. User study

The user study was done on a class of 21 high school students majoring in Latin, 15 to 16 years old. The students were divided into two groups, one of which was a control group following the same lesson, but without the robot. In this group, the images were also shown on a tablet, but the spoken sentences were played by the tablet. The students were randomly assigned to these two groups, resulting in a robot group of 10 students and a tablet group of 11 students. First, the students took a written multiple choice test of the 20 vocabulary words. Then, the students played 60 rounds of the game. This amounted to around 10 minutes of practice. When a student indicated that the game was finished, they took the same test as before playing the game. Additionally, they filled in a small questionnaire about their experience during the study and some basic demographic data. As this was a group of Dutch speaking students, the instructions as well as the test and questionnaire were in Dutch.

This study was organised to answer the two research questions: was there a learning effect and was this different between the two groups. Figure 3 illustrates the results of both groups on both the pre- and post-test. Here, it is clear that there is a learning effect of playing the game. This was also evaluated using the two-sided Wilcoxon signed rank test on the pre- and post-test scores of all 21 students. This test shows that there is a significant difference between the pre- and post-test scores ($V = 5, p < 0.001, n = 21$). Looking at Figure 3, it is clear that this difference corresponds to an increase in the scores. It can be concluded that the use of the application led to a significant learning effect.

Looking at Figure 3, no clear difference can be seen between both groups. To evaluate if there is a statistically significant difference between the two groups, the Wilcoxon rank sum or Mann-Whitney U test was performed on the normalized learning gain of the students in both groups. This test showed no significant difference between the two groups ($W = 59.5, p = 0.7656, n_1 = 10, n_2 = 11$). From this result, it can be concluded that the presence of the Furhat robot when playing the proposed game has no significant effect on the learning effect of students.

VI. Conclusions

The tutoring system proposed in this master's dissertation enables students to practice vocabulary of a second language with a social robot. The game is visually grounded, using pictures to represent the vocabulary words, so no translation to the students' mother tongue is necessary. The content of the game can be AI-generated in real time, so no two rounds of the game have to be the same. The robot acts as a tireless tutor, so it can be used as support for teachers, providing the students with more one-

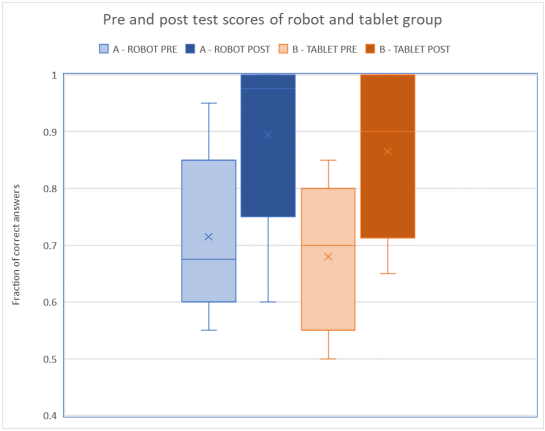


Fig. 3: Distribution of scores in pre- and post-test for students in the robot group (blue) and the tablet group (orange).

on-one practice. The implementation of this application was discussed, as well as the results of a user study performed with a group of high school students. The study showed a clear learning effect of the students after using the proposed tutoring system. A control group that practiced without the presence of the social robot showed a similar learning effect, and there was no statistically significant difference between the two groups. A possible explanation for this is that the game in its current form, did not resemble a natural social interaction closely enough for the positive effect of social robots that has been shown in other research to appear. In future work, an extension of the game could be developed where the student and the robot can have a visually grounded conversation in order to differentiate between the images. Still, the game proposed here shows great potential for the combination of social robots and generative AI to have a positive impact in the field of second language education.

References

[1] C. Leiter, R. Zhang, Y. Chen, J. Belouadi, D. Larionov, V. Fresen, and S. Eger, "Chatgpt: A meta-analysis after 2.5 months," *arXiv preprint arXiv:2302.13795*, 2023.

[2] T. Belpaeme, J. Kennedy, A. Ramachandran, B. Scassellati, and F. Tanaka, "Social robots for education: A review," *Science robotics*, vol. 3, no. 21, p. eaat5954, 2018.

[3] O. Mubin, C. J. Stevens, S. Shahid, A. Al Mahmud, and J.-J. Dong, "A review of the applicability of robots in education," *Journal of Technology in Education and Learning*, vol. 1, no. 209-0015, p. 13, 2013.

[4] N. Randall, "A survey of robot-assisted language learning (rall)," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 9, no. 1, pp. 1–36, 2019.

[5] S. Alharbi, M. Alrazgan, A. Alrashed, T. Alnomasi, R. Almojel, R. Alharbi, S. Alharbi, S. Alturki, F. Alshehri, and M. Almojil, "Automatic speech recognition: Systematic literature review," *IEEE Access*, vol. 9, pp. 131858–131876, 2021.

[6] J. K. Westlund and C. Breazeal, "The interplay of robot language level with children's language learning during storytelling," in *Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction extended abstracts*, pp. 65–66, 2015.

[7] M. Alemi, A. Meghdari, and M. Ghazisaedy, "Employing humanoid robots for teaching english language in iranian junior high-schools," *International Journal of Humanoid Robotics*, vol. 11, no. 03, p. 1450022, 2014.

[8] S. Al Moubayed, J. Beskow, G. Skantze, and B. Granström, "Furhat: a back-projected human-like robot head for multiparty human-machine interaction," in *Cognitive Behavioural Systems: COST 2102 International Training School, Dresden, Germany, February 21-26, 2011, Revised Selected Papers*, pp. 114–130, Springer, 2012.

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[10] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[11] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[12] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the association for computational linguistics*, vol. 5, pp. 135–146, 2017.

[13] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," *arXiv preprint arXiv:1712.09405*, 2017.

[14] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," *Advances in neural information processing systems*, vol. 13, 2000.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[16] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *et al.*, "Improving language understanding by generative pre-training," 2018.

[17] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[18] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[19] OpenAI, "Gpt-4 technical report," 2023.

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[21] J. Agnese, J. Herrera, H. Tao, and X. Zhu, "A survey and taxonomy of adversarial neural networks for text-to-image synthesis," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, no. 4, p. e1345, 2020.

[22] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans, *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in Neural Information Processing Systems*, vol. 35, pp. 36479–36494, 2022.

[23] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," 2022.

[24] `https://www.midjourney.com/home/`. Accessed on May 24, 2023.

[25] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Advances in neural information processing systems*, vol. 27, 2014.

[26] L. Barrault, O. Bojar, M. R. Costa-jussà, C. Federmann, M. Fishel, Y. Graham, B. Haddow, M. Huck, P. Koehn, S. Malmasi, C. Monz, M. Müller, S. Pal, M. Post, and M. Zampieri, "Findings of the 2019 conference on machine translation (WMT19)," in *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, (Florence, Italy), pp. 1–61, Association for Computational Linguistics, Aug. 2019.

[27] R. Harding, *English for Everyone: Level 1: Beginner, Course Book*. Dk Publishing, 2016.

[28] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022.

[29] `https://huggingface.co/runwayml/stable-diffusion-v1-5`. Accessed on May 24, 2023.

[30] A. T. Corbett and J. R. Anderson, "Knowledge tracing: Modeling the acquisition of procedural knowledge," *User modeling and user-adapted interaction*, vol. 4, pp. 253–278, 1994.

[31] Z. Pardos, Y. Bergner, D. Seaton, and D. Pritchard, "Adapting bayesian knowledge tracing to a massive open online course in edx," in *Educational Data Mining 2013*, Citeseer, 2013.

[32] `https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2`. Accessed on May 24, 2023.

[33] T. Schodde, K. Bergmann, and S. Kopp, "Adaptive robot language tutoring based on bayesian knowledge tracing and predictive decision-making," in *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 128–136, 2017.

[34] `https://github.com/openai/gpt-3/tree/master/dataset\_statistics`. Accessed on May 24, 2023.

[35] `https://prompthero.com/stable-diffusion-prompt-guide`. Accessed on May 24, 2023.

Contents

List of Figures

List of Tables

List of Acronyms

AI Artificial Intelligence.

BKT Bayesian Knowledge Tracing.

HRI Human-Robot Interaction.

LLM Large Language Model.

NLP Natural Language Processing.

PFA Performance Factor Analysis.

RALL Robot-Assisted Language Learning.

RNN Recurrent Neural Network.

Chapter 1

Introduction

Learning a second language in school often starts with a group of students being taught grammar and vocabulary by a teacher in front of the class. The more natural way to learn a language is through social interactions and conversation, but there is not enough funding and too few teachers for this kind of extensive one-on-one practice of the language.

As generative AI such as large language models gained impressive momentum since late 2022 [1], this master's dissertation can propose a novel solution to this problem through the use of social robots as second language tutor, powered by generative AI. These social robots could bring support to teachers and offer one-on-one practice to the students. The tutor will never get tired of practice and will always be in a good mood, which can not be expected of teachers. As language learning is a social experience, the social appearance of the robots and their embodiment make them well suited as tutors.

In this work, a tutoring system is proposed, where the content is fully generated by AI: large language models are used for the text while a generative text-to-image model provides the visual content. As this content can be generated in real time, this system allows for constant adaptation of the difficulty as well as the content to the needs of the student.

For the application to adapt itself to the needs of the student, it must be able to estimate these needs. To do this, a Bayesian student modeling technique is used. This enables the robot tutor to present the student with content of the right difficulty, offering easier practice when the student needs it and increasing it when the student is ready for a challenge.

Combining the social abilities of their embodiment with the generative AI that is becoming increasingly better into an adaptive system that provides the student with

what they need, might allow robot tutors to be the perfect support for teachers in the classroom.

1.1 Research questions

In this master's dissertation, the choice was made to implement second language tutoring through game play. This game play provides a way for students to practice foreign language vocabulary with a social robot tutor. This leads to two main research questions. The first one is on the learning effect of the game that is developed within this master's dissertation.

Research question 1: What are the learning outcomes when learning a second language together with a social robot driven by generative AI?

The second research question is specifically on the effect of the presence of the social robot, and how this affects the way students learn.

Research question 2: Does the presence of a social robot affect the vocabulary acquisition of students?

1.2 Outline

The next chapter, Chapter 2, brings a detailed explanation of the fields of social robots and of natural language processing, both indispensable for the development of the game. Then, Chapter 3 follows with an overview of the game in question, starting with the inspiration for the game and ending with the requirements for the development of this game. Then, in Chapter 4, the implementation of the game is discussed, including an overview of the models that were used. Then, in Chapter 5, the results are discussed, including the outcome of a user study performed to test this application. Finally, Chapter 6 ends this master's dissertation with the conclusion and discussion.

Chapter 2

Background

In this chapter, an overview of the background needed to use social robots in a language educational context will be given. This chapter is divided into two sections. The first will focus on social robots; what are they, what is their role in education and specifically language education? As will be discussed later, one of the ways that social robots communicate with humans is through natural language. Therefore, when working on social robots, the field of Natural Language Processing (NLP) is often involved. Therefore, this field is introduced in the second part of this chapter. This section is divided according to five tasks within the field of NLP that are relevant for this master's dissertation. These tasks are (1) word embeddings, (2) language modeling through large language models, (3) text-to-image or image generation from text, (4) translation and (5) visual dialog.

2.1 Human-Robot Interaction and Education

Social robots are robots that communicate with humans using the same communication channels as humans do when communicating with each other: verbally and non-verbally. The field that researches this communication between humans and robots, is called Human-Robot Interaction (HRI). The goal of these social robots is usually some form of cooperation, where the robot is often a partner to the human. This is in contrast to traditional robotics, where they are often seen as a kind of tool that can be used in situations that are not well suited for humans [2].

This section starts with a small history of language teaching methods. Then, a general oversight of some influential social robots will be given. After this, the use of technology in education, and then more specifically the use of social robots in education will be discussed, after which the more specific field of language education follows.

2.1.1 Language education

Throughout history, there have been many methods and ideas on how to teach a student a second language. Before looking at the role of social robots in language education, a limited history of foreign language education will be given, based on a paper by Nagy [3].

In the 19th century, the **Grammar Translation Method** was introduced. It was also called the classical method and focused mostly on morphology and syntax. The focus was mostly on written language such as literature, not on communication or conversation. Then, as an alternative, the **direct method** appeared. It focused much more on oral performance and communication, as a counterbalance to the focus on literature of the Grammar Translation Method. Practice happened through repetition and drills, while the use of translation and the students' mother tongue was prohibited. Then, a reaction to these two movements was the **reform movement**. The focus of this method was on the spoken language and familiarizing the students with the sounds of the foreign language. The students were immersed in the foreign language, using small groups and native speakers, while evaluation was in the form of conversation and interviews.

In the 20th century, many more methods appeared. The **reading approach** focused on recognition, with grammar only being taught for reading comprehension. Translation was used in this method, with discussion of the material in the students' first language. The **Audio-lingual** method banned the mother tongue and focused on oral skills and habit formation, with vocabulary and grammar being taught in context. The **audio-visual method** introduced linguistic varieties and started lessons through visual presentations. The **oral-situational approach** introduced new words or grammar organised around situations, with a focus on oral speech before written text. There was little to no use of the mother tongue, and drills and repetition were used for practice. The **cognitive approach** stated that language learning is rule acquisition, not habit formation and making errors is inevitable. It taught vocabulary in context with little use of the students' mother tongue. **Krashen's natural approach** believed that to learn a language, listening comes first and the rest will follow, students must be exposed to meaningful input to learn, and the aim of language is communication, so teaching grammar directly is useless. The method of **total physical response** states that understanding precedes speaking, so in lessons, the teacher should give commands to follow. When the learners are ready to speak, they can give each other commands as well. It also believed that stress should be reduced in class. In the **silent way**, the teacher is silent most of the time, except

for pronunciation. It is a structural method that uses building blocks and it focuses on self-correction and learner autonomy. In **Community language learning**, there is a focus on group work, where the class group is in a circle or U-shape. The students define what and when to learn, according to their needs. In **suggestology**, there is a focus on relaxation before class, e.g., through yoga. There is no correction of the students' mistakes, and the method uses dialogues, dramatized texts, songs and games, without use of the students' mother tongue. Then, **communicative language teaching** quickly became one of the most influential language teaching methods. It is an umbrella term for many methods, but the aim is to develop the learners' communicative competence, first through imitation, then free production. In this method, meaning is more important than form or structure and grammar is taught only when necessary. Lessons in this method often consist of group or pair work and are often interactive.

In the 21st century, experts mostly moved away from the idea that there is one perfect method. It is believed that the language teaching method should be based on the context, the student and the teacher. Forms of grammar and vocabulary are still presented, the use of the mother tongue and translation is reintroduced, and practice is done through repetition, drilling and often in a task-based way as well. The idea here was to correct the deficiencies of communicative language teaching, while still applying the good techniques.

2.1.2 Social robots

One of the first social robots ever designed was Kismet [4], developed at the Massachusetts Institute of Technology (MIT). Kismet consisted of a head and neck, with the possibility to move its eyes, eyebrows, lips and neck, as shown in Figure 2.1. It had very limited control software, which was used to extract some basic social signals such as sound amplitude and emotions based on prosody, and visual information such as motion and people appearing before it. The combination of this limited hardware and software still worked surprisingly well to create the illusion of a social presence. [5]

In 2006, the Nao robot was first sold. It was designed by Aldebaran Robotics, and is a small (58cm tall) humanoid robot, as shown in Figure 2.2. Due to its small size, affordability, robustness and easy-to-use software, it is one of the most popular humanoid robots in the world and it is used very often in HRI research[7].

In 2014, SoftBank Robotics, previously Aldebaran Robotics, sold the first Pepper robot. It is a 1.2m tall wheeled humanoid robot, as pictured in Figure 2.3. Pepper was

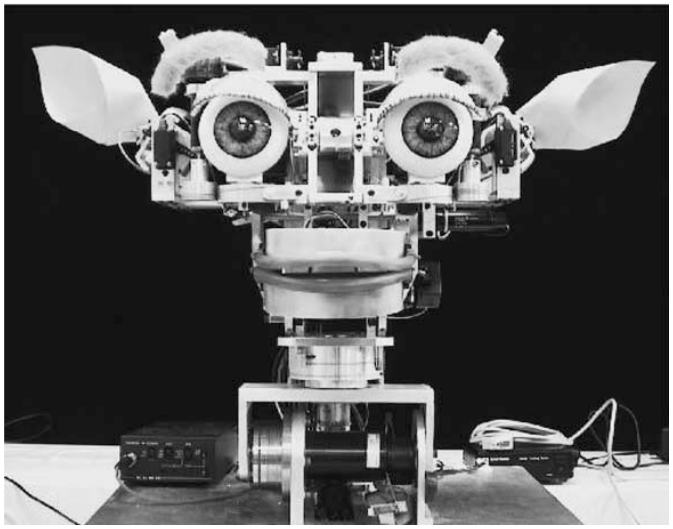


Figure 2.1: Kismet, one of the earliest examples of a social robot

Source: Adapted from [6]

Figure 2.2: The Nao robot, a small, humanoid robot often used in HRI research.

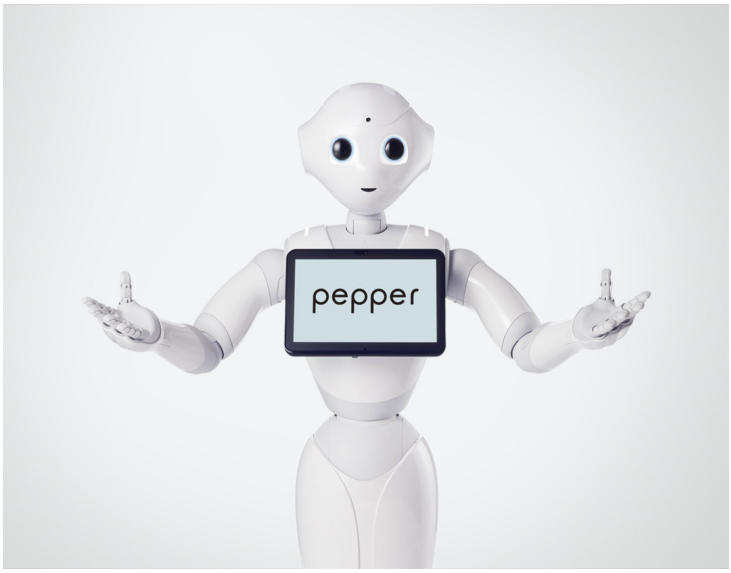


Figure 2.3: Pepper: a 1.2m tall humanoid robot, designed to have a pleasant appearance.

first marketed as a B2B robot that could lighten the load in businesses while also attracting customers, but later it was also offered as a B2C product. Pepper has three omnidirectional wheels and 17 joints, allowing expressive body language and smooth movements. It was designed to have a pleasant appearance, with a focus on safety, affordability, interactivity and good autonomy. Pepper has use cases in home as well as public environments, and research has been done in many application areas, such as elder care and education [8].

When using robots in applications where social interaction is key, there is a need for a human face that looks and moves realistically. In social interactions, human lips carry much information on speech and intonation, while eyes and their gaze show much about the attention and affect of a person. As technology advances, the creation of human faces has mainly taken two paths: digital animated faces versus physical, mechanical heads. Much progress has been made in the design of digital faces, but this is not easily transferred to robotics, where mechanical movements directed by digital signals rarely result in smooth, human-like movements. This effect is often referred to as the uncanny valley [9].

In 2011, Furhat Robotics introduced the Furhat: the first commercially available robot with a back-projected face, as seen in Figure 2.4. This is an interesting combination of a digital animated face, projected on a physical, three dimensional robot. The Furhat consists of a head with a neck, and can perform some basic natural movements such as

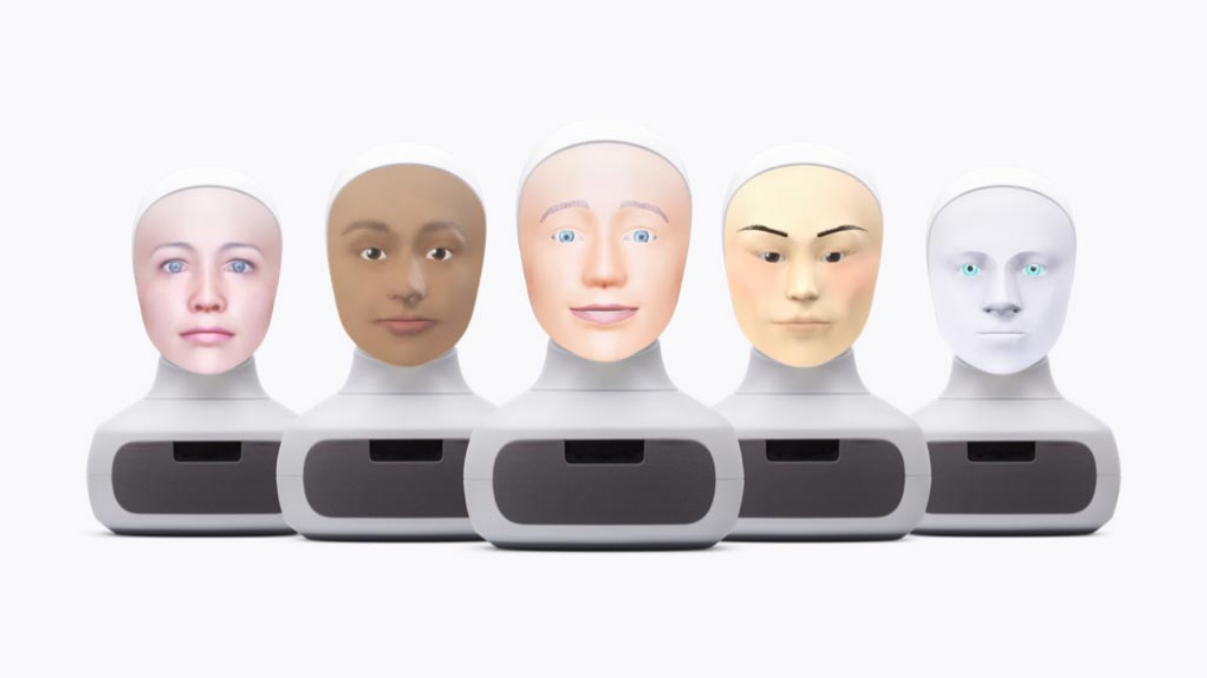


Figure 2.4: Furhat: a robotic head with a back-projected face.

Source: Adapted from [10]

nodding, shaking its head and raising its eyebrows. An important advantage of the back-projected face, is that it is very customizable: the gender, skin color, size of its features and amount of makeup can all easily be adjusted, allowing use in various circumstances [9].

Social robots are expected to have many applications. Many of these are still in the research phase, with some first commercial applications appearing. Most notable application areas for social robots are the service industry, education, entertainment, healthcare and robots as personal assistants. **Service** robots are often used to attract attention as well as to support human workers in the service industry. In **education**, social robots have proven to be useful in assisting. The robot can take on various roles and can be applied in many educational fields. An example of how social robots are used in education is the L2TOR project, where an embodied digital learning environment was developed, in which a social robot tutor helps children learn a second language [11]. Robots in **entertainment** generally either belong to exhibitions or the performing arts, or they are some form of toy or pet robot for children or adults. A well-known example of a robot for entertainment is Ameca, by Engineered Arts, shown in Figure 2.5. It is made

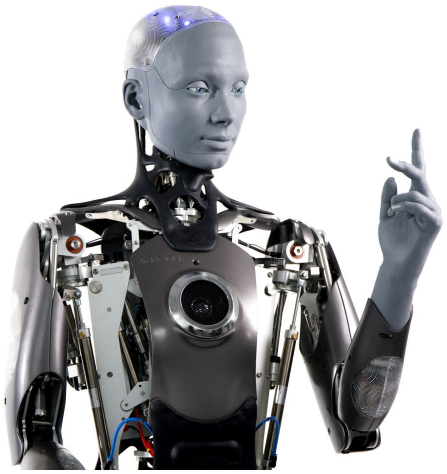


Figure 2.5: Ameca, an entertainment robot by Engineered Arts.

to move in a smooth, lifelike way and has impressive facial expressions. It can be bought or rented for use as an attraction at events. [12]

In **healthcare**, social robots can be used to offer support, education or diversion, in order to improve healthcare and therapy outcomes. Related are socially assistive robots, that are often targeted to older adults. An example of this is Paro, which is a baby seal robot that is used in research on people with dementia. It is an interactive social robot used for therapeutic purposes. Paro is pictured in Figure 2.6.

Figure 2.6: Paro, the therapeutic baby seal robot.

Social robots as **personal assistants** are a logical extension to smart-home assistants such as Google Assistant or Amazon Alexa. When social robots are used in this way, they add a social presence to home assistants, while often not adding other physical functionality. Sometimes, they are motorized, enabling them to move around with the user. [5]

The most important challenges that social robots face in the coming years are mostly not hardware related, but are about the software that runs on them. Social robots create the expectation that they can communicate and move around like a human can, but their autonomous control systems and their artificial intelligence are not capable of this yet. Challenges in hardware exist as well, such as speed of movement and battery life [5]. What the effect will be of the great advancements in the field of natural language processing, such as ChatGPT and it's massive amounts of media attention [1] on the progress in the field of social robots will be discovered in the future. But, before these technologies based on written text can be easily integrated into social robots, advancements in speech-to-text technology are necessary [13].

2.1.3 Technology in education

As schools often cope with limited funding, shortages of teachers and growing classrooms, technology can offer relief by supporting teachers and providing one-on-one tutoring and adaptive exercises.

This is often in the form of Intelligent Tutoring Systems (ITS). These are computer systems that guide learners through exercises, while providing hints, feedback or explanations where necessary, personalized to the needs of the student. They are often based on artificial intelligence and require expert knowledge to make. The use of these ITS has been researched extensively and the effectiveness has been shown [14].

An advantage of these ITS and how artificial intelligence is incorporated in them, is that these systems can adapt the content and difficulty to each individual student. In the field of educational psychology, Vygotsky says that a student learns best while in the zone of proximal development. This is where exercises are just too difficult for a student to solve on their own, but it is achievable with guidance [15].

To achieve this, the ITS must have an estimation of the skill of each student. How this can be achieved, is researched in the field of student modeling or learner modeling. Here, the goal is to use a model that can estimate the skill and knowledge of a student. Two well known models often used to achieve this are Bayesian Knowledge Tracing (BKT) and Performance Factor Analysis (PFA).

Bayesian knowledge tracing models take a probabilistic approach to student modeling, and are a special case of hidden Markov models. Here, student knowledge is a latent

variable, and the observable process is the correctness of the exercises done by the student. Learning is thus modelled by a discrete transition from the unknown to the known state. The model provides an estimate of the probability that a student will solve an exercise on each skill correctly. The actual outcome of the exercise is then used as input to the update equations, which update the internal parameters of the model. Like this, the model uses the observable process of the answer of the student to model the latent variables of the actual skills of the student. A detailed explanation of the internal workings of the BKT model are given in Chapter 4. Performance factor analysis is a model based on a logistic function, where previous data about the performance of the student is used to compute a skill estimate. This estimate is then transformed using a logistic function into an estimated probability that the student will answer correctly [16].

BKT models can be used to model many different kind of skills. Schodde et al. used it to trace the student knowledge and choose the next exercise in a game setting for learning foreign language words with a robot tutor [17]. Pardos et al. used a BKT-based student model in a massive open online course on circuit design [18]. Kasurinen et al. used a BKT model for estimating student knowledge in a programming course [19]. Adaptations to the classical BKT system are often made. For example, Yudelson et al. researched the effect of using student-specific parameters [20], while Qiu et al. investigated the effect of time passing on student knowledge by modeling the probability that a student forgets learned knowledge [21].

In more recent years, deep learning has entered the field of student modeling. This lead to recurrent neural networks being introduced for student modeling, which is called Deep Knowledge Tracing, and was found to outperform previously used models. [22]

Using these student modeling techniques, ITS can provide personalized guidance to students in order to optimize their learning, in school settings where resources are often too limited for personalized education for every student.

2.1.4 Social robots in education

The use of social robots in education builds on the same concept as an ITS, with one important difference: the intelligent tutor is an embodied agent. This leads to a much higher development cost, and must therefore be justified. The advantages of social robots over other forms of technology are partially due to their embodied nature, which enables them to interact with the physical world. Next to this, the embodied nature of the

technology leads to interactions being perceived as social, which can also be beneficial for learning. Lastly, research shows that there are increased learning gains when students are interacting with embodied social robots over virtual agents. The learning effects from the use of social robots can be divided into affective and cognitive outcomes. Affective outcomes of the robot on students are mostly related to the emotional state of the student, with benefits such as improved motivation and reduced anxiety. The cognitive outcomes correspond to increased performance and higher test scores. [23]

When using social robots in education, there are multiple roles the robot can take on. They can be used as a teacher or tutor in addition to the teacher, where they are posed as an expert in relation to the student. The robots can also take on the role of a peer. Then, they are often perceived as less intimidating, as they are learning together with the student. Lastly, the robot can also be presented as a novice, leading to the student taking on the role of instructor. One of the advantages of the robot not being presented as an expert, is that there is more understanding for possible mistakes, as these can also be expected from peers and novices. [24, 23]

An important effect within HRI, that must also be considered when looking at the efficacy of robots in education, is the novelty effect. In education, this effect describes the tendency for students to have better (cognitive or affective) results when first using new technology such as a robot as tutor, due to interest and excitement for this new technology. These improved results are not expected to necessarily last long, so more long-lasting research is needed on this. [25]

There are some obvious challenges in the use of social robots over non-embodied ITS: the additional hardware of the robot has a higher cost, it might need regular maintenance, there is a higher threshold for teachers to get used to this technology and distribution and installation of these embodied tutors is not obvious. Next to this, social robots in education sometimes present themselves as very human-like, which leads to very high expectations of the ability of the robot.

Another challenge encountered when using social robots in education is how robot tutors decide what action to take. When a student is struggling, should the robot immediately come to the rescue, or should the student be challenged to try harder? There is no definitive answer to this kind of questions for human tutors, let alone social robots. A last challenge within the field of social robots for education is how students perceive the

robot. It is important for the students to feel comfortable around the robot. This leads to considerations such as using a smaller robot with young children, as not to intimidate them. [23]
The social appearance of the robot also leads to specific expectations that the robot will understand speech and social signal without issues. Both of these have undergone strong improvements over the last years, but these technologies are still imperfect. [24]. Especially the performance of speech recognition when working with children is still insufficient [26, 13]. A lot of research on the effect of social robots has been done, with many promising results, while also uncovering many technical challenges. Up until now, most studies used the robot tutors in restricted scenarios. Here, the affective and cognitive outcomes are generally positive [23].

2.1.5 Social robots in language education

Learning a language is a very important part of growing up. Research shows that early language skills are a predictor of later academic success, while bilingualism has been linked to many positive outcomes such as higher IQ scores and better economic opportunities and a protective effect against dementia [27, 28, 29, 30, 31]. Next to this, learning a new language is a useful skill in a more globalized world [32].

Learning a language is an inherently social act [33]. Children learn their first language from their parents and the people around them, and learning a second language opens the door to communication with a larger, more varied group of people. Just as in general education, using robots in language education leads to many design choices such as the form, role, amount of personalization and abilities of the robot. Research suggests that social robots are able to help students with vocabulary acquisition, while more research is needed to compare social robots with other technologies in teaching other aspects of language. It has also been demonstrated that robots aid learning when used next to a human teacher. Lastly, research shows that robots have a positive effect on the learners' affective state. [26]

Additionally, robots have the advantage of being more aware of their surroundings than virtual tutors, because their embodiment enables the addition of sensors. They interact with students the way people interact with each other and they can be customized to the specific needs of the student. Next to this, learning a language involves a lot of repetition. This may lead to fatigue and boredom in human tutors or teachers, while a robot tutor

does not suffer from this. [34]

In conclusion, results are promising, but more research and advancements in the related technology are needed for the optimal use of social robots in language education. Most important here is automatic speech recognition: its performance is not always good enough for fluent conversations in perfect conditions. With noisy backgrounds, multiple people in a conversation and speech by people who do not yet master the language, speech recognition performance is an important bottleneck for social robots in education [13]. Additionally, using robots for long-term interactions must be researched further. The high cost of social robots is also a barrier for the frequent use of social robots in a classroom. Teachers and users who do not have a technical background must also be able to comfortably use the robots in their environment, which may also pose a challenge. [26]

2.2 Natural Language Processing

To use social robots for second language tutoring, there is a need for some form of language processing in the robot. This field is called natural language processing (NLP). In this section, some relevant NLP tasks and models will be discussed. First, the general concept of word embeddings is introduced. Then, we will continue with large language models (LLMs), which can be used for many purposes. Then, the multimodal task of image generation or text-to-image will be discussed, after which we will go over another important task in the field of NLP: translation. Finally, the field of visual dialog, and its tasks and models will be discussed.

2.2.1 Word embeddings

When performing any task within the field of NLP, it is necessary to represent words or their meaning in a way that can be used by a computer. In the early days of NLP, this was done by using the index of a certain word within the vocabulary used. The problem with this approach is that there is no notion of similarity between words; synonyms were not represented in a more similar way than words that have no relation. Because of this, the idea of transforming words to an embedding space arose. Here, words are represented by a vector in a multidimensional space, with the assumption that vectors that are closer together, also have a more similar meaning and words with a similar difference vector also have a similar relation [35].

Generally, this transformation is done by some form of neural network. The model

is trained by letting it perform a task where the meaning of the words is important for success. One task that is often used is the skip-gram objective, where the goal is to predict the words that surround a given word. Another one is the continuous bag-of-words (CBOW) objective, where a certain word is predicted, given the surrounding words. Both of these were used at Google by Mikolov et al. to develop Word2Vec [36, 35], a well-known technique to train neural networks for the use of word embeddings. Other well known techniques are GloVe [37], where the co-occurrence frequency of words is also taken into account and FastText [38, 39], where parts of words are also considered.

2.2.2 Large language models

One of the most fundamental tasks in NLP is that of language modeling: predicting whether a given sequence of words is likely or not, or, in a more probabilistic view, finding the joint probability function over sequences of words. This task is very difficult to solve due to the so-called curse of dimensionality: the options grow exponentially with the length of the sentence, where the base is the size of the vocabulary, usually a very large number. This task was often attempted using statistical models, but is quickly infeasible due to the aforementioned dimensionality problem. The first well known attempt to tackle this problem using neural networks was by Bengio et al [40]. Here, the model jointly learned distributed representations of each word, or word embeddings, as well as the probability function for word sequences.

Another type of model strongly linked to language modeling is sequence-to-sequence models. These are models that have a sequence, often a natural language sequence, as input as well as output, e.g., recurrent neural networks for translation. These sequence-to-sequence models often use an encoder and decoder that consist of complex recurrent or convolutional neural networks, possibly linked by an attention mechanism. Attention mechanisms were designed to improve upon models that consist of an encoder and decoder. A strong limitation in these models is the following: the encoder transforms the sequence into a fixed length vector, that is then used by the decoder to form a new sequence. This means that the encoder must compress all information within the sentence into a fixed-length vector, regardless of the length of the sentence. Research has shown that the performance of this encoder-decoder model deteriorates with the length of the sentence, especially for sentences longer than the ones in the training corpus. An attention mechanism allows the model to align itself: the decoder is able to search the source sentence

for the positions with the most relevant info and using this to generate the next word [41].

Then, in 2017, Vaswani et al. introduced the transformer [42]. This neural network fully replaces the recurrent or convolutional layers with self-attention layers, resulting in better results with significantly lower training costs. With over 70,000 citations as reported on Google Scholar, this paper transformed the world of NLP.

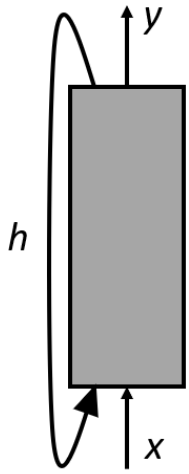


Figure 2.7: Schematic representation of a recurrent neural network.

To understand the reason for the success of the transformer, the limitations of recurrent neural networks (RNN) must first be understood. RNNs are mostly used with sequences of data, where the parts of these sequences are the input for the same node in a sequential way. This node outputs a state, which represents the data that has passed through the RNN until then. This state is then given as additional input when the next part of the sequence is processed. This way, the full sequence can be processed as a whole. Figure 2.7 gives a schematic representation of this, where x is the input, y is the output and h the state. This leads to a strong limitation: the output of the first part of the sequence is needed as input for the next, making parallellization impossible. This leads to strong limits on the amount of data that can be processed in a reasonable amount of time, regardless of the available computing power.

With the emergence of the transformer, this limitation could be avoided. The transformer was designed for the processing of sequential data as well, but its architecture enables it to process all these sequential data in parallel. This is due to the use of self-

attention, where parts of the sequence can influence each other in a parallel matter. Figure 2.8 shows the original architecture of the transformer as proposed in 2017.

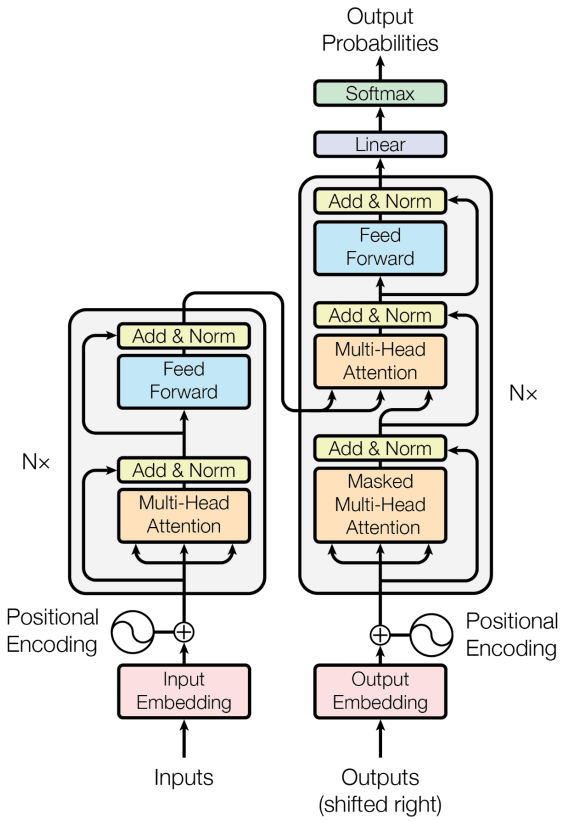


Figure 2.8: The original architecture of the transformer.

Source: Adapted from [42]

The existence of the transformer suddenly made it possible to train neural networks on larger amounts of data, as the bottleneck of sequential processing was removed, enabling the emergence of Large Language Models (LMM). These often have billions or more parameters and are trained on large amounts of unsupervised data. They are often trained using a certain language modeling task, after which they can be used for a variety of other NLP tasks, with little to no extra training. A few influential models will be discussed here.

In 2018, OpenAI published GPT: a family of large language models called Generative Pre-trained Transformers. Here, the goal is to learn universal representations that can be easily transferred to other tasks with little adaptation. This is done using a combination of unsupervised pre-training and supervised fine-tuning. Quickly, its successors arose,

with GPT-2 in 2019 , GPT-3 in 2020 and GPT-4 in 2023, with each model resulting in a significantly better performance than the previous, partially due to a strongly increased amount of parameters: GPT-1 started out with 117M parameters, GPT-2 has 1.5B parameters and GPT-3 has 175B parameters. The amount of parameters of GPT-4 has not been published. These generative models with large amounts of parameters, trained on large corpora opened the door for completing tasks with very little or no training on the specific task. Few-shot learning arose, where it is possible to provide the LLM with few examples of the task at hand, after which the model can complete further instances of this task. Even more extreme is zero-shot learning, where the LLM is able to complete unseen NLP tasks, due to the natural language understanding gained in its general training. [43, 44, 45, 46]

In December 2022, OpenAI released ChatGPT [47]. It is a chatbot built on top of GPT-3.5 (an improved version of GPT-3) and later GPT-4. It was fine-tuned using a mixture of reinforcement learning and supervised learning, called Reinforcement Learning from Human Feedback [48]. It was made available to the public and has since gotten large amounts of attention and has brought the topic of NLP into the public eye [1].

A second influential family of transformer models was introduced by Google researchers in 2018: Bidirectional Encoder Representations from Transformers (BERT) [49]. While BERT models are bidirectional, previous models used only context from one side of the target word. These models are trained using two unsupervised tasks. The first is a form of language modeling, where a percentage of the input words are masked and have to be predicted by the model. The second is next sentence prediction, where the model is given two sentences and must predict if these sentences follow each other, or the second is just a random sentence. From this paper, many more models within the BERT family arose.

2.2.3 Text-to-image

A text-to-image model is a form of multimodal NLP model. Multimodal refers to the fact that it relies on multiple modalities: it takes language as an input, but the output is visual. More specifically, text-to-image models take as input a natural language sequence, e.g. a description, and return an image with similar semantic meaning to the sequence. Most of the popular text-to-image models fall under the category of deep generative models, which combine classical generative models with neural networks and have as goal to estimate

the joint probability distribution of the data and its labels. [50]

As text-to-image models take a natural language sequence as input, part of their recent success can be attributed to the progress in large language models. The text input is first processed by a LLM, after which it is used in image generation. These models are trained on immense amounts of data found on the internet, often with the caption of the image as label. This leads to many ethical concerns regarding artists whose data has been used for training these models. They did not have the possibility to opt out from this, while the resulting technology might threaten their livelihoods. [51]

Some of the popular text-to-image models are diffusion models. As input, these models take images consisting of random noise. Then, guided by the processed natural language sequence, they iterate over the image many times. In each iteration, some noise is removed to bring the image semantically closer to the text input, until a detailed, sometimes even photo realistic image remains. An example of these diffusion text-to-image models is Imagen by Google [52].

In 2021, OpenAI published the first version of DALL-E, after which the second version, DALL-E 2 , followed in 2022 [53]. These models build on OpenAI's GPT systems, as discussed in Section 2.2.2. They use a transformer, which takes as input concatenated image and text tokens. This transformer is a decoder-only model, that autoregressively models its input [54]. Both of these models were made available to the public, with a limited amount of free use. Due to this and their performance, the models have gotten a considerable amount of media attention, both positive and negative. [51]

Other well known text-to-image models are Stable Diffusion, an open source diffusion model [55] and Midjourney, published in beta by the research lab Midjourney [56].

2.2.4 Translation

The core task of translation models is machine translation: the model must find the most probable sentence in the target language, given a sentence in the original language. In the beginning of machine translation, statistical models were used. These were built over many years using domain knowledge. Later, neural models were used, strongly improving the performance. The first occurrence of neural machine translation was by Sutskever et al. at Google [57]. Since the arrival of the transformer, as discussed in Section 2.2.2, these have dominated the field of machine translation [58].

Important in the field of machine translation is how to evaluate the models, as there is typically no one single correct translation. One proposed and widely used metric for the automatic evaluation of machine translation is BLUE: the BiLingual Evaluation understudy [59]. This metric compares the machine translation to one (or several) human translations. It gives a score based on n-gram precision, which is the occurrence of sequences of n words in both phrases. BLUE is one of many existing metrics for evaluating machine translation.

2.2.5 Visual dialog

Visual dialog is a task within the intersection of NLP and computer vision. This task requires a chatbot to hold a conversation in natural language about an image. Other tasks strongly related to visual dialog are (1) image captioning, where a description of an image must be generated, (2) visual question answering, where a model must generate an answer given an image and a question related to it and (3) visual question generation, where a model must generate questions, often to discriminate between images.

The formal definition of the task of visual dialog is that the model must generate an answer, given an image, a history of the dialog and a follow-up question. Das et al. presented a dataset for the task of visual dialog, called VisDial [60], while a dataset for visual question answering was presented by Antol et al. [61]

The task of visual dialog is especially useful in the field of social robots, as the often anthropomorphized body of the robot leads to the expectation that it can see and discuss what it sees. Apart from this, it can also be useful in aiding visually impaired people and in the analysis of large amounts of data.

In the same paper that the VisDial dataset was presented, Das et al. also presented a family of neural models for visual dialog. The encoders consists of (1) a late fusion model, that embeds image, history and question separately, after which they are fused, (2) a hierarchical recurrent encoder, using 2 RNNs, one on the dialog level and one for question-answer pairs and (3) a memory network that treats previous question-answer pairs as facts to store in memory. Two decoders were also proposed, a generative and a discriminative model [60].

In 2021, Kim et al. [62], presented a transformer that is pretrained on vision and

language data, in order to reach a better performance in downstream tasks. This model avoids convolutional layers and region supervision, in order to strongly improve its speed and come to a simpler model. It has been demonstrated on the tasks of visual question answering, as introduced above, and the task of Natural Language for Visual Reasoning [63], where the correctness of a statement in relation to an image must be assessed.

A lot of research is being done in the field of visual dialog and the intersection of NLP and computer vision in general. Many interesting ideas and promising models are surfacing, to overcome the remaining challenges. One of the important challenges that remains is that the available datasets for visual dialog are still limited, compared to the amount of data often used in e.g., LLMs [62]. Another is the difficulty in evaluating the performance of these visual dialog models [64]. Overcoming these challenges and arriving at high quality visual dialog systems are important steps for a multimodal social robot that can interact with people in a natural way.

Chapter 3

Visual Game Playing

In this master's dissertation, the choice was made to implement second language tutoring through game play. This game play provides a way for students to practice foreign language vocabulary with a social robot tutor, driven by generative AI. Specifically, the choice for visual game playing was made: the student plays a game with a social robot, where language is learned in a visually grounded way. In this chapter, the inspiration for this game will first be discussed. Then, the final shape of the game will be explained. Finally, we will go over the requirements that were needed to make the game, which will then be discussed in greater detail in Chapter 4.

3.1 Inspiration

The original inspiration for the game came from a paper by Das et al [65]. This paper discusses how two chatbots are trained using deep reinforcement learning to play a visually grounded game. The game consists of the chatbots communicating with visual questions and answers in order to reach a common goal of agreeing on which picture they are discussing. One chatbot knows the picture, the other one has a set of (similar) pictures to choose from. Then, the two chatbots communicate using natural language until reaching agreement.

From this paper, the idea sprang to implement this sort of game, where one player is a social robot and the other is a student, learning the language in which they communicate. In the game described here, the social robot and student can take on both roles: the one describing the picture and the one guessing it. This means that the robot takes on the role of a peer.

Benefits

The game as described above could provide many benefits to a student learning a language. By the choice of images, it could be used to learn vocabulary around a specific theme in a natural way. As the game is visually grounded, there is no need to provide translations to the mother tongue of the student. As the use of the student's mother tongue is a topic under discussion [66, 67], it can be seen as a benefit that this can be omitted with a game of this form.

As discussed in Chapter 2, language learning is inherently social [26]. Therefore, using an embodied agent such as a social robot can provide benefits over using a non-embodied intelligent tutor. The use of a social robot has benefits compared to a human teacher due to the robot tutor never becoming tired, bored or irritable. Due to this, the robot could play this game for as long as wanted by the student, and with as many students consecutively as needed.

Problems

There are some factors that make the aforementioned game unrealistic to realize within a master's dissertation. Firstly, current technology might still be too limited. The use of speech recognition is on the rise as the technology progresses, but even in perfect circumstances, mistakes often occur. When using speech recognition technology with atypical populations, its performance is still far from perfect. These atypical populations include children, people with a different mother tongue, a strong accent or students still in the progress of learning the language, all of which can be applicable to this situation. To facilitate the use of speech recognition in these circumstances, it could be possible to adapt existing speech recognition on specific groups, but this is still subject to more research [26, 13].

Next to this, the implementation of this game would require well performing visual question generation and visual question answering. The combination of these strong requirements and the technological limitations, lead to the full implementation of this being out-of-scope for a master's dissertation.

Simplified version of game

Within this master's dissertation, the choice for a simplified version of the game described above was made. As will be described in more detail in the following section, the game consists of a social robot presenting the user with a set of images, as well as a description of one of the images in the goal language. Then, the student has to indicate which image fits the description best. This simplified game eliminates the use of speech recognition, while still keeping the benefits of being visually grounded. A limitation of this game is that the student only practices the goal language in a passive way.

In this simplified version of the game, the robot takes on the role of a tutor, as only the robot speaks sentences in the foreign language, and the student only listens to these in order to guess the correct image.

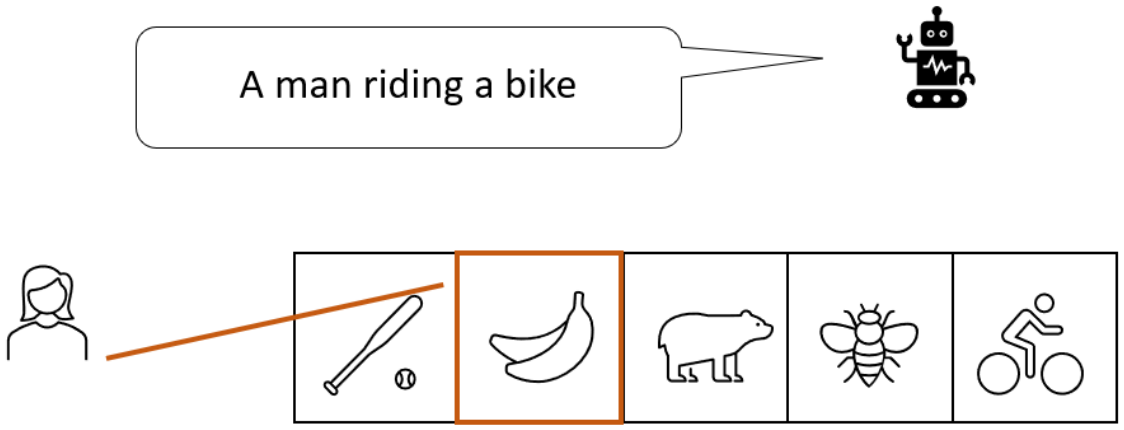
Within this simpler game, a stronger focus was put on the contents of the game and how they could be obtained. As described in the coming sections as well as Chapter 4, the content of the game is made using generative models, opening up the possibility of completely novel content, personalized to the strengths and needs of the student.

The aim of this game is to help students learn vocabulary through game playing with a social robot, in a visually grounded way, so there is no need for translations. The social robot is a tireless tutor that offers personalized content at the level of the student.

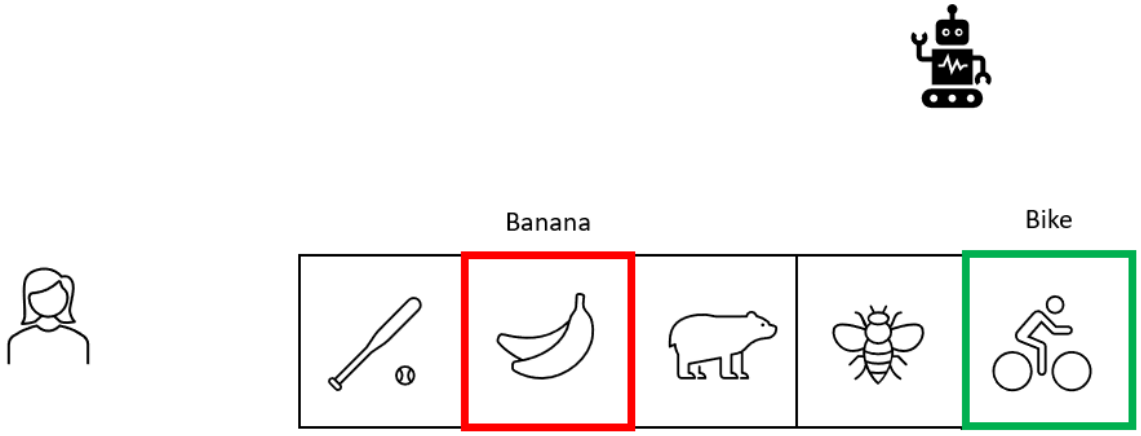
3.2 Game setup

This section will describe the form of the game. The basis of the game is a vocabulary list, that contains all of the words that the student is practicing. The game consists of multiple rounds, in each of which one of the vocabulary words is practiced. The social robot presents the student playing a game with a sentence in the goal language. This sentence contains the word that is being practiced. Then, a set of five images is shown on a (preferably touch) screen. One of these images corresponds to the sentence that was said by the robot and thus contains the word that is being practiced. The other four images are considered distractors. They depict scenes that contain other words from the vocabulary list. Which words are chosen for the distractors will be discussed in the following section. Then, the student has to indicate which image corresponds best to the description spoken by the robot. If the student chooses the correct image, a green

border appears around this image and the vocabulary word in the goal language is shown above it. If the student chose the wrong image, a red border appears around it, while a green border appears around the correct image. Above both images, the corresponding vocabulary word is shown in the goal language. This way, the student gets a chance to learn both words.



(a) A description of the image is spoken by the robot, the user indicates which one they think corresponds best



(b) The correct and incorrect answers are indicated by green and red respectively, while their corresponding vocabulary word is shown

Figure 3.1: Schematic representation of the form of the game.

Figure 3.1 shows a schematic representation of the game, with English as the goal language. The robot speaks a description, "A man riding a bike", which corresponds to the fifth image. The user wrongly indicates the second image. Then, the right and wrong image are shown using a respectively green and red border, while the words corresponding

to these images are shown above in English, the goal language.

3.3 Adjustments of difficulty

In order for the game to stay challenging, the difficulty should match the student's knowledge. For this to be possible, student modelling is used. The implementation details can be found in Chapter 4. What is required for adjusting the difficulty of the game, is an estimate of how good the student knows every one of the vocabulary words. This estimate should then be adjusted after each exercise. When a student answers an exercise correctly, the student model is updated positively for this one word. When a student gets an exercise wrong, it can be assumed the students did not know the word that was being practiced, as well as the word that was wrongly indicated. To clarify, using the example of Figure 3.1, the student is here assumed to not know the words "banana" and "bike". Then, the model is updated negatively for these two words. What this entails exactly will be discussed in Chapter 4.

As the student model provides estimates of the student's skill, the game should be adjusted according to this. This happens with the choice of distractors. When a word is not well known, very different distractors from the word that is being practiced are used. As the estimate of the student's knowledge of this word increases, the distractors become more similar to the word that is being practiced. The student model expresses its estimate of the user's skill as a probability that the next exercise involving this word will be answered correctly. These probabilities are divided into three categories: easy, medium and hard. This corresponds to how difficult the exercise involving this word should be. Then, the other vocabulary words are ordered by how similar they are to the word that is being practiced. These are divided into the same three categories. Then, the distractors are randomly chosen from the category that corresponds to the user's level.

3.4 Requirements

To allow a student to play the game described above, there are some requirements. First, a vocabulary list of words that the student should learn should be provided, for example by the teacher. The vocabulary list used here was taken from an English course book [68]. Then, descriptions containing these words as well as images fitting to these descriptions

are needed. Both of these are obtained through generative AI models. If the goal language is not English, the descriptions should be translated, as they were generated in the goal language. This is a result of the used text-to-image model being intended for usage in English, as indicated on their Huggingface page [69, 70]. In order to adjust the difficulty levels of the game, a student model is needed. Here, because of its intuitive parameters and simple implementation, a Bayesian model is used. Lastly, to choose which distractors are used, a way to model word and sentence similarity is necessary. This is done using a transformation of the words and sentences to an embedding space. All of the models mentioned above are described in detail in Chapter 4.

Chapter 4

Implementation

In this chapter, a more detailed discussion of the implementation of the game will be given. First up is the way the words within a round of the game are chosen. This pertains to the distractors as well as the word that is being practiced. Then, a closer look is taken at the generation of the descriptions, with a focus on the prompts used and the similarity checks. Then, the image generation, with style choices and the handling of generation errors is discussed. Finally, an overview of the implementation choices of the user interface is given. In each section, if relevant, the used models are also discussed in detail.

All of this was implemented in Python. The Furhat robot is usually programmed using Kotlin, but a Python API is also available. Here, the Python API was chosen as this was easiest to integrate with the rest of the code and the models.

4.1 Choice of words

In this section, the choice of the practice word as well as the distractors is discussed. Before diving into the implementation choices that were made here, two models are described in detail. First, the the BKT model used for student modeling is discussed, as this influences the choice of the practice word as well as the distractors. Then, the model used to calculate sentence and word similarity is discussed, as this is used for adjusting the difficulty while choosing distractor words. Finally, the implementation is discussed, including the usage of these models.

4.1.1 Bayesian Knowledge Tracing

In this section, the student model used will be discussed in more detail. As discussed in Chapter 3, this model is needed to adjust the difficulty based on the skills of the student.

As the game is based on a vocabulary list, the words in this list will be considered the skills the students is learning. Therefore, the student model must continuously keep an estimation of the student's knowledge of each word in the list, which is then updated with each round of the game. Thus, each of these rounds will be considered as one exercise.

To formalize, the student model must provide n probabilities, with n the size of the vocabulary list. These probabilities indicate the estimate of the student model that the student will answer the next exercise correctly.

Parameters

The model used here is a Bayesian Knowledge Tracing model [71]. It is a special case of a hidden Markov model, where the latent variables are the student's knowledge of each skill and the observable data are the correctness of the exercises. These latent variables are binary variables: is this skill mastered by the student or not? The observations are binary variables as well: did the student solve this exercise correctly or not? Internally, the model keeps the variables θ_i, which represents the probability that the student knows word i. The model also keeps some global parameters, that are the same for all words:

1. P_i: the probability that a skill is already learned (*initial*)
2. P_l: the probability that the student will learn on next practice opportunity (*learn*)
3. P_s: the probability that the student will make a mistake despite knowing (*slip-up*)
4. P_g: the probability that the student will answer correctly despite not knowing (*guess*)

It is also possible to choose the initial probability as separate values per skill, e.g., based on prior knowledge of the student skills. This was not done here, as this extension would introduce a non-negligible delay in the execution of the user study.

Update equations

First, the variables θ are set to the initial probability P_i. Then, as the student completes an exercise on skill i, these variables are updated. This update happens, based on the correctness of the exercise c.

$$\text{if } c = 1 : \theta_i' = \frac{\theta_i(1 - P_s)}{\theta_i(1 - P_s) + (1 - \theta_i)P_g} \tag{4.1}$$

$$\text{if } c = 0 : \theta_i' = \frac{\theta_i P_s}{\theta_i P_s + (1 - \theta_i)(1 - P_g)} \tag{4.2}$$

The update equations are shown in Equations 4.1 and 4.2, while the structure of the BKT model is graphically shown in Figure 4.1 [16]. The update equations are easily intuitively understood when looking at this schematic representation.

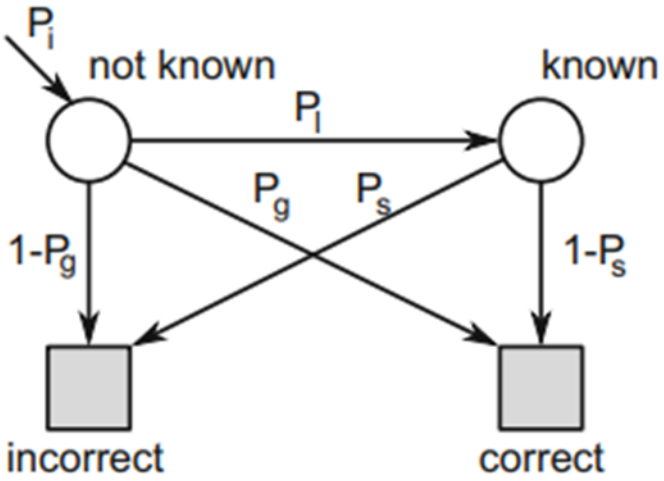


Figure 4.1: The structure of a BKT model.

Source: Adapted from [16]

Prediction equations

While the model internally keeps a prediction of whether the student has mastered a skill, this does not equal the probability that the student will solve an exercise on this skill correctly, as the student can guess the correct answer while not knowing it, or a slip-up can occur on an exercise the student would usually solve correctly. Therefore, the prediction on the correctness of the exercise must be calculated using the prediction equation, as shown in Equation 4.3 [72]. This equation can once again be easily understood intuitively: the probability that the student will answer correctly is the sum of the probability that they did not know, but guessed well and the probability that they did know, and a slip-up did not occur.

$$P_{correct,i} = P_g(1 - \theta_i) + (1 - P_s)\theta_i \tag{4.3}$$

4.1.2 Sentence and word similarity

In the game as described in Chapter 3, the difficulty is adjusted based on the similarity of the distractors. To do this, a model for the similarity of two natural language sequences is needed. This is done by first calculating the embeddings of these sequences, after which a distance metric is applied to them.

The embeddings were calculated using the *all-MiniLM-L6-v2* model as published on Huggingface [73], that is intended for use with sentences and short paragraphs, but, after some initial tests on relevant words and sentences, seems to perform adequately on word similarity. The distance metric used here is the Euclidean distance between the vectors of the calculated embedding. This is not a normalized vector, so when it was used in an absolute way, the limit was chosen through trial and error. Another metric that is often used is cosine similarity, which measures the angular distance between the vectors. Research shows that for text similarity, cosine similarity outperforms Euclidean distance in some cases [74]. Therefore, in further research, it is preferable to use cosine similarity instead of the euclidean distance metric.

4.1.3 Implementation

As stated in Chapter 3, the choice of the words within a round of the game is personalized to the skills of the user. Of the five words that are used within one round, one is the word that is being practiced, and appears in the spoken sentence. This word will be referred to as the practice word, while the other four are the distractors.

All of the aforementioned words come from a vocabulary list. This list must contain words that can be presented visually due to the design of the game. Within the proof-of-concept of this master's dissertation, a vocabulary list on *Clothes and accessories* from the book *English for Everyone: Level 1: Beginner, Course Book* was chosen [68]. The words in this list as well as their Spanish translations are shown in Table 4.1. This theme was well suited for the game as it contains words that are easily visually represented.

Both the choices of the practice word and of the distractors are influenced by the BKT student model. This model was discussed in detail in the beginning of this section, where its parameters were also listed. The values of these parameters were chosen partially from literature, and partially based on the structure of the game. In short, the four parameters are (1) P_i, the probability that a skill is initially learned, (2) P_l, the probability of the student learning on next practice opportunity, (3) P_s, the probability of a slip-up and (4)

English	**Spanish**	**English**	**Spanish**
t-shirt	camiseta	socks	calcetines
blouse	blusa	boots	botas
shirt	camisa	shoes	zapatos
dress	vestido	sandals	sandalias
skirt	falda	sneakers	zapatillas
pants	pantalón	scarf	bufanda
jeans	jeans	hat	sombrero
jacket	chaqueta	gloves	guantes
coat	abrigo	belt	cinturón
raincoat	impermeable	bag	bolso

Table 4.1: The English words in the vocabulary list and their Spanish translations.

P_g, the probability of a student guessing correctly while not knowing. The values of P_i, P_l and P_s were chosen from literature [18], while P_g was chosen as 0.20, as there are five answer options. The values are shown in Table 4.2.

P_i	P_l	P_s	P_g
0.20	0.10	0.15	0.20

Table 4.2: The values for the BKT parameters.

The student model gives an estimate of the probability that a student will answer correctly: $P_{correct}$, with $P_{correct,i}$ as that probability for an exercise on the word with index i. These values are used as metric for student knowledge when making the choice of words. The practice word for the next round of the game is always chosen as the word with the lowest estimate of student knowledge by the student model, which is seen as the least well known word. If there are multiple such words, one is chosen at random.

Then, based on the estimate of student knowledge of the practice word, the level of this

round is chosen as easy, medium or hard. This level was chosen by dividing the estimate using the bounds as shown in Table 4.3. These were chosen based on the requirement that a new word would always start in the easy category, but the categories are of similar sizes. With the parameters discussed above, an exercise with a new word is estimated to be correct with a probability of $P = 0.33$.

Easy	Medium	Hard
0-0.40	0.40-0.70	0.70-1.00

Table 4.3: The boundaries for the levels of the rounds, based on BKT-estimates.

The distractors used are based on the level of the round. The idea is that the difficulty increases when the distractors are more similar to the practice word. An example from the vocabulary list: if the practice word is *shirt*, similar distractors would be *t-shirt*, *blouse*, while less similar distractors could be *shoes*, *belt*.

When choosing distractors, first, the possible distractors of the correct level are chosen. This is done by sorting all of the words in the vocabulary list based on their difference to the practice word, in the ascending order, which is calculated in the embedding space, as described before. The first word of this sorted list is the practice word itself, so this is deleted. The resulting list is then split into three equal parts, which correspond to the easy, medium and hard level. Now that a list of words in the correct level is known, the distractors must be chosen. This is done by choosing the four least well known words of this list, based on the estimates of the student model. The idea behind this is that using lesser known words as distractor leads to a smaller chance that the student gets the exercise correct by exclusion. Additionally, when the student answers the exercise incorrectly, both the practice word and the wrong guess are shown on the screen. This appearance of the word is also a chance for the student to learn it. Therefore, using the least well known words adds some practice of these words.

The idea of basing the difficulty of the game on the BKT model was inspired by work done by Schodde et al. [17]

4.2 Description generation

In this section, the generation of the descriptions is discussed. First, the choice and usage of the LLM is discussed, then the model used for translation follows. Then, the implementation of these models within this master's dissertation is discussed.

4.2.1 Language model

As seen in Chapter 3, the game contains descriptions related to a certain vocabulary word that the student is learning. These descriptions are generated using a large language model (see Chapter 2), more specifically a variant of the GPT-3 family, which are transformer-based models [44]. These models can be used through a paid API provided by OpenAI. This model was chosen because of its good performance, as well as for the ease of use of the API, so no LLM must be run locally.

Models such as GPT-3 are zero- or few-shot learners: it is not necessary to fine-tune the model for the specific task it must perform. Due to the natural language understanding gained from pre-training, it is able to complete the task with no or just a few examples. The input of such a model can therefore be a natural language sequence, often called a prompt. In the case of few-shot learning, this prompt contains the examples as well as the task. Often, it is enough to describe the task in natural language. In the case of description generation based on a word, a prompt could be: "Generate a description using the word {*word*}." Based on the specific requirements of the description, additions can be made.

When starting this master's dissertation, the newest generation of models in the GPT-family was GPT-3. This generation consists of multiple models, with different performances and different pricing. After trying out the available versions for the generation of descriptions of some example words, it seemed that the performance of the *text-curie-001*-model and *text-davinci-003*-model were sufficient. As the *curie*-model is ten times cheaper than the *davinci*-model, this was used further.

Later on, in March 2023, the model behind ChatGPT was released: *gpt-3.5-turbo*. This model had increased performance for a lower price than the previously mentioned *curie* model, so the switch was made. As this model is optimized for chat, it takes an additional input called *system*. This is meant to give the chatbot initial instructions on its function. It can be used to influence the tone and writing style of the generated language.

How this was used here is discussed in the section on implementation.

Later that same month, the roll-out of GPT-4 started as well. As this model at the time was only available through a waitlist and the performance of the previous model was sufficient, the switch was not made.

4.2.2 Translation

Most LLMs and text-to-image models perform best when the text input given to them is in English, and they are often even intended to be used with English only. This means that if the language the student is learning is not English, there is a need for a translation model. In this master's dissertation, the focus was on Spanish as goal language. The first approach that was tried out was to generate English sentences and use these for the generation of images. Then, when the descriptions were to be presented to the student while playing the game, they were translated.

At first, the descriptions were translated using a model published on Huggingface that can be run locally [75]. After having a native Spanish speaker evaluate the results of this model, it was concluded that the translations were not always correct, often with grammatical inaccuracies or unnatural constructions. This was partially due to the translation model's imperfections, but also because the generated sentences were at times constructed in a manner that is very typical for the English language, but doesn't translate to Spanish very well.

In order to prevent presenting grammatically incorrect sentences to a student learning the language, the switch was made to Microsoft Azure's translator [76], which is available as an API that is free to use for students. After a comparison of these results to the previous model's translation by the same native Spanish speaker, it was concluded that there were few grammatical errors left, but some sentences still contained weird constructions or were overly complicated due to difficult to translate English constructions.

As the resulting translations were still not completely satisfying, another approach was taken. GPT-3.5 was trained on mainly English data, but other languages are present as well [77]. Because of this, a different approach was possible: the sentences were generated in Spanish by the LLM, after which they were translated to English by the Microsoft Azure translator, to ensure the best results when using the text-to-image model. This leads to simpler, more natural sentences in Spanish, as they aren't translated from English. Furthermore, if the translation should introduce grammatical errors, these are never shown

to the student. The final results of the translation and its possible inaccuracies will be discussed in Chapter 5.

4.2.3 Implementation

Given the language model described before and the words that were chosen as described in the previous section, the next step is to generate a description for every one of these words. After these are generated, a check is done on the similarities of these sentences. If two sentences are too similar, the corresponding images might differ very little. Then, the game might be too difficult, so one of them is regenerated, until no two sentences are too similar.

As discussed above, the sentences are generated using GPT-3.5, which takes two prompts as input: one *system* prompt which can be used to specify the role of the LLM, as well as influence the style of the generated text. The specific task is also given as input in the form of a prompt. In the task prompt, the vocabulary word is filled in, as well as the theme of the vocabulary list, which is *clothes and accessories*.

It is important for the generated sentences to fulfil certain criteria: the sentence must be short enough, not too difficult in structure and vocabulary and they must not contain other words from within the theme, as this can cause confusion. These criteria were added to the system prompt after the generated sentences did not seem to match them. The GPT-3.5 model tends to generate context surrounding the sentences, which is unwanted. Therefore, it was added to the system prompt that the system may only speak Spanish, and to the task prompt that only the sentence itself must be provided. In the task prompt, it is mentioned that a description of a picture is needed, in an attempt to achieve visual sentences. It is also said that the vocabulary word must be in the sentence, as sometimes, sentence were generated that described the word, without containing the word. Lastly, the theme of the word is also mentioned in the task prompt, to avoid confusion with synonyms with a completely different meaning. An example of this was encountered when testing with words from the theme *Farm animals*. One of the words in the theme was *chick*, which was of course meant to refer to a small chicken, but can easily occur in a different context with the meaning *young woman*. Therefore, adding the theme to the prompt leads to sentences incorporating the correct, appropriate meaning. All of these things were taken into account, resulting in the prompts that are given below.

- System prompt: *You generate the descriptions in textbooks for learning words. The*

descriptions are short, simple sentences with easy words for new students. The book has a theme, and within this theme, only the specified word is used. You speak only Spanish.

- Task prompt: *Generate a short one sentence description in Spanish of a picture that contains a* {**word**}*. It is in a book for learning the vocabulary for* {**theme**}*. The sentence must contain the word* {**word**}*. Only provide the sentence itself.*

After all of the sentences have been generated, the similarities are calculated two by two, using the embeddings and similarity metric as discussed in Section 4.1. Two sentences are deemed too similar of the metric goes below 10.0. This value was chosen based on some small tests with generated sentences, and is a balance between not using too similar sentences that make the game too difficult, and the time it costs to often regenerate sentences. If two sentences are too similar, one of the similar sentences is added to a list. After checking all sentence pairs, all of the sentences in the list are regenerated, after which the similarity check is performed again. This continues until no two sentences are too similar. In practice, as the 20% most similar words in the vocabulary list are never chosen as distractors, two sentences were rarely too similar according to this metric.

4.3 Image generation

In this section, the generation of images is discussed. First, the choice of model is introduced, after which the implementation and usage of that model within this master's dissertation is discussed.

4.3.1 Text-to-image model

A second requirement for the game as described in Chapter 3 is the generation of images, given the descriptions that were generated as discussed above. This is an example of the text-to-image task as described in Chapter 2.

Text-to-image models take a natural language sequence as input. As in large language models, this is also referred to as a prompt. The style of the image can then be influenced by adjusting the prompt. This is called prompt engineering. In the game, the image must be generated based on the description, so that will form the basis of the prompt. How exactly this is done and what style choices were made and implemented using prompt

engineering will be discussed in the next section.

The first model that was tested in this master's dissertation was DALL-E by OpenAI [53], which can be accessed through a paid API. After some initial tests using descriptions generated as discussed above, the performance of this model seemed sufficient for this application, but, as the available API is paid, the price of usage went up quickly. Stable diffusion [69], as discussed in Chapter 2, is an open source model that can be run locally, with a similar performance as DALL-E in initial tests. Due to this, the model is free, and there is more control on the speed, as the hardware on which it is running can be adjusted freely. These advantages lead to the further usage of Stable Diffusion throughout this master's dissertation.

4.3.2 Implementation

Using this text-to-image model and the descriptions that were generated as described in the previous section, images can be generated. As discussed before, Stable Diffusion takes a prompt as input. This prompt could consist of the generated description, possibly with a style modifier added. When no style is specified, models like Stable Diffusion tend to generate images in a photo realistic style. As these generated images often contain some mistakes, the photo realistic style can result in strange, impossible and even creepy images.

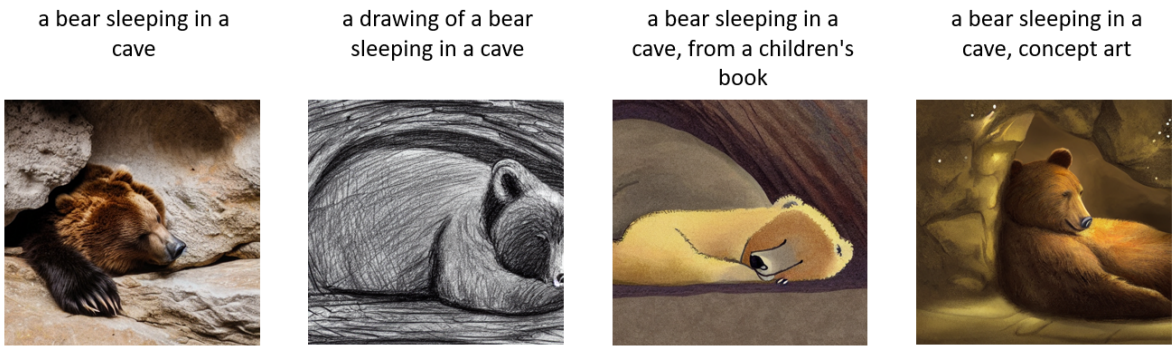


Figure 4.2: Example of influence of style description in prompts using Stable Diffusion.

As depicted in Figure 4.2, there are many possible style modifiers that lead to results of varying success. To avoid overly realistic and therefore creepy images, it could help to specify that it should be a drawing. This easily results in very simplified drawings. A second attempt was to specify that the image is from a children's book. This leads to a

more appropriate style than when the drawing style is used, but the image is still very minimalist and simplified. There are many tricks within the field of prompt engineering that can lead to the desired styles. One with good results for this application was adding the style modifier *concept art* to the generated descriptions, as suggested in many prompt engineering guides and blogs (e.g., [78]). An example is shown in Figure 4.2. This style modifier was therefore used for the generation of images throughout this master's dissertation. This was the final prompt:

- Image prompt: {**description**}, *concept art*

The text-to-image model used contains a safety check that all generated images contain only appropriate content. Otherwise a black image is returned. This check regularly returns what appears to be false positives, leading to black images in the game. This can be handled in two ways. The safety check can be disabled, but then it might occur that actual inappropriate content is used within the game. The other option is to check if the image is black, by looking at the pixel values, after which the image can be regenerated. As this second option was easy to implement and does not introduce the risk of presenting the user with inappropriate content, this method was used.

4.4 User interface

When all content is generated for a round of the game, it must be presented to the user. A few choices had to be made on how this is presented, which will be explained here.

First of all, the decision on whether the description is shown as written text had to be made. Showing the text makes the game easier, especially if the user knows a language related to the one that is being practiced. It also enables the user to learn the spelling of words while practicing. A disadvantage is that the user will then mostly look at the screen to read the text, which lessens the amount of social interaction with the robot. The language that was taught in this application was Spanish. The application was tested in the Dutch speaking part of Belgium. French is one of the other official languages of the country, so the general population, including the test group, has a basic knowledge of the language. As both French and Spanish are Romance languages, the choice was made not to show the text on the screen. Then, the game was not too easy, and social

interaction with the robot was stimulated. Another effect of this choice, is that the student is practicing their listening skills, while reading skills are practiced as well when the sentence is shown. To enable the user to at least learn the spelling of the words they are practicing, the words corresponding to the correct and possibly incorrect image are shown above them at the end of the round.

In first tests of the application, it became clear that users tended to look at the images while the robot is speaking, instead of looking at the robot as in a social interaction. This lead to the design choice of showing the images on screen with a delay, letting them appear around the time the robot finished its sentence.

Another adaptation that was made to enhance the social interaction was some subtle interaction and feedback from the robot. After the description is spoken by the robot, it looks at the screen where the images appear. This indicates to the user that the robot is finished speaking and interaction with the screen is now required. Then, the user indicates what image they choose, by tapping this image on the touch screen. If the user chose the correct image, a small and subtle smile appears on the face of the robot. If the user chose incorrectly, a subtle frown appears.

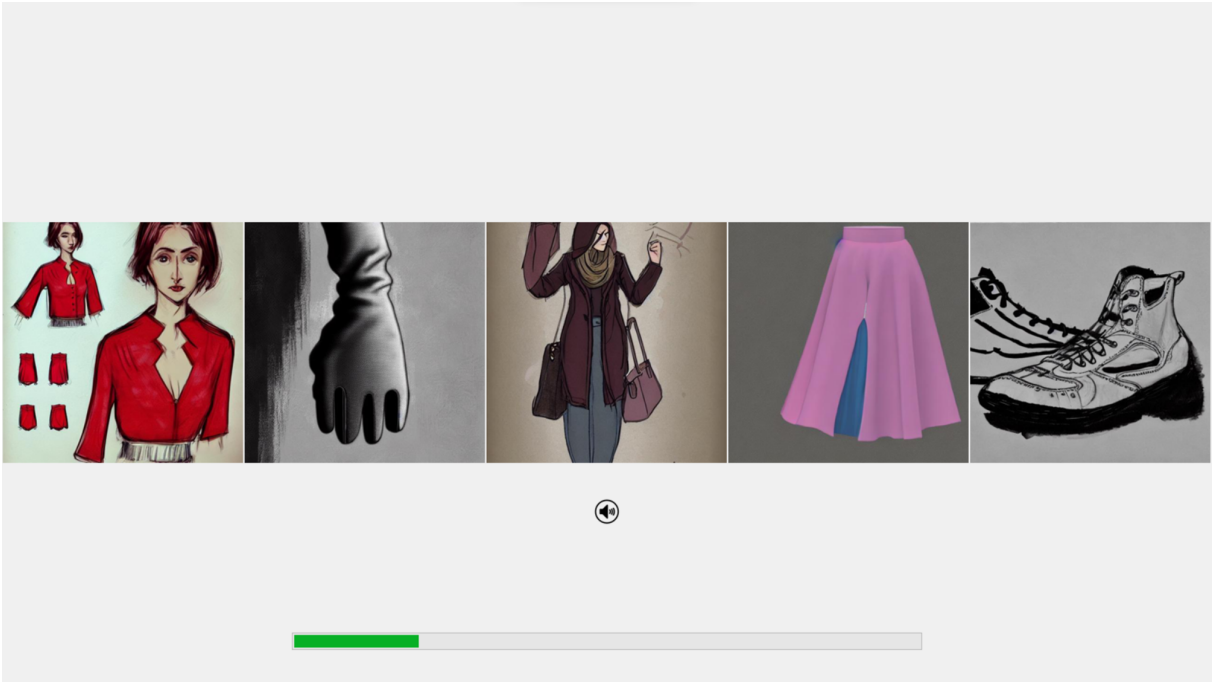
As the written description is not shown on screen, understanding the Spanish sentence from the first try might be difficult for a new user. Therefore, the user always has the opportunity to let the robot repeat the sentence by tapping a speaker icon on the screen.

When the user chooses an image, some feedback is shown on the screen. All irrelevant images disappear, while the correct image is shown with a green border, and, if there is one, the incorrect image is shown with a red border.

A last design choice that was made was to show a progress bar on the screen that indicates how far the user is within the set of exercises. First tests indicated that it is unpleasant for the user to be presented with exercises without any indication of progress or how many rounds were left. The amount of rounds played is of course adjustable, but in the user study, 60 rounds were used, which corresponds to about 10 minutes of playing the game.

All of these choices were implemented in the user interface, which was made using the Python library TKinter [79].

In Figure 4.3, an example of the user interface described above is given. In the first image, the screen after the robot has spoken the description is shown. All five images appear on screen, as well as the speaker button for repeating the description and the progress bar. The description heard here could be "Los guantes son para mantener las manos calientes.", corresponding to the second image of the game. In Figure 4.3 (b), the screen is shown after the user has tapped the fourth image. On the screen, the incorrectly chosen image as well as the correct one are shown, with corresponding red and green borders. Above both images, the corresponding word is shown.



(a) The beginning of a round, after the robot has spoken the description.

(b) The second part of the round, after the user has tapped the fourth image

Figure 4.3: Screenshots of a round of the game.

Chapter 5

Results

In this chapter, the results of this master's dissertation will be discussed. The integrated system was evaluated in a user study. In this, a group of participants took a written test on their Spanish vocabulary knowledge, to establish their level of Spanish vocabulary before receiving a lesson. After this, they practiced Spanish Vocabulary using the application. Then, the students took a post-test, which allows us to quantify the learning gain. In order to avoid technical problems due to unreliable connections to remote servers and variable delays, the descriptions, translations and images used in this study where generated beforehand. This also means that all students encountered the same content when practicing, though not necessarily in the same order. Therefore, the discussion of the results of this master's dissertation can be split into two parts. First, the data that was generated for this user study is evaluated in Section 5.1. Lastly, the user study, including its set-up, process, limitations and results are discussed in Section 5.2.

5.1 Generated data

From its conception, the application was designed for use with real-time generated data. Therefore, first the time constraints of generating the data in real time are discussed. As the data for the user study was generated beforehand, it is possible to investigate its quality, which also gives us some insight in the data that could be generated in real time. The quality of the generated data is discussed in following sections.

As discussed in the previous chapters, the generated data consists of descriptions, translations of these descriptions and images. The descriptions were generated in Spanish and then translated to English. These English descriptions were then used as input to the text-to-image model. Therefore, the evaluation of the generated data is done in three

parts. First, the correctness of the Spanish sentences and their translations to English is discussed. Then, the appropriateness of these sentences for use within the game will be evaluated. Finally, the last section is about how well the generated images correspond to the sentences from which they were generated.

The vocabulary list used here consisted of 20 words within the theme *Clothes and accessories*, as discussed in Chapter 4. For the user study, it would be undesirable if each word always occurred with the same description and image. Therefore, for each word, five descriptions, translations and images were generated, to ensure variation in the visual and auditory stimuli. The game play in the study consisted of 60 rounds. In each round, 5 images were presented together with a single description. In total, 100 images and descriptions were generated, so on average, each image appeared three times. This was of course often as distractor, which means that the corresponding descriptions were not presented to the user.

5.1.1 Real time generation of data

If the data of the game is generated in real time, the introduction of large delays would have a detrimental effect on the learning experience. Within a round, the vocabulary words are first chosen. Then, from these words, descriptions are generated using a LLM and translated using Microsoft Azure's translator, after which their similarity is checked using word embeddings. Then, the images are generated with Stable Diffusion on a remote server, after which they are displayed. This means that all data of a round of the game must be generated within that round. The delay due to the generation and translation of the descriptions must therefore be negligible. As the images are shown only after the sentence has been spoken, this can take as long as it takes to speak the generated description.

To obtain all descriptions, they must first be generated and translated, checked for similarity and, if needed, new descriptions must be generated and translated. Therefore, the time to build a single exercise is variable. The generation and translation of sentences is done remotely through an API call, but the similarity check is done locally, so it is influenced by the hardware on which the model runs. The generation of images is also done locally. This is the part that introduces most of the delay, and it is also strongly influenced by the specifications of the hardware. To get an idea of the possible speed-up through improved hardware, Table 5.1 shows the timing when run on two different GPUs

of the IDLab GPULab, in seconds. The time for generating a round is split into three parts: the generation, translation and similarity check of the descriptions (including the choice of words), the generation of images and the display. The total time is also reported.

GPU	NVIDIA GeForce GTX 1080 Ti	Tesla V100-SXM3-32GB
Descriptions	5.45s	5.20s
Images	68.25s	15.10s
Display	5.38s	2.73s
Total	79.62s	23.35s

Table 5.1: Time needed to generate the parts of a round of the game.

5.1.2 Translations

The generated Spanish sentences and their English translations were evaluated by a native Spanish speaker. These were divided into three categories: correct, partially correct and incorrect. Table 5.2 contains a summary of the correctness of the translations. The partially correct translations were grammatically correct but unusually phrased. An example of a partially correct and of an incorrect sentence are given below.

- Partially correct: Es: Los pantalones son de mezclilla azul. En: The pants are blue denim. *Mezclilla* refers to the material, and is normally not used in this way.
- Incorrect: Es: La mujer lleva un hermoso bufanda alrededor de su cuello. En: The woman wears a beautiful scarf around her neck. The Spanish sentence should be *La mujer lleva* ***una hermosa*** *bufanda alrededor de su cuello.*

Correct	95
Partially correct	2
Incorrect	3

Table 5.2: Correctness of the generated translations used in the user study.

5.1.3 Description generation

As the English sentences are used for the image generation, the generated descriptions were evaluated after translation to English. They were evaluated on four criteria by five people with Dutch as mother tongue but with a sufficient knowledge of the English language. The criteria used are listed here:

1. Is the sentence appropriate within the theme 'clothes and accessories'? (0: not appropriate, 5: perfectly appropriate)
2. Does the sentence contain other words from the theme 'clothes and accessories', such as other pieces of clothing or accessories? (0: many other words, 5: no other words)
3. Is the sentence unnecessarily long? (0: unnecessarily long, 5: short)
4. Is the sentence simple enough for someone with a limited knowledge of the language? (0: very complicated, 5: simple enough)

The mean score and standard deviation of the sentences on each criterion are given in Table 5.3.

Criterion	Mean	Standard deviation
1	4.99	0.1
2	4.748	0.713
3	4.612	0.654
4	4.428	0.808

Table 5.3: Average score and standard deviation of the sentences on the four criteria.

Another interesting characteristic of the descriptions is how many words in the sentence describe visual properties of the word. For example, in the sentence *'The dress is red.'*, both *dress* and *red* give a visual hint. If the student would not know the word *dress*, but does know the word *red*, they could deduce from the images which one is correct, without knowing the practice word itself. Therefore, if the sentences contain a lot of visual descriptors, the game could become too easy. When designing the prompts for the LLM, no instructions were given on the amount of visual descriptors the sentences

could contain. On average, the generated sentences used in the game contain 2.17 visual descriptors. Table 5.4 contains the number of sentences that contain one, two or three visual descriptors. An example of each group is given below.

- One descriptor: Sandals are a type of footwear ideal for sunny days. Descriptors: **Sandals**
- Two descriptors: The shirt is green. Descriptors: **Shirt, green**
- Three descriptors: The blouse is red and has short sleeves. Descriptors: **Blouse, red, short sleeves**

Number of descriptors	Number of sentences
1	15
2	53
3	32

Table 5.4: Number of sentences per number of descriptors within the sentence.

5.1.4 Image generation

The images used in the user study were generated based on the descriptions as described in Chapter 4. In general, the performance of text-to-image models is impressive, but far from perfect. Therefore, the factual correctness of the images compared to the sentences they were generated from is discussed here. The images were divided into three categories: correct, partially correct and incorrect. In the correct images, what is described in the sentence is clearly portrayed in the image. In the partially correct images, most of the sentence is correctly portrayed, but some details are incorrect or not portrayed clearly. In the incorrect images, what is shown in the picture is clearly not what is in the sentence. This is summarized in Table 5.5.

Figure 5.1 shows illustrative examples. Figure 5.1a shows an incorrect image: the gloves are described to be black but the image contains white gloves. Figure 5.1b contains a correct image of a pink dress. Figure 5.1c shows a partially correct image, where the jacket is said to be yellow, but in the image it is black as well as yellow. Lastly, Figure

Correct	83
Partially correct	7
Incorrect	10

Table 5.5: Correctness of the images generated for the user study.

5.1d is also partially incorrect: the blouse is red but has black buttons, unlike the white buttons in the description.

5.2 User study

In this section, the study meant to test the application will be discussed. First, the set-up of the study and the process through which the students went within the study will be discussed. After this, a small recap of the limitations of the generated data within the study is given. Lastly, the resulting data from the study will be discussed.

5.2.1 Set-up and process

The study recruited 21 high school students majoring in Latin, of which 11 were 15 years old and 10 were 16 years old. As the data was collected from minors, consent was received from their parents. The students were divided into two groups, of which one practiced the vocabulary using the setup as described and the other group acted as a control group by playing the same game, but without the robot. In this group, the images were also shown on a tablet, but the spoken sentences were played by the tablet. The set-ups of the two groups are shown in Figure 5.2. The students were randomly assigned to these two groups, resulting in a robot group of 10 students and a tablet group of 11 students. The students were sent to the location of the study two-by-two by their teacher. First, they took a written multiple choice test of the 20 vocabulary words as discussed in chapter 4, as well as five color terms that often appeared in the descriptions. After about ten minutes, the students moved into the classrooms where the study was set up. Here, they played 60 rounds of the game, with or without robot. This amounted to around 10 minutes of practice. The content of these rounds was randomized and adjusted to the performance of the student as discussed in Chapter 4. When a student indicated that the game was finished, they took the same test as before practicing. Additionally, they filled in a small

(a) Incorrect: These gloves are black.

(b) Correct: The dress is pink.

(c) Partially correct: The waterproof jacket is yellow.

(d) Partially correct: The blouse is red and has white buttons.

Figure 5.1: Correctness of generated images and their descriptions.

questionnaire about their experience during the study and some basic demographic data. The test that was used as pre- and post-test as well as this questionnaire are added to the appendix. After this, they rejoined their class and were asked not to share anything about the study with the students that did not yet participate. As this was a group of native Dutch speaking students, the instructions as well as the test and questionnaire were in Dutch.

The study ran during two separate class periods. After all the students had done the study, the class was gathered for a debriefing and a small introduction on social robots, and there was an opportunity to ask questions.

(a) Set-up of the study with the Furhat robot.

(b) Set-up of the study without the Furhat robot.

Figure 5.2: Example of the set-up of the two groups of the study.

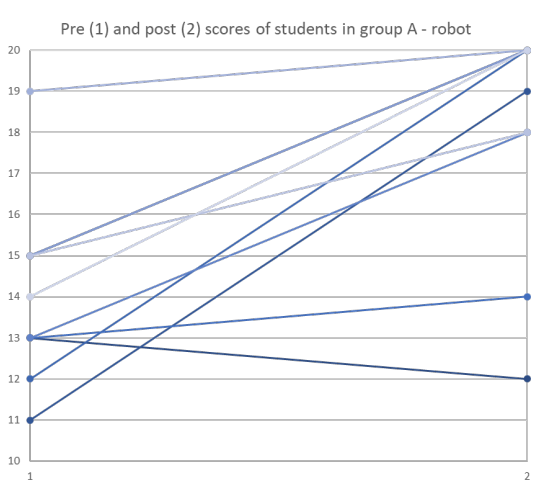
5.2.2 Results

In this section, the results of the user study will be discussed. This is divided into three parts: (1) the evolution of the students' scores between pre- and post-test, (2) the difference in learning effect between the two groups and (3) the correctness of the student model in comparison to the post-test scores of the students.

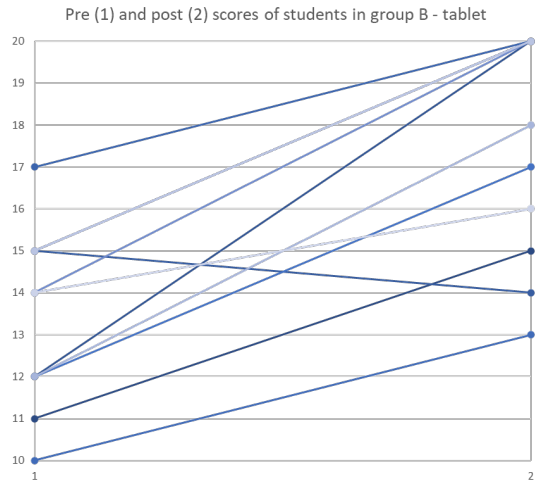
Learning effect

The learning effect on the Spanish vocabulary of the students in the user study can be evaluated through the evolution in the students' scores on the pre- and post-test. Figure 5.3 shows the evolution per student from the pre-test, at x-value 1, to the post test, at

x-value 2, for the two groups. The y-axis shows the score out of 20, which corresponds to the 20 vocabulary words. Each line in the graph represents one participant and shows the evolution between their pre-test score (1) and their post-test score (2). The colors that were questioned on the tests are not included in this graph. This graph shows a clear increase in test scores for most of the students. It can also be noted that the ceiling effect takes place in this data: many of the students achieved the highest possible score on the post-test. This might mean that the students' scores would have increased even more if this had been possible. A relevant metric to assess the effectiveness of an educational intervention is the normalized learning gain. It is calculated by dividing the learning gain by the maximum possible learning gain. This metric was calculated for the pre- and post-test scores of all students, and led to a mean normalized learning gain of 0.69, with a standard deviation of 0.40.



(a) Evolution of scores in pre- and post-test of group A - ROBOT.



(b) Evolution of scores in pre- and post-test of group B - TABLET.

Figure 5.3: Evolution of students between pre- and post-test.

Figure 5.4 shows four boxplots of the scores of the students in pre- and post-test, for both groups separately. The boxplots show the minimum and maximum value through the whiskers, the first and third quartile through the boundaries of the box, the second quartile or median through the line in the box and the mean through the cross mark. Looking at these plots, a clear increase in score can be seen. The ceiling effect is also apparent here, which is indicated by the boxes reaching the maximum score of 20.

To evaluate the learning effect of the application, the two-sided Wilcoxon signed rank

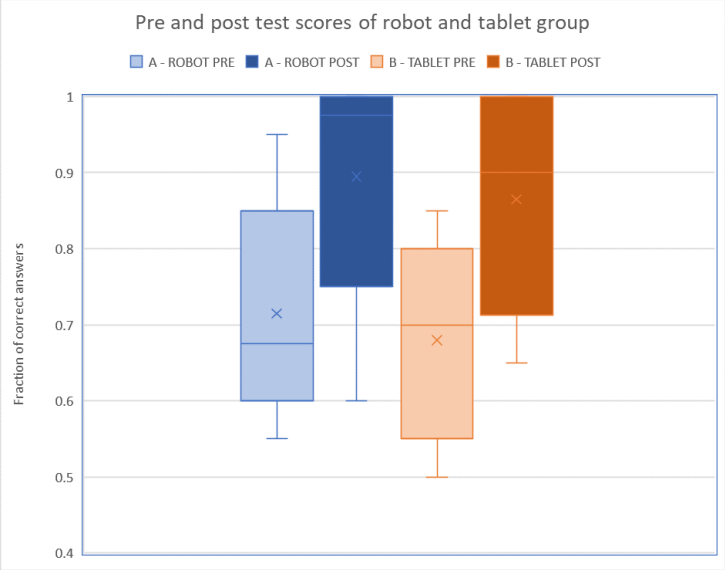


Figure 5.4: Distribution of scores in pre- and post-test for students in the robot group (blue) and the tablet group (orange).

test was performed on the pre- and post-test scores of all 21 students. This test shows that there is a significant difference between the pre- and post-test scores ($V = 5, p < 0.001, n = 21$). Looking at Figure 5.4, it is clear that this difference corresponds to an increase in the scores. It can be concluded that the use of the tutoring application led to a significant learning effect.

Effect of robot

The use of the application was shown to have a significant learning effect for students of both groups together. Now, the difference between both groups will be discussed.

In Figure 5.3, the evolution per student is shown for both groups. No immediate difference between these two graphs can be seen. In Figure 5.4, the distribution of post- and pre-test scores is shown for the two groups. The graph shows that the student in the tablet group started with a lower score in general, and also seemed to attain a lower score on the post-test. The ceiling effect makes it difficult to draw conclusions from this graph, as many of the students obtained the highest score.

To check for a difference between the two groups, the Wilcoxon rank sum test (also known as the Mann-Whitney U test) was performed on the differences between pre- and post test for the students in both groups. This test showed no significant difference

between the difference in pre- and post-test scores of the two groups ($W = 55, p = 1, n_1 = 10, n_2 = 11$). Here, the normalized learning gain is also a relevant metric to assess the effectiveness of the educational intervention, with and without robot. The normalized learning gain of both groups was calculated, after which the Wilcoxon rank sum test was performed on this metric. This test showed no significant difference between the normalized learning gain of both groups ($W = 59.5, p = 0.7656, n_1 = 10, n_2 = 11$).

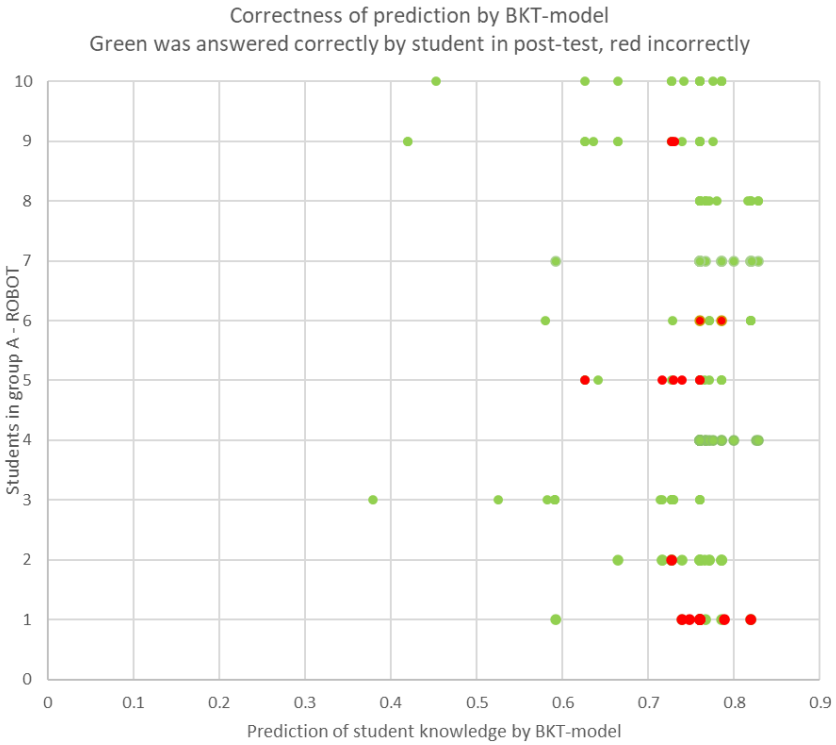
From these result, it can be concluded that the presence of the Furhat robot when playing the proposed game has no significant effect on the learning effect of students.

Student modeling

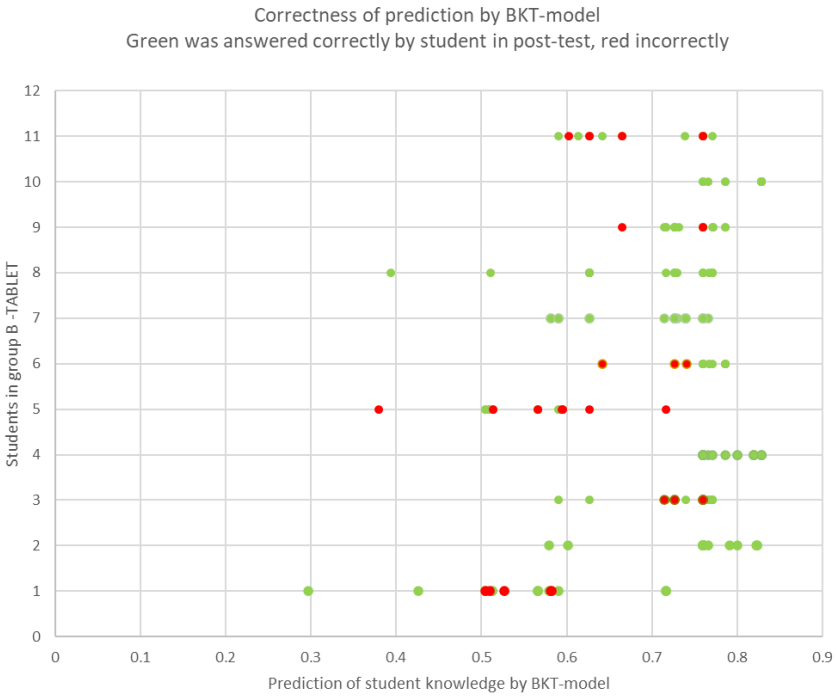
During the game, the content of the next round is based on the estimation of the student's skill by the BKT student model. One of the outputs of the BKT-model is the prediction of the correctness of exercises on the words within the vocabulary list. If the student model is a correct representation of the student's knowledge, this prediction at the end of the tutoring should correspond to the students' performance on the post-test taken right after. Important to notice here is that the student model bases its prediction on the exercises within the lesson, and that the exercises within the test are of a different form. This might impact the performance of the predictions.

Figure 5.5 contains a visual representation of the BKT predictions and the actual post-test answers for the two groups. The x-axis contains the probabilities generated by the BKT-model. The y-axis shows the individual students. Per student, there are 20 dots in the graph. Each of these dots corresponds to a word in the vocabulary list. The location of the dots on the x-axis represents the BKT-models prediction on the probability that this student would answer an exercise on this vocabulary word correctly. The color of the dot indicates whether the student answered the exercise corresponding to this dot correctly on the post-test, with a green dot showing a correct answer, and a red dot an incorrect answer. If the BKT-model would produce perfect predictions, all of the red dots would be located at the left side of the graph and all of the green dots would show up on the right. By looking at these graphs, an obvious pattern does not immediately appear.

In order to draw conclusions about the correctness of the BKT predictions, some statistical tests were performed. The predictions of the BKT-model on words whose exercises were answered correctly on the post-test were compared to the ones whose exercises were answered incorrectly. Looking at Figure 5.5, this corresponds to comparing the position



(a) Correctness of BKT prediction of group A - ROBOT.



(b) Correctness of BKT prediction of group B - TABLET.

Figure 5.5: Results of BKT student model. The x-axis shows the prediction of the student model, the y-axis shows the different students and the colors show the correctness of the answer in the post-test.

	Train	Test
Accuracy	78.3%	65.5%
Precision	19.7%%	16.7%
Recall	39.4%	30.8%
F1-score	26.3%	21.6%

Table 5.6: The relevant metrics for the performance of the BKT-model.

on the x-axis of the green dots and the red dots. If the predictions are good, these positions should come from a significantly different distribution. To test this, the Wilcoxon rank sum tests was performed on the data of the robot group, the tablet group and of both groups together. When performed on the robot group, the test lead to insignificant differences in the distributions ($W = 2012, p = 0.2103, n_1 = 181, n_2 = 19$). When performed on the tablet group, the test lead to a significant difference ($W = 3817, p = 8.002\text{E}{-}05, n_1 = 193, n_2 = 27$). When the test was performed on the data of both groups together, a significant difference was found ($W = 11663, p = 6.228\text{E}{-}05, n_1 = 374, n_2 = 46$).

It is not obvious to draw conclusions from these statistical tests, but, as the model is used for predictions, it is more useful to look at the accuracy of these predictions. This can be done by fitting a linear classifier on the predictions of the BKT, and to compare the outcome of this to the post-test values. Important to note is that there is a strong class imbalance here: on the post-test, there are many more correct answers than incorrect ones. Because of this, the linear model used here is a logistic regression model with balanced class weights. The data is divided into a train and test split of 80/20 and the model is then fit on the training data. This leads to a prediction boundary of 0.7060415: a BKT prediction of at least this value is predicted to lead to a correct answer, and everything below this value is predicted to be answered incorrectly.

In Table 5.6, the relevant metrics for the performance of the predictions of the BKT-model are reported. First, the accuracy is reported, but as the classes are strongly imbalanced, the more relevant precision, recall and F1-score are also given. These metrics are calculated with incorrect answers as positive label, as this is the minority class.

		Predicted	
		c	**i**
True	**c**	51	20
	i	9	4

Table 5.7: Confusion matrix of the predictions of the BKT-model after applying a linear classifier, with c and i meaning *correct* and *incorrect*.

In Table 5.7, the confusion matrix for the BKT predictions is given. The columns represent the predicted value, where c means it was predicted to be answered correctly and i incorrectly. The rows represent the true value of the correctness of the students' answers on the post-test, with the same meaning for c and i as before. As this table indicates, the model incorrectly predicts 20 exercises to be answered incorrectly, and incorrectly predicts 9 exercises to be answered correctly, while only four of the true incorrect answers were predicted accurately. This in combination with the low F1-score in Table 5.6, indicates that the predictive power of the BKT-model was not satisfactory.

Chapter 6

Conclusion

Starting this chapter is a discussion of the most important results, including their shortcomings. Then, in the section on future work, the possible extensions and improvements on this master's dissertation will be presented. In the last section, the conclusion follows.

6.1 Discussion

In the first part of Chapter 5, the generated data was discussed. In the generated sentences, the translations and the generated images, mistakes or imperfect results occurred. This is to be expected, as all of these things were generated using a model without any feedback loop present: there was no check on the appropriateness of the sentences and images or on the correctness of the translations. The text-to-image model and the LLM were used through prompt engineering. These models have very impressive performances, but they were not trained or fine-tuned on this specific application. Still, looking at the numbers in Chapter 5, their performance was quite good.

Of the generated translations, 95% were evaluated as correct and natural translations, with only 3% of the translations being marked incorrect. The evaluation of the generated sentences on the four criteria resulted in scores all between 4 and 5 out of 5, with only one being below 4.5. Of the generated images, 83% were evaluated to be factually correct, and 7% were evaluated to be partially incorrect. These results seemed to be sufficiently good for a clear learning effect as measured in the user study.

The learning effect of the two groups of students was shown to be statistically significant. When looking at the results of the students on both tests shown in Figure 5.3, it can be seen that almost all students have a clear increase in score after practicing with

the proposed application. In both groups, one student has a lower score on the post test, twice with a difference of one. A reasonable explanation for this is that the student guessed correctly on some questions in the pre-test, as this was a multiple choice test, and did not have such luck on the post-test. Looking at these graphs and the statistical results, the first research question as proposed in Chapter 1, can be answered: using the proposed tutoring system has a significant learning effect on the students' second language vocabulary acquisition.

The second research question was about the effect of the presence of the social robot during the lesson. As was shown in Chapter 5, there was no statistically significant difference on the increase in scores between the two groups. A possible explanation for this is that the game in its simplified form, as discussed in Chapter 3, did not resemble a natural social interaction closely enough for the positive effect to appear of social robots that has been shown in other research. This could be further investigated through the implementation of the original game idea, where there is a true visual conversation about the images and both the student and the robot can take on both roles. If the above explains the lack of difference between those groups, it is reasonable to expect that this advanced kind of game would introduce a difference in learning effect on account of the robot.

To implement such an extended game, there are some technical requirements. Well-functioning text-to-speech is necessary, specifically for people who do not master the language they are speaking, as this is the target group for this application. Next to this, the content spoken by the robot must be generated. In a conversation about images, this is mostly visual question generation and answering, as well as image captioning. All of these are a part of the field of visual conversation, as discussed in Chapter 2. Progress in these two fields, in combination with the LLMs and text-to-image that were already used here, could lead to an interesting and successful form of second language tutoring with social robots and generative.

Another result that was discussed in Chapter 5, is the output of the BKT-model as prediction for the scores on the post-test. Here, it was concluded that the final values of the BKT-model were not good for predicting the correctness of the answer on the relevant exercise. To explain this result, there are two possibilities to consider. It could be that the BKT-estimates were never correct, not during the game either, or it could be that the

predictions of the performance in the game did not translate very well to the post-test.

The second option is relevant to consider, as the students' skills were tested in a very different way in the post-test, than they were during the exercises of the game. First of all, in the game, the words were spoken out loud (and then shown on the screen after the exercise), while in the post-test they were written out. Secondly, the post-test asked for a translation, while translations never occurred in the game, as only the connection between the foreign language word and images of the object was trained. These differences might result in the bad predictions of the student model.

The other option is that the BKT-models predictions were never accurate, not even during the game. This could be due to the parameter choices. For example, letting the pre-test influence the initial estimates of the student model might have strongly increased the performance of the model. The other parameters, such as the guess, slip and learn probability might have also caused the insufficient performance, as they were not trained on this specific game but based on values from literature. This could be verified by training these parameters on part of the data gathered in the user study, and validating of the performance increase on the other data. Another approach could be to look at the accuracy of the predictions made during the game. This was considered out of scope for this master's dissertation. Another possible influence on the performance of the student model is that, as the vocabulary list contained 20 words, each of these words did not occur very often in the game. Longer game play might have led to increased performance, as more data would have been available.

Another limitation to consider about the proposed application as it is presented now, is that there are strong restrictions on what words can be used in the vocabulary list. The words used here, within the theme of *clothes and accessories*, all represent objects that can be represented visually and are very easy to recognize. Especially in the earlier stages of learning a language, many of the words that are taught fall into this category, but as the student's knowledge of the language progresses, more abstract concepts must be taught. The tutoring system as it is presented here might not suffice for learning these words. Also interesting to note is that the proposed application can only be used to teach nouns and adjectives. Due to the static nature of generated images, it is harder to express actions, relations and verbs. An extension of the system to include videos might ease this restriction. The current limited vocabulary that can be taught introduces a limitation

in the usefulness and the width of the applicability of the tutoring system. Despite of this, there are a lot of words that can be represented visually, for which this application is still useful. It might be that student gain implicit knowledge when practicing with the proposed application, such as the grammatical structure of the target language and its pronunciation, but this needs to be investigated.

6.2 Future work

As discussed in the previous section, the data that was generated was presented to the user without some sort of feedback or revision of the quality. This led to a small percentage of the data being suboptimal. This also means that there was no real control on how difficult the game was, and that the level was determined through extensive testing beforehand, but was locked during the game, except for the adaptations based on the student model. An interesting improvement on the application would be to implement this kind of feedback loop, where the generated descriptions are checked for appropriateness and difficulty before they are passed to the text-to-image model, after which the generated images are checked on how much they correspond to the given description.

Another suboptimal result was the prediction accuracy of the BKT-model on the post-test scores. As discussed in the previous section, there are some ways to investigate and improve this result. The first step here would be to further investigate the gathered data to see if the student model predictions were accurate during the game itself. If this is not the case, the BKT-model parameters could be improved by training them on the gathered data, which should lead to better results if the study would be repeated. If the model predictions were good during the game, it would mean that the way of testing the students' knowledge after practice was not representative for the way the vocabulary was practiced. Then, if the study were to be repeated, this test should be adjusted to be more similar to the game. This could mean that the words are spoken out loud and their meaning is shown using pictures instead of translations.

An interesting extension relates to the limitation of what words can be used, as discussed in the previous section. In this master's dissertation, all vocabulary words used, during development, testing and the user study, represented objects or animals that could

be shown. It might also be possible to use different classes of words, such as prepositions of spatial relationships between words (on, under, next to, above, ...) or tenses of verbs (he is running, he has run, ...), but this is only possible if the used generative models have sufficient performance on these details. To correctly picture these spatial prepositions or verb tenses, some reasoning is necessary, which might be too difficult for the text-to-image model. Further investigation on the use of these kinds of word classes within the tutoring system is necessary and might lead to interesting results.

As mentioned in earlier chapters, one of the advantages of using AI-generated content, is that this content can be adjusted to the student. As the content used in the user study was generated beforehand, this was not possible here, so an interesting extension could be to personalize the game to each student. A way of doing this that was investigated in the beginning of this master's dissertation is to ask a question about the interests of the student before the start of the lesson, and use this info as the theme in which the words are shown. For example, if the student would say that they really liked animals, and the vocabulary list provided by the teacher was about sports, the generated images could picture animals playing the different kinds of sports, by adding the extra info to the prompts of the generative models. After some more research on how exactly this could be done, this could form an interesting extension to the application.

A final extension that could be done is to return to the initial idea of the game, as discussed in Chapter 3. Here, the game consists of a full visual conversation about a set of images, where one of the players must try to find out which image they are talking about. Then, both roles of the game could be fulfilled by the student and the robot in an alternating way. With this game, the student would not only be passively practicing their listening skills, but also practice their speech. As this more complex version of the game would lead to a more natural social interaction, this could lead to a noticeable positive influence of the presence of the social robot, while this was not the case in the current version of the game. Nevertheless, the implementation of the full visual conversations within the game requires advances in speech recognition technology and much more research on what models could be used and in what way, but it is a promising extension.

6.3 Conclusion

The tutoring system proposed in this master's dissertation enables students to practice vocabulary of a second language with a social robot, where all content is AI-generated. The application is visually grounded, using pictures to represent the vocabulary words, so no translation to the students' mother tongue is necessary. The content of the application is AI-generated in real time, so no two rounds have to be the same. The descriptions are generated using a large language model, and are then fed to a text-to-image model to generate the visual content.

The robot acts as a tireless tutor, so it can be used as support for teachers, providing the students with more one-on-one practice. The implementation of this tutoring application was discussed, as well as the results of a user study performed with a group of high school students. The study showed a clear learning effect of the students after playing the proposed game. A control group that used the application without the presence of the social robot showed a similar learning effect, and there was no statistically significant difference between the two groups. There are many possible extensions and improvements of the application, that show great promise for a way to allow social robots to help in the classroom with second language tutoring.

Much of the technology used in this application is very new, with improvements constantly arising. The generated data is not yet consistently of high enough quality, as the generated images still contain mistakes, the translations are not perfect and the large language models do not follow instructions perfectly yet, but it can be expected that these flaws will become less frequent with improving technology. As this happens and text-to-speech technology improves, social robot tutors fully driven by generative AI can become a useful and supportive addition to the classroom.

Bibliography

[1] Christoph Leiter, Ran Zhang, Yanran Chen, Jonas Belouadi, Daniil Larionov, Vivian Fresen, and Steffen Eger. Chatgpt: A meta-analysis after 2.5 months. *arXiv preprint arXiv:2302.13795*, 2023.

[2] Cynthia Breazeal, Kerstin Dautenhahn, and Takayuki Kanda. Social robotics. *Springer handbook of robotics*, pages 1935–1972, 2016.

[3] Imola Katalin Nagy et al. In between language teaching methods: do we need (to know about) methods at all? *Acta Universitatis Sapientiae, Philologica*, 11(3):119–139, 2019.

[4] Cynthia Breazeal. *Designing sociable robots*. MIT press, 2004.

[5] Christoph Bartneck, Tony Belpaeme, Friederike Eyssel, Takayuki Kanda, Merel Keijsers, and Selma Šabanović. *Human-robot interaction: An introduction*. Cambridge University Press, 2020.

[6] Cynthia Breazeal. Toward sociable robots. *Robotics and autonomous systems*, 42(3-4):167–175, 2003.

[7] Adam Robaczewski, Julie Bouchard, Kevin Bouchard, and Sébastien Gaboury. Socially assistive robots: The specific case of the nao. *International Journal of Social Robotics*, 13:795–831, 2021.

[8] Amit Kumar Pandey and Rodolphe Gelin. A mass-produced sociable humanoid robot: Pepper: The first machine of its kind. *IEEE Robotics & Automation Magazine*, 25(3):40–48, 2018.

[9] Samer Al Moubayed, Jonas Beskow, Gabriel Skantze, and Björn Granström. Furhat: a back-projected human-like robot head for multiparty human-machine interaction. In *Cognitive Behavioural Systems: COST 2102 International Training School,*

Dresden, Germany, February 21-26, 2011, Revised Selected Papers, pages 114–130. Springer, 2012.

[10] https://furhatrobotics.com/. Accessed on May 24, 2023.

[11] Tony Belpaeme, James Kennedy, Paul Baxter, Paul Vogt, Emiel EJ Krahmer, Stefan Kopp, Kirsten Bergmann, Paul Leseman, Aylin C Küntay, Tilbe Göksun, et al. L2tor-second language tutoring using social robots. In *Proceedings of the ICSR 2015 WONDER Workshop*, 2015.

[12] https://www.engineeredarts.co.uk/robot/ameca/. Accessed on May 24, 2023.

[13] Sadeen Alharbi, Muna Alrazgan, Alanoud Alrashed, Turkiayh Alnomasi, Raghad Almojel, Rimah Alharbi, Saja Alharbi, Sahar Alturki, Fatimah Alshehri, and Maha Almojil. Automatic speech recognition: Systematic literature review. *IEEE Access*, 9:131858–131876, 2021.

[14] James A Kulik and JD Fletcher. Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of educational research*, 86(1):42–78, 2016.

[15] SA McLeod. What is the zone of proximal development? 2012.

[16] Radek Pelánek. Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques. *User Modeling and User-Adapted Interaction*, 27:313–350, 2017.

[17] Thorsten Schodde, Kirsten Bergmann, and Stefan Kopp. Adaptive robot language tutoring based on bayesian knowledge tracing and predictive decision-making. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, pages 128–136, 2017.

[18] Zachary Pardos, Yoav Bergner, Daniel Seaton, and David Pritchard. Adapting bayesian knowledge tracing to a massive open online course in edx. In *Educational Data Mining 2013*. Citeseer, 2013.

[19] Jussi Kasurinen and Uolevi Nikula. Estimating programming knowledge with bayesian knowledge tracing. *ACM SIGCSE Bulletin*, 41(3):313–317, 2009.

[20] Michael V Yudelson, Kenneth R Koedinger, and Geoffrey J Gordon. Individualized bayesian knowledge tracing models. In *Artificial Intelligence in Education: 16th*

International Conference, AIED 2013, Memphis, TN, USA, July 9-13, 2013. Proceedings 16, pages 171–180. Springer, 2013.

[21] Yumeng Qiu, Yingmei Qi, Hanyuan Lu, Zachary A Pardos, and Neil T Heffernan. Does time matter? modeling the effect of time with bayesian knowledge tracing. In *EDM*, pages 139–148, 2011.

[22] Qi Liu, Shuanghong Shen, Zhenya Huang, Enhong Chen, and Yonghe Zheng. A survey of knowledge tracing. *arXiv preprint arXiv:2105.15106*, 2021.

[23] Tony Belpaeme, James Kennedy, Aditi Ramachandran, Brian Scassellati, and Fumihide Tanaka. Social robots for education: A review. *Science robotics*, 3(21):eaat5954, 2018.

[24] Omar Mubin, Catherine J Stevens, Suleman Shahid, Abdullah Al Mahmud, and Jian-Jie Dong. A review of the applicability of robots in education. *Journal of Technology in Education and Learning*, 1(209-0015):13, 2013.

[25] Weijiao Huang, Khe Foon Hew, and Luke K Fryer. Chatbots for language learning—are they really useful? a systematic review of chatbot-supported language learning. *Journal of Computer Assisted Learning*, 38(1):237–257, 2022.

[26] Natasha Randall. A survey of robot-assisted language learning (rall). *ACM Transactions on Human-Robot Interaction (THRI)*, 9(1):1–36, 2019.

[27] Greg J Duncan, Chantelle J Dowsett, Amy Claessens, Katherine Magnuson, Aletha C Huston, Pamela Klebanov, Linda S Pagani, Leon Feinstein, Mimi Engel, Jeanne Brooks-Gunn, et al. School readiness and later achievement. *Developmental psychology*, 43(6):1428, 2007.

[28] Dale Walker, Charles Greenwood, Betty Hart, and Judith Carta. Prediction of school outcomes based on early language production and socioeconomic factors. *Child development*, 65(2):606–621, 1994.

[29] Tracy McKee Agostin and Sherry K Bain. Predicting early school success with developmental and social skills screeners. *Psychology in the Schools*, 34(3):219–228, 1997.

[30] Olusola O Adesope, Tracy Lavin, Terri Thompson, and Charles Ungerleider. A systematic review and meta-analysis of the cognitive correlates of bilingualism. *Review of educational research*, 80(2):207–245, 2010.

[31] Ellen Bialystok, Fergus IM Craik, and Morris Freedman. Bilingualism as a protection against the onset of symptoms of dementia. *Neuropsychologia*, 45(2):459–464, 2007.

[32] Albert Saiz and Elena Zoido. Listening to what the world says: Bilingualism and earnings in the united states. *Review of Economics and Statistics*, 87(3):523–538, 2005.

[33] Jacqueline Kory Westlund and Cynthia Breazeal. The interplay of robot language level with children's language learning during storytelling. In *Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction extended abstracts*, pages 65–66, 2015.

[34] Minoo Alemi, Ali Meghdari, and Maryam Ghazisaedy. Employing humanoid robots for teaching english language in iranian junior high-schools. *International Journal of Humanoid Robotics*, 11(03):1450022, 2014.

[35] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[36] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[37] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[38] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.

[39] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. Advances in pre-training distributed word representations. *arXiv preprint arXiv:1712.09405*, 2017.

[40] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

[41] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[43] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[44] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[45] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[46] OpenAI. Gpt-4 technical report, 2023.

[47] https://help.openai.com/en/articles/6825453-chatgpt-release-notes, 2022. Accessed on May 24, 2023.

[48] https://openai.com/blog/chatgpt, 2022. Accessed on May 24, 2023.

[49] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[50] Jorge Agnese, Jonathan Herrera, Haicheng Tao, and Xingquan Zhu. A survey and taxonomy of adversarial neural networks for text-to-image synthesis. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(4):e1345, 2020.

[51] Avijit Ghosh and Genoveva Fossas. Can there be art without an artist? *arXiv preprint arXiv:2209.07667*, 2022.

[52] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[53] https://openai.com/product/dall-e-2. Accessed on May 24, 2023.

[54] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021.

[55] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

[56] https://www.midjourney.com/home/. Accessed on May 24, 2023.

[57] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

[58] Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. Findings of the 2019 conference on machine translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy, August 2019. Association for Computational Linguistics.

[59] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[60] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 326–335, 2017.

[61] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015.

[62] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021.

[63] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs, 2019.

[64] Unnat Jain, Svetlana Lazebnik, and Alexander G Schwing. Two can play this game: Visual dialog with discriminative question generation and answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5754–5763, 2018.

[65] Abhishek Das, Satwik Kottur, José MF Moura, Stefan Lee, and Dhruv Batra. Learning cooperative visual dialog agents with deep reinforcement learning. In *Proceedings of the IEEE international conference on computer vision*, pages 2951–2960, 2017.

[66] Luke Prodromou. From mother tongue to other tongue. *Retrieved on August*, 20:2007, 2002.

[67] Manoj Kumar Yadav. Role of mother tongue in second language learning. *International Journal of research*, 1(11):572–582, 2014.

[68] Rachel Harding. *English for Everyone: Level 1: Beginner, Course Book*. Dk Publishing, 2016.

[69] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022.

[70] https://huggingface.co/runwayml/stable-diffusion-v1-5. Accessed on May 24, 2023.

[71] Albert T Corbett and John R Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4:253–278, 1994.

[72] Brett van De Sande. Properties of the bayesian knowledge tracing model. *Journal of Educational Data Mining*, 5(2):1–10, 2013.

[73] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2. Accessed on May 24, 2023.

[74] Victor U Thompson, Christo Panchev, and Michael Oakes. Performance evaluation of similarity measures on similar and dissimilar text retrieval. In *2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K)*, volume 1, pages 577–584. IEEE, 2015.

[75] https://huggingface.co/Helsinki-NLP/opus-mt-en-es. Accessed on May 24, 2023.

[76] https://azure.microsoft.com/en-us/products/cognitive-services/translator. Accessed on May 24, 2023.

[77] https://github.com/openai/gpt-3/tree/master/dataset_statistics. Accessed on May 24, 2023.

[78] https://prompthero.com/stable-diffusion-prompt-guide. Accessed on May 24, 2023.

[79] https://docs.python.org/3/library/tkinter.html. Accessed on May 24, 2023.

Appendices

Appendix A

The multiple choice test used as pre- and post-test for the students in the user study.

Duid de juiste Spaanse vertaling aan

Vergeet de achterkant niet.

Kleding en accessoires

1. **t-shirt**
   - pantalón
   - falda
   - chaqueta
   - sombrero
   - camiseta
2. **blouse**
   - blusa
   - camisa
   - impermeable
   - falda
   - vestido
3. **overhemd**
   - camisa
   - abrigo
   - pantalón
   - sombrero
   - zapatos
4. **jurk**
   - abrigo
   - bufanda
   - pantalón
   - vestido
   - calcetines
5. **rok**
   - impermeable
   - zapatillas
   - falda
   - bufanda
   - zapatos
6. **broek**
   - bufanda
   - impermeable
   - pantalón
   - sombrero
   - chaqueta
7. **spijkerbroek**
   - falda
   - zapatos
   - guantes
   - impermeable
   - jeans
8. **jasje**
   - chaqueta
   - sombrero
   - cinturón
   - bufanda
   - vestido
9. **jas**
   - falda
   - abrigo
   - calcetines
   - jeans
   - chaqueta
10. **regenjas**
    - falda
    - calcetines
    - vestido
    - blusa
    - impermeable
11. **sokken**
    - chaqueta
    - calcetines
    - zapatos
    - abrigo
    - vestido
12. **laarzen**
    - cinturón
    - chaqueta
    - jeans
    - botas
    - impermeable
13. **schoenen**
    - cinturón
    - falda
    - chaqueta
    - blusa
    - zapatos
14. **sandalen**
    - vestido
    - camiseta
    - sandalias
    - impermeable
    - bufanda
15. **sportschoenen**
    - blusa
    - sombrero
    - chaqueta
    - zapatillas
    - bufanda
16. **sjaal**
    - camisa
    - bufanda
    - falda
    - sandalias
    - zapatos

17. **hoed**
    - bufanda
    - jeans
    - guantes
    - sombrero
    - camisa
18. **handschoenen**
    - zapatos
    - guantes
    - botas
    - impermeable
    - bufanda
19. **riem**
    - camiseta
    - zapatillas
    - cinturón
    - vestido
    - guantes
20. **tas**
    - guantes
    - abrigo
    - botas
    - bolso
    - zapatillas

Kleuren

1. **blauw**
    - blanco
    - amarillo
    - rojo
    - negro
    - azul
2. **rood**
    - blanco
    - amarillo
    - rojo
    - negro
    - azul
3. **zwart**
    - blanco
    - amarillo
    - rojo
    - negro
    - azul
4. **wit**
    - blanco
    - amarillo
    - rojo
    - negro
    - azul
5. **geel**
    - blanco
    - amarillo
    - rojo
    - negro
    - azul

Appendix B

The questionnaire about the experience of the students during the user study and some demographic data.

Vragenlijst ervaring

1. Wat is je leeftijd? …….
2. Wat is je gender?
    - Man
    - Vrouw
    - Ander: ……..
    - Dat zeg ik liever niet
3. Hoe goed schat je je Spaans in? Geef een getal van 0 (ik spreek geen Spaans) tot 5 (Spaans is mijn moedertaal).

4. Heb je ooit al eens Spaans geleerd? Indien wel, waar, van wie en hoeveel?

5. Wat vond je van het spelletje? Was het gebruiksvriendelijk? Was het duidelijk wat je moest doen?

6. Het thema was ‘kleding en accessoires’. Vond je dit interessant? Waren dit woorden die je graag wou leren?

7. Als je het spel met de robot gespeeld hebt, wat vond je van de robot?

8. Heb je nog opmerkingen?