UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

Jayr Alencar Pereira

A Method for Adapting Large Language Models for Communication Card Prediction in Augmentative and Alternative Communication Systems

Recife

2023

Jayr Alencar Pereira

A Method for Adapting Large Language Models for Communication Card Prediction in Augmentative and Alternative Communication Systems

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação.

Área de Concentração: Inteligência Computacional

Orientador (a): Robson do Nascimento Fidalgo

Coorientador (a): Cleber Zanchettin

Recife

2023

Catalogação na fonte
Bibliotecária Nataly Soares Leite Moro, CRB4-1722

P436m Pereira, Jayr Alencar

A method for adapting large language models for communication card prediction in augmentative and alternative communication systems / Jayr Alencar Pereira. – 2023.

113 f.: il., fig., tab.

Orientador: Robson do Nascimento Fidalgo.

Tese (Doutorado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2023.

Inclui referências.

1. Inteligência computacional. 2. Comunicação aumentativa e alternativa. 3. Pranchas de comunicação. 4. Construção de frases. 5. Predição de pictogramas. 6. Colourful semantics. I. Fidalgo, Robson do Nascimento (orientador). II. Título

006.31 CDD (23. ed.) UFPE - CCEN 2023 –140

Jayr Alencar Pereira

“A Method for Adapting Large Language Models for Communication Card Prediction in Augmentative and Alternative Communication Systems”

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovada em: 18/07/2023.

____
Orientador: Prof. Dr. Robson do Nascimento Fidalgo

BANCA EXAMINADORA

____
Prof. Dr. Geber Lisboa Ramalho
Centro de Informática/ UFPE

____
Prof. Dr. Adriano Lorena Inacio de Oliveira
Centro de Informática/ UFPE

____
Prof. Dr. Filipe Carlos de Albuquerque Calegário
Centro de Informática / UFPE

____
Prof. Dr. Rodrigo Frassetto Nogueira
Faculdade de Engenharia Elétrica e de Computação / UNICAMP

____
Prof. Dr. André Carlos Ponce de Leon Ferreira de Carvalho
Instituto de Ciências Matemáticas e de Computação / USP

ACKNOWLEDGEMENTS

I would like to first thank God, as J.R.R. Tolkien said, the co-creator, for everything. He has allowed this to happen in my life during these past fuor years.

To my family. My wife Luana, who is always by my side in every moment, for all the support and trust. My children, Teresa, Catarina, João, and Mariana, bring new meaning to my life daily, teaching me things I never dreamed of. My mother supported and encouraged me from the beginning. My father, from whom I learned that life is tough but beautiful. My siblings, for their patience and support during this time. And also to my in-laws and brothers-in-law for all their support and help.

To my teachers, from kindergarten until now. Especially to my advisor Robson Fidalgo, who trusted me to complete this work and made me believe I could go further. To the Graduate Program in Computer Science professors at the Center for Informatics of UFPE, who have given me a new perspective on the world. Also, to the team at NeuralMind.ai, in the persons of professors Rodrigo Nogueira and Roberto Lotufo.

To my friends. Especially those I made during this period and those whose bonds of friendship have strengthened.

To the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) for the financial support.

Finally, to all those who, in some way, contributed for me to achieve this goal, my most sincere thanks!

ABSTRACT

Augmentative and Alternative Communication (AAC) systems assist individuals with complex communication needs to express themselves. Communication cards are a popular method used in AAC, where users select cards and arrange them in sequence to form a sentence. However, the limited number of cards displayed and the need to navigate multiple pages or folders can hinder users' communication ability. To overcome these barriers, various methods, such as vocabulary organization, color coding systems, motor planning, and predictive models, have been proposed to aid message authoring. Predictive models can suggest the most probable next cards based on prior input. Recent advancements in Artificial Intelligence (AI) and Machine Learning (ML) have shown potential for improving the accessibility and customization of AAC systems. This study proposes adapting large language models to communication card prediction in AAC systems to facilitate message authoring. The proposed method involves three main steps: 1) adapting a text corpus to the AAC domain by either converting it into a corpus of telegraphic sentences or incorporating features that enable the exploration of visual cues; 2) fine-tuning a transformer-based language model using the adapted corpus; and 3) replacing the language model decoder weights with an encoded representation of the user's vocabulary to generate a probability distribution over the user's vocabulary items during inference. The proposed method leverages that transformers-based language models, such as Bidirectional Encoder Representations from Transformers (BERT), share the weights of the input embeddings layer with the decoder in the language modeling head. Therefore, the plug-and-play method can be used without additional training for zero-shot communication card prediction. The method was evaluated in English and Brazilian Portuguese using a zero-shot setting and a few-shot setting, where a small text corpus was used for fine-tuning. Additionally, the impact of incorporating additional features into the training sentences by labeling them with the Colourful Semantics structure was assessed. The results demonstrate that the proposed method's models outperform models pre-trained for the task. Moreover, the results indicate that incorporating Colourful Semantics improves the accuracy of communication card prediction. Thus, the proposed method utilizes the transfer learning ability of transformers-based language models to facilitate message authoring in AAC systems in a low-effort setting.

Keywords: augmentative and alternative communication; message authoring; sentence construction; pictogram prediction; colourful semantics.

RESUMO

Os sistemas de Comunicação Aumentativa e Alternativa (CAA) auxiliam indivíduos com necessidades complexas de comunicação a se expressarem. Um recurso comum em CAA é o uso de cartões de comunicação, que o usuário pode selecionar e organizar em sequência para formar uma frase. No entanto, o número limitado de cartões exibidos e a necessidade de navegar por várias páginas ou pastas podem dificultar a construção de mensagens. Para superar essas barreiras, vários métodos foram propostos, como organização de vocabulário, sistemas de chaves de cores, planejamento motor e modelos preditivos. Os modelos preditivos podem sugerir os cartões mais prováveis para completar uma frase. Avanços recentes em Inteligência Artificial (IA) mostram potencial para melhorar a acessibilidade e a personalização dos sistemas de CAA. Este estudo propõe um método para adaptar modelos de linguagem para predição de cartões de comunicação em sistemas de CAA para facilitar a elaboração de mensagens. O método proposto envolve três etapas: 1) adaptar um *corpus* de texto ao domínio da CAA, convertendo-o em um corpus de frases telegráficas ou incorporando recursos que permitem a exploração de pistas visuais; 2) ajustar um modelo de linguagem baseado em *transformers* usando o *corpus* adaptado; e 3) substituir os pesos do decodificador do modelo de linguagem por uma representação codificada do vocabulário do usuário para gerar uma distribuição de probabilidade sobre os itens de vocabulário do usuário durante a inferência. O método proposto aproveita que modelos de linguagem baseados em *transformers*, como o *Bidirectional Encoder Representations from Transformers* (BERT), compartilham os pesos da camada de *embeddings* de entrada com o decodificador no cabeçalho de modelagem de linguagem. Portanto, o método pode ser usado sem treinamento adicional para a predição de cartões de comunicação. O método foi avaliado em Língua Inglesa e Língua Portuguesa do Brasil usando configurações *zero-shot* e *few-shot*, em que um pequeno corpus de texto foi usado para o ajuste fino. Além disso, foi avaliado o impacto da incorporação de recursos adicionais nas frases de treinamento, rotulando-as com a estrutura do *Colourful Semantics*. Resultados mostram que o método proposto supera modelos pré-treinados e que a inclusão de *Colourful Semantics* melhora a precisão da predição de cartões.

Palavras-chave: comunicação aumentativa e alternativa; pranchas de comunicação; construção de frases; predição de pictogramas; colourful semantics.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AAC	Augmentative and Alternative Communication
AI	Artificial Intelligence
ARASAAC	Aragonese Portal of Augmentative and Alternative Communication
ARES	Context-AwaRe Embeddings of Senses
AUROC	Area Under the Receiver Operating Characteristic
BERT	Bidirectional Encoder Representations from Transformers
BERTimbau	Pretrained BERT Models for Brazilian Portuguese
brWaC	Brazilian Web as Corpus
CACE	Augmentative Communication and Environment Control
CCN	Complex Communication Needs
CHILDES	Child Language Data Exchange System
CS	Colourful Semantics
DL	Deep Learning
GPT	Generative Pre-trained Transformer
InVeRo	Intelligible Verbs and Roles
LLM	Large Language Model
LM	Language Model
ML	Machine Learning
MLM	Masked Language Modeling
MRR	Mean Reciprocal Rank
MWE	Multi Word Expression
NLP	Natural Language Processing
OOV	out-of-vocabulary
PODD	Pragmatic Organization Dynamic Displays
POS	Part-Of-Speech

PrAACT	Predictive Augmentative and Alternative Communication with Transformers
RAPT	Renfrew Action Picture Test
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
RQ	Research Question
SemCHILDES	Semantic CHILDES
SO	Specific Objective
SRL	Semantic Role Lebaling
StArt	State of the Art through Systematic Review
SVO	Subject-Verb-Object
TAM	Technology Acceptance Model
UTAC	Unitat de Tècniques Augmentatives de Comunicació
ViT	Vision Transformer
VOCAS	Voice Output Communication Aids

CONTENTS

1 INTRODUCTION

This chapter presents this thesis, highlighting its context, motivation, objectives, research questions, scope delimitation, target audience, and contributions. Finally, the structure of the other chapters is presented.

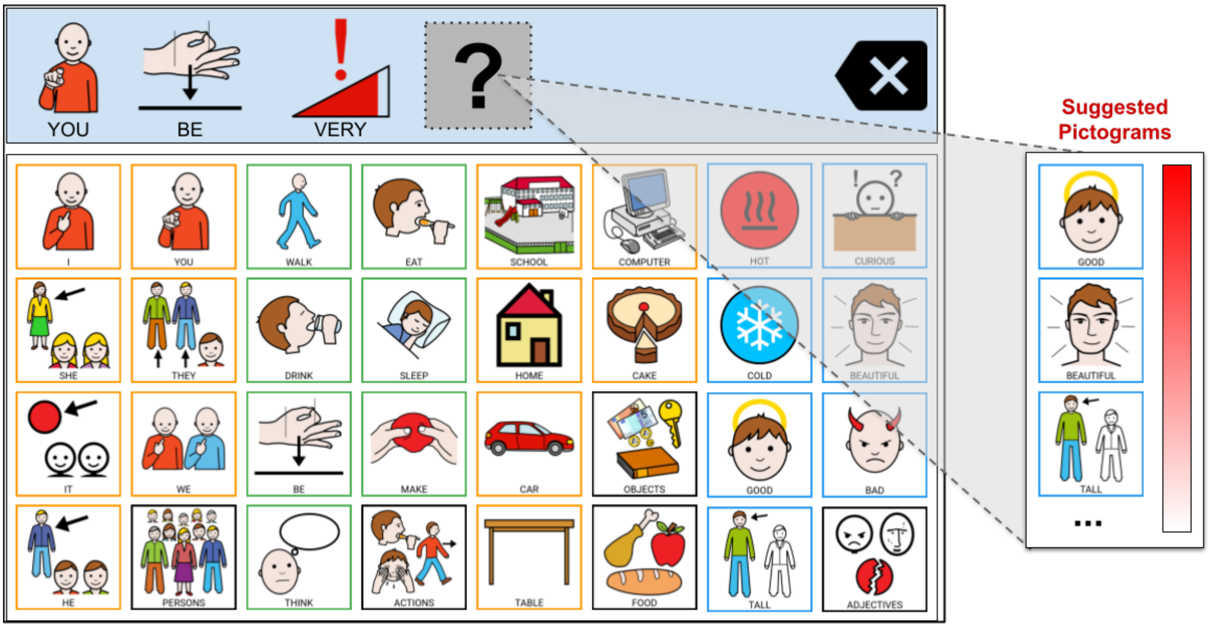
1.1 CONTEXT AND MOTIVATION

The field of Augmentative and Alternative Communication (AAC) aims to provide individuals with Complex Communication Needs (CCN), as those resulting from conditions involving autism, cerebral palsy, and developmental disabilities, with methods to supplement or replace spoken language. These methods can include communication boards, sign language, and speech-generating devices. The goal of AAC is to enable individuals to effectively communicate their wants, needs, and ideas (BEUKELMAN; LIGHT, 2020; American Speech-Language-Hearing Association, n.d.). A common approach in AAC is using communication cards, also known as pictograms, which are graphical representations of concepts, such as actions, objects, people, animals, descriptions, or places, that can be selected and arranged in sequence to form a sentence. An example of an AAC system is illustrated in Figure 1. These systems typically include a content area displaying the available cards for selection and a phrase area displaying the selected cards arranged to form the sentence. They have been shown to enable children and adults with CCN to communicate and participate in a wide range of environments and activities (MCNAUGHTON et al., 2019; CHUNG; CARTER, 2013).

The use of AAC boards for communication by individuals with CCN has been found to present certain barriers or difficulties, as previous research highlighted (PEREIRA et al., 2019; DONATO; SPENCER; ARTHUR-KELLY, 2018; JUDGE; TOWNEND, 2013; BAXTER et al., 2012). To effectively support message construction, these systems need to facilitate card selection, for example, through strategies such as paging or organizing cards into categories, as illustrated in Figure 1. However, these strategies can also present challenges. The user's vocabulary may not fit within the limited cards displayed on the first screen, and the need to navigate multiple pages or categories can make communication more difficult. Additionally, cards irrelevant to the desired message can distract the user during the search process.

In AAC, strategies to facilitate message authoring through card selection have been pro-

Figure 1 – Example of AAC system using a predictive communication card suggestion model. Predictive models may act on the background of these systems to suggest cards to complete sentences in construction. Notice that the cards have different border colors. This is a color coding approach, usually used in AAC systems, in which the cards are labeled according to their part of speech or function (e.g., green for verbs). The cards with black borders are folders that contain sub-cards (e.g., actions).



Source: Pereira et al. (2022)

posed to support individuals with CCN, such as cerebral palsy or autism. These strategies include vocabulary organization methods, color coding systems, motor planning approaches, and predictive models (FRANCO et al., 2018). Predictive models can suggest the most probable next cards based on prior input, as illustrated in Figure 1. These models receive the sentence in construction as input and return the most probable cards to complete the sentence.

Recent advancements in Artificial Intelligence (AI) and Machine Learning (ML) have significantly improved high-tech AAC systems. As outlined by Elsahar et al. (2019), the integration of AI in AAC systems can enhance accessibility to high-tech devices, increase the speed of output generation, and improve the customization and adaptability of AAC interfaces. Sennott et al. (2019) also highlights the potential benefits of incorporating AI into AAC systems, specifically mentioning the use of Natural Language Processing (NLP) techniques for tasks such as word and message prediction, automated storytelling, voice recognition, and text expansion. The incorporation of AI in AAC systems raises questions about the potential use of AI to assist in the creation of grammatically correct, semantically meaningful, and comprehensive messages within these systems.

In AAC, card prediction is typically treated as a NLP problem. Previous studies on this task have employed techniques such as n-gram models (HERVÁS et al., 2020; GARCIA; OLIVEIRA;

MATOS, 2016), or knowledge bases (PEREIRA; FRANCO; FIDALGO, 2020; MARTíNEZ-SANTIAGO et al., 2015). However, the main disadvantage of these previous works is that they freeze the vocabulary after training the models. For example, the vocabulary of an n-gram model is based on the words (or expressions) that occur in the training corpus. This means that different AAC users with different vocabularies may be unable to use these models effectively. The vocabulary of an AAC user can vary depending on many factors, such as age, gender, culture, and personal preferences. Moreover, AAC users can acquire new words and increase their verbal repertoire (LORAH et al., 2015). This poses a challenge for models that are based on fixed vocabularies. When a new word needs to be added to the model's vocabulary, retraining the entire model can be computationally expensive and time-consuming. Therefore, there is a need for adaptive models that can efficiently incorporate new words into the existing vocabulary without the need for extensive retraining.

Recent advancements in NLP have shifted towards using Large Language Models (LLMs) based on transformer architecture (VASWANI et al., 2017). These have demonstrated state-of-the-art performance in a wide range of NLP tasks. Language Models (LMs) like Bidirectional Encoder Representations from Transformers (BERT) (DEVLIN et al., 2019), and Generative Pre-trained Transformer (GPT) series (RADFORD et al., 2019; BROWN et al., 2020; OPENAI, 2023) can be used for card prediction as they can understand the context and provide accurate predictions. These models are trained on large amounts of text data, allowing them to understand the relationships between words and phrases, which can aid in predicting appropriate pictograms. Furthermore, these LLMs can also be used in few- or zero-shot settings for AAC card prediction. In few-shot learning, the model is trained on a small amount of data. Zero-shot learning involves applying the model to a task or domain it has not seen during training without additional training data (XIAN; SCHIELE; AKATA, 2020). The advantage of using these models in few- or zero-shot settings is that they can generalize to new or unseen situations. This reduces the need for collecting and annotating large amounts of data, which can be time-consuming and expensive. Instead, the models can leverage the knowledge learned from their pre-training on large text corpora to make accurate predictions in new contexts.

Another strategy for facilitating message authoring in AAC systems is the use of color coding systems, such as the Fitzgerald Keys (FITZGERALD, 1949) or the Colourful Semantics (CS) system (BRYAN, 2003). These systems group cards according to their grammatical or semantic role, respectively. CS is a therapeutic tool that employs colors and questions such as *Who?*, *What Doing?*, and *What?* to facilitate sentence construction and promote understanding of well-structured and meaningful sentences in natural language among children with CS. Previous studies have demonstrated the effectiveness of using this tool in treating children with speech difficulties. For instance, (BRYAN, 2003) reported that after eight weeks of treatment, a child could identify and use semantic roles in constructing phrases while narrating a story and demonstrated the ability to construct more complex sentences after a few months. Similarly, Bolderson et al. (2011), and Christopoulou et al. (2021) observed improved participant communication performance. Despite its effectiveness, there is a lack of AAC systems that integrate CS for communication.

Given this context, the motivation for this work is driven by the following points:

1. **Enhancing AAC Systems**: Integrating advanced AI techniques, such as transformer-based LMs, can enhance AAC systems by providing more accurate and context-aware card predictions. This can improve the efficiency and effectiveness of message authoring, facilitating communication for individuals with CCN.

2. **Adaptive Vocabulary Handling**: Existing AAC systems often struggle to accommodate individual users with unique vocabularies. By leveraging transformer-based LMs, we aim to develop models that efficiently incorporate new words into the vocabulary without extensive retraining. This adaptive vocabulary handling will enable AAC systems to adapt to the evolving communication needs of users.

3. **Few-shot and Zero-shot Learning**: The ability of transformer-based LMs to generalize to new or unseen situations in few-shot or zero-shot settings can significantly reduce the burden of data collection and annotation in AAC system development. Leveraging pre-trained models can save time and resources while providing accurate predictions in diverse contexts.

4. **Integration of Colourful Semantics**: Integrating CS into AAC systems can promote the development of well-structured and meaningful sentences in natural language. Incorporating CS principles can enhance message construction's syntactic and semantic aspects in AAC systems.

By addressing these motivations, we aim to contribute to advancing AAC systems, making communication more accessible and efficient for individuals with CCN.

1.2 OBJECTIVES AND RESEARCH QUESTIONS

The main objective of this study is to propose a method for communication card prediction in AAC systems using LLMs to enhance message authoring by accommodating the variable vocabularies of AAC users. To reach this primary objective, we have the following Specific Objectives (SOs):

- SO-1: Propose and assess the performance of a transformer-based neural network model for enhancing communication card prediction in high-tech AAC systems.
- SO-2: Determine the proposed method's efficacy with minimal or no additional training.
- SO-3: Evaluate the effectiveness of integrating Colourful Semantics (CS) into communication card prediction models for AAC systems and compare the performance of these models to those that do not incorporate CS.

To accomplish these objectives, this work is guided by the following Research Questions (RQs):

- RQ-1: How can a transformer-based neural network be adapted to improve communication card prediction in AAC systems, considering the variability of users' vocabularies?
- RQ-2: What is the effectiveness of the proposed method for adapting transformer-based neural networks in terms of communication card prediction accuracy, and how does it compare to models that require additional training?
- RQ-3: Does incorporate Colourful Semantics (CS) into communication card prediction models for AAC systems improve their accuracy, and how does this improvement compare to models without CS?

1.3 SCOPE DELIMITATION

This work is delimited into four aspects:

1. *Language* – The experiments in this work are performed in Brazilian Portuguese and English;

2. *Experimental design* – At this stage, the models are compared using intrinsic metrics like Top-n accuracy, Mean Reciprocal Rank (MRR) and Entropy@K, which can be automatically computed;

3. *Type of AAC systems* – this work focus on *high-tech* AAC systems based on pictographic symbols, as shown in Figure 1.

4. *Target audience* – formed by children with complex communication needs who cannot write conventionally nor use a conventional keyboard (e.g., QWERTY) to communicate. Regarding written language, the public of this proposal can or not be literate. In the case of a literate child, cognitive deficits may compromise the use of written language, and, in this case, AAC is seen as a complementary resource (i.e., facilitator of communication). In the case of a non-literate child, AAC is considered an alternative resource for communication because it is based on a graphics system;

1.4 DOCUMENT STRUCTURE

The remaining chapters of this proposal are structured as follows:

Chapter 2 – Background: presents the theoretical foundations necessary to support this work

Chapter 3 – Related Work: presents the works related to this research.

Chapter 4 – Methodology: presents the method proposed in this research and the experiments performed.

Chapter 5 – Experiments: presents the details of the performed experiments.

Chapter 6 – Results: presents the preliminary results.

Chapter 7 – Conclusions: presents the final considerations on the main topics covered in this proposal, including the contributions achieved and the indications of future works.

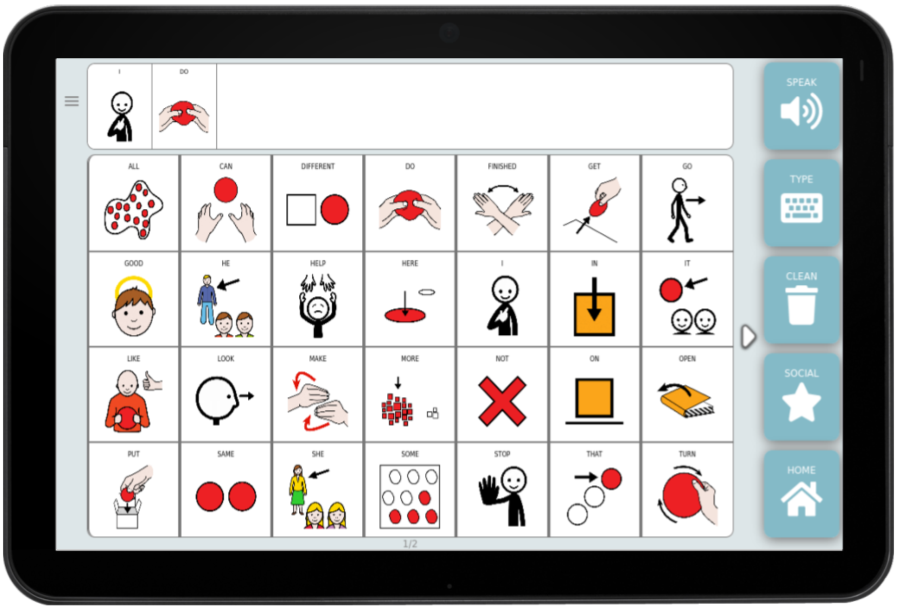
2 BACKGROUND

2.1 AUGMENTATIVE AND ALTERNATIVE COMMUNICATION (AAC)

According to Beukelman and Light (2020), approximately 97 million people in the world may benefit from AAC. These people constitute a heterogeneous population regarding diagnosis, age, location, communication modality, and extent of AAC use (American Speech-Language-Hearing Association, n.d.). They generally have limitations on gestures, oral, or written communication, causing functional communication and socialization problems. AAC users include not just people with CCN but also children who are at risk for speech development, individuals who require AAC to supplement or clarify their speech, and individuals who require AAC to support comprehension (e.g., those with degenerative cognitive and linguistic disorders such as Alzheimer's disease), and those with temporary conditions (BEUKELMAN; LIGHT, 2020).

Considering the external support that can be used on AAC interventions, non-technological systems are often referred to as *low-tech* (e.g., paper-craft cards, objects, and communication books), whereas technological systems are referred to as *high-tech* (e.g., speech-generating devices, or AAC applications installed in smartphones or tablets). The use of *high-tech* AAC systems helps the user to express feelings and opinions, develop understanding, reduce frustration in trying to communicate, and have a greater power of choice (BEUKELMAN; LIGHT, 2020). These systems allow users to construct sentences by selecting communication cards from a grid and arranging them in sequence. Figure 2 presents an example of *high-tech* AAC system with a content grid (bottom large rectangle), and a sentence area (tiny top rectangle), where cards are arranged in sequence. High-tech AAC systems have gained ground recently. The advent of mobile devices like iPad, iPhone, and Android smartphones and tablets facilitated the release of low-cost systems (LORAH; TINCANI; PARNELL, 2018; LORAH et al., 2022). By searching in the Apple App Store and Google Play Store for "alternative communication", we can find a variety of applications for AAC. Most apps promote communication using communication cards, similar to the one shown in Figure 2. Studies have demonstrated the positive effect of these devices' usage by people with CCN (HOLYFIELD; LORAH, 2022; HUGHES; VENTO-WILSON; BOYD, 2022). Holyfield and Lorah (2022) showed that using *high-tech* AAC is more pleasant for children with multiple disabilities versus *low-tech* and that the communication using the platform the children prefer (i.e., *high-tech*) may be more efficient.

Figure 2 – Example of *high-tech* AAC system using pictograms. Generally, these systems contain (1) a content area (large rectangle at the bottom) with the available pictograms for selection and (2) a phrase area (tiny rectangle at the top) that presents the selected pictograms arranged to form the sentence.



Source: The author (2023)

2.2 MESSAGE AUTHORING IN AAC

AAC uses various tools and techniques to support the communication of individuals with CCN. In the case of *high-tech* aided AAC, the pictographic images in the communication cards act as visual support to the user. The picture gives meaning to the words in the user's vocabulary. Such pictograph systems are applied to individuals who are illiterate because of age or disability and allow communication for people with low cognitive levels or at very early stages (PALAO, 2019). Many pictogram databases are available online that can be used in such systems. We can cite the ARASAAC(PALAO, 2019) database, which makes more than 30 thousand pictograms available.

Many of the available *high-tech* AAC systems organize the pictograms in grids, as shown in Figure 2. The vocabulary organization depends on the user's needs and preferences. Some may use categories to organize the cards, while others prefer multiple pages. Anyway, such systems must allow and facilitate card selection for sentence construction (FRANCO et al., 2018). Among the strategies that can be used to facilitate message authoring in AAC, we can list the vocabulary organization, the usage of color coding systems, and the usage of a word, card, or pictogram prediction technique.

2.2.1 Vocabulary Organization

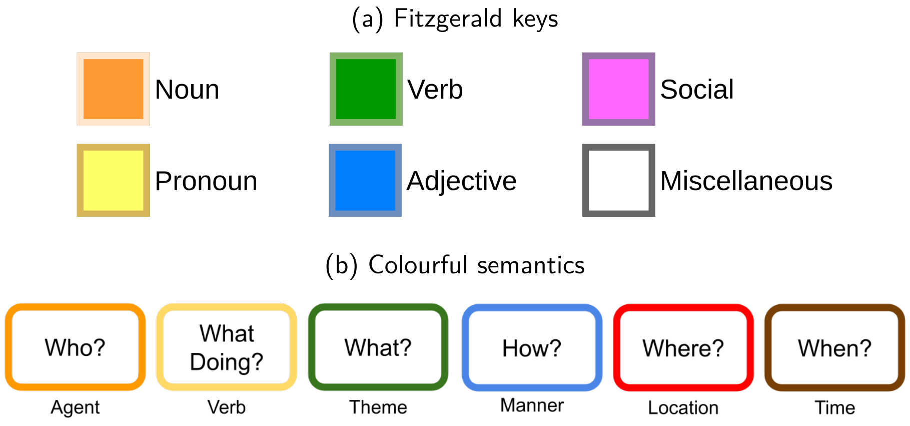
An initial strategy for facilitating card selection consists of organizing the system grid. There are different approaches to grid organization (BEUKELMAN; LIGHT, 2020): 1) **schematic or activity organization** – the symbols are organized according to event schema within the individual's day (e.g., routines, activities); 2) **taxonomic organization** – involves grouping symbols according to superordinate categories (e.g., the cat is an animal); 3) **semantic-syntactic organization** – the symbols are organized according to the part of speech and their semantic relationships; 4) **pragmatic organization** – Pragmatic Organization Dynamic Displays (PODD) (PORTER, 2007) combines different vocabulary organization strategies, such as activity and taxonomic organization in addiction to navigational features (e.g., go back, go forward); 5) **alphabetical organization** – cards organized in sequence by their labels like a personal dictionary; 6) **chronological organization** – often include a single column or row to represent the sequence or chronology of activities (e.g., brushing teeth, having breakfast); and 7) **idiosyncratic organization** – considers that each user may have a personal organization approach that may or not include some elements of the other approaches.

2.2.2 Color Coding Systems

According to Franco et al. (2018), pictogram selection in a robust AAC may include color-coding systems and a motor planning protocol. The most used color coding system is a modification (MCDONALD; SCHULTZ, 1973) of the Fitzgerald Key (FITZGERALD, 1949). The system groups pictograms into six colors regarding their Part-Of-Speech (POS) or grammatical role, as shown in Figure 3a. In addition to using colors, the system suggests organizing the pictograms from left to right according to their POS or role (BEUKELMAN; LIGHT, 2020).

Another color coding system is the CS (BRYAN, 2003). CS is a therapeutic tool developed to help children with CCN develop the construction and understanding of written or spoken sentences. The purpose of this system is to support the development of syntactic structures through a semantic script (HETTIARACHCHI, 2015). The script is composed of a color key system associated with key questions (i.e., Who? What Doing? What? Where? What Like?) that help the individual to understand the semantic role of each constituent of a phrase, as illustrated in Figure 3b. Colors act as a visual aid to indicate the grammatical structure. While questions help to link this structure (syntactic) to its meaning (semantics) (LAW et al., 2012).

Figure 3 – Color coding systems. (a) the Fitzgerald Keys system, which color cards according to their grammatical class (e.g., noun). (b) Colourful Semantics, which color cards according to the role they can have in a sentence.



(a) Fitzgerald keys

(b) Colourful semantics

Source: The author (2023)

CS differs from other color coding systems by identifying the semantic roles of the constituents of a sentence, which are more significant than the syntactic functions (i.e., subject, verb, and objects) for individuals with language difficulties (BOLDERSON et al., 2011). A semantic role is a property that denotes the role played by a word or phrase concerning the predicate it modifies in a sentence. For example, in the sentence "The boy ate popcorn", "boy" is the Agent of the verbal predicate "ate", while "popcorn" is the Theme. In CS, the roles Agent, Theme, Recipient, Manner, Description, Place, and Time are used. According to Bryan (2003), these roles are associated with colors and questions with the intention of: (i) make visual discrimination between each semantic role; (ii) further establish the relationship between the question and the semantic role; (iii) associate each type of phrase with a visual sequence of colors; and (iv) alert the child when he omitted a semantic role. The author used this tool to treat a 5-year-old child who had difficulties planning sentences and ordering and remembering words. Her goals during this treatment were: 1) to teach the identification of semantic roles in written sentences; and 2) encourage the use of knowledge of semantic roles and their functions to create sentences with the following predicate-argument structures: a) verb+agent+theme; b) verb+agent+place; c) verb+agent+theme+place; and d) verb+theme+description.

Several studies demonstrate the effectiveness of using CS in treating children with speech difficulties. In the first application made by Bryan (2003), for example, after eight weeks of treatment, the child was able to identify and use semantic roles in the construction of sentences during storytelling, and after a few months, she could notice an advance in the construction of more complex sentences than those initially taught. Bolderson et al. (2011)

applied CS to 6 children aged between 5 and 6 years for nine weeks. After this period, there was a significant increase in the average length of sentences produced by the children and in the metrics extracted from the Renfrew Action Picture Test (RAPT) (RENFREW, 2016) tests. Recently, we assessed the acceptance of caregivers to using a *high-tech* AAC system based on CS with their students, patients, or children (PEREIRA; PEREIRA; FIDALGO, 2021). The proposed system uses CS as a script for guiding sentence construction. The proposal was evaluated using the Technology Acceptance Model (TAM) (DAVIS, 1985). The results demonstrate that caregivers recognize the usefulness of such a proposal.

2.2.3 Card Prediction

Card prediction can facilitate message authoring in AAC systems by suggesting relevant communication cards as the user selects the cards to compose a sentence. This can save the user time and effort and improve the overall usability of the AAC system. According to Beukelman and Light (2020), such prediction techniques may offer many potential benefits to AAC users: 1) reduce the number of selections required to construct a sentence, thereby decreasing the effort for individuals; 2) provide spelling support for users who cannot accurately spell words; 3) provide grammatical support; and 4) may increase communication rate. The literature presents a growing number of published studies that use computational resources and techniques to perform pictogram or word prediction in AAC systems, driven by the increasing use of AI in the field (SENNOTT et al., 2019).

2.3 LANGUAGE MODELING

A LM is a model that assigns probabilities to sequences of words (JURAFSKY; MARTIN, 2019). Consider the sentence “Brazil is a beautiful ____” and wonder what the best word to complete it is. Most people will choose words like “country”, “place”, or “nation”, for they are the most probable among those that make sense. This human decision is so natural that we do not think about how it happens. However, regarding LMs, deciding what word to use to complete a sentence depends on the probabilities learned from a training text corpus. For example, for an n-gram LM, the most probable word is the one that occurs most frequently following the word “beautiful” in the training corpus. The same model can also assign a probability to an entire sentence and predict that the sentence “Brazil is a beautiful country” has a higher probability of appearing in a text corpus than the same words in a different order.

An n-gram LM is the simplest model that assigns probabilities to sentences and sequences of words (JURAFSKY; MARTIN, 2019). The aim is to predict the next word based on the $n-1$ preceding words. The model uses relative frequency counts to estimate the probability of each word in a vocabulary V to be the next in the sequence h. Given a large text corpus, count the number of times the sequence h is followed by the word $w \in V$. This way, in a bi-gram model ($n = 2$), the probability of the word "country" completing the sequence "Brazil is a beautiful ____" is defined by the equation 2.1, where C is the function that counts the occurrence of words or sequences in the corpus. Since this is a bi-gram model, only the last preceding word is considered in the equation, which can be simplified to $P(country|beautiful)$ or $P(w_n|w_{n-1})$.

$$P(country|Brazil\ is\ a\ beautiful) = \frac{C(beautiful\ country)}{C(beautiful)} \tag{2.1}$$

The probability of an entire sequence can be estimated using the chain rule:

$$\begin{aligned} P(w_{1:n}) &= P(w_1)P(w_2|w_1)P(w_3|w_{1:2})...P(w_n|w_{1:n-1}) \\ &= \prod_{k=1}^{n} P(w_k|w_{1:k-1}) \end{aligned} \tag{2.2}$$

The assumption that the probability of the next word depends only on the previous word is called Markov assumption (JURAFSKY; MARTIN, 2019). Markov models assume it is possible to predict the probability of a future unit (e.g., next word) by looking only at the current state (e.g., last preceding word). However, language is a continuous input stream highly affected by the writer/speaker's creativity, vocabulary, language development level, etc. If we ask two different persons to describe the same scene from a picture in a single sentence, there is a probability of both constructing sentences with a similar sense but using different words or ordering them differently. Besides, in a written text, the occurrence of a specific word may depend not only on the $n-1$ preceding but on the entire context, which can be the sentence, the paragraph, all the text, or the text's aim or topic. Still, n-gram models produce strong results for relatively small corpora and was the dominant LM approach for decades (GOLDBERG; HIRST, 2017). Among the LMs that do not make the Markov assumption, we can highlight those based on Recurrent Neural Networks (RNNs) (ELMAN, 1990) and on the Transformers architecture (VASWANI et al., 2017), presented in Section 2.5. Both may rely on the usage of word embeddings (cf. Section 2.4) for feature extraction.

2.4 WORD EMBEDDINGS

Word embeddings is a method to represent words using real-valued vectors to encode their meaning, assuming that words with similar meanings may be closer to each other in the vector space (JURAFSKY; MARTIN, 2019). Mikolov et al. (2013a) proposed the skip-gram model (a.k.a. *word2vec*), which learns high-quality vector representations of words from large amounts of text. The quality of the learned vectors allows similarity calculations between words and even operations such as $King - Man + Woman = Queen$, or $Madrid - Spain + France = Paris$. This means that by subtracting the vector of the word *Man* from the vector of the word *King* and summing it with the vector of the word *Woman*, the result vector is closer to the vector of the word *Queen* than any other vector (MIKOLOV; YIH; ZWEIG, 2013; MIKOLOV et al., 2013a). These vectors can also capture synonymy with quality, for words with similar meanings can have similar vector representations.

The Skip-gram model's training objective is to find word vectors useful for predicting the surrounding words in a sequence or a document (MIKOLOV et al., 2013b). This way, the model is trained using a self-supervised approach, which avoids the need for any hand-labeled supervision signal (JURAFSKY; MARTIN, 2019). Given a sequence of words $w_1, w_2, ..., w_n$, the model attempts to maximize the average log probability calculated according to Equation 2.3), where c is the training context size of words that are surrounding the center word w_t. A large c results in more training examples and can result in a high accuracy but may require more training time (MIKOLOV et al., 2013b). The basic Skip-gram formulation defines $P(w_{t+j}|w_t)$ using the softmax function, as in Equation 2.4, where v_w and $v_w^{'}$ are the input and output vectors of w, and W is the vocabulary size. This formulation is impractical for the cost of computing the gradient of $logP(w_O|w_I)$ is proportional to the vocabulary size, which can be large. Mikolov et al. (2013b) suggests using the hierarchical softmax (MORIN; BENGIO, 2005) as an efficient approximation of the full softmax. This way, the neural network behind skip-gram learns the best vector representation for each word in a vocabulary. The final model output is a dictionary with $\{word : vector\}$ pairs.

$$\frac{1}{n}\sum_{t=1}^{n}\sum_{-c\leq j\leq c, j\neq 0} logP(w_{t+j}|w_t) \tag{2.3}$$

$$P(w_O|w_I) = \frac{\exp({v_{w_O}^{'}}^{\top} v_{w_I})}{\sum_{w=1}^{W}\exp({v_w^{'}}^{\top} v_{w_I})} \tag{2.4}$$

There are a set of other word embeddings approaches with the same aim: to provide vector representation to words. We can classify skip-gram as a model that provides static embeddings, for the representation of a word will be the same indifferently of the context it occurs. For example, the word *bat* has a different meaning in the sentences *He can't bat the ball* and *Batman dress like a bat*. However, in a static word embedding model, it has the same vector. The transformers architecture (VASWANI et al., 2017) overcomes this problem by adding context to the embeddings. In Section 2.5, we present the architecture and how it can be used to produce contextualized word embeddings.

Scarlini, Pasini and Navigli (2020) proposed Context-AwaRe Embeddings of Senses (ARES) as a semi-supervised approach to producing sense embeddings for all the word senses in WordNet. WordNet (MILLER, 1995) is a lexical database that groups nouns, verbs, adjectives, and adverbs into sets of synonyms (a.k.a. synsets). Each synset expresses a distinct concept with its glossary definition and lexical relationships (e.g., meronym, hyperonym, and hyponym). The linking between the synset and the words it groups is defined by word senses. In WordNet, a word-sense is identified by a sense-key, for example *person%1:03:00::*. The sense-key is represented by *lemma%lex_sense*, where *lemma* is the text of the word or collocation as found in the WordNet (e.g., *person* or *playing_period*). And lex_sense is encoded as *ss_type:lex_filenum:lex_id:head_word:head_id*, where *ss_type* is the synset type, that corresponds to its part-of-speech (1 for nouns, 2 for verbs, 3 for adjectives, 4 for adverbs, and 5 for satellite adjectives[1]), *lex_filenum* represents the name of the lexicographer file containing the synset, *lex_id* is a two-digit decimal integer that, when appended onto lemma, uniquely distinguishes a sense within a lexicographer file, *head_word* is the lemma of the first word of a satellite's head synset, and *head_id* is a two-digit decimal integer that, when appended onto head_word , uniquely distinguishes the sense of *head_word* within a lexicographer file, the same as *lex_id*. More information can be found in the WordNet Documentation[2].

For each word-sense in WordNet, the ARES construction method finds its occurrences in a text corpus and computes its embeddings using BERT, which considers all the context (i.e., the entire sentence) for producing the word embeddings. As a word sense may occur in more the one sentence, the authors average the embeddings BERT produced. Besides, the method also computes the embedding representation of the word senses' glossary definition using BERT.

[1] In WordNet, adjectives are arranged in clusters containing head and satellite synsets. For example, the synset for *gone* (definition: *no longer retained*) is satellite for *lost* (definition: *no longer in your possession or control*).

[2] <https://WordNet.princeton.edu/documentation/senseidx5wn>

For this, the authors average the BERT representations for the words in the sense definition. For example, the representation for word-sense *person%1:03:00::* is the average vector of the representation produced by BERT for each token in its definition: *a human being*. The final representation is a 2048 dimensions real-valued vector, which we can divide into two sides: 1) a contextualized (first 1024 positions), with vectors extracted from the usage examples, and 2) a gloss-based (last 1024 positions), with vectors computed from glossary definition.

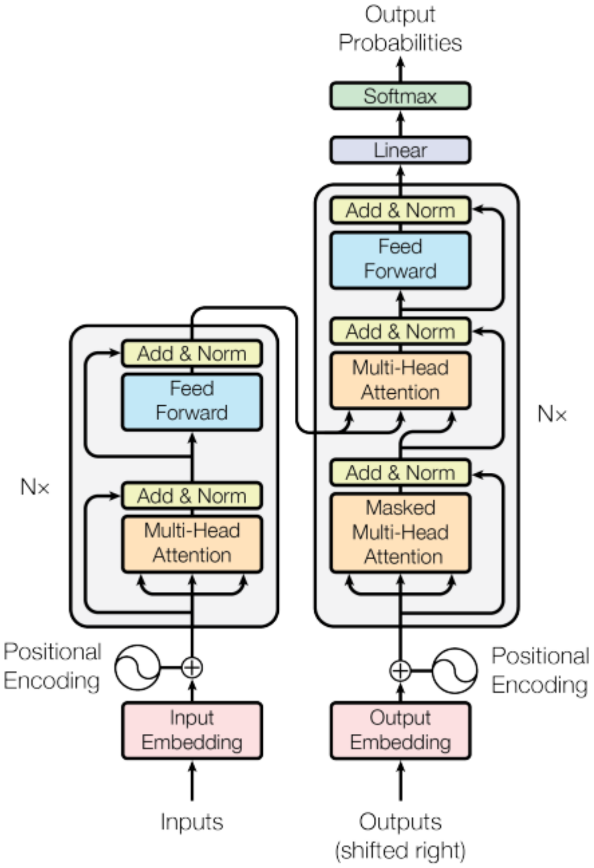
2.5 TRANSFORMERS

The Transformers architecture is a neural network model that has become increasingly popular in NLP tasks due to its impressive performance. The architecture was introduced in the paper "Attention is All You Need" by Vaswani et al. (2017). It uses a combination of self-attention and feedforward neural networks to process sequential data, such as sentences or documents, and extract useful features from them. The Transformer architecture has been used in many applications, including machine translation, language modeling, and text classification.

The key feature of the Transformer architecture is the attention mechanism, which allows the model to focus selectively on specific parts of the input sequence while processing it. The attention mechanism computes a weighted sum of the input sequence, where the relevance of each input element to the model's current state determines the weights. In other words, the attention mechanism learns to assign different importance to different parts of the input sequence, depending on the context. The self-attention mechanism in Transformers is particularly powerful because it allows the model to attend to any part of the input sequence, not just the adjacent elements. This means the model can capture long-range dependencies between the sequence elements, particularly useful for NLP tasks (JURAFSKY; MARTIN, 2019).

The Transformer architecture consists of an encoder and a decoder, as show in Figure 4. The encoder processes the input sequence while the decoder generates the output sequence. Both the encoder and decoder are composed of a stack of identical layers, each containing two sub-layers: a multi-head self-attention mechanism and a feedforward neural network. The self-attention mechanism in each layer allows the model to attend to different parts of the input sequence at different positions. At the same time, the feedforward network processes the output of the self-attention mechanism to produce the final output of the layer.

Figure 4 – The Transformers architecture. Transformers exclusively utilize self-attention mechanisms to compute input and output representations. The architecture's attention mechanism allows it to capture long-range dependencies within sequences efficiently, enabling highly parallelizable computations.



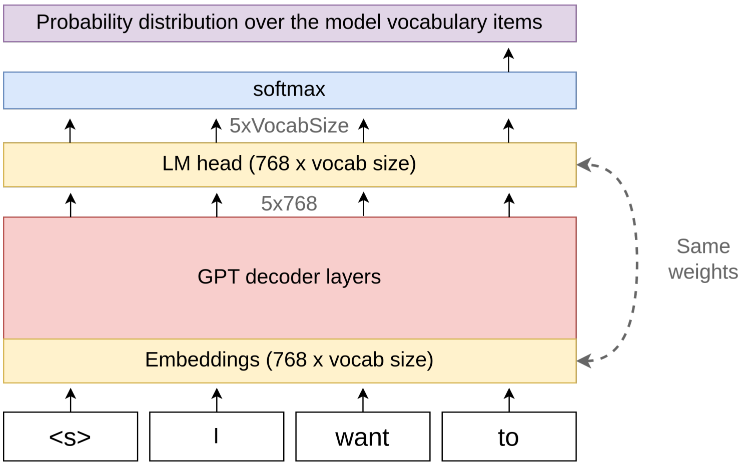
Source: Vaswani et al. (2017)

2.5.1 GPT

GPT (RADFORD et al., 2018; RADFORD et al., 2019; BROWN et al., 2020) is an auto-regressive generative language model that stands for Generative Pre-trained Transformer. This model uses the Transformers architecture to learn word representation that transfers with little adaptation to a wide range of tasks (RADFORD et al., 2018). The main task is to predict the next word in a given sequence and then learn the best vectorial word representations. These representations perform downstream tasks like sentiment analysis, machine translation, etc.

Figure 5 shows the basic architecture of GPT. The text input passes throughout an embedding layer to be transformed into real-valued vectors, which are then input into the transformer blocks. The base version of the model uses a 768-dimensional state for word embeddings. The model vocabulary is a Byte Pair Encoding vocabulary that splits words into subwords. A 12-layered model was used with 12 attention heads in each self-attention layer, i.e., they used 12 transformer blocks. The output of the last block is used as input for the downstream task

Figure 5 – Overview of the GPT Architecture - a Deep Learning Model for Natural Language Processing (NLP). The model consists of multi-layered transformers that leverage unsupervised learning to understand and generate human-like text. Its state-of-the-art performance on various language tasks makes it a popular choice for advancing NLP research.


Probability distribution over the model vocabulary items
softmax
5xVocabSize
LM head (768 x vocab size)
5x768
GPT decoder layers
Embeddings (768 x vocab size)
Same weights
<s>
I
want
to

Source: The author (2023)

head. For language modeling, the task head is a linear layer with a softmax activation function that transforms the output of the transformer blocks into a probability distribution over the vocabulary. The softmax function ensures that the sum of the probabilities of all vocabulary tokens equals 1.0. During inference, the token with the highest probability is selected as the predicted token. The model is trained using a maximum likelihood objective. Given the previous tokens in the sequence, the goal is to maximize the probability of predicting the correct token at each time step.

GPT-3 (BROWN et al., 2020), demonstrates that LLMs are few-shot learners. This model and its rivals (e.g., Google PaLM (CHOWDHERY et al., 2022), and DeepMind GOPHER (RAE et al., 2021)) promoted a revolution in most of the NLP-related tasks for not huge amounts of annotated data are necessary to a downstream task. GPT-3 was trained with 100 times more data than its predecessor GPT-2. Its large version has 175 billion parameters. A large amount of training data and the number of parameters make GPT-3 powerful in performing on-the-fly tasks on which it was never explicitly trained. Among these tasks, we can cite machine translation, math operations, writing code, etc.

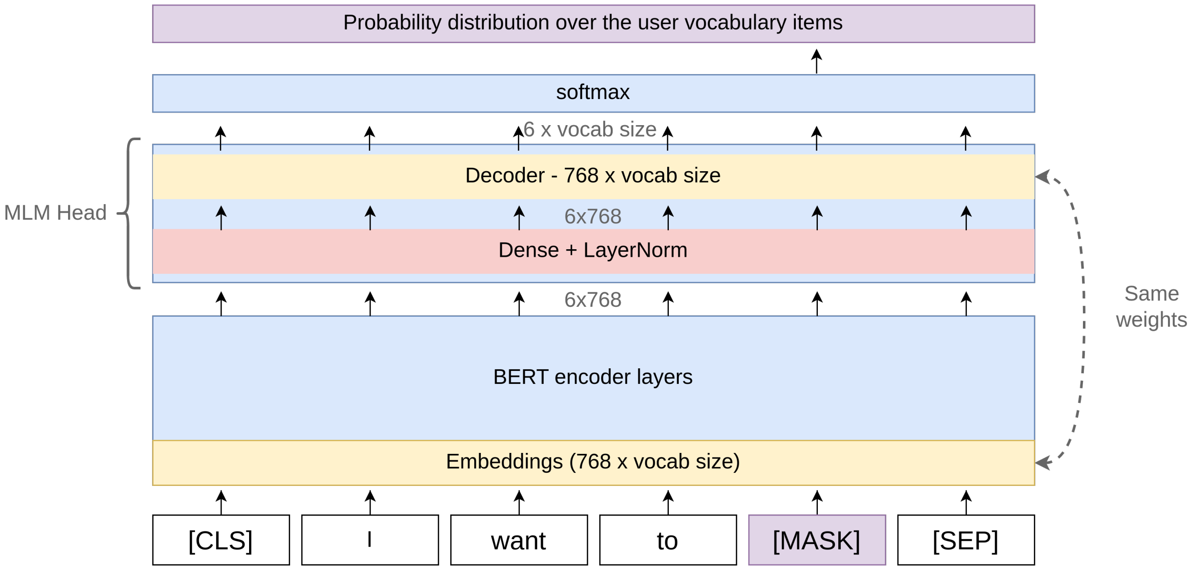
2.5.2 BERT

BERT is a language representation model that stands to Bidirectional Encoder Representations from Transformers (DEVLIN et al., 2019). This model uses the attention mechanism

(VASWANI et al., 2017) to learn contextual relations between tokens (words or sub-words) in unlabeled texts by joint conditioning on both left and right contexts in all layers of the model. Unlike directional models, which process the input in sequence (left-to-right or right-to-left), BERT processes the entire sequence simultaneously. Thus, it allows the model to learn the word's context based on all neighborhoods, left and right. Figure 6 shows an overview of the model, which receives a sequence of tokens as input and generates a representation for each token, which is then used for downstream tasks such as text classification, question answering, and language generation.

As shown in Figure 6 an embedding layer transforms the input text into vector representations. BERT uses a Word Piece Embeddings (WU et al., 2016a), which, to improve handling of rare words, divide words into a limited set of common sub-word units (e.g., "Playing" into "Play#" and "#ing"). This way, before inputting text to BERT, it has to be tokenized. Tokenization is splitting a sentence into tokens, which can be words or word pieces in the case of BERT. During training, the data generator randomly chooses 15% of the token positions for prediction. For example, if the i-th token is chosen, it is replaced with (1) the $[MASK]$ token 80% of the time, (2) a random token 10% of the time, or (3) the unchanged i-th token 10% of the time. The model attempts to predict the i-th token based on the contextual information

Figure 6 – BERT learning strategy. An embedding layer encodes the input tokens into representation vectors, which are inputted to the encoder layers. The output is a sequence of vectors, each corresponding to the input tokens in the same positions. The MLM head uses the output vectors to produce a probability distribution over the model vocabulary. The hidden state size depends on the model version: 768 for BERT-base and 1024 for BERT-large. Notice that the embeddings layer and the decoder layer share weights and that the number 6 in the figure refers to the input sequence length.


Probability distribution over the user vocabulary items
softmax
6 x vocab size
MLM Head
Decoder - 768 x vocab size
6x768
Dense + LayerNorm
6x768
Same weights
BERT encoder layers
Embeddings (768 x vocab size)
[CLS]
I
want
to
[MASK]
[SEP]

Source: The author (2023)

provided by the non-masked, generating a contextualized representation for each.

The language modeling task consists of predicting the next words to be added in a text of size N, where the preceding context $[w_1, ..., w_{i-1}]$ influences the next word probability $p(w_i)$. BERT allows word prediction through Masked Language Modeling (MLM). So, it is not necessarily a next-word prediction, for the mask can be put in any part of the sentence. Placing the mask token at the end of the sequence makes simulating the next word prediction possible. MLM predicts the masked tokens in a given sentence. As shown in Figure 6, the classification layer on top of the encoder output multiplies the output vectors by the embeddings matrix, transforming them into vocabulary dimension, allowing the MLM task. The MLM head consists of a softmax classifier that outputs a probability distribution over the vocabulary for each masked token. During training, the model is optimized to minimize the loss between the predicted and actual tokens.

BERT uses a cross-entropy loss function calculated over the 15% tokens chosen for prediction. Cross-entropy is the average number of bits required to store the information in a variable if an estimated probability distribution q is used instead of the true distribution p. In language modeling, p is the real distribution of the language, while q is the distribution estimated by the model. It is not possible to know the real p. However, given a long sequence of words W (i.e., a large N), it is possible to approximate the per-word cross-entropy using Shannon-McMillan-Breiman theorem (ALGOET; COVER, 1988) (cf. Equation 2.5).

$$H(p, q) \approx -\frac{1}{N} log_2 q(W) \tag{2.5}$$

This way, given a sequence of tokens W of length N and a trained language model P, the cross-entropy is approximated as follows:

$$H(W) = -\frac{1}{N} log_2 P(W) = -\frac{1}{N} log_2 P(w_1, w_2, ..., w_N) \tag{2.6}$$

BERT was adapted for different tasks and different languages. One such adaptation is Pretrained BERT Models for Brazilian Portuguese (BERTimbau) (SOUZA; NOGUEIRA; LOTUFO, 2020), a variant of BERT pre-trained on a large corpus of Brazilian Portuguese text data, the Brazilian Web as Corpus (brWaC) (FILHO et al., 2018). BERTimbau is designed to perform well on a wide range of NLP (NLP) tasks in Brazilian Portuguese, such as sentence textual similarity, recognizing textual entailment, and named entity recognition.

2.5.3 Transfer Learning

Transfer learning is a machine learning technique that involves leveraging knowledge gained from one task to improve the performance of another related, or unrelated task (LU et al., 2015; PAN; YANG, 2010). Transfer learning has shown great success in NLP applications, particularly using large language models such as GPT, BERT, and others. In transfer learning for NLP, a pre-trained language model is fine-tuned on a specific downstream task with a smaller dataset. This fine-tuning process allows the model to adapt to the specific task by updating its parameters while retaining the knowledge learned from the pre-training process.

One of the main benefits of transfer learning is allowing for the development of high-performing models with fewer data and computational resources. This is particularly useful when labeled data is scarce or costly to obtain. Transfer learning also reduces the time and effort required to develop a custom model from scratch. Pre-trained models have already learned a significant amount of linguistic knowledge that can be applied to downstream tasks. The attention mechanism is an essential component of transfer learning in NLP, particularly in transformer-based models such as GPT and BERT. The attention mechanism allows the model to focus on specific parts of the input sequence relevant to the task while ignoring irrelevant information. This improves the model's ability to capture long-range dependencies and understand the context of the input text.

Formally, transfer learning can be defined as follows (LU et al., 2015),

Definition 2.5.1. *Given a source domain D_s, a learning task T_s, a target domain D_t, and a learning task T_t, transfer learning aims to enhance the learning of the target predictive function $f_t(\cdot)$ in D_t using the knowledge in D_s and T_s where $D_s \neq D_t$ or $T_s \neq T_t$.*

The above definition considers the case where there is a source domain D_s, and a target domain D_t, which is the most popular in literature (PAN; YANG, 2010). Each domain D consists of a feature space χ and a probability distribution $P(X)$, where $X = x_1, ..., x_n \in \chi$. A task $T = \{Y, f(\cdot)\}$ consists of a label space Y and a predictive function $f_t(\cdot)$, also written as $P(y|x)$. Besides, in the definition, $D_s \neq D_t$ implies that either $\chi_s \neq \chi_t$ or $P_s(X) \neq P_t(X)$. Similarly, the condition $T_s \neq T_t$ implies that either $Y_s \neq Y_t$ or $f_s(\cdot) \neq f_t(\cdot)$. That is, the source and target domains may have different feature spaces or probability distributions, and the source and target tasks may have different label spaces or predictive functions. However, the source and target domains are related, for some explicit or implicit relationships exist between their feature spaces.

Zero-shot learning is a type of transfer learning in which a model is trained on a source task and then applied to a target task with no training data (XIAN; SCHIELE; AKATA, 2020). In zero-shot learning, the model is expected to generalize to new tasks or categories not seen during training. This is achieved by leveraging the shared knowledge, and representations learned from the source task to make predictions on the target task. For example, a model pre-trained on a large corpus of text can be used to classify sentiment in a specific domain, such as customer reviews for a new product, even if the model has never seen data from that domain before.

Few-shot learning involves training a model on a small amount of data to recognize new categories or tasks. This is achieved by fine-tuning a pre-trained model on the few examples available for the new task or category. Few-shot learning is useful when collecting large amounts of labeled data is challenging or expensive. For example, few-shot learning can train a model to recognize new types of animals with limited labeled data. Zero-shot and few-shot learning are examples of transfer learning as they both involve leveraging knowledge learned from a pre-existing task to improve performance on a new task. These approaches have been successfully applied in various NLP tasks like sentiment analysis, question answering, and text classification.

3 RELATED WORK

This chapter presents the results of a systematic mapping study previously published in (PEREIRA et al., 2022). The study aimed to select works that propose card or pictogram prediction methods in AAC. Section 3.1 presents the method for selecting works. While Section 3.2 presents the selected works.

3.1 WORK SELECTION

The systematic mapping study presented in this chapter aims to analyze the scientific proposals for communication card prediction in *high-tech* AAC systems concerning the computational techniques and methods used for prediction and the methods and metrics used to evaluate the proposals. For this, four RQs were defined, each aimed at a different research facet. Table 1 shows the RQs and their related research facet. The facets were designed to help to answer the research questions and obtain a broad view of the current status of research in the field. They serve to classify the articles obtained from the screening criteria.

RQ-1 (*Prediction method*) aims to identify the study's computational method or technique used for pictogram prediction. This information is essential to understand the field evolution over time regarding the methods employed to attack the task. RQ-2 (*Prediction unit*) aims to identify the prediction unit, which is important to understand how the method makes predictions. This question is important because the definition of pictogram may not be the same among the studies. In AAC, a pictogram is a picture+label pair. The label is generally a word or expression a text-to-speech application will speak. And the picture or photo is the visual support for the user to understand its meaning. This question aims to identify what the study uses to perform prediction: the label, the image, the pair image+label, etc. RQ-3 (*Evalua-*

Table 1 – Research Questions used in the Mapping Study.

#	Question	Facet
RQ1	What are the computational methods/algorithms/artifacts used for pictogram prediction?	Prediction method
RQ2	What is the prediction unit?	Prediction unit
RQ3	How the proposal quality is assessed?	Evaluation method
RQ4	What evaluation metric is used?	Evaluation metric

Source: The author (2023)

tion method) aims to identify the method used to evaluate the proposal quality. The way the proposal is considered may indicate the approach maturity. For example, an automatic (intrinsic) evaluation may show that the approach is in an initial stage of development (JURAFSKY; MARTIN, 2019). RQ-4 (*Evaluation metric*) investigates what metrics the studies used for evaluation. This information clarifies how the proposal is evaluated.

3.1.1 Data Sources and Search Strategy

Chen, Babar and Zhang (2010) suggest using a search string in scientific databases to combine terms of interest to extract as many related studies as possible and avoid the inclusion of unrelated studies in the results. Figure 7 presents the used search string. AAC stands for Augmentative and Alternative Communication, which can also be found as Supplementary and Alternative Communication. AAC systems can also be called voice output devices, communication boards, or Voice Output Communication Aids (VOCAS). The used string includes all these terms to increase the search range. The term “word prediction” was included in the string because different studies may treat pictograms differently. Some consider that the word on its label better represents a pictogram. Besides, pictogram prediction supports sentence construction in AAC, similar to message composition and authoring.

The search string presented in Figure 7 was employed to query eight scientific databases, namely IEEE Xplore, ACM Digital Library, Google Scholar, PubMed, Science Direct, Scopus, Springer, and Taylor & Francis Online[1]. The study period encompassed publications from 2015 to 2022, and data collection was carried out between May and June 2022. In total, 467 studies were retrieved and subsequently organized using the State of the Art through Systematic

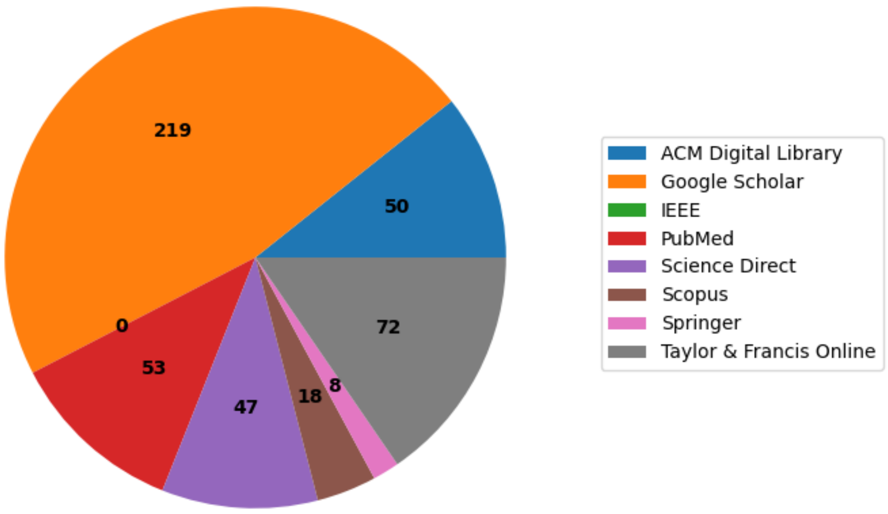
Figure 7 – Search string used in the Mapping Study. AAC can also be found as Supplementary and Alternative Communication. AAC systems can also be called voice output devices, communication boards, or VOCAS.

(“alternative communication” OR “AAC” OR “voice output devices” OR
“communication boards” OR “voice output communication aids” OR “VOCAS”)
AND (“sentence construction” OR “pictogram prediction” OR “pictogram suggestion”
OR “predictive composition” OR “word prediction” OR “message composition” OR
“message authoring”)

Source: The author (2023)

[1] The string may undergo some modifications based on the database search format.

Figure 8 – Studies returned in the mapping study grouped by sources.



Source: The author (2023)

Review (StArt) tool (FABBRI et al., 2016). The distribution of these studies across different sources is depicted in Figure 8. Notably, databases associated with health-related domains, such as PubMed and Taylor & Francis Online, contained more articles. This observation aligns with the clinical nature of AAC, which often involves research conducted by speech therapists and other healthcare professionals. However, the increased adoption of high-tech AAC and AI in this domain has led to studies appearing in technology-oriented sources like ACM Digital Library. Despite being a specialized database, IEEE Xplore yielded no relevant articles. Furthermore, we identified and excluded 47 duplicated studies using the StArt duplicates classification process.

3.1.2 Study Selection

To assess the relevance of the studies to be included in the final results, we applied the criteria presented in Table 2. Notice that these are exclusion criteria, meaning that the mapping results exclude the studies that fall on at least one of them. We opted to include only primary studies as they may fit better the research questions. This criterion avoids including editorials, keynotes, biographies, opinions, tutorials, workshop summary reports, progress reports, posters, thesis, dissertations, book chapters, panels, or literature mappings or reviews, which may not propose new approaches for pictogram prediction. Some studies focus on word-based systems in the AAC field. We excluded these studies because they may present a word or character

Table 2 – Mapping study exclusion criteria.

#	Criteria
E1	The study is written in a language other than English;
E2	The study is not a primary study;
E3	The study is not of Augmentative Alternative Communication field;
E4	The study focuses on AAC but does not use any strategies for pictogram suggestion;
E5	The study focus on word prediction with no pictogram;

Source: The author (2023)

prediction techniques that cannot perform pictogram prediction.

The procedure for applying the criteria consisted of screening the studies' titles, keywords, and abstracts. In some cases, accessing the study's full text was necessary as insufficient information is provided in the abstract to decide. It is required when studies are about AAC and mention prediction but do not specify if it is about words or pictograms in the abstracts. Two researchers performed the screening procedure to avoid individual biases. Any uncertainties or discrepancies were resolved through a researcher's meeting, where discussions and consensus were reached to ensure the rigor and reliability of the screening process.

3.1.3 Data Extraction

For data extraction, we applied the keywording technique, as proposed by (PETERSEN et al., 2008). The method assigns labels or keywords to concepts found in the study's text. Some open codes would be obtained, which have to be put into an overall structure. The categories' codes may be merged or renamed (PETERSEN; VAKKALANKA; KUZNIARZ, 2015). According to (PETERSEN; VAKKALANKA; KUZNIARZ, 2015), the process may only be applied to the paper's abstract. However, if the abstracts are unclear, the method may consider the paper's introduction, conclusion, or other parts. We applied keywording to the papers' full text to fit the research questions better. This way, the labels we code while reading the papers help answer the research questions in Table 1.

3.2 WORK PRESENTATION

This study analyzed 248 papers retrieved using the search strings presented in Figure 7. However, applying the criteria shown in Table 2, only eight studies were included in the

Table 3 – List of studies included in the mapping study after applying selection criteria.

Title	Author and Year	Venue
A semantic grammar for beginning communicators	(MARTíNEZ-SANTIAGO et al., 2015)	Knowledge-Based Systems
Context-aware communicator for all	(GARCÍA et al., 2015)	International Conference on Universal Access in Human-Computer Interaction
An augmentative and alternative communication tool for children and adolescents with cerebral palsy	(SATURNO et al., 2015)	Behaviour & Information Technology
Evaluating pictogram prediction in a location-aware augmentative and alternative communication system	(GARCIA; OLIVEIRA; MATOS, 2016)	Assistive Technology
Compositional Language Modeling for Icon-Based Augmentative and Alternative Communication.	(DUDY; BEDRICK, 2018)	Association for Computational Linguistics Meeting
Predictive composition of pictogram messages for users with autism	(HERVÁS et al., 2020)	Journal of Ambient Intelligence and Humanized Computing
A semantic grammar for augmentative and alternative communication systems	(PEREIRA; FRANCO; FIDALGO, 2020)	International Conference on Text, Speech, and Dialogue
PictoBERT: Transformers for next pictogram prediction	(PEREIRA et al., 2022)	Expert Systems with Applications

Source: The author (2023)

final results. In Table 3, we present the included studies, their references, and publishing venue. Notice that there are three studies published in conferences (GARCÍA et al., 2015; DUDY; BEDRICK, 2018; PEREIRA; FRANCO; FIDALGO, 2020), and five published in journals. Besides, most venues are from the Computer Science field, except for (GARCIA; OLIVEIRA; MATOS, 2016), published in a multidisciplinary journal. AAC is a multidisciplinary field (BEUKELMAN; LIGHT, 2020), and the participation of the Computer Science community in this field is due to the need to improve AAC interventions to maximize communication for individuals with CCN (LIGHT; MCNAUGHTON, 2012) by using mobile applications. Besides, word or pictogram prediction may involve NLP techniques, which rely on machine learning and statistical analysis (SENNOTT et al., 2019), fields generally populated by computer scientists.

Table 4 presents the results of applying the keywording technique (cf. Section 3.1.3), which generated 22 keywords along the five studied facets. Next, we discuss these results regarding each research facet.

Prediction Method: We identified five methods used to perform pictogram prediction in the studies. We can say that the most common methods are those based on knowledge bases: semantic grammar (2 studies), concept network (1 study), and direct graph (1 study).

Table 4 – Studies included in the mapping study after applying keywording.

Study	Facets and Keywords				
	Prediction Method	**Prediction Unit**	**Evaluation Method**	**Evaluation Metric**	**Outcomes**
Martínez-Santiago et al. (2015)	Semantic Grammar	Pictogram sense	Automatic	None	No baseline
García et al. (2015)	concept network	Pictogram label	None	None	not reported
Saturno et al. (2015)	Direct graph	Pictogram label	Quasi experiment	Number of Pictograms, Time	Positive
Garcia, Oliveira and Matos (2016)	n-gram	Pictogram label	Automatic	Keystroke saving	Positive
Dudy and Bedrick (2018)	Deep learning	Pictogram related words	Automatic	MRR, Top-n Accuracy	No baseline
Hervás et al. (2020)	n-gram	Pictogram label, Pictogram POS	Quasi experiment	Time, Number of Pictograms, Top-n Accuracy	Positive
Pereira, Franco and Fidalgo (2020)	Semantic Grammar	Pictogram sense	Automatic	Precision	No baseline
Pereira et al. (2022)	Deep learning	Pictogram sense	Automatic	Perplexity, Top-n accuracy	Positive

Source: The author (2023)

Two studies using statistical language models based on n-grams (HERVÁS et al., 2020; GARCIA; OLIVEIRA; MATOS, 2016). These studies trained bi-gram language models by using pre-defined text corpora. Another characteristic they have in common is that they enrich the models' knowledge with the user's actual usage. Two other approaches employed deep learning models (PEREIRA et al., 2022; DUDY; BEDRICK, 2018). They used neural networks trained with synthetic text corpora generated from natural language text samples. The literature suggests that neural networks-based language models may perform better than statistical models (GOLDBERG; HIRST, 2017). Besides, Pereira et al. (2022) compared their model with knowledge-based approaches and demonstrated improvements. Their models outperformed the semantic grammar in predicting the correct pictogram to complete a sentence. However, neural networks may require more computational resources than statistical models or knowledge bases, making their deployment difficult in production.

Prediction unit: The analyzed studies used four types of prediction units: the pictogram label, the label's POS, the label's set of related words, and the label's word sense. As discussed in Section 2.1, in *high-tech* AAC systems, each pictogram has an associated label or caption, which can be a word or a multi-word expression. Some analyzed studies consider this label enough for making pictogram prediction (GARCÍA et al., 2015; SATURNO et al., 2015; GARCIA; OLIVEIRA; MATOS, 2016; HERVÁS et al., 2020). This way, they perform a word prediction and do not take care of polysemic words. For example, the English word "bat" can have many meanings (e.g., "nocturnal mouselike mammal" or "a club used for hitting a ball") and, similarly, many related pictograms in a given vocabulary. In addition to the label, Hervás et al. (2020) opted to use its POS tag as a prediction unit. This approach involves annotating the text with words grammatical classes (i.e., POS), such as nouns, verbs, and adjectives. They trained a bi-gram language model using the sequence of POS tags as training data. The aim is to suggest the pictograms labeled with the predicted POS tag to the user. The authors compared the two approaches and noticed that the prediction improvement based on POS sequencing is unclear. Dudy and Bedrick (2018) treated a pictogram as a set of synonyms. For a given pictogram, they look for the labels used in the Symbolstix database[2] and generate a real-valued vector using pre-trained word embeddings vectors. For example, if a pictogram has four associated words, the authors get the words' vectorial representation in the embeddings matrix and average them. The result is used as the pictogram vectorial representation. The authors used these vectors as input to their recurrent neural network. Other studies followed an approach similar to Schwab et al. (2020), which consider that a pictogram is better represented by a concept from a dictionary (e.g., person: a human being) (MARTíNEZ-SANTIAGO et al., 2015; PEREIRA; FRANCO; FIDALGO, 2020; PEREIRA et al., 2022). This approach assumes that the concept links the pictogram label and its figure. Martínez-Santiago et al. (2015) used concepts from the FrameNet database, Pereira, Franco and Fidalgo (2020) used WordNet synsets (a set of synonyms with a glossary definition, e.g., a person is a human being), and Pereira et al. (2022) used WordNet word-senses (a link between a word and a synset). For more details about the differences between a synset and a word sense, refer to WordNet documentation[3]. Pereira et al. (2022) encodes each word-sense to a real-valued vector using the embeddings constructed by Scarlini, Pasini and Navigli (2020). Approaches based on concepts (synsets, word-senses) may fit better polysemic words. However, it may require a prepossessing step in the prediction pipeline. An example is Pereira et al. (2022), which parsed the text corpus for word-sense disambiguation, and Dudy and Bedrick (2018), which requires the preexistence of a list of words for each pictogram. On the other hand, approaches that use labels may not need a preprocessing step, but it does not treat polysemy.

[2] <https://www.n2y.com/symbolstix-prime/>

[3] <https://wordnet.princeton.edu/documentation/wngloss7wn>

Evaluation method: The analyzed studies performed two types of experiments for evaluating their proposals: automatic evaluation and quasi-experiments. One of the papers only presents the proposal and some usage examples but does not carry out an assessment (GARCÍA et al., 2015). First, we describe the studies that performed an automatic evaluation. Martínez-Santiago et al. (2015) evaluated the semantic grammar at each step of its construction. They tested how well the controlled language (i.e., set of sentences) fits into the semantic grammar. Garcia, Oliveira and Matos (2016) ran several software simulations to measure the performance of the different pictogram prediction approaches they proposed. They evaluated the models over a set of sentences indicated by specialists as adequate for the AAC domain. Dudy and Bedrick (2018) used the synthetic text corpus they created to evaluate their models. They divided the corpus into a five-fold split and computed the model performance in each fold. Pereira, Franco and Fidalgo (2020) assessed the quality of the predictions made by their semantic grammar by using it to reconstruct subject+verb+object sentences extracted from the CHILDES database (MACWHINNEY, 2014). All the experiments performed by Pereira et al. (2022) were automatic. They used part of the synthetic text corpus they built to assess the quality of the proposal on predicting pictograms to complete the sentences. Besides, they asked practitioners to inform examples of sentences usually constructed in AAC systems and evaluated the models' ability to complete them. Two studies performed a quasi-experiment involving humans. Saturno et al. (2015) analyze a student's performance through a dialogue with and without using the proposed AAC system. They observed the efficiency and satisfaction of using the system with predictions. The student is a child with complex communication needs. In Hervás et al. (2020), a teacher working with autistic children with complex communication needs participated in the experiments, which involved reproducing the children's conversations in the class over five weeks in the AAC tool. This way, most of the studies used an automatic evaluation and assessed their proposal quality without the participation of actual AAC users. This situation can be explained by the difficulties of accessing people with complex communication needs, but it also indicates that the field is more exploratory than experimental.

Evaluation metric: Two studies did not report the used evaluation metrics (MARTíNEZ-SANTIAGO et al., 2015; GARCÍA et al., 2015). Saturno et al. (2015) assessed the number of pictograms used by the experiment participant to construct the proposed sentences and the time spent. Hervás et al. (2020), which also performed a quasi-experiment, used the same metrics and a top-n accuracy, indicating whether the model on top-n predicted the participant's pictogram used. Top-n accuracy was also used by Dudy and Bedrick (2018) and Pereira et al. (2022), the two approaches based on deep learning. These approaches used other most common metrics in the NLP field. Dudy and Bedrick (2018) used MRR, generally used to assess information retrieval systems quality, where is wanted to the best item to be in a higher position in the ranking. Pereira et al. (2022) used a metric called Perplexity, which indicates how surprised a language model is when exposed to a new text distribution. In other words, it quantifies how well the model can predict new, unseen data. Pereira, Franco and Fidalgo (2020) evaluated their proposal's precision for reconstructing the sentences from a corpus. And finally, Garcia, Oliveira and Matos (2016) assessed the system's quality in saving keystrokes. This way, there is no consensus on the metric most adequate for the task. However, top-n accuracy is the most used metric among the analyzed studies. As mentioned in Section 2.1, AAC systems use to present pictograms in a grid. This way, we can say top-n accuracy measures how accurate the system is in predicting the pictograms that will be shown in a grid of size n.

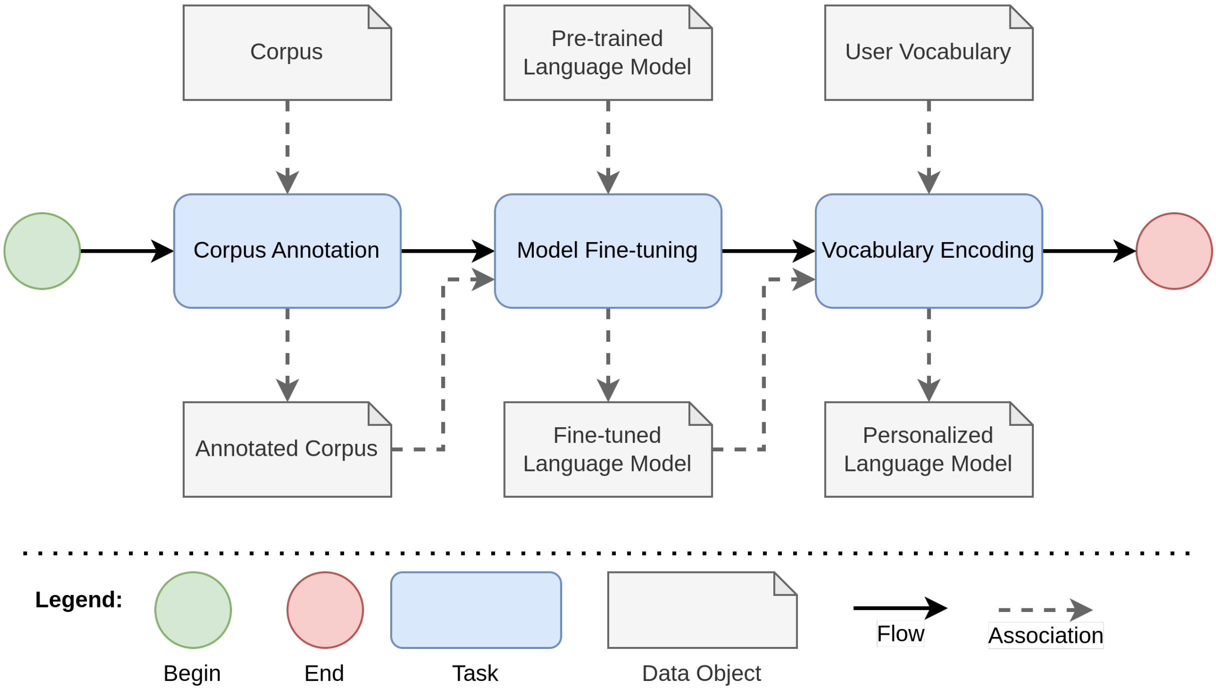
3.3 CHAPTER CONCLUSIONS

In conclusion, the main disadvantage of the works presented in this chapter is that they freeze the vocabulary after training the models. For example, the vocabulary of an n-gram model is based on the words (or expressions) that occur in the training corpus. These models may not be effective for users with varying vocabulary needs, and updating them can be computationally expensive and time-consuming. Even state-of-the-art models, such as PictoBERT, based on BERT, have this limitation. Our proposed approach overcomes these limitations by allowing for easy adaptation to different vocabularies and user needs without extensive retraining. It also requires only a small amount of text data for fine-tuning and is plug-and-play, avoiding the need for training or fine-tuning large language models. This approach represents a significant advance in state-of-the-art communication card prediction. It provides a more efficient and adaptive solution to accommodate the varying vocabularies of different users. It also allows for more flexibility in updating and maintaining the models, making them easier to use and more accessible for those who need them. As such, it can potentially improve the communication and quality of life for AAC users.

4 PRAACT: PREDICTIVE AUGMENTATIVE AND ALTERNATIVE COMMUNICATION WITH TRANSFORMERS

In this chapter, we present the proposed method for adapting LLMs to communication card prediction in AAC, named Predictive Augmentative and Alternative Communication with Transformers (PrAACT). As illustrated in Figure 9, the method comprises three main steps: Corpus Annotation, Model fine-tuning, and Vocabulary Encoding, each with specific inputs and outputs. By customizing the language model to fit the user's vocabulary, the proposed method facilitates message authoring in AAC systems, leveraging the power of transfer learning from LLMs like BERT and GPT. Besides, the proposed method allows for easily adapting transformers-based LLMs to different vocabularies and user needs without extensive retraining. This approach leverages the power of pre-trained language models like BERT and GPT, which are effective in many NLP tasks. In the upcoming sections, we will provide a detailed explanation of each step in the method.

Figure 9 – PrAACT: method for adapting LLMs to communication card prediction in AAC systems. The method consists of three main steps: Corpus Annotation, Model Fine-tuning, and Vocabulary Encoding. The outputs of each step feed into the subsequent step, resulting in a customized language model for efficient message authoring in AAC systems.



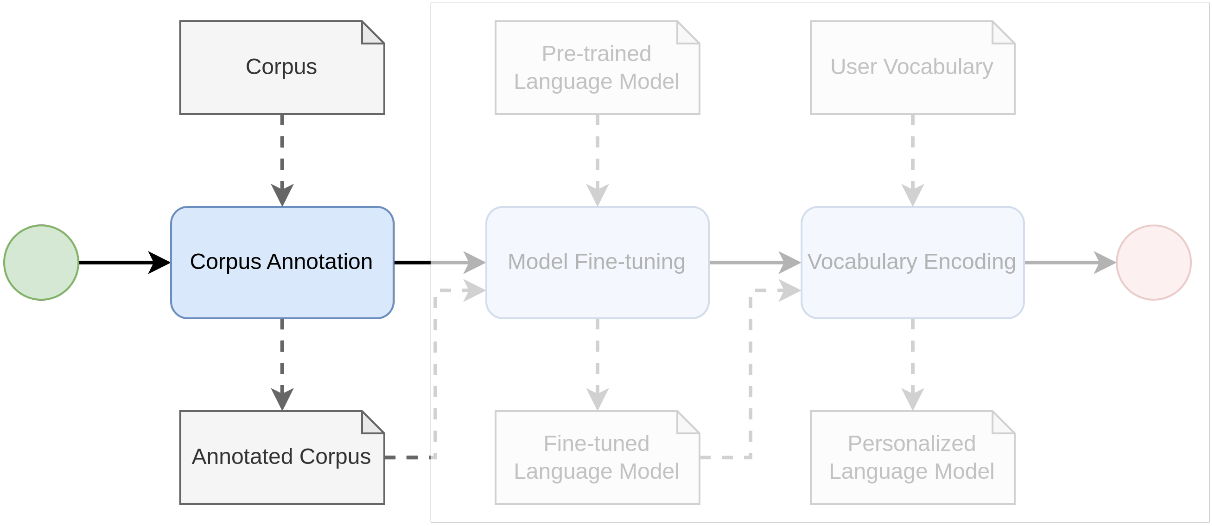
Source: The author (2023)

4.1 CORPUS ANNOTATION

The Corpus Annotation step of the proposed method is crucial for adapting LLMs to communication card prediction in AAC systems. As shown in Figure 10, it inputs a natural language corpus and outputs an annotated corpus. In this context, it is worth considering that AAC has specific characteristics that differ from natural languages, such as limited vocabulary, reduced sentence complexity, and reliance on semantic and visual cues. Therefore, the Corpus Annotation step is a domain adaptation process that requires a corpus for AAC or with similar characteristics. The corpus must contain conversational sentences with simple structures because individuals with communication impairments often have limited language abilities and rely on visual cues to communicate. Thus, the corpus needs to reflect the communication needs of the target audience, which generally requires simple, concise language. Additionally, AAC systems are typically designed to provide users with limited vocabulary options. Complex language structures may make locating and selecting appropriate communication cards more challenging. Therefore, using conversational sentences with simple structures in the AAC corpus ensures that the language model can accurately predict the most suitable communication cards for the user.

Obtaining an adequate corpus for AAC systems can be challenging. One option is to extract texts from the WEB or books. Still, these texts may contain complex sentence constructions, idiomatic expressions, and unfamiliar vocabulary unsuitable for the context of AAC. Another

Figure 10 – The Corpus Annotation, essential for adapting language models to communication card prediction in AAC systems, requires a corpus that reflects the needs of individuals with communication impairments.



Source: The author (2023).

option is to use specialized corpora developed for AAC, which are often limited in size and scope. Therefore, careful consideration must be given to the selection and preparation of the corpus for the Corpus Annotation step to ensure that it is appropriate for the domain adaptation process of the LLMs used in AAC systems. In this work, we conducted experiments in English and Brazilian Portuguese. For the English language, we utilized the AACText and SemCHILDES corpora. AACText (VERTANEN; KRISTENSSON, 2011) is a corpus comprising around 6,000 sentences of fictional AAC-like communications, divided into training, testing, and development sets. The authors of AACText utilized Amazon Mechanical Turk to construct this corpus, creating numerous messages that model conversational AAC. The SemCHILDES dataset, on the other hand, was used for pre-training PictoBERT (PEREIRA et al., 2022) for pictogram prediction. It is a large corpus of North American English comprising 955,489 sentences from the Child Language Data Exchange System (CHILDES) database (MACWHINNEY, 2014). As for Brazilian Portuguese, we used the AACptCorpus (PEREIRA et al., 2023a), and the construction method is detailed in Section 5.1.2. Chapter 5 presents more details about how these corpora are used in this study.

The annotation of the corpus can be performed in various ways, depending on the specific needs of the user or group. Two examples of annotation approaches include (a) transforming the natural language sentences into telegraphic sentences that use only keywords, such as verbs, nouns, adverbs, and adjectives, with the lemmas of the words; or (b) annotating it with grammatical structures of the sentences to create semantic scripts like CS (cf. Section 3). CS (BRYAN, 2003) is a therapeutic tool developed to help individuals with CCNs understand and develop the construction of written or spoken sentences. The tool uses a color-coded system associated with key questions, such as “Who?” “What Doing?” “What?” “Where?” and “What Like?” help individuals understand the semantic role of each constituent of a phrase. Colors act as visual aids that indicate the sentence’s grammatical structure, while questions help link this structure to its meaning. This semantic script can annotate the corpus and create a dataset for fine-tuning the LLM for AAC communication card prediction.

The following subsections outline two methods for processing the natural language corpus: one for transforming it into a corpus of telegraphic text (cf. Section 4.1.1) and another for annotating the corpus using the semantic roles derived from CS (cf. Section 4.1.2).

4.1.1 Natural Language Text to Telegraphic Text

Telegraphic language is a style of speaking that simplifies the expression of ideas by only using the most necessary content words while leaving out grammatical function words (like determiners, conjunctions, and prepositions) and inflectional endings. Telegraphic language is crucial in AAC systems. Individuals with CCNs often face challenges in processing and producing complex linguistic structures. By simplifying the expression of ideas and using only essential content words, telegraphic language enables individuals to convey their messages more effectively. Besides, telegraphic language helps to reduce cognitive load and enhance communication efficiency.

This section presents a method for transforming natural language sentences into telegraphic sentences. The proposed method aims to create a telegraphic version of a sentence by identifying and removing non-content words and inflectional endings. The resulting sentence contains only the essential content words needed to convey the meaning of the original sentence in a more straightforward and accessible manner. The method uses POS tagging and morphological analysis to identify the non-content words and inflectional endings to be removed.

First, the sentences must be parsed to extract the words' lemmas and POS tags to transform the natural language corpus into a telegraphic language corpus. The POS tags identify the word's function in a sentence, such as whether it is a noun, verb, or adjective. The lemmas are the word's basic form and help reduce the vocabulary's size by grouping different forms of the same word. For example, the verb "running" is reduced to its lemma "run". Once the sentences have been parsed and annotated with the POS tags and lemmas, the resulting telegraphic sentences will contain only the essential words needed to convey the sentence's meaning. Only nouns, pronouns, verbs, adjectives, and adverbs are kept. This process transforms sentences like "I ate a cake at school this morning" into "I eat cake school morning". This adaptation will help the model to learn a new language distribution so it will be prepared to process sentences in which prepositions, articles, and verb inflections are omitted.

Various NLP tools can be used to transform the corpus into telegraphic language. In this study, experiments for English and Brazilian Portuguese were performed (cf., Chapter 5). Spacy NLP (HONNIBAL; MONTANI, 2017) tool was used for the English experiments. Spacy is a popular open-source software library that provides efficient and scalable NLP functionalities in Python. It has become a preferred choice for many researchers and developers due to its accuracy, speed, and ease of use. Additionally, Spacy provides pre-trained models that can be used for various NLP tasks, such as POS tagging and lemmatization.

For Portuguese, the Stanza NLP tool (QI et al., 2020) was used. Stanza NLP is another open-source NLP tool that can be used for various natural language processing tasks, including POS tagging and dependency parsing. It is also available in Python and supports multiple languages, including Portuguese. Like Spacy, Stanza provides pre-trained models for these tasks, trained on large datasets, allowing for accurate and efficient analysis of text data. Stanza also offers a user-friendly interface that allows easy customization of the preprocessing pipeline, making it a valuable tool for researchers and developers in NLP. Stanza's Portuguese models are highly accurate and efficient, providing a comprehensive set of linguistic annotations that can be used for various NLP tasks. However, it's important to note that this method is generic. Different parsers and tools can be used depending on the specific language and the intended use of the AAC system.

4.1.2 Using the Colourful Semantics structure

In addition to annotating the corpus with POS tags and lemmas to transform it into telegraphic language, other annotations can also be performed, depending on the AAC system developer's intentions or the user needs. For example, the CS (cf. Section 2.2.2) annotation can be added to the corpus to allow training models that use semantic scripts to help the user understand the sentence meaning. This annotation is based on a color key system and key questions that guide the individual in understanding the syntactic structure of a phrase and linking it to its meaning. Including such annotations can enhance the effectiveness of the LLMs used in AAC systems and facilitate message authoring. However, the specific annotations used in the corpus depend on the intended use of the AAC system and the user's needs.

Before using CS it is important to consider that different verbs can have different structures that individuals with CCNs may not easily understand. For instance, while verbs such as "to eat" are typically transitive and follow the structure of Agent+Verb+Theme, verbs like "to be" are copulas and follow the structure of Theme+Verb+Description. This can confuse people with CCNs as the sequence of CS roles will change, and then the sequence of colors will also change. To make it easier for individuals with CCNs, a simplified structure such as Subject+Verb+Object may be more appropriate, even for copular verbs like "to be". This way, mapping semantics roles to syntactic functions is essential in annotating a corpus with

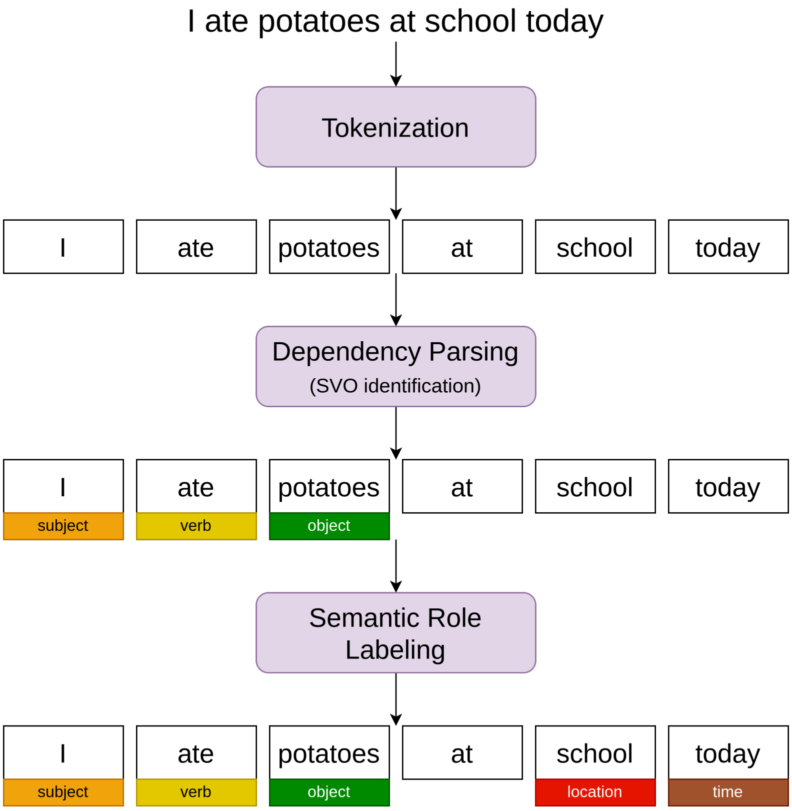
CS roles.

We propose a three-step pipeline for annotating the sentences with the CS roles, as depicted in Figure 11. The pipeline comprises the following steps: tokenization, dependency parsing, and semantic role labeling. In the tokenization step, the sentences are tokenized, breaking them into individual words or tokens. This process is essential for preparing the text for further analysis in the subsequent steps. We approach the other two steps in the following paragraphs.

In the second step, dependency parsing identifies the subject-verb-object structure within the sentences. This NLP technique plays a crucial role in capturing the syntactic organization of the sentences by examining the interdependencies between words. It facilitates the identification of grammatical relationships, including subject-verb and verb-object. For this purpose, the Portuguese model from Stanza and the English model from SpaCy can harness their respective linguistic resources and capabilities;

In the third step, Semantic Role Lebaling (SRL) identifies the adverbial complements, such as location, time, and manner. This technique, which surpasses mere syntax analysis, is crucial in identifying the semantic roles of words within a sentence, including agent, pa-

Figure 11 – Illustration of the three-step pipeline method for annotating sentences using Colourful Semantics roles.



Source: The author (2023)

tient, and instrument. By establishing the syntactic relationships between words, SRL assigns them corresponding semantic roles, contributing to a deeper comprehension of the sentence's intended meaning. The Intelligible Verbs and Roles (InVeRo) semantic parser (CONIA et al., 2020) can be employed for both languages to perform this analysis. The InVeRo parser is particularly renowned for its multilingual capabilities, facilitating the process of semantic role labeling across different languages. This feature proved advantageous for our study, which involved experiments conducted in English and Portuguese. Leveraging the capabilities of the InVeRo parser, we successfully extracted and labeled the semantic roles associated with each constituent of the sentences in both languages, thereby enhancing our understanding of the sentence's overall structure and meaning.

This three-step pipeline annotates the corpus with syntactic and semantic information, merging in the CS roles (e.g., who, what doing, what, when, how, where). For example, consider the sentence "I ate potatoes at school today". The dependency parsing technique identifies the subject "I", the verb "ate", and the object "potatoes". The adverbial complement "at school" will be identified and labeled as the location semantic role, and "today" as the time semantic role by the SRL technique. This allows the method to assign the semantic roles of "who" (I), "what doing" (eat), "what" (potatoes), "where" (at school), and "when" (today) to the different constituents of the sentence, according to the CS framework.

In this method, it is important to determine how to treat auxiliary verbs (e.g., "be", "have", and "do"), which can have different functions and meanings in a sentence. In this study, we decided to label auxiliary verbs as "what doing" in the same way as main verbs, as they also contribute to the action or state described in the sentence. This includes modal verbs such as "can" and "should", which indicate possibility or necessity and can be crucial for conveying meaning in specific contexts. This way, the sentence "I want to eat the cake", for example, is annotated as *who: I*, *what doing: want, eat*, *what: cake*. By treating auxiliary verbs in this way, we aim to capture the whole semantic meaning of the sentence while ensuring that the predicted communication cards accurately reflect the intended message.

4.2 MODEL FINE-TUNING

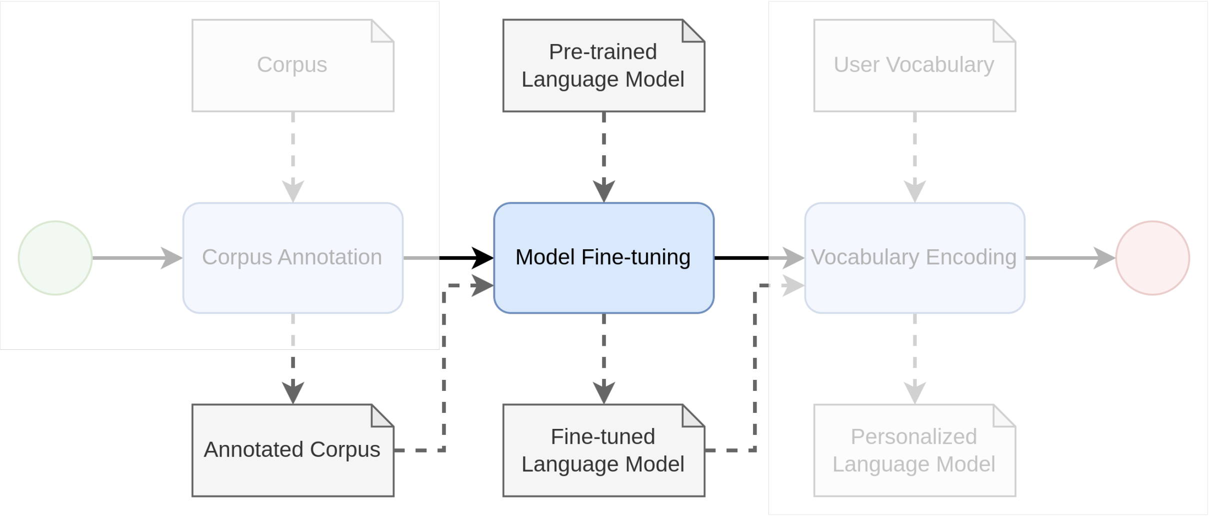
This section focuses on the fine-tuning step of PrAACT. After the corpus has been annotated with CS roles and transformed into a corpus of telegraphic language, as described in Section 4.1, the next step is to train a language model to predict the most appropriate communication cards to complete a sentence in construction. This section discusses the inputs and techniques used to fine-tune the model.

As shown in Figure 12, the inputs of this step are the corpus annotated in the Corpus Annotation Step (cf. Section 4.1) and a pre-trained language model that allows transfer learning. The annotated corpus serves as the training data for the model, and the pre-trained language model is used as a starting point for transfer learning. These models are large neural networks trained on vast amounts of text data that have been learned to predict the next word in a sentence given the previous words or to predict a masked word in a sentence like "Paris is the [MASK] of France" (cf. Section 2.5). They can be fine-tuned on a smaller, task-specific dataset, such as our annotated corpus, to make predictions on our target task.

Transfer learning is important because it allows us to use the pre-trained models' language understanding abilities (cf. Section 2.5.3). Pre-trained language models have acquired a vast knowledge of words' language structure and semantic representation, rendering them valuable resources for developing communication card prediction models with higher accuracy. Subsequently, fine-tuning the pre-trained models on our annotated corpus adjusts the models to the specific task, potentially enhancing the overall performance.

Different models can be used for communication card prediction in this step. In Chapter 3, we cite the models used in recent works in this field, which include n-gram language models, knowledge bases, and RNN. However, recent research has shown that transformer-based models are the best alternative in terms of performance, generalization to unseen text, and adaptation

Figure 12 – Model fine-tuning step. The inputs of this step are the corpus annotated in the Corpus Annotation Step and a pre-trained language model for transfer learning. The annotated corpus serves as the training data for the model, while the pre-trained language model is used as a starting point.



Source: The author (2023)

to different users and scenarios (PEREIRA et al., 2022). Transformer-based models, such as BERT and GPT, have performed state-of-the-art NLP tasks, including language generation and understanding. These models use an attention mechanism to process the input sequence, which allows them to capture long-term dependencies and context (cf. Section 2.5). The following sub-sections present the challenges of adapting a transformer-based LM for communication card prediction in AAC.

4.2.1 How to represent a communication card

One of the challenges of using a transformer-based LM for this task is how to encode a communication card to use it as input for a deep neural network. As mentioned in Section 2.5, transformers models like BERT encode the words in a sentence using word embeddings, which are numerical representations of words that capture semantic and syntactic similarities between them (cf. Section 2.4). As discussed in Chapter 3, recent works that perform card prediction to aid message authoring have employed various methods to represent communication cards. One popular approach is representing communication cards through a concept rather than solely relying on the caption (MARTíNEZ-SANTIAGO et al., 2015; DUDY; BEDRICK, 2018; PEREIRA; FRANCO; FIDALGO, 2020; PEREIRA et al., 2022). This involves associating the communication card with a specific concept, such as a synset in WordNET (MILLER, 1995) or a specific ontology, which can help to better capture the meaning and context of the communication card beyond just the caption. Other methods have been employed, such as representing the communication card through the POS of the caption or a set of synonyms for the caption. These different approaches demonstrate the ongoing efforts to develop more effective methods for predicting communication cards and improving the accessibility of AAC systems.

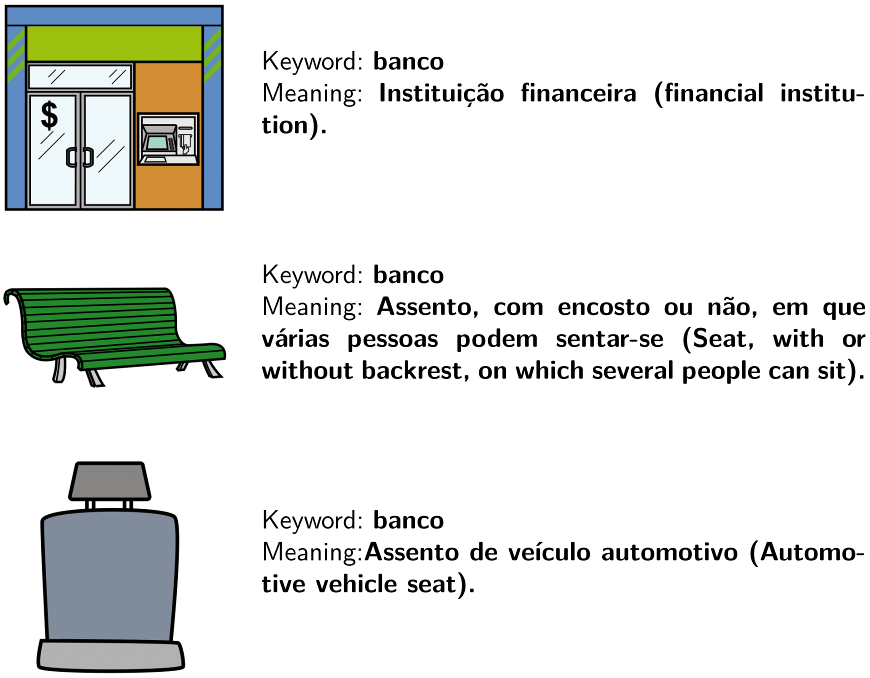
In this section, we outline the method we applied to address how to represent best a communication card for input to a transformer neural network. Our approach thoroughly analyzed various representation methods, including the caption, glossary definition, and synonym approaches discussed in Chapter 3. We also considered additional factors, such as the visual features of the pictogram. For this, we perform an exploratory experiment using Vision Transformer (ViT) to encode the communication cards' images.

4.2.1.1 *Preliminary Experiments*

To evaluate the efficacy of different methods for representing communication cards, we fine-tuned BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020), a Brazilian Portuguese version of BERT, using a synthetic corpus of Brazilian Portuguese AAC-like communications (PEREIRA et al., 2023a). Before fine-tuning, however, it was necessary to transform the text-based corpus into a communication cards-based corpus. This transformation involved linking each word or Multi Word Expression (MWE) in the corpus to its corresponding pictogram using a unique identifier. We utilized the Brazilian Portuguese set of pictograms available in the ARASAAC database, which provides a list of keywords and corresponding glossary definitions for each pictogram. The ARASAAC dataset provides a diverse set of pictograms covering various communication contexts, each with keywords that can be used as captions for a pictogram-based AAC system. The keywords also have meaning descriptions, similar to a dictionary. Figure 13 illustrates three different pictograms with the keyword "banco" and how each pictogram has a different meaning.

The need for disambiguation arises when a single term corresponds to multiple pictograms. To resolve this, we utilized the Unsupervised Nearest Neighbors algorithm, specifically the ball tree algorithm, in conjunction with BERTimbau embeddings to identify the most relevant

Figure 13 – Examples of ARASAAC pictograms for the word "banco", its keywords and meanings.



Keyword: **banco**
Meaning: **Instituição financeira (financial institution).**

Keyword: **banco**
Meaning: **Assento, com encosto ou não, em que várias pessoas podem sentar-se (Seat, with or without backrest, on which several people can sit).**

Keyword: **banco**
Meaning:**Assento de veículo automotivo (Automotive vehicle seat).**

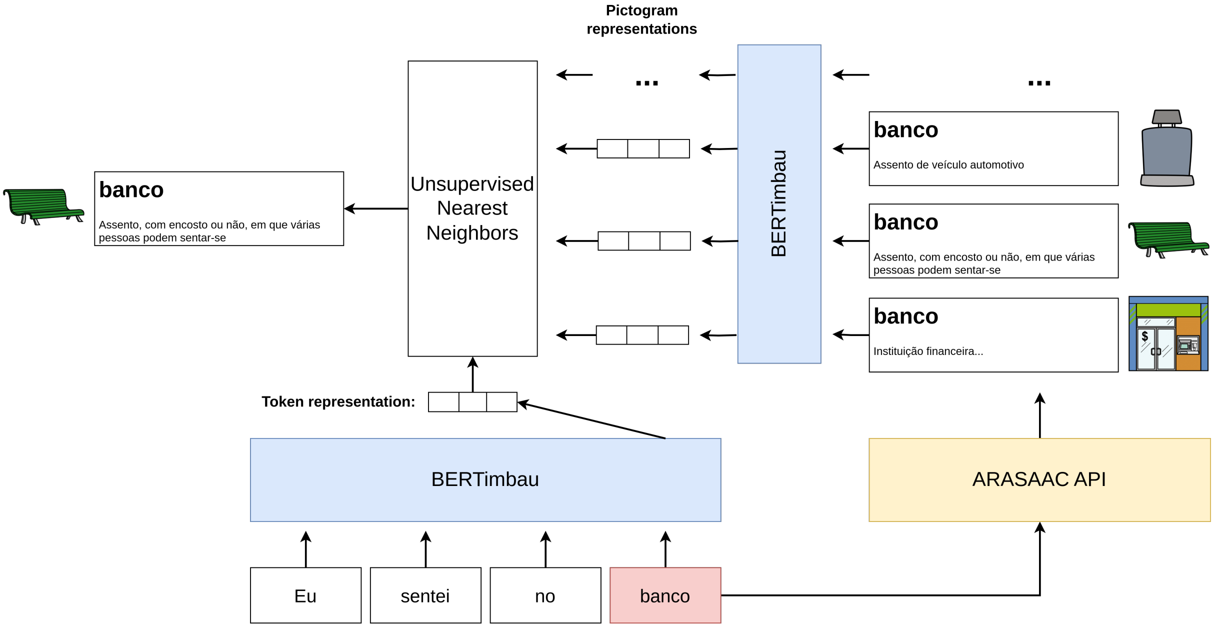
Source: The author (2023)

pictogram for each word in a given sentence. The ball tree algorithm, a variant of the Nearest Neighbors algorithm, is particularly adept at handling high-dimensional data. It divides the data into nested hyper-spheres, or "balls", facilitating quicker nearest neighbor queries.

In our study, we employed the ball tree algorithm to identify the most similar items, or pictograms, for each term in the corpus. The algorithm computes the distance between the BERTimbau embeddings of the terms and the pictograms, assigning each term to the pictogram with the smallest distance, i.e., the nearest neighbor. This Unsupervised Nearest Neighbors algorithm enabled us to associate each term in the corpus with the corresponding pictogram, thereby creating a suitable corpus for fine-tuning BERTimbau in this context. This method ensures that each term is linked to the most relevant pictogram, enhancing the accuracy and efficiency of the disambiguation process. To tokenize the sentences in the corpus, we utilized all the keywords in the ARASAAC vocabulary, including MWEs such as "fazer xixi" (pee) or "café da manhã" (breakfast). A multi-word expression tokenizer was employed to handle these expressions, resulting in a tokenized sentence like $S_t = \{ele,\ querer,\ fazer\ xixi\}$. The sentence was also lemmatized to match the lemmas used as keywords in the ARASAAC database. We searched for matching pictograms for each token in the ARASAAC database and performed disambiguation when more than one pictogram was found.

Our approach to disambiguation is similar to that of Scarlini, Pasini and Navigli (2020). As discussed in Section 2.5.2, each encoder layer of BERT outputs a representation of each word that is a contextualized embedding influenced by the other words in the sentence. These embeddings can provide meaningful vectorial representations of words or sentences applicable to various tasks (see Section 2.5.2). Typically, the vector produced by BERT for the special token [CLS] (appended at the start of each sequence) serves as the sentence representation. Conversely, the word representation can be derived by accessing the vector at its corresponding position. For instance, if a word is located second in the sequence following the [CLS] token, the output vector at the second position can be used as the word representation. In our study, we utilized BERTimbau for encoding the concatenated pictogram definitions. The summation of the vectors from the last four encoder layers of BERT for the token [CLS] was deemed the pictogram representation. Similarly, using the same strategy, we applied BERTimbau to encode the target token (i.e., the token requiring disambiguation), thereby obtaining a vector and a roster of encoded candidate pictograms for each token in the sentence. To determine the most appropriate pictogram for each token, we employed the Unsupervised Nearest Neighbors with the ball tree algorithm. Figure 14 illustrates this process using the sentence "eu sentei no

Figure 14 – Illustration of the process for automatically selecting a pictogram from ARASAAC to the word “banco” in the context “eu sentei no banco” (I sat on the bench).


Pictogram representations
...
...
banco
Assento de veículo automotivo
banco
Assento, com encosto ou não, em que várias pessoas podem sentar-se
banco
Instituição financeira...
BERTimbau
Unsupervised Nearest Neighbors
banco
Assento, com encosto ou não, em que várias pessoas podem sentar-se
Token representation:
BERTimbau
ARASAAC API
Eu
sentei
no
banco

Source: The author (2023)

banco” (I sat on the bench) as an example and using the word “banco” as the target word. Notably, the search for potential pictograms is conducted on ARASAAC, and BERTimbau is employed to encode the meaning of both the target word and the pictogram. Subsequently, the Unsupervised Nearest Neighbors algorithm with the ball tree algorithm selects the pictogram with the closest representation to the target word.

To fine-tune BERTimbau for communication card prediction in the context of this preliminary experiment, it is necessary to modify both the model vocabulary and the input embedding layer. BERT and BERTimbau rely on a vocabulary based on WordPiece (WU et al., 2016b), which breaks words into a limited set of sub-word units (e.g., “elefante” to “ele”, “##fante”). However, in this experiment, sub-word tokenization is unnecessary, as the tokens must be unique identifiers that cannot be divided into sub-units. We created a vocabulary of identifiers for ARASAAC pictograms. Since each pictogram has a unique identifier, we used a word-level tokenizer that splits sentences into tokens using white spaces as a delimiter. For example, the sentence illustrated in Figure 15 is tokenized as “6481 31141 16713”.

Changing the vocabulary requires changing the embeddings layer, also. Intuitively, we tell the model that we changed the vocabulary to use a new language, and the new embedding vectors represent the terms in this new language. In our experiments, we extract embeddings from four sources: 1) the pictogram caption (i.e., word or expression); 2) the pictogram caption synonyms; 3) the pictogram glossary definition from ARASAAC; and 4) the pictogram image.

We use the input embeddings layer from BERTimbau as a basis for the caption embeddings. Formally, given a vocabulary V composed of words and MWEs $(w_1, ..., w_n)$, the BERTimbau original embedding $B \in \mathbb{R}^{h \times D_b}$, where h is the size of the hidden state and D_b is the BERTimbau vocabulary size, and given a new embeddings matrix $P \in \mathbb{R}^{h \times D_v}$, where $D_v = |V|$, for each token t_i in V, populate P with the t_i embeddings from B. For MWEs, the embeddings of each token are extracted from BERTimbau's embeddings layer to a matrix $E \in \mathbb{R}^{h \times n}$, where h is the dimensionality of the embedding, and n is the number of tokens in the expression. We use the mean vector $\overline{E}$ as the expression's embedding representation. We use an approach similar to (DUDY; BEDRICK, 2018) for caption synonyms. First, we search in ARASAAC for the list of keywords for each pictogram. The pictogram representation is the average of the embeddings of its keywords retrieved from BERTimbau's original embeddings layer.

We use the definitions from ARASAAC concatenated with keywords to generate embeddings from the pictogram definition. A pictogram in ARASAAC lists keywords, each with a definition. We concatenate this list as

$$keyword_0||definition_0||...||keyword_n||definition_n \tag{4.1}$$

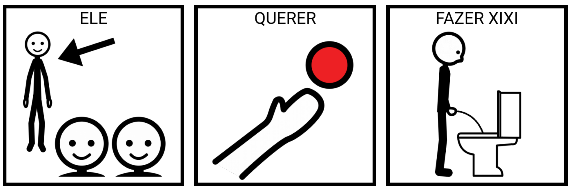
, which we use to compute the pictogram vector using two extraction methods. The first extraction method considers the mean vector of the definition extracted from B (i.e., BERTimbau input embeddings). The second method uses the BERTimbau last encoders layer outputs for the $[CLS]$ token[1]. We also computed representations from pictogram images using a ViT model pre-trained on ImageNet-21k (14 million images, 21,843 classes) and fine-tuned on ImageNet 2012 (1 million images, 1,000 classes) (DOSOVITSKIY et al., 2020)[2].

We fine-tune with a batch size of 768 sequences with 13 tokens (768 * 13 = 9,984 tokens/batch). Each data batch was collated to choose 15% of the tokens for prediction, following the same rules as BERT: If the i-th token is chosen, it is replaced with 1) the

[1] BERT tokenizer adds the $[CLS]$ token at the beginning of the processed sentences. This token output representation is generally used as input for classification models.

[2] Available at <https://huggingface.co/google/vit-base-patch16-224>

Figure 15 – The sentence *Ele quer fazer xixi* (he wants to pee) represented using ARASAAC pictograms.



Source: The author (2023)

Table 5 – Evaluation results of comparing the manners to represent a communication card. Results are sorted by $ACC@1$. The K values for $AAC@K$ represent different grid sizes commonly used in AAC boards.

Method	PPL	ACC@1	ACC@9	ACC@18	ACC@25	ACC@36
Pictogram captions	15.433	0.237	0.530	0.620	0.657	0.702
Pictogram synonyms	14.282	0.225	0.511	0.604	0.647	0.698
Pictogram definition [input embeddings mean]	23.368	0.209	0.492	0.580	0.627	0.673
Pictogram image + synonyms	122.407	0.042	0.169	0.220	0.255	0.293
Pictogram definition [mean last layer]	22.496	0.019	0.122	0.206	0.246	0.295
Pictogram image	106.130	0.007	0.037	0.078	0.112	0.146
Pictogram image + caption	89.685	0.007	0.038	0.076	0.111	0.146
Pictogram definition [CLS] last layer	89.107	0.003	0.062	0.117	0.153	0.203

Source: The author (2023)

$[MASK]$ token 80% of the times, 2) a random token 10% of the times or 3) the unchanged i-th token 10% of the times. We use the same optimizer as BERT (DEVLIN et al., 2019): Adam, with a learning rate of 1×10^{-5} for all model versions, with $\beta_1 = 0.9$, $\beta_2 = 0.999$, L2 weight decay of 0.01, and linear decay of learning rate. Fine-tuning was performed in a single 16GB NVIDIA Tesla V100 GPU for 200 epochs for the captions and synonyms versions and 500 epochs for the other versions. The definition- and image-based versions require more training time because the input vectors are from a different vectorial space than the BERTimbau embeddings. The model needs more time to adjust the parameters to these new vectors.

4.2.1.2 Results

Table 5 presents the results obtained by testing each version of the models in terms of top-k accuracies (ACC@K) and perplexity (PPL). ACC@K is a performance evaluation measure to assess the accuracy of predictions generated by the model. The ACC@K metric is calculated by checking if the correct prediction is within the top K predictions made by the model. If the correct prediction is within the top K, then the ACC@K is 1; otherwise, it is 0. The ACC@K value is then averaged over all instances to give a proportion or percentage of correct predictions within the top K predictions. A higher ACC@K value indicates better accuracy, with a perfect score of 1.0 indicating that the correct prediction is always within the top K predictions and a lower score indicating lower accuracy. Various K values were utilized to assess the accuracy of the predictions, including 1, 9, 18, 25, and 36, which represent different grid sizes commonly found in AAC boards.

Perplexity (generally noted as PPL or ppl) is the inverse probability assigned to the test set by the language model, normalized by the number of unique words in the vocabulary (JURAFSKY; MARTIN, 2019). Intuitively, if a model gives a high probability to the test set, the information there is not surprising to the model. Thus, it has lower perplexity, indicating a good comprehension of language. For example, for a test set $W = w_1, w_2, ..., w_N$:

$$PP(W) = P(w_1, w_2, ..., w_N)^{-\frac{1}{N}} = \sqrt[N]{\frac{1}{P(w_1, w_2, ..., w_N)}} \tag{4.2}$$

In which the probability of W can be expanded with the chain rule:

$$PP(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i|w_1, ..., w_i - 1)}} \tag{4.3}$$

Where $P(w_i|w_1, ..., w_i - 1)$ is the probability of the i-th token given the previous $i - 1$ (i.e., the context). Thus, considering a bigram model, we have:

$$PP(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i|w_i - 1)}} \tag{4.4}$$

Notice that because of the inverse in Equation 4.3, the higher the conditional probability of the word sequence, the lower the perplexity.

Perplexity can also be obtained by exponentiating the cross-entropy (Equation 2.6). In doing so, we consider perplexity as the average number of words encoded using $H(W)$.

$$PP(W) = 2^{H(W)} = 2^{-\frac{1}{N} log_2 P(w_1, w_2, ..., w_N)} \tag{4.5}$$

Perplexity does not properly apply to BERT MLM, as in BERT, the cross-entropy is calculated only for masked tokens. However, BERT also assigns the probability of a given sentence to exist in a test set by assigning the probability of each word when no masked token is inputted into the model. From the sentence probability, we can calculate the cross-entropy and the perplexity.

The results show that the model in which the embeddings were calculated using the pictogram caption synonyms has the lowest perplexity. This means that this model better understands how the language present in the test set works. Consequently, it can perform better generalization in diverse scenarios, making it more adaptable and versatile. Conversely, the model that extracted embeddings solely from the pictograms' captions demonstrated superior accuracy. Accuracy is a direct reflection of how correct the predictions of a model are. In this case, the higher accuracy indicates that this model was more successful in making correct predictions. However, the difference between these two models across all metrics may not be substantial enough to indicate one as superior to the other definitively. It's important to note that the choice of the best model can depend on the specific requirements of the task at hand. For instance, if the task prioritizes the general understanding of language and adaptability to different scenarios, the model with lower perplexity might be more suitable. On the other hand, if the task values the correctness of individual predictions, the model with higher accuracy might be the better choice. Therefore, selecting the best model should be guided by the specific objectives and constraints of the task.

Regarding the models in which the pictogram definitions were used to compute embeddings, using the mean vector of the definition extracted from the BERTimbau input embeddings were shown to be more effective. Using the BERTimbau outputs as the definition representation did not show good results, with higher perplexities and lower accuracies. Fine-tuning BERTimbau using these embeddings may require more training data and time, for the vectors are from a vectorial space different from the model's original. The same happens to the models using embeddings computed from pictogram images and their combinations. Based on these models' training and validation loss curves, there is still space for improvement, as the measures keep falling even after 500 epochs.

4.2.1.3 Preliminary Experiments Conclusions

Therefore, based on the metrics presented in Table 5, the best way to represent a pictogram in the proposed method is using the pictogram caption or its synonyms. Which of these two approaches to use depends on the vocabulary characteristics. For example, it is impossible to use synonyms if no synonyms dataset is available. However, if the same caption is used for two different pictograms in a vocabulary, it may be difficult for the model to disambiguate them. Using the pictograms' concept, as in PictoBERT (PEREIRA et al., 2022), can solve these problems. However, it should be noted that for some languages, such as Portuguese, a well-established lexical database like Princeton WordNET (MILLER, 1995) for English may not be available. In such cases, using pictogram definitions could be considered as an alternative. However, the results showed that using definitions performed worse than using only captions or synonyms. Moreover, encoding communication cards based on their definition might require more time and resources than using only captions. In this work, we present a novel approach that can solve this problem efficiently (cf. Section 4.3)

4.2.2 Model Adaptation

The experimental results in Section 4.2.1 showed that using captions is sufficient to represent a communication card for performing predictions using a BERT model. This way, communication card prediction can be compared to a word prediction task. The main difference between word prediction and communication card prediction in AAC is the presence of MWEs. MWEs are common in communication cards as they convey complex meanings more concisely than individual words. For instance, the expression "Good morning" is often represented by a single pictogram in communication cards. Similarly, the expression "I want to go to the bathroom" can be represented by a single pictogram that conveys the same meaning. In this Section, we detail how to adapt BERT to perform communication card prediction in AAC systems considering the MWEs.

The main difference between the changes made in BERT that we describe in this section and the changes described in Section 4.2.1.1 is that in this section, we do not modify the entire embeddings matrix of BERT. Instead, we add the tokens representing MWEs to its existing vocabulary. This approach assumes that MWEs can be effectively represented as single tokens in the model vocabulary. By doing so, we can capture the meaning of MWEs more accurately and avoid tokenizing them into their constituent parts, which can result in a loss of semantic information. For example, in the Portuguese sentence "Eu gosto de café da manhã" (I like breakfast), "café da manhã" is a multi-word expression that should be treated as a single token in the model vocabulary. By adding it to the vocabulary, we ensure that the model can learn to represent its meaning more accurately.

To add MWEs to the BERT vocabulary, we first extracted MWEs with two or three words from the ARASAAC vocabulary for Brazilian Portuguese. We limited it to three words because expressions with more than three words may contain complete sentences in which prediction is not required. Next, we tokenized each MWE and computed the mean vector representation of its constituent tokens from BERT's original input embeddings. We then added these MWE tokens to the model vocabulary, along with their corresponding vector representations. This allowed us to handle MWEs in the input sentences and enabled BERT to provide better representations for multi-word expressions in the communication card prediction task. This approach allows for the BERT model to handle MWEs as well as out-of-vocabulary (OOV) words using the WordPiece tokenization algorithm. Adding the MWEs to the model's vocabulary, BERT can represent these expressions as a single token. They can use its pre-trained WordPiece tokenization algorithm to handle OOV words. This way, the model can handle more complex language expressions, such as idiomatic expressions and domain-specific terminology, which are often challenging for AAC systems.

When using CS to predict communication cards, it is necessary to include the tags representing the structural roles in the model's vocabulary, such as "<who>" and "</who>". This is done by averaging the embeddings of each token within the tag. Using CS may be more adequate for bidirectional models like BERT because they can better capture the relationships between the different structural elements. For instance, consider the sentence "<who> I </who> <verb> drink </verb> <what> water </what>". In this example, BERT can more easily understand that "I" is the subject acting for "drink" and that "water" is the object of the verb. Thus, incorporating the structural role tags of CS in the model's vocabulary can improve the model's ability to predict communication cards.

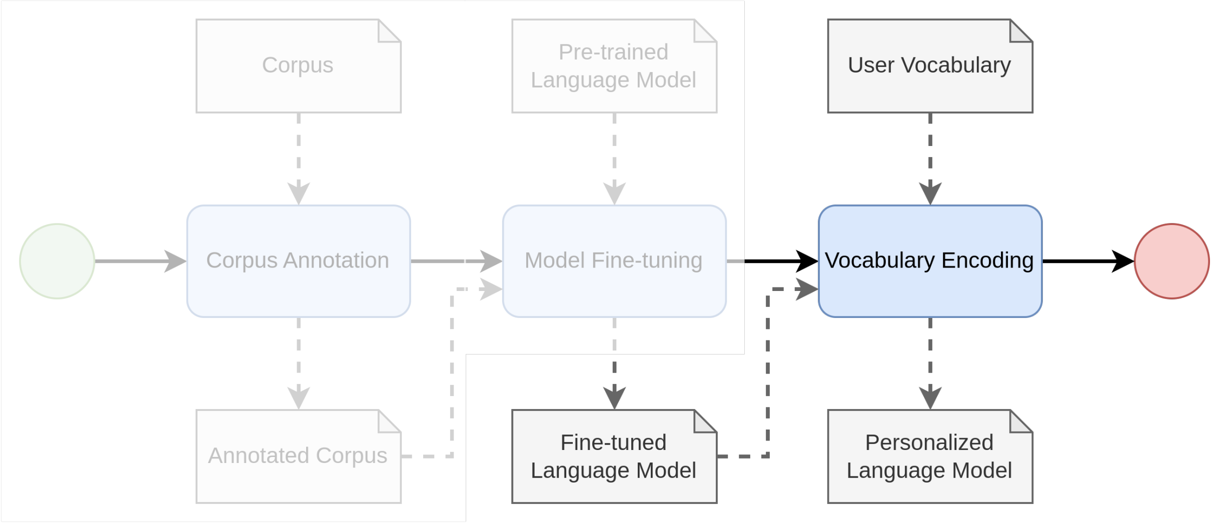
4.3 VOCABULARY ENCODING

This section focuses on the Vocabulary Encoding step, the most critical step in PrAACT. It enables language models to be used in a zero-shot setting, allowing users to communicate flexibly and adaptably without requiring pre-defined sentences. As shown in Figure 16, the inputs for this step are the fine-tuned language model produced in the previous step or a pre-trained model without fine-tuning, and the user vocabulary, including the communication cards with their caption.

This step takes advantage of the fact that transformer-based models like BERT share weights between the input embeddings layer and the decoder layer in the MLM head (cf. Section 2.5.2). In BERT, the decoder layer is a linear layer that predicts the masked tokens given the context. Specifically, during training, BERT masks some input tokens and trains the model to predict the original tokens. The decoder layer receives the word representations outputted by the encoder layers of the transformer (hidden states) for the masked tokens. It produces logits, which are then transformed into probabilities by a softmax function. During inference, the decoder layer generates new tokens given the hidden states, making it an important part of the BERT's MLM head model. We can change the model decoder layer to force it to produce probabilities to a different vocabulary by changing the weight matrices.

This modification involves updating the decoder layer to produce logits for the new vocabulary, which represents the unnormalized probabilities for each vocabulary item. As depicted

Figure 16 – The Vocabulary Encoding step enables users to communicate flexibly and adaptively without predefined phrases or sentences by taking advantage of shared weights in the input embeddings layer and decoder layer of transformer-based models like BERT.

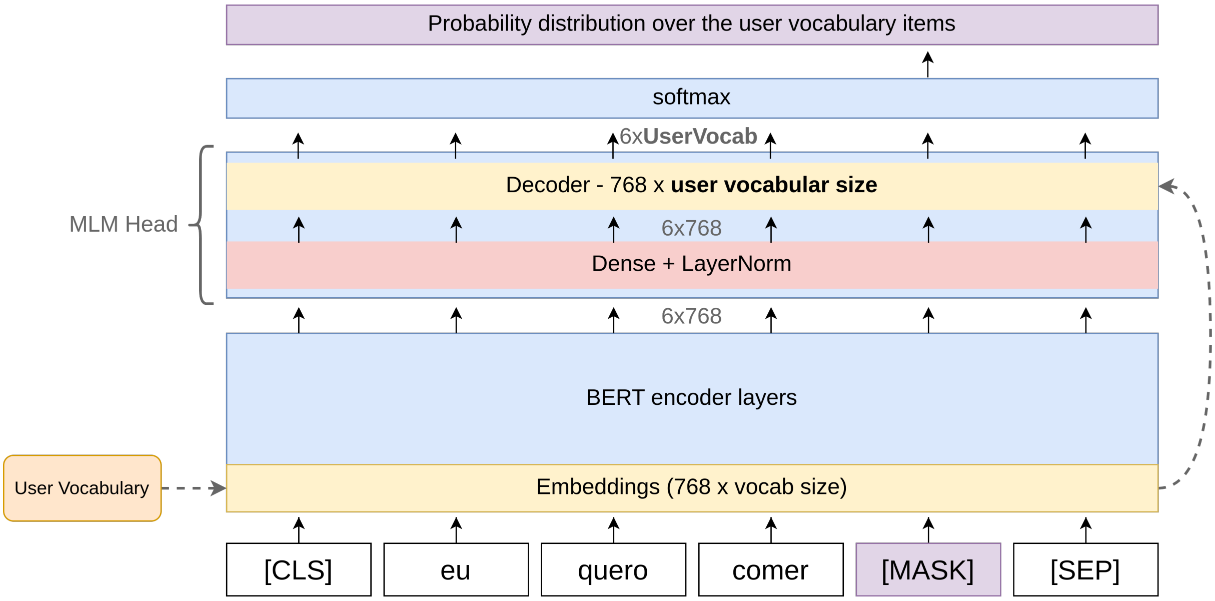


Source: The author (2023)

in Figure 17, the user vocabulary items are passed through the embedding layers to generate an embedding vector for each item. These resulting embeddings replace the original decoder layer's weights, thereby enabling the model to generate new tokens from the user vocabulary. Subsequently, the softmax function is applied to these logits, transforming them into a probability distribution over the user vocabulary. This softmax function normalizes the logits and produces a probability distribution, where each item's probability corresponds to its likelihood of being the correct output.

To generate the logits for the new vocabulary, we first encode the user vocabulary into vectors from the same vectorial space as the model's original input embeddings. We use the model's original embedding layer to perform this encoding. Each vocabulary item is passed through the embedding layer to extract its representation. The proposed method in this work enables the addition of MWEs to the model vocabulary, as mentioned in Section 4.2.2. However, the current section presents a method step that allows for the adjustment of the model to a new specific vocabulary, which may include new MWEs due to user preferences, regional variations, and other factors. For MWEs that are not in the model vocabulary, we combine the representations of the multiple words that make up the expression. For example, if the verbal expression "wake up" is not included in the model vocabulary, it will be tokenized as ("wake", "up"), and the communication card vector is the combination (e.g., average) of the vectors of "wake" and "up". This produces a $h \times |V|$ matrix, where h is the model's hidden states size and $|V|$ is the user vocabulary size. This matrix is used as the decoder layer weights, allowing

Figure 17 – Vocabulary Encoding using the Embeddings Layer. The user vocabulary items are passed through the embedding layers to generate an embedding vector for each item. The resulting embeddings replace the original decoder layer's weights, allowing the model to generate new tokens from the user vocabulary. The number 6 refers to the input length, and the number 768 refers to the BERT-base hidden states size


Probability distribution over the user vocabulary items
softmax
6xUserVocab
MLM Head
Decoder - 768 x user vocabular size
6x768
Dense + LayerNorm
6x768
BERT encoder layers
User Vocabulary
Embeddings (768 x vocab size)
[CLS]
eu
quero
comer
[MASK]
[SEP]

Source: The author (2023)

it to output logits for the new vocabulary.

The same modification can be applied to transformer-based language models like GPT-2. Unlike BERT, GPT-2 is trained as a left-to-right autoregressive language model, predicting the next token given the context. GPT-2's decoder is the language modeling head, which shares weights with the embeddings layer, allowing for generating new tokens given some context. The language modeling head is a linear layer followed by a softmax function that outputs the probabilities of the next token given the context. By replacing the language modeling head with a linear layer with the user vocabulary size and the weights computed from the embeddings of the user vocabulary, we can generate tokens from the user vocabulary. This modification ensures the new vocabulary has vectors from the same embedding space as the model's original input embeddings.

5 EXPERIMENTS

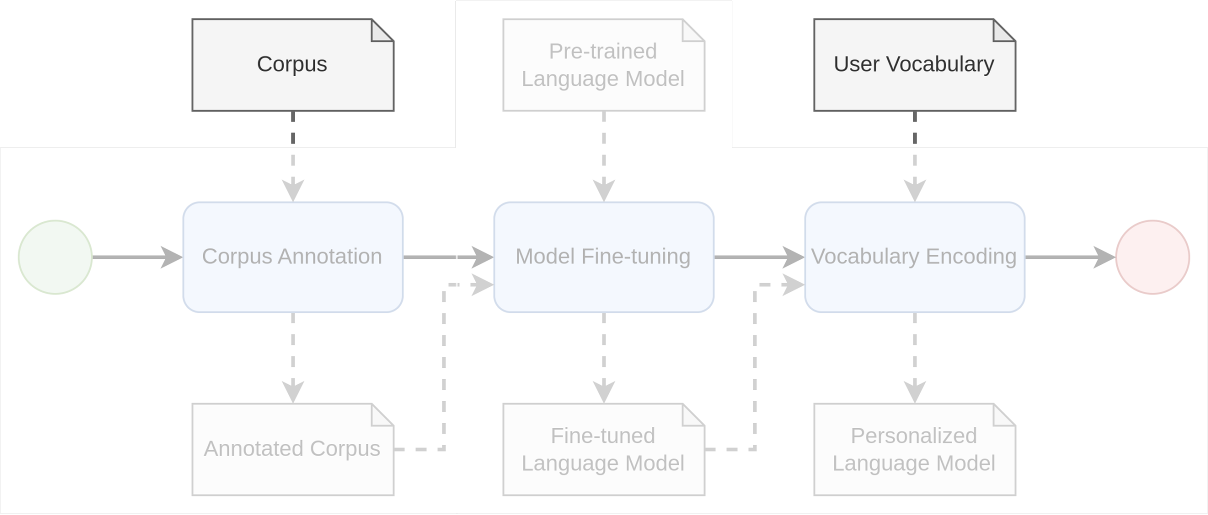
In this chapter, we present the experiments to evaluate the quality of the models produced following the PrAACT method. The performed experiments aim to 1) compare the models constructed using the method proposed in this work, which requires little or no additional training rather than a model pre-trained in the task, and 2) compare a model that uses the CS's semantic roles during predictions with a model that does not. Section 5.1 presents the datasets used and the annotation details. Section 5.2 details the models used in the comparison and the fine-tuning or pre-training implementation details. In Section 5.3, we present the details of a relevance ranking evaluation, which assesses the ability of models to prioritize relevant communication cards when completing a sentence related to a specific topic (e.g., feeding). Finally, in Section 5.4, we present the details of the completion evaluation to assess our models' quality in predicting appropriate communication cards to complete a given sentence. All the results and discussions are presented in Chapter 6.

5.1 DATASETS

The proposed method is evaluated in two distinct languages: English and Brazilian Portuguese. This categorization allows for properly classifying each dataset in this section according to the corresponding language. For each language, we employed two types of datasets, namely: 1) an AAC controlled vocabulary that serves as the user vocabulary input for the proposed method, as described in Section 4.3; and 2) a text corpus utilized for fine-tuning the models, as detailed in Section 4.3. These datasets form two of the three inputs of the proposed method, while the pre-trained language model, discussed in Section 5.2, constitutes the third input, as shown in Figure 18.

Furthermore, we include the corpus employed for training PictoBERT (PEREIRA et al., 2022) for comparison purposes. Table 6 provides an overview of the datasets used in the study. It presents information on each dataset's language, size, description, and objective. We give more details about each dataset in the following subsections.

Figure 18 – Proposed method's data input. The corpus is used for fine-tuning the pre-trained model. That is, adapting the model to the AAC context. The user vocabulary is used for adapting the model to the user's daily communication.



Source: The author (2023)

5.1.1 Portuguese Controlled Vocabulary

For Brazilian Portuguese, we used the vocabulary constructed by AAC specialists from ComunicaTEA[1], a Brazilian association formed by parents of children with CCN aimed at helping other families to have access to and use AAC tools. The vocabulary was constructed using the Reaact AAC Platform[2] and is freely available for testing[3].

Figure 19 shows the first page of the used vocabulary. Each communication card has a

[1] <https://comunicatea.com.br/>

[2] <https://reaact.com.br/>

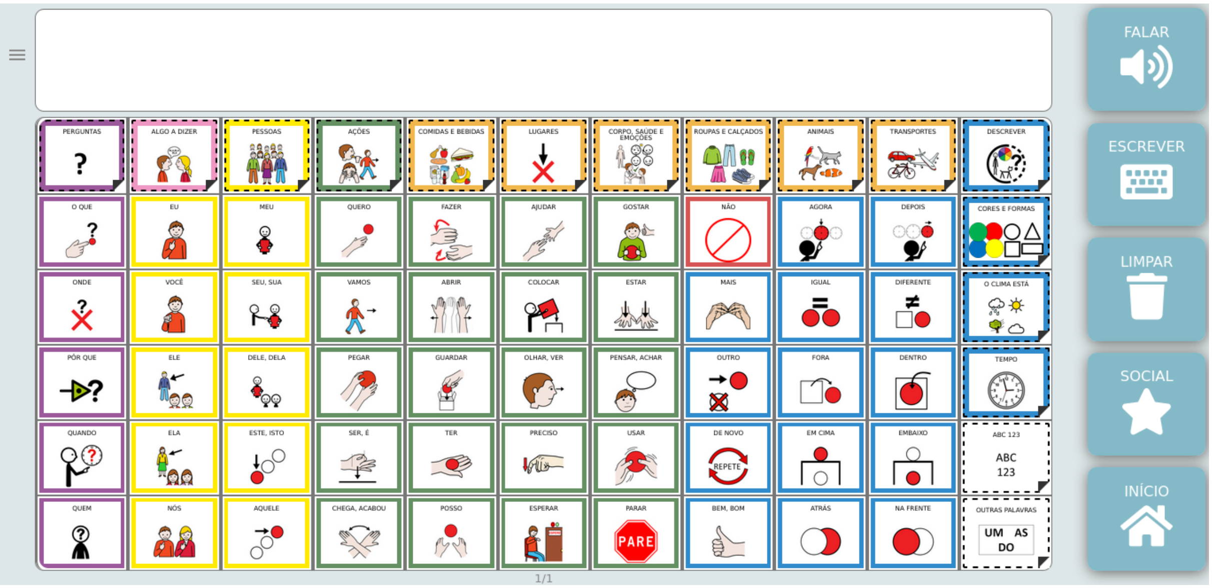
[3] Access <https://login.reaact.com.br/login> and click at "Testar Flipbook".

Table 6 – Summary of the datasets used in the experiments performed in this study. PT-Br stands for Brazilian Portuguese and EN for English.

Dataset	**Lang.**	**Size**	**Description**	**Objective**
ACCptCorpus	PT-Br	13K sentences	Composed of AAC-like communications annotated with the CS structure.	Fine-tune Brazilian Portuguese models.
ComunicaTEA vocabulary	PT-Br	978 communication cards	Common use vocabulary constructed by AAC specialists.	Assess the Brazilian Portuguese models' quality.
AACText	EN	7K sentences	Composed of AAC-like communications transformed into telegraphic language.	Fine-tune English few-shot models.
SemChildes	EN	955K sentences	Sub-set of CHILDES corpus annotated with word-senses.	Used for pre-training Picto-BERT.
CACE-en vocabulary	EN	715 communication cards	English version of the CACE-UTAC vocabulary translated from Spanish.	Assess the English models' quality.

Source: The author (2023)

Figure 19 – Screenshot of the first page of the ComunicaTEA vocabulary implemented in the Reaact AAC Tool. Notice that communication cards have a picture and a caption and that cards with dashed border lines are folders that contain related communication cards.

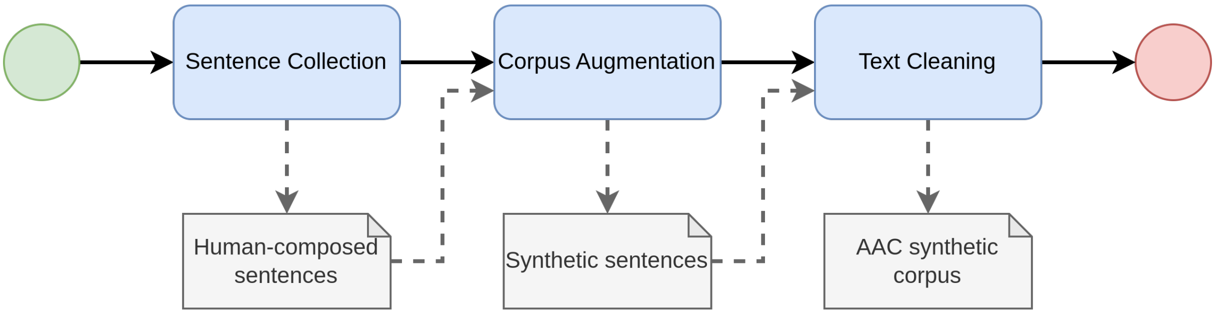


Source: The author (2023)

picture (pictogram or photo) and a caption with the word or expression represented by the picture. The first screen presents the most frequently used cards to the user, along with folders marked with dashed border lines such as “pessoas”, “ações”, “comidas e bebidas”, “lugares” and “animais”.

We remove from the vocabulary the folders “Algo a dizer” (something to say) and “Perguntas” (questions) and their respective communication cards. These folders contained cards with complete sentences or self-contained expressions, such as “o que está fazendo?” (what are you doing?) and “Tive uma ideia” (I got an idea), which do not require prediction and may be used alone. The resulting vocabulary consisted of 978 communication cards organized in 22 folders, including the first page as a folder. The cards from the excluded folders were not considered in the final count, as they are not used in sentence construction. Moreover, certain communication cards in the vocabulary have a comma separating the words in their captions, indicating that they can be used with multiple words. We included separate copies of such cards in the vocabulary to account for this, each with a distinct caption split. An example is the card with the caption “bem, bom” in Figure 19, which is transformed into two cards: one with “bem” as caption and the other with “bom”.

Figure 20 – Overview of the method used for constructing AACptCorpus, a synthetic corpus for AAC.



Source: The author (2023)

5.1.2 Portuguese Corpus

To our knowledge, Brazilian Portuguese has no corpus of the AAC domain. Existing approaches developed for English cannot simply be translated to Portuguese due to the two languages' distinct grammatical and syntactic structures. For example, Portuguese has a more complex system of verb conjugation, gender and number agreement, and word order than English. This section describes the method used to construct a corpus of sentences resembling those utilized in Brazilian Portuguese AAC systems. The method encompasses three key phases outlined in the sub-sections: sentence collection, corpus augmentation, and text cleaning, as depicted in Figure 20. We name the produced corpus ACCptCorpus (PEREIRA et al., 2023a).

5.1.2.1 Sentence Collection

For sentence collection, we invited 18 professionals, including speech therapists, psychologists, and parents of children with CCN, to inform the sentences they consider the most commonly constructed in different contexts using high-tech AAC. Each participant answered a questionnaire asking them to construct sentences about home, school, kitchen, and leisure contexts and sentences free of context. We collected 667 unique sentences, which we now refer to as human-composed.

This number of sentences may not be enough to cover AAC communication. However, they can be used as a reference to generate similar sentences based on similarity in vocabulary and structure (syntactic and semantics). As a vocabulary, we used the ARASAAC dataset (PALAO, 2019), which provides a diverse set of pictograms covering a variety of communication contexts. Each pictogram has a set of keywords, which can be used as captions for a pictogram-based AAC system. We use the Brazilian Portuguese ARASAAC list of keywords[4] as our vocabulary, removing punctuations and keywords with more than three tokens (e.g., *Material de estimulação sensorial*). We used the human-composed sentence as examples for generating new sentences with similar structures.

5.1.2.2 Text Augmentation

For corpus augmentation, we used GPT-3 (BROWN et al., 2020)[5] with a few-shot learning approach. We provide some examples to GTP-3 in the form of text prompts and ask it to produce new similar examples by completing our prompts. GPT-3 is an LLM trained to predict the next word. It is a powerful tool for text generation, for it can complete text prompts like the one shown in Figure 21. For composing the prompts, we shuffled the vocabulary items and divided them into groups of 20. Then, we randomly selected five words (or expressions) from each group and used them to search for example sentences from the human-composed set. We sample from 3 to 6 example sentences for each group and use them as few-shot examples.

[4] Available at <https://api.arasaac.org/api/keywords/br>

[5] We used `text-davinci-002` available via the OpenAI API.

Figure 21 – GPT-3 text prompt used for sentence generation.

```
Generate new Portuguese sentences using the words in this vocabulary: “delas”, “vizinho”, “avó”, “médico”, “bebê”, “pai”, “professor”, “policial”, “garota”, “profissões”, “primas”, “irmã”, “crianças”, “rapaz”, “avô”, “de vocês”, “motorista”, “filho”, “dentista”, “adulto”.
##
Example 1: eu tenho um filho e uma filha.
##
Example 2: eu vi meu filho feliz.
##
Example 3: nós gostamos delas.
##
Example 4: meu avô foi trabalhar.
##
Example 5: você é um grande professor.
##
Example 6: nós vamos seguir o professor.
##
Example 7:
```

Source: The author (2023)

5.1.2.3 Text Cleaning

We performed a data cleaning step on the automatically generated corpus. This step included (a) removing sentences containing offensive content, using the method proposed by (LEITE et al., 2020); (b) removing sentences with higher perplexities according to BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020), a Portuguese Brazilian version of BERT, selecting those in the first quartile to be removed; and (c) removing sentences with less than three or more than 10 tokens. To compute perplexity, the model utilizes a copy of the input sentence as the label and assigns a probability to each word. From this, the cross-entropy and perplexity can be derived (PEREIRA et al., 2022).

5.1.2.4 The Constructed Corpus

Table 7 presents the details of the produced corpus regarding the number of words and sentences. It provides information such as the total number of words, the number of unique words, the average sentence length, and the number of sentences. This information can be useful for understanding the overall structure and composition of the corpus.

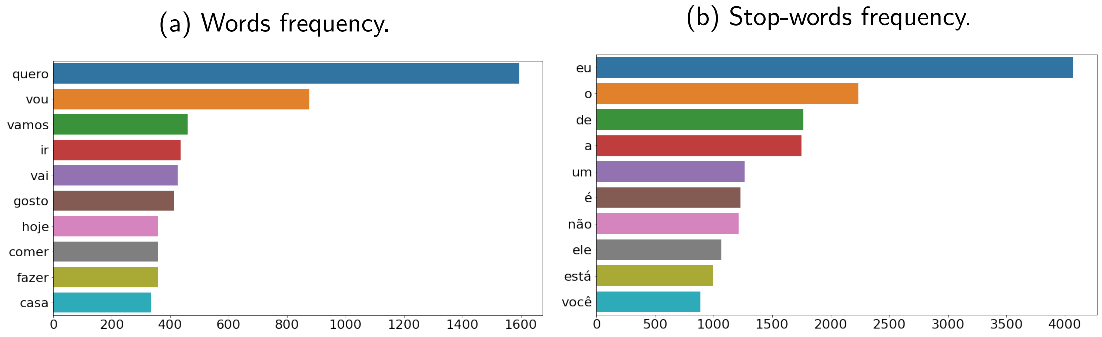
Figure 22 presents a chart that displays the frequency of words in the corpus, with a separate section for stop words, sorted by frequency. The chart provides an overview of the most common terms used in the corpus. It can help identify patterns or trends in the language used. The most frequent word (excluding stopwords) in the corpus is “quero” (i.e., “I want”), which indicates that the corpus might be focused on expressing wants or desires. In AAC, it is common for users to express their needs and desires, which makes the presence of the word “quero” not surprising. The chart also displays the frequency of stop words, which are the words that are not semantically meaningful, such as “o”, “a”, “de”, etc. Stop words in high frequency indicate that the corpus contains many common, everyday languages rather than specialized or technical ones. Overall, the chart in Figure 22 can be a useful tool for analyzing

Table 7 – Portuguese dataset summary.

Words		**Sentences**				
Total	**Unique**	**Total**	**Max Length**	**Min Length**	**Mean Length**	**Most Frequent Length**
89572	4758	13796	11	3	6	6 (3432 times)

Source: The author (2023)

Figure 22 – Frequency distribution of words in the constructed corpus.



(a) Words frequency.

(b) Stop-words frequency.

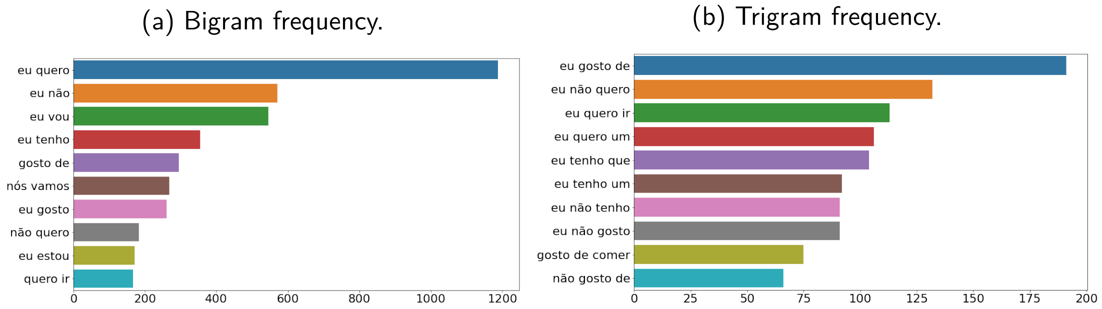
Source: The author (2023)

the language used in a corpus and gaining insight into the topics and themes it covers.

The chart in Figure 23 displays the frequency of word combinations, specifically bigrams and trigrams, in the corpus. Bigrams are combinations of two words, such as “I am,” and trigrams are combinations of three words, such as “I am going.” The chart is sorted by frequency, with the most frequent bigrams and trigrams appearing at the top. This type of analysis is useful for identifying common phrases and idiomatic expressions used in the corpus and understanding the relationship between words in the language. Additionally, it can provide insight into the style and tone of the text, such as whether it is formal or informal. Overall, the chart in Figure 23 can be a valuable tool for understanding the language used in the corpus at a deeper level. For example, the most frequent bigram is “eu quero” (I want), indicating that the corpus might be focused on expressing wants or desires. Additionally, it can be used to identify patterns in the language, such as specific conjunctions or prepositions, which can further inform the analysis of the corpus.

In Figure 24, we can see the human-composed corpus’s word and bigram frequency distributions. This figure is useful for comparing the distribution of the generated corpus with the human-composed corpus. The comparison of the generated corpus with the human-composed

Figure 23 – Frequency distribution of N-gram in the constructed corpus.

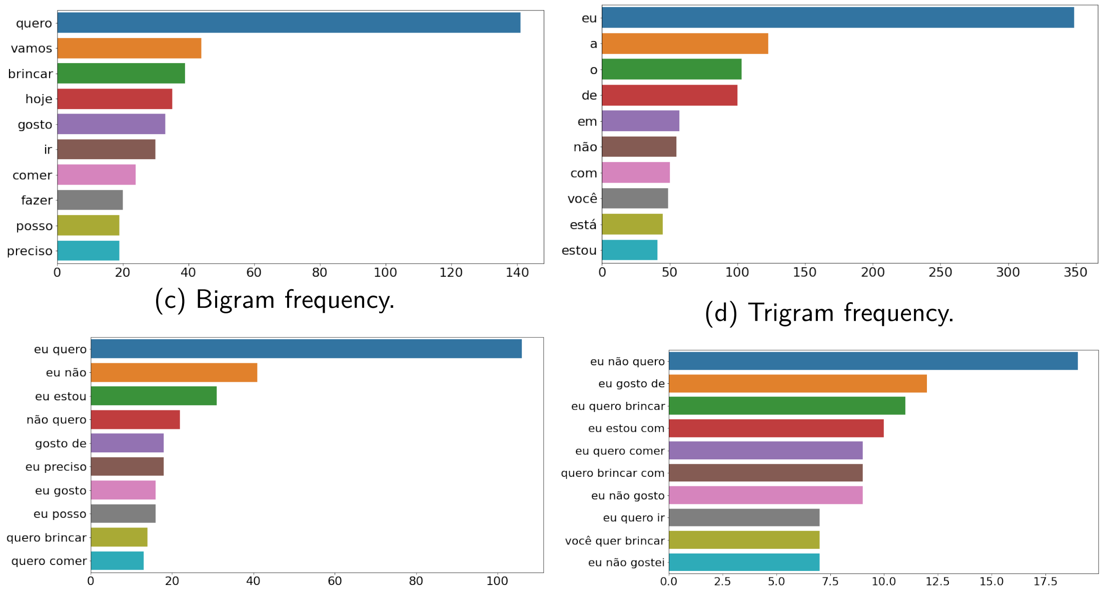


(a) Bigram frequency.

(b) Trigram frequency.

Source: The author (2023)

Figure 24 – Word and n-gram frequency distribution in the human-composed corpus.



(a) Words frequency.

(b) Stop-words frequency.

(c) Bigram frequency.

(d) Trigram frequency.

Source: The author (2023)

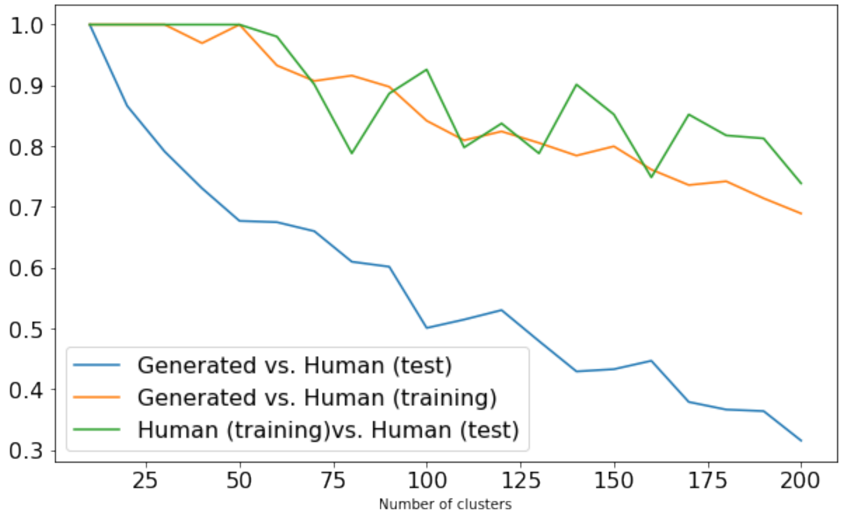
corpus is divided into four categories: words, stop-words, bigrams, and trigrams. When analyzing the distributions, we notice that the human-composed and generated corpus have similar distributions. Additionally, the top 10 most frequent words and stop-words of the human-composed corpus have a similar presence in the generated corpus.

The coverage of the constructed corpus can be used to evaluate the quality and representativeness of the automatically generated sentences. It is defined as the fraction of the sentences generated by the text augmentation method assigned to the same cluster as at least one human-composed sentence. To quantify the coverage, we employed a clustering-based approach to generate sentence embeddings for both the human-composed and augmented corpora. We used the k-means clustering algorithm to group the sentence embeddings into clusters. We used BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020) to generate sentence embeddings using the average vector outputted by the last 4 encoder layers to the $[CLS]$ token.

To evaluate the coverage of the generated corpus, we collected an additional 203 sentences from AAC specialists. This set is referred to as the test set of the human-composed corpus. The original 667 sentences collected from the specialists constitute the training set of the human-composed corpus. The test set provides a means of measuring the quality and reliability of the generated corpus by comparing its content with the human-composed sentences.

The line chart in Figure 25 depicts the coverage ratio of three different scenarios: the blue line represents the coverage ratio of the automatically generated corpus over the test set of

Figure 25 – Coverage of the automatically generated corpus over the human-composed sentences.



Source: The author (2023)

the human-composed corpus. The orange line represents the automatically generated corpus coverage ratio over the human-composed corpus training set. Finally, the green line represents the coverage ratio of the test set of the human-composed corpus over the training set.

As the number of clusters increases from 10 to 200, we can observe that the blue line (coverage of the automatically-generated corpus over the test set of the human-composed corpus) decreases deeper than the other two lines. This can be explained by the fact that the human-composed corpus is smaller than the generated one, leading to a decrease in coverage as the number of clusters increases. However, it is important to note that both the orange and green lines remain relatively stable throughout the range of the number of clusters, showing that the coverage of the auto-generated corpus over the training set and the test set of the human-composed corpus over the training set, respectively, is not significantly affected by the number of clusters.

The results demonstrate that the generated corpus is semantically similar to the original human-composed corpus, with a coverage ratio of up to 0.7 for the training set of the human-composed corpus, even when a large number of clusters is used. The coverage ratio is slightly lower but still significant for the test set of the human-composed corpus, remaining up to 0.5 with fewer than 130 clusters utilized.

5.1.2.5 Corpus Annotation

For annotating the Portuguese Corpus (ACCptCorpus), we utilized the Portuguese model available in the Stanza NLP tool. This model allowed us to turn the corpus into a telegraphic format, making it easier to analyze and extract key information. Furthermore, we also employed the CS roles to provide a more detailed annotation of the corpus. To extract the syntactic structure and semantic roles from the corpus, we used dependency parsing and SRL. Dependency parsing allowed us to extract the subject+verb+object structure of each sentence. We used the Portuguese model from Stanza NLP (QI et al., 2020). SRL was used to extract adverbial complements such as location, time, and manner. This allowed us to identify and label the semantic roles associated with each sentence constituent. For SRL, we used the InVeRo semantic parser (CONIA et al., 2020). By combining these two techniques, we could annotate our corpus with both syntactic and semantic information, which merged in the CS roles (e.g., who, what doing, and what). For example, let us consider the sentence "Eu comi pipoca na escola hoje" (I ate popcorn at school today) from AACptCorpus. The dependency parsing technique identified the subject "eu" (I), the verb "comi" (ate), and the object "pipoca" (popcorn). The adverbial complement "na escola" (at school) was identified and labeled as the location semantic role, "hoje" (today) as the time semantic role by the SRL technique. This allowed us to assign the semantic roles of "who" (eu), "what doing" (comi), "what" (pipoca), "where" (na escola), and "when" (hoje) to the different constituents of the sentence, according to the CS framework.

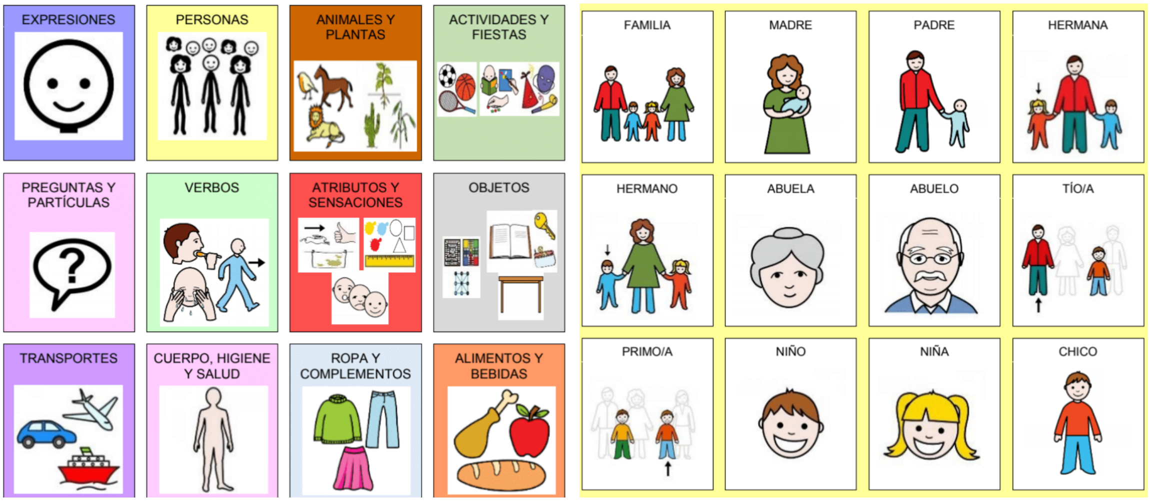
5.1.3 English Controlled Vocabulary

For the English version of the vocabulary, we utilized a translated version of the CACE-UTAC vocabulary [6] originally in Spanish. The translation of CACE posed fewer challenges than the translation of ComunicaTEA vocabulary since it only employs ARASAAC pictograms, enabling us to use the ARASAAC API to obtain translations. Additionally, it facilitated mapping the communication cards in the vocabulary to WordNet concepts. ARASAAC provides a mapping between pictograms and synsets, which is advantageous in testing the PictoBERT model based on WordNet concepts.

The translation process of the CACE-UTAC vocabulary to English involved searching for

[6] <https://www.utac.cat/descarregues/cace-utac>

Figure 26 – Communication cards in CACE-UTAC vocabulary. (a) presents dome of the original CACE categories before translation. (b) show the communication cards inside the "persons" folder.



(a) Example of CACE folders.

(b) Cards inside the *Persons* folder.

Source: The author (2023)

ARASAAC pictograms that matched the ones used in CACE and listing the words related to that pictogram in English. It is important to note that ARASAAC has multiple words associated with a single pictogram in each language, which resulted in a list of potential translations for each communication card. A human expert then assessed each list and chose the ARASAAC word that was the best translation for the original Spanish caption.

The CACE-UTAC vocabulary used in this study, similarly to ComunicaTEA, is organized into folders that group communication cards by semantic categories, such as "foods" or "descriptors". The vocabulary comprises a total of 24 folders and 715 pictograms. However, since CACE is originally in Spanish and contains some regional expressions, we excluded some of the cards that might not be used in other Spanish-speaking countries or may not have a direct translation in English. Figure 26 shows a few examples of CACE folders and communication cards in their original Spanish version.

5.1.4 English Corpora

In this section, we present two corpora for English. The first is the AACText (VERTANEN; KRISTENSSON, 2011) corpus, which consists of 6K sentences that resemble AAC-like communications. The second corpus is used for pre-training the PictoBERT model, the Semantic CHILDES (SemCHILDES), which is not used in our experiments but is still relevant to mention.

The AACText corpus consists of fictional AAC-like communications comprising approximately 6K sentences, divided into training, test, and development sets. The authors used Amazon Mechanical Turk to construct the corpus to create many messages that model conversational AAC. Workers were asked to invent messages using a scanning-style AAC interface for communication. The authors found that their crowdsourced collection better modeled conversational AAC than datasets based on telephone conversations or newswire text. The authors also leveraged their crowdsourced messages to intelligently select sentences from larger Twitter, blog, and Usenet datasets. We use the AACText corpus for fine-tuning the English few-shot models. To adapt the AACText corpus for use in our communication card prediction method, we first transformed it into a telegraphic language corpus. To achieve this, we removed sentences containing commas, which are typically not used in pictogram-based AAC systems. We then used the Spacy English parser to extract the POS and lemmas of the words in the remaining sentences. We included two versions of each sentence in the corpus: the telegraphic and the natural language version. While AAC communications are generally telegraphic, users may use natural language or include prepositions or verb inflections. This resulted in 7K sentences for the training set and 1K for both test and validation.

The SemCHILDES dataset was created by Pereira et al. (2022) for pre-training PictoBERT for pictogram prediction. The SemCHILDES is a large corpus (955,489 sentences) of North American English from the CHILDES database (MACWHINNEY, 2014). To use it for pre-training PictoBERT, the authors labeled part of CHILDES with word-senses using SupWSD (PAPANDREA; RAGANATO; BOVI, 2017) and attributed a sense key to each content word (verb, nouns, adjectives, and adverbs). Functional words (e.g., pronouns and prepositions) were kept in their original form, and the result is a large word-sense-labeled dataset suitable for training a language model. An advantage of SemCHILDES over other word-sense labeled datasets is that it comes from conversational data, which is more similar to the type of language used in AAC communication. The authors also annotated part of the British English corpus of CHILDES with semantic roles, allowing for the fine-tuning of PictoBERT to perform pictogram prediction based on the CS semantic structure.

5.2 MODELS

In this section, we present the models used in our experiments, depicted in Table 8. We used three types of models: pre-trained, fine-tuned, and zero-shot. The pre-trained model we used is PictoBERT, pre-trained for pictogram prediction in the AAC context. The fine-

Table 8 – Summary of the models used in the experiments performed in this study.

Model	Type	Lang.	Training Size	Dataset
PictoBERT	Pre-trained	EN	955K sentences	SemCHILDES
BERT-AAC few-shot	Fine-tuned	EN	7K sentences	AACText
BERT-AAC zero-shot	Zero-shot	EN	-	-
GPT-2-AAC	Zero-shot	EN	-	-
BERTptAAC	Fine-tuned	PT-br	13K sentences	AACptCorpus
BERTptCS	Fine-tuned	PT-br	13K sentences	AACptCorpus w/ CS roles

Source: The author (2023)

tuned models, which we also call few-shot models, are those that we fine-tuned from models pre-trained for word prediction like BERT. Zero-shot models are those for which we do not have any adjustments on their weights but change only language modeling head weights with the controlled vocabulary embeddings. We detail the models, the modifications, and the implementation details in the following subsections.

5.2.1 PictoBERT

PictoBERT (PEREIRA et al., 2022) is a model trained for predicting the next pictogram in a sentence constructed using AAC boards. The model is an adaptation of the BERT in which the input embeddings have been modified to allow word-sense usage instead of words, considering that a word-sense represents a pictogram better than a simple word. PictoBERT outperforms the previously used n-gram models and knowledge bases for the same task. The model can also be fine-tuned to adapt to different users' needs, making transfer learning its main characteristic.

The vocabulary for PictoBERT is derived from word senses and functional words from SemCHILDES. The sense keys (e.g., `person%1:03:00::`) from WordNet represent word senses in the corpus while functional words (e.g., pronouns) are in their original form (e.g., I). To create PictoBERT, the BERT vocabulary was changed to a Word Level based, using SemCHILDES to train a Word Level tokenizer. The embeddings layer in BERT was also modified using ARES embeddings (SCARLINI; PASINI; NAVIGLI, 2020) to replace the original BERT embeddings. The ARES embeddings were chosen because they were computed using BERT and are in the same vectorial space as the BERT's original embeddings. For each position in the vocabulary occupied by a word-sense, the vector referent to its sense-key in ARES is inserted into the PictoBERT embedding layer. The original BERT embedding is used for functional words.

In addition, multi-word expressions that are not in WordNet are represented using BERT's tokenizer and averaging their embeddings from BERT. Finally, PictoBERT is trained using transfer learning from BERT's pre-trained weights.

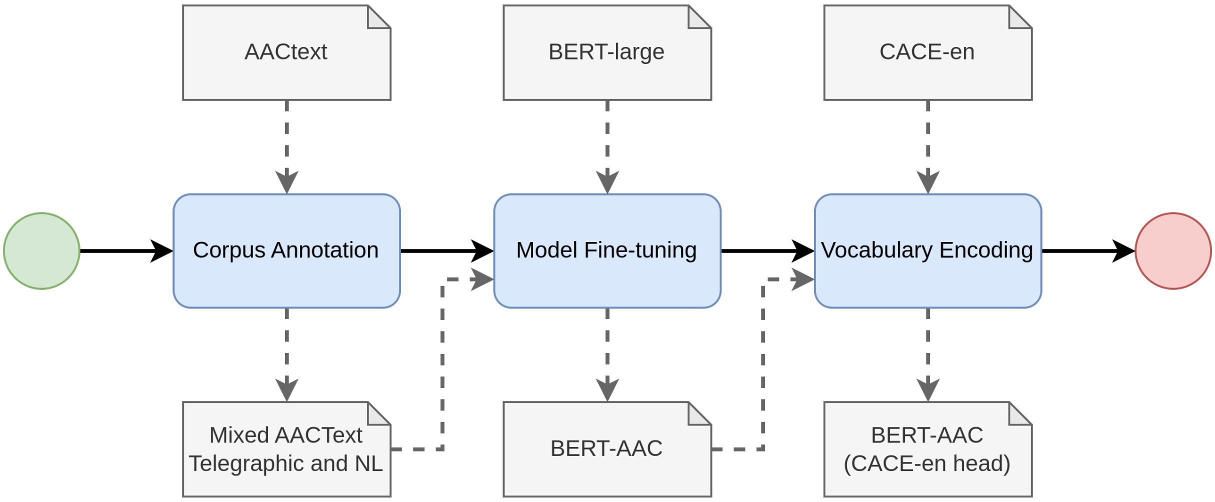
PictoBERT was trained using the North American part of the SemCHILDES dataset, divided into 98/1/1 splits for training, validation, and testing. The training and validation sets were used for pre-training, with a batch size of 128 sequences, each containing 32 tokens. Each data batch was collated to select 15% of the tokens for prediction using the same rules as BERT. Specifically, the selected token was replaced with [MASK] token 80% of the time, a random token 10% of the time, and the original token 10% of the time. PictoBERT was trained for 500 epochs using the Adam optimizer, with a learning rate of 1×10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, L2 weight decay of 0.01, and linear decay of the learning rate. The training was conducted on a single 16GB NVIDIA Tesla V100 GPU, with each epoch taking 20 minutes. The total training time for each version would be approximately 166.7 hours (500 epochs x 20 minutes/epoch = 10,000 minutes = 166.7 hours) or approximately 7 days, assuming continuous training. The model weights were adjusted using the training split, while the validation split was used to monitor the model's performance during training.

5.2.2 Fine-tuned models

This sub-section presents the fine-tuned models, modifications, and implementation details. The objective of fine-tuning was to adjust the pre-trained models to communication card prediction considering the particularities of the domain. This includes considering the structure of the language used, for example. The fine-tuning examples shown in this section are instantiations of PrAACT, the method proposed in this work and presented in Chapter 4. Therefore, the process for fine-tuning each model is illustrated by extending the diagram in Figure 9.

Figure 27 illustrates PrAACT instantiated to construct BERT-AAC, a BERT-based model for communication card prediction in English AAC systems. As presented in Chapter 4, PrAACT consists of three main steps: Corpus Annotation, Model Fine-Tuning, and Vocabulary Mapping. In this instantiation, the Corpus Annotation step takes AACText as input. It outputs a mixed corpus with telegraphic and natural language sentences, which reflects the particularities of the language used in AAC systems. The Model Fine-Tuning step takes the mixed corpus and the pre-trained BERT-large as input and fine-tunes the model to predict communication cards in the AAC domain using MLM. Finally, the Vocabulary Mapping step receives the fine-

Figure 27 – PrAACT adapted to construct BERT-AAC, a version of BERT-large for communication card prediction in English AAC systems.



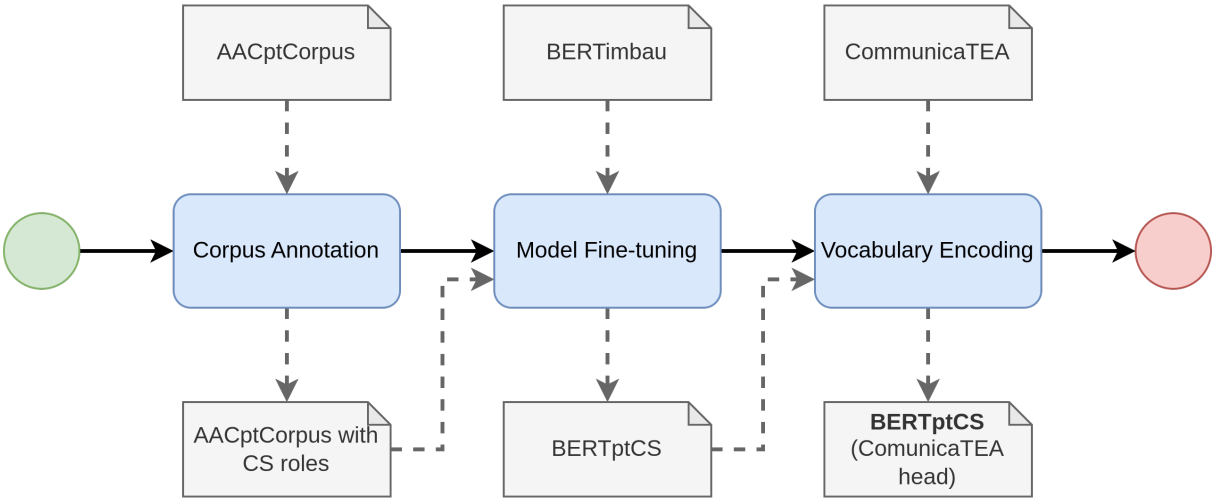
Source: The author (2023)

tuned model and the CACE-en vocabulary as input. It outputs a BERT-like model that can predict communication cards using CACE-en vocabulary cards. The model input is raw text (telegraphic or not) with the $[MASK]$ token on the place where the communication card prediction is needed, such as the end of a sentence: "I want [MASK]".

BERT-AAC was fine-tuned with a batch size of 128 sequences with 20 tokens (192 * 20 $=$ 2,560 tokens/batch). Each data batch was collated to choose 15% of the tokens for prediction, following the same rules as BERT: If the i-th token is chosen, it is replaced with 1) the $[MASK]$ token 80% of the time, 2) a random token 10% of the time or 3) the unchanged i-th token 10% of the time. We use the same optimizer as BERT (DEVLIN et al., 2019): Adam, with a learning rate of 1×10^{-5} for all model versions, with $\beta_1 = 0.9$, $\beta_2 = 0.999$, L2 weight decay of 0.01, and linear decay of learning rate. Fine-tuning was performed in a single 16GB NVIDIA Tesla T4 GPU for 50 epochs, each taking approximately 1 minute.

For the vocabulary mapping step, the BERT-large tokenizer is used to tokenize the captions of each communication card in the CACE-en vocabulary. Next, the input embeddings layer of the BERT-large model is used to extract a vector representation for each token in the captions. The communication card embedding vector is then obtained by taking the mean of the representations of its caption tokens. For instance, the card with the caption "work out" is tokenized as ["work" and "out"], and its vector representation is calculated as the mean of the embeddings of "work" and "out". Thus, $e(\text{"work out"}) = \frac{e(\text{"work"})+e(\text{"out"})}{2}$, where e is the embeddings matrix, and 2 is the number of tokens in the caption. The decoder layer in the BERT MLM head is replaced with a Linear transformation layer that maps the hidden states

Figure 28 – PrAACT adapted to construct BERTptCS, a version of BERT that performs communication card prediction in Brazilian Portuguese considering the CS (e.g., who? what doing? what?). This illustration also applies to BERTptAAC, which does not use CS.



Source: The author (2023)

output of BERT's encoders into the output classes, where the number of classes is equal to the CACE-en vocabulary size. The softmax function is applied to the linear layer output at inference time to convert them into a probability distribution over the CACE-en vocabulary. Each value represents the probability of each item in the vocabulary replacing the $[MASK]$ token in a sentence like "I want to eat [MASK]".

Figure 28 depicts adapting PrAACT to construct BERTptCS, a Brazilian Portuguese version of BERT that predicts communication cards using the CS framework. The process of constructing BERTptCS is similar to the one used for BERT-AAC. First, the AACptCorpus is annotated using CS roles, which involve identifying a sentence's subject, verb, object and adverbial complements (i.e., location, manner, and time). We used Stanza NLP to extract Subject-Verb-Object (SVO) structures and InVeRo for semantic parsing, as mentioned in Section 5.1.2.5. The resulting annotated corpus contains sentences such as "<quem> eu </quem> <verbo> querer comer </verbo> <o_que> pipoca </o_que>", which indicate the subject ("eu"), verb ("querer comer"), and object ("pipoca").

We employed BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020) as the pre-trained model as input for the fine-tuning step. To incorporate the CS roles, such as "<quem>" (who) and "<o_que>" (what), into the original vocabulary of the model, we added corresponding vectors to the embedding layers that represent these new tokens. To achieve this, we first tokenized the CS roles and then captured the representation of each token from the BERTimbau original embedding layer. The role representation was obtained as the mean vector of its constituent tokens. For instance, the role "<o_que>" was tokenized as "<", "o", "_", "que", ">", and its corresponding representation vector was computed as the mean vector of these tokens. The reason for using the vector representations of tokens like "<" and ">" is to prevent the model from generalizing the role's meaning to similar tokens. If we just used the representations of "o" and "que" tokens, the model might generalize the role "<o_que>" to other contexts where "o" and "que" are used together, even when they do not represent the same role. By adding the special tokens "<" and ">", we ensure that the model learns to associate these representations only with their intended roles, avoiding potential confusion.

BERTptCS was fine-tuned using a batch size of 384 sequences with 33 tokens (384 * 33 = 12,672 tokens/batch). Each data batch was collated to choose 15% of the tokens for prediction, following the same rules as BERT: If the i-th token is chosen, it is replaced with 1) the $[MASK]$ token 80% of the time, 2) a random token 10% of the time or 3) the unchanged i-th token 10% of the time. We use the same optimizer as BERT (DEVLIN et al., 2019): Adam, with a learning rate of 1×10^{-5} for all model versions, with $\beta_1 = 0.9$, $\beta_2 = 0.999$, L2 weight decay of 0.01, and linear decay of learning rate. Fine-tuning was performed in a single 16GB NVIDIA Tesla T4 GPU for 50 epochs.

The vocabulary encoding process used for BERTptCS was performed similarly to that of BERT-AAC, where mean vectors were utilized. The input data consisted of communication cards in the ComunicaTEA vocabulary, and each card's caption was tokenized using the BERTimbau tokenizer. Next, we extracted a vector representation for each token in the captions from the BERTimbau input embeddings layer. The communication card embedding vector was then computed as the mean of the representations of its caption tokens. This method ensures that the resulting embeddings contain information from all the words in the caption and can represent the communication card accurately. We replace the weights of the BERTimbau MLM head decoder layer with the encoding resulting embeddings. This way, the model can produce a probability distribution over the ComunicaTEA vocabulary for completing a sentence like "<quem> eu </quem> <verbo> ir </verbo> <onde> [MASK] </onde>".

For the construction of BERTptACC, the process is similar to that of BERTptCS. However, in BERTptACC, we do not use CS tags (e.g., <quem> </quem>) in the training text. Instead, we keep the words in the corresponding roles positions: <who> <verb> <what> <how> <where> <when> to ensure the usage of direct order sentences. Unlike BERTptCS, there is no modification in the model's input vocabulary for BERTptACC. Additionally, the training dataset for BERTptAAC contains more minor sequences compared to BERTptCS, requiring fewer computational resources.

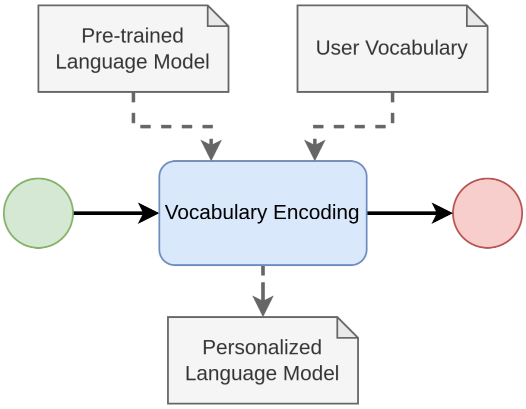
To fine-tune BERTptAAC, we used a batch size of 512 sequences with 17 tokens (512 * 17 $=$ 8,704 tokens/batch). Each data batch was collated to choose 15% of the tokens for prediction, following the same rules as BERT: If the i-th token is chosen, it is replaced with 1) the $[MASK]$ token 80% of the time, 2) a random token 10% of the time or 3) the unchanged i-th token 10% of the time. We use the same optimizer as BERT (DEVLIN et al., 2019): Adam, with a learning rate of 1×10^{-5} for all model versions, with $\beta_1 = 0.9$, $\beta_2 = 0.999$, L2 weight decay of 0.01, and linear decay of learning rate. Fine-tuning was performed in a single 16GB NVIDIA Tesla T4 GPU for 50 epochs.

5.2.3 Zero-shot models

In Figure 29, we present the part of PrAACT that enables using pre-trained large language models without additional training. This approach is known as a zero-shot approach as it requires no training for the specific task at hand. The model is a modified version of a pre-trained model to produce a probability distribution over the communication cards' vocabulary.

Adapting the pre-trained models involves replacing the decoder layer in the language modeling head with the embeddings matrix that represent the communication cards in a given vocabulary. We evaluated this approach with two large pre-trained models, BERT-large and GPT-2, and used the CACE-en vocabulary. For BERT-large, we utilized the same method as for BERT-AAC to encode the vocabulary and replace the decoder, and we named the adapted model BERT-AAC zero-shot. For GPT-2, we computed the communication card embeddings using the model's input embeddings layer and updated the Linear layer used as the language modeling head with the resulting vectors. We named this model GPT-2-AAC.

Figure 29 – Proposed zero-shot approach for adapting pre-trained language models to perform communication-aware language tasks without additional training. The approach involves modifying the decoded layer in the language modeling head, enabling models like BERT and GPT-2 for card prediction.



Source: The author (2023)

5.3 RELEVANCE RANKING EVALUATION

The Relevance Ranking Evaluation aims to assess the ability of different models to prioritize relevant communication cards when completing a sentence related to a specific topic. Specifically, considering the sentence “I want to go _____”, which must be completed by a location, we evaluate how well the models rank the cards related to the location topic compared to other cards. This experiment aims to measure the ability to identify and rank the most relevant cards for a given sentence, such as identifying the most relevant location cards to complete the sentence. The results of this experiment provide insights into how the model can be further improved to enhance its usability for individuals with communication impairments.

For the Relevance Ranking Evaluation, we focused on assessing only the English models due to the availability of a pre-trained model (i.e., PictoBERT) to compare against. This experiment evaluated the quality of the few-shot and zero-shot models produced by the proposed method against a pre-trained model. Specifically, we compared PictoBERT (pre-trained) with BERT-AAC few-shot, BERT-AAC zero-shot, and GPT-2-AAC zero-shot models. The goal was to assess the ability of these models to rank communication cards related to a specific topic (e.g., location) when completing a given sentence. This experiment provides insights into the quality of models produced by the proposed method and their potential for improving the communication of individuals with CCN.

We use the test set of the SemCHILDES corpus for the Relevance Ranking Evaluation. The ground-truth cards used in this experiment were selected from the CACE-en vocabulary. To pre-process the dataset, we selected only the sentences where the last token was in the CACE-en vocabulary and belonged to one of the four folders: “food”, “beverage”, “places”, “time”, and “attributes”. An example of a sentence that would be considered for this evaluation is “I want to eat _____” where the model needs to predict a food as the relevant card. This way, the cards within the folder “foods” are considered the ground-truth references. The model’s task, in this case, is to attribute higher probabilities to the cards within the folder “foods” than to any other card in the vocabulary. Using PictoBERT in this scenario requires mapping the controlled vocabulary to the word senses and functional words in the models’ vocabulary. This is done as described in Section 5.1.3.

In this experiment, we used the Area Under the Receiver Operating Characteristic (AUROC) as the evaluation metric to measure the ability of the models to prioritize the relevant communication cards when completing a sentence. This metric evaluates the model’s ability to correctly rank the relevant cards higher than others. To calculate the AUROC score, the items in the CACE-en vocabulary labeled with the folder “food”, “beverage”, “places”, “time”, and “attributes” were considered as the ground truth references (A). In contrast, the remaining items were considered as B. The probability scores for all the items in B was calculated using the model, and the items in B were sorted by their probability scores in descending order. The true positive and false positive rates were calculated, and the AUROC scores were calculated for the ROC curve. A higher AUROC score indicates better model performance in correctly ranking the relevant communication cards.

The AUROC can be expressed mathematically as:

$$AUROC = \int_0^1 TPR(FPR)^{-1} dFPR, \tag{5.1}$$

where TPR is the true positive rate, defined as the fraction of items in A that are correctly ranked, and FPR is the false positive rate, defined as the fraction of items not in A that are incorrectly ranked. The integral is taken over the FPR from 0 to 1. The integrand numerator represents the tangent line’s slope to the ROC curve at a given point. The denominator is the horizontal distance between the given and the point (0,1). The AUROC score ranges from 0 to 1, where a score of 1 indicates perfect performance and a score of 0.5 indicates random guessing.

5.4 COMPLETION EVALUATION

The Completion Evaluation experiment aimed to assess the quality of the models in predicting appropriate communication cards to complete a given sentence. Through this experiment, we aimed to identify the best-performing model and potential areas for further improvement in the prediction of communication cards. In this experiment, we compare the performance of models trained with and without incorporating the roles of CS. Specifically, we compared the accuracy of models trained on sentences with and without these roles. The goal was to assess whether incorporating CS roles improves the prediction accuracy of communication cards.

In this experiment, we used two models, BERTptCS and BERTptAAC. We chose these models because they were trained on the same dataset but with different features. BERTptCS incorporates CS roles while BERTptAAC does not. By comparing the performance of these two models, we could assess the impact of including CS roles in the training data on the prediction accuracy of communication cards. Additionally, since both models are based on the same BERT architecture and were trained in the same language (i.e., Brazilian Portuguese), we could isolate the impact of the inclusion or exclusion of CS roles on the model's performance.

For this experiment, we used the test set from the AACptCorpus, consisting of 667 sentences constructed by humans. To evaluate the ability of the models to predict appropriate communication cards to complete a given sentence, we masked the last token of each sentence, excluding the Colourful Semantics roles tags. To illustrate, a sentence like "<quem> eu </quem> <verbo> comer </verbo> </o_que> pipoca </o_que>" is transformed into "<quem> eu </quem> <verbo> comer </verbo> </o_que> [MASK] </o_que>". For the BERTptAAC model, we mask the sentence as follows: "eu comer [MASK]".

As mentioned in Chapter 3, previous work on communication card prediction has used keystroke saving, MRR, top-k accuracy, and perplexity as metrics for automatically evaluating pictogram prediction models. However, perplexity may not be the most suitable metric to assess the quality of models in a sentence completion task. Perplexity is commonly used to evaluate language models based on their ability to predict the next word given some context. However, in a sentence completion task where the model is asked to predict the missing word in a given sentence, the focus is not only on the likelihood of the predicted word but also on whether it is the correct word that makes sense in the context. Still, it does not take into account the appropriateness or relevance of the predicted items. Additionally, keystroke saving is unsuitable for our experiment, as the methods we are comparing will not change the AAC system grid or folders, so the number of selections to construct a given sentence should be the same. Therefore, in this experiment, we use top-k accuracy (ACC@K) and MRR to assess the quality of the predictions made by the models. For ACC@K, we use different values of K (1, 9, 18, 25, 36) to simulate the grid sizes in AAC systems. ACC@K measures the proportion of times that the correct communication card appears within the top K predicted cards.

In addition to top-k accuracy and MRR, we add Entropy@K to our metrics set, as it provides insight into the diversity of the predicted pictograms, which is essential in AAC scenarios where users may have a limited vocabulary. Entropy@K measures the uncertainty of the top-K items suggested by a model. The metric calculates the entropy of the probability distribution of the top-K predictions. This way, the higher the Entropy@K score, the more uncertain the model's predictions are. Entropy@K can be calculated as:

$$Entropy@K = -\frac{1}{K}\sum_{i=1}^{K} log(p(y_i|X)), \tag{5.2}$$

where K is the number of predictions to consider, $p(y_i|X)$ is the predicted probability (in log scale) of the i-th pictogram given the input sentence X. This equation measures the entropy or "surprise" of the model's predictions up to the top K items for each input. A lower score indicates more confident and consistent predictions across different inputs.

6 RESULTS

This chapter presents the findings obtained from the experiments conducted to evaluate the proposed models' performance in the prediction of communication cards. This chapter is divided into three sections: Relevance Ranking Evaluation, Completion Evaluation, and Models Analysis. The Relevance Ranking Evaluation (cf. Section 6.1) assesses the models' ability to prioritize relevant communication cards when completing a sentence related to a specific topic. The Completion Evaluation (cf. Section 6.2) compares the performance of models trained with and without incorporating Colourful Semantics roles. Finally, in the Models Analysis (cf. Section 6.3), we present an analysis comparing the models' predictions.

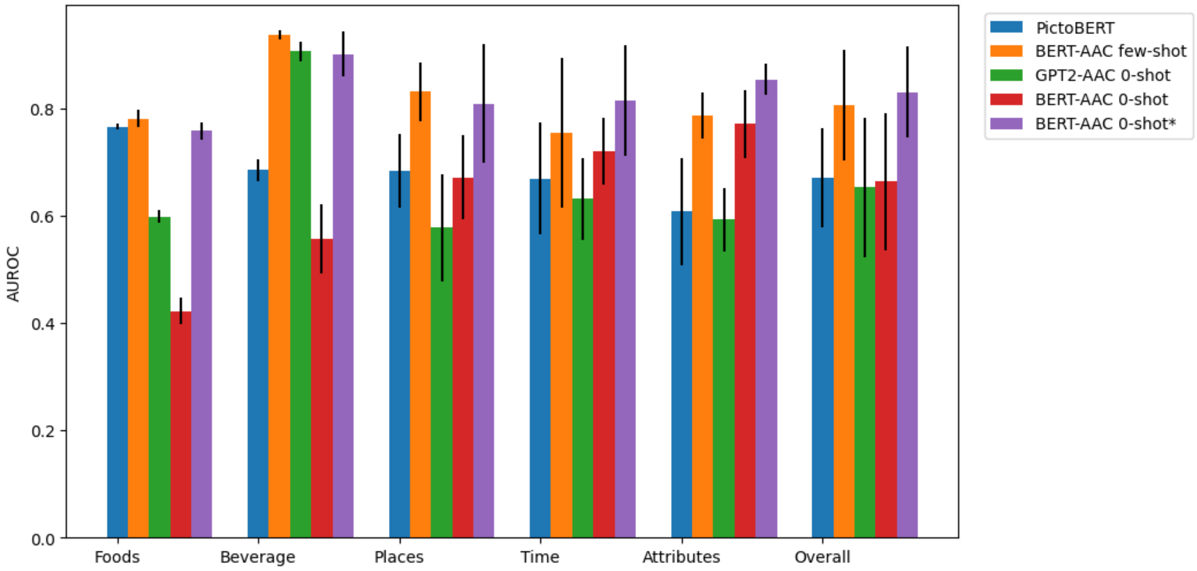
6.1 RELEVANCE RANKING EVALUATION RESULTS

Figure 30 shows the average AUROC and the standard deviation (Std) of different models' predictions to complete sentences from four different topics: Foods, Beverage, Places, Time, and Attributes. The models are evaluated on each topic separately.

The results presented in Figure 30 indicate that the BERT-AAC few-shot model, constructed using all PrAACT's stepts, outperformed the pre-trained PictoBERT on all topics evaluated. Although the PictoBERT was pre-trained with a larger corpus of 955K sentences, the BERT-AAC few-shot model, which was fine-tuned with only 7K sentences, demonstrated higher accuracy in completing sentences related to foods, beverages, places, time, and attributes. This demonstrates the efficacy of the proposed method. Additionally, implementing PictoBERT requires annotating the user vocabulary with word-sense keys from WordNet or other electronic lexicons, adding an extra disambiguation step. In contrast, BERT-AAC encodes the user vocabulary using the model input embeddings layer without requiring external resources. Therefore, the proposed method offers a more efficient and straightforward approach to communication card prediction without sacrificing accuracy.

The results of the zero-shot adapted models showed that they were slightly below the performance of the pre-trained models overall. This suggests that while zero-shot adaptation is a promising technique, there is still room for improvement. Regarding communication card prediction, using a pre-trained model is currently the most effective approach. However, zero-shot adaptation could be helpful in situations with specific vocabulary or context requirements

Figure 30 – AUROC average (AVG) and standard deviation (Std) of models' predictions to complete sentences from four different topics: Foods, Beverage, Places, Time, and Attributes. The model marked with an * is the BERT-AAC zero-shot evaluated with a (.) at the end of each sentence.

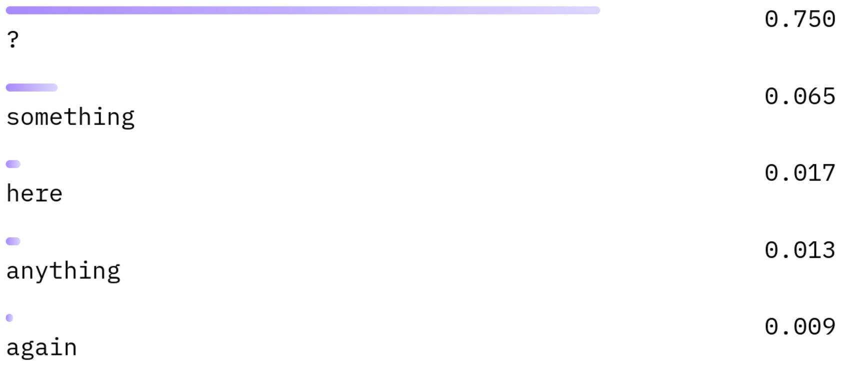


Source: The author (2023)

that a pre-trained model does not cover. Overall, these findings highlight the potential of both pre-training and zero-shot adaptation for improving the accuracy of communication card prediction models. In addition, these findings suggest that zero-shot models can be a good alternative when a large dataset for pre-training or fine-tuning is not available. The proposed method has demonstrated the ability to fit different vocabularies of different users without requiring additional training. This is a crucial advantage for applications that cater to diverse user groups with varying communication needs. Overall, the results suggest that zero-shot models can provide a viable and efficient solution for communication card prediction, especially when data availability is limited.

The results in Figure 30 show that the zero-shot version of BERT-AAC, denoted with a *, outperformed both the PictoBERT and the fine-tuned BERT-AAC in two topics: Time and Attributes. This happened because we added an ending dot to each sentence in the test set to improve the model's performance. BERT is a bidirectional model, and providing right-side context should enhance its ability to fill in the mask. For instance, if the sentence is “I want to eat [MASK]”, BERT should assign a high probability to punctuation tokens to fill the mask. However, in a usage scenario, adding a dot at the end of the sequence may not be appropriate, especially if the user wants to construct a question. Figure 31 shows the predictions of bert-large-uncased for the sentence “Do you want to eat [MASK].”, with the dot at the end of the sequence. We can observe that the model assigned a higher probability to the question

Figure 31 – Top-5 predictions of bert-large-uncased for the sentence "Do you want to eat [MASK]." with a dot at the end of the sequence. Screenshot taken from the Huggingace model card: <https://huggingface.co/bert-large-uncased>.



```
?              0.750
something      0.065
here           0.017
anything       0.013
again          0.009
```

Source: The author (2023)

mark "?" despite the added dot, which indicates that the dot may interfere with the model's performance. We present more comparisons of models' predictions in Section 6.3.

6.2 COMPLETION EVALUATION RESULTS

Table 9 shows the results of the top-n accuracy (ACC@K) and Mean Reciprocal Rank (MRR) of BERTptCS and BERTptAAC. The results demonstrate that both models, BERTptCS and BERTptAAC, perform better at higher values of K, as indicated by the increasing values of ACC@K. Additionally, BERTptCS outperforms BERTptAAC in all ACC@K metrics, with the most significant difference being at ACC@1. However, when looking at MRR, we can see that BERTptCS also outperforms BERTptAAC, but with a smaller margin. Yet, on average, BERTptCS is better at ranking the correct communication card prediction in the top positions of the list of candidates compared to BERTptAAC. Overall, these results suggest that using the CS structure can improve the accuracy of the communication card prediction model.

Table 10 presents the results of Entropy@K for the BERTptCS and BERTptAAC models. The entropy measures the uncertainty of the distribution of the predicted communication cards.

Table 9 – Results of top-n accuracy (ACC@K) and Mean Reciprocal Rank (MRR) for the BERTptCS and BERTptAAC models on the evaluation dataset.

Model	ACC@1	ACC@9	ACC@18	ACC@25	ACC@36	MRR
BERTptCS	0,50	0,74	0,81	0,85	0,87	0,58
BERTptAAC	0,45	0,69	0,77	0,80	0,84	0,53

Source: The author (2023)

Table 10 – Results of Entropy@K for BERTptCS and BERTptAAC.

Model	Entropy@1	Entropy@9	Entropy@18	Entropy@25	Entropy@36
BERTptCS	0,31	19,73	36,06	47,74	79,91
BERTptAAC	1,02	22,68	45,57	59,42	95,43

Source: The author (2023)

The lower the entropy, the more confident the model is in its predictions. The table shows that both models have higher entropy scores as the value of K increases. BERTptCS has lower entropy scores than BERTptAAC for all values of K, which indicates that BERTptCS is better at predicting the correct communication card, as it has a more concentrated distribution of probabilities. Additionally, BERTptCS has an entropy of 0.31 for Entropy@1, indicating that it makes a very confident prediction for the top candidate. In contrast, BERTptAAC has an entropy of 1.02, indicating it is less confident in its top prediction. This suggests that incorporating the CS structure in the training process can lead to a more accurate and confident communication card prediction model.

The results of the experiments show that BERTptCS outperforms BERTptAAC in terms of all metrics evaluated. Specifically, the ACC@K metric shows that BERTptCS has higher accuracy than BERTptAAC for all values of K (1, 9, 18, 25, and 36). Furthermore, the MRR values of BERTptCS are also higher than BERTptAAC, indicating that BERTptCS provides more relevant and accurate predictions. In addition, the Entropy@K metric shows that BERTptCS produces more uniform distributions of predicted tokens across all positions than BERTptAAC. These results demonstrate that incorporating the CS structure into the fine-tuning process of BERT improves the accuracy and relevance of the model's predictions and the uniformity of the predicted tokens' distribution. Therefore, we conclude that BERTptCS is a better model than BERTptAAC for predicting communication cards.

6.3 MODELS' PREDICTIONS ANALYSIS

This section aims to present a qualitative analysis of the predictions made by the different models to complete example sentences. The analysis aims to understand better how each model works and identify potential errors or limitations. In the following subsections, we present the analysis for both the English models (cf. Section 6.3.1) and the Portuguese models (Section 6.3.2). The qualitative analysis provides valuable insights into how the models perform and can help guide further improvements to their architecture and training.

6.3.1 English Models

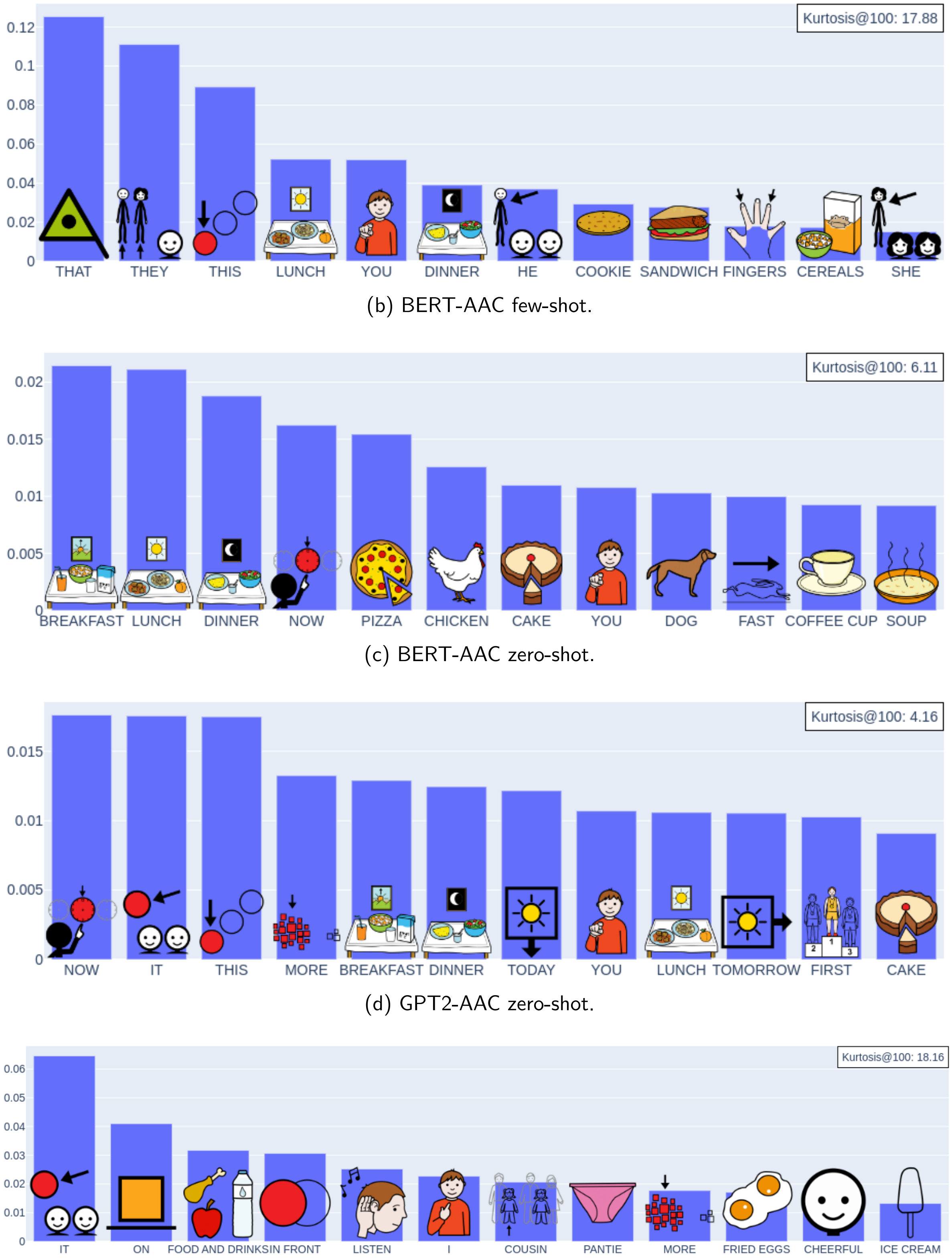
This section presents a qualitative analysis of the English models for communication card prediction. These models include PictoBERT, BERT-AAC in zero and few-shot settings, and GPT2-AAC in the zero-shot setting. We analyze the models' ability to predict the most appropriate communication cards based on the context of an incomplete sentence. Our analysis provides insights into the strengths and weaknesses of each model and highlights the challenges of adapting large language models to the specific needs of AAC systems. By comparing and contrasting the performance of different models, we aim to guide researchers and practitioners in the field of AAC who seek to develop more accurate and efficient communication aids.

In Figure 32, we present the top-12 predictions made by four English models for communication card prediction: PictoBERT, BERT-AAC zero and few-shot, and GPT2-AAC. The sentence to be completed was the telegraphic "I want eat". The predictions made by all models are meaningful and relevant for the given sentence, suggesting that they have learned relevant associations between words and concepts. However, we observed that GPT2-AAC assigned high probabilities to cards with captions that may not typically be used to complete such a sentence, such as "pantie", "listen", or "cousin", which may indicate that this model has some limitations in capturing the context and structure of AAC-related language.

PictoBERT is another model that assigned high probabilities to cards not usually used to complete the tested sentence, "I want eat." This could be attributed to the characteristics of the training corpus. As mentioned in Section 5.1.4, the corpus used for training PictoBERT is based on transcriptions of children's speech from CHILDES. The texts in CHILDES may contain figurative language from storytelling or reading children's books, which could explain the high probabilities assigned to items like *she*, *they*, *he*, and *fingers* (PEREIRA et al., 2022). It is possible that these words were used in the training corpus in contexts related to eating, resulting in the high probabilities assigned to them in the sentence completion task.

The BERT-AAC models accurately predicted suitable communication cards to complete the sentence "I want eat [MASK]". The models made meaningful predictions, except for the word "you" in the top-12 predicted cards, which may not be a usual completion for the given sentence. For the few-shot model, it was noticed that most of the top-12 communication cards represent eatable things such as "lunch" and "pizza". Other cards are adverbs that can describe

Figure 32 – English models' predictions for completing the sentence "I want eat".



(a) PictoBERT.

(b) BERT-AAC few-shot.

(c) BERT-AAC zero-shot.

(d) GPT2-AAC zero-shot.

Source: The author (2023)

the action, such as “fast”, “now”, and “outside”. Overall, the models show promising results in predicting suitable communication cards for the given sentence, which can aid individuals with communication difficulties in expressing their needs and desires related to food.

The BERT-AAC zero-shot predictions shown in the figure refer to the sequence “I want eat [MASK].”, with a dot in after the masked token. The predictions are meaningful in this case. We can notice that the model has given high probabilities to cards referring to pronouns such as “it” and “this.” It also assigned high probabilities for adverbs such as “now,” “more,” and “today,” as well as the word “first,” which is also an adverb in the given context, i.e., sentence. However, there are fewer eating-related cards in the top-12 compared to BERT-AAC few-shot predictions.

In Figure 32, we also present the Kurtosis@100, which is a measure of flatness of the probability distributions obtained from the probabilities assigned to the top-100 cards. Kurtosis is a statistical measure that indicates the degree of tailedness or peakedness of a probability distribution. It measures the relative amount of data in the tails of a distribution compared to the data near the mean. A higher kurtosis value indicates that the distribution has more of its data in the tails, and a lower kurtosis value indicates that the distribution is flatter. The Kurtosis@100 formula used in this work is shown below:

$$\frac{\sum_{i=1}^{n}(p_i - \frac{1}{n})^4}{n} - 3, \tag{6.1}$$

where n is the number of items in the distribution, p_i is the probability of the ith item, and the term $\frac{1}{n}$ is the expected probability for each item in a uniform distribution. The formula subtracts 3 from the result to obtain a kurtosis value of 0 for a uniform distribution.

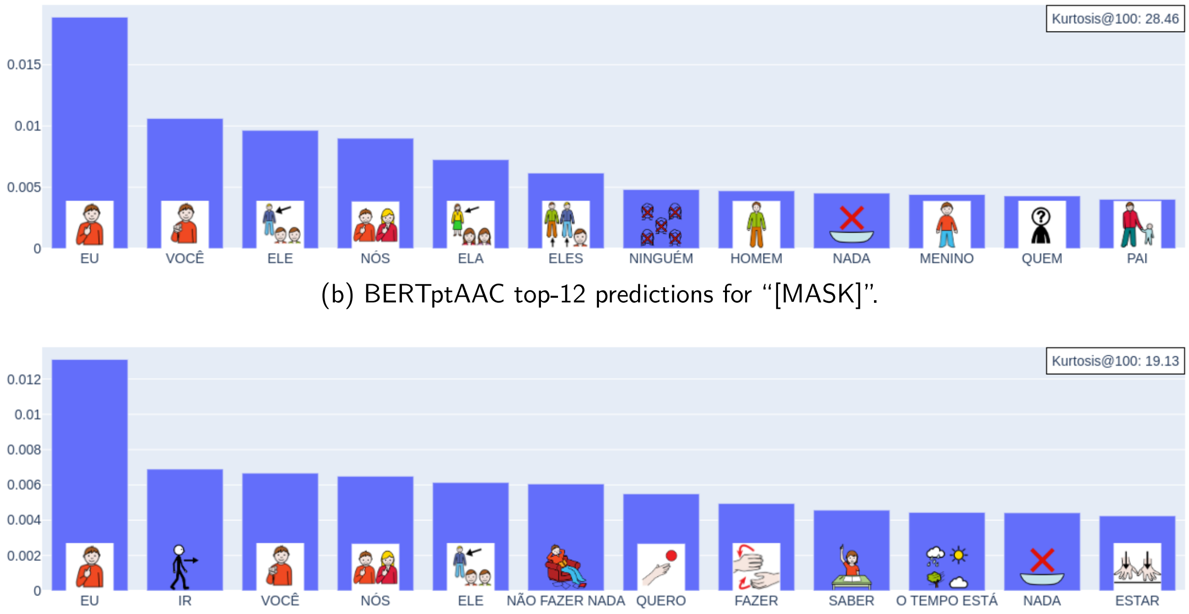
From the figure, it can be observed that PictoBERT predictions and GPT2-AAC zero-shot have the most peaked distributions. In contrast, BERT-AAC models have flatter distributions. A flat distribution in BERT-AAC models may indicate that the model is unable to predict a specific word to fill the mask confidently, and therefore, it is assigning a relatively equal probability to all possible words. However, in the context of communication card prediction models, a flat distribution can indicate that the model is not overfitting to a particular set of examples. Therefore, it is producing more diverse and potentially more generalizable predictions. On the other hand, a peaked distribution may indicate that the model is overfitting to the training data and cannot generalize well to new examples.

In conclusion, the English models for communication card prediction, including PictoBERT, BERT-AAC in zero and few-shot settings, and GPT2-AAC in the zero-shot setting, have demonstrated their ability to predict appropriate communication cards based on the context of the incomplete sentence. However, there are still challenges in adapting large language models to the specific needs of AAC systems, and each model has its strengths and weaknesses. By comparing and contrasting the performance of different models, this analysis can guide researchers and practitioners in developing more accurate and efficient communication aids. It is important to note that the training corpus's characteristics can affect the models' performance, as seen with PictoBERT. Overall, the results suggest that these models have the potential to aid individuals with CCN in expressing their needs and desires related to food and eating.

6.3.2 Portuguese Models

This section analyzes the two Brazilian Portuguese models, BERTptCS and BERTptAAC. The analysis focuses on examining the output of the models and identifying patterns and trends in the predictions. Through a qualitative analysis, we aim to understand better how each model performs regarding predicted tokens' accuracy, relevance, and distribution. This analysis provides insights into the strengths and limitations of each model, which can be used to guide future development and improvement of AAC systems. The analysis is conducted considering the communication cards present in the ComunicaTEA vocabulary.

Figure 33 – Portuguese models' predictions for the beginning of sentence construction.



(a) BERTptCS top-12 predictions for "<quem> [MASK] </quem>".

(b) BERTptAAC top-12 predictions for "[MASK]".

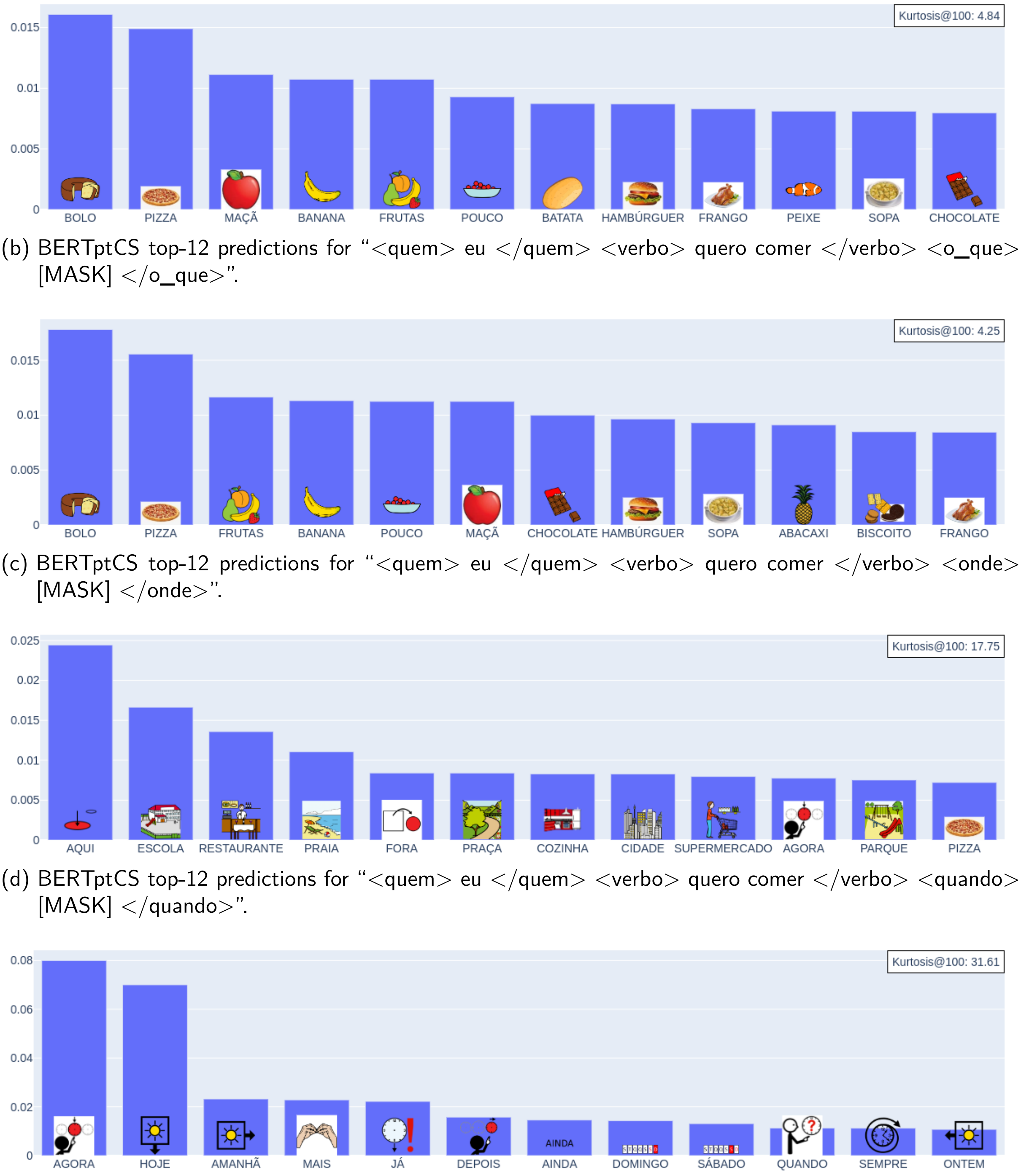
Source: The author (2023)

Figure 33 shows examples of predictions performed by BERTptCS and BERTptAAC, simulating the beginning of sentence construction. The communication cards in the figure are composed of the pictogram images, the caption, and the probability the model has given to each card. Analyzing the predictions, we notice that the predictions made by BERTptCS consist mainly of agentive words, which can play the role of the agent in a sentence (e.g., "eu", i.e., "I"). While the predictions made by BERTptAAC are a mix of verbs and pronouns, as the model has no information on how the user wants to initiate the construction of the sentence, either by the subject or by the verb. The use of CS roles brings more context to the prediction, and this can be verified in the example shown, which simulates the user choosing the first sentence communication card. Notice that if the user prefers to start from the verb, they can do so, and the model will predict the most suitable communication card to fill the mask in "<verbo> [MASK] </verbo>". Examples like this highlight the importance of using CS roles, as they can treat the sentence components differently, providing more accurate predictions.

The Kurtosis@100 values for both BERTptCS and BERTptAAC at the beginning of sentence construction are quite high, which may indicate that both models are overfitting to the training data and cannot generalize well to new examples. The small number of training examples could cause this overfitting, but it could also be due to the characteristics of the training corpus used for fine-tuning. The corpus has many sentences that begin with the word "eu", which is the most frequent word. However, it is worth noting that the human-composed sentences, which AAC specialists informed as the ones they consider common in AAC, also have many sentences starting with "eu" (cf. Section 5.1.2). This suggests that this may be a characteristic of the AAC domain.

In Figure 34, we present examples of predictions for inserting the verb object complement in the sentence "eu quero comer" (I want to eat). The predictions made by BERTptCS and BERTptAAC are quite similar if we consider using a <o_que> mark for the CS model. Both the models, BERTptCS and BERTptAAC, assigned high probabilities to eatable things when predicting the verb object complement for the sentence "eu quero comer" (I want to eat). The predictions made by BERTptAAC for inserting the verb object complement in the sentence "eu quero comer" are affected by the structure of the training corpus. If we compare BERTptAAC to its equivalent English version in this work, the BERT-AAC few-shot, we notice that both models assigned high probabilities to non-eatable things to complete the sentence "I want to eat [MASK]". BERTptAAC was fine-tuned using a version of AACptCorpus that does not use CS roles but maintains the words in the sentences following the CS order (e.g., who, what doing, what, how, where). Therefore, it is expected that in the training corpus, complements for "what" occur with high frequencies.

Figure 34 – Portuguese models' predictions for verb completion.



(a) BERTptAAC top-12 predictions for "eu quero comer [MASK]".

(b) BERTptCS top-12 predictions for "<quem> eu </quem> <verbo> quero comer </verbo> <o_que> [MASK] </o_que>".

(c) BERTptCS top-12 predictions for "<quem> eu </quem> <verbo> quero comer </verbo> <onde> [MASK] </onde>".

(d) BERTptCS top-12 predictions for "<quem> eu </quem> <verbo> quero comer </verbo> <quando> [MASK] </quando>".

Source: The author (2023)

One advantage of BERTptCS over BERTptAAC is its ability to make more accurate predictions using other roles besides the complement. For example, suppose the user wants to construct a sentence indicating where he/she wants to eat. Using BERTptAAC as we trained it would require the user first to inform what they want to eat, or they would see communication cards for locations with a considerably lower probability. However, with CS, the user can provide more context to the model, which is useful for making more accurate predictions, as demonstrated in Figure 34 and our experiments in Section 6.2. For example, Figure 34c shows the predicted complements for the location's role. All the communication cards suggest locations except for the card with "Pizza". Figure 34d shows the top-12 predicted complements for the time role. All the predicted complements for time are consistent, except for "mais" (more), which must have occurred together with time complements during training. It is important to note that the mapping process can sometimes group unrelated words, leading to unexpected predictions. Also, slight differences in the training data can lead to differences in the model's predictions, as seen with the prediction of "mais" in this example. It's worth noting that designing the user interface to facilitate the interaction using CS is beyond the scope of this work.

6.4 USAGE GUIDELINES: HOW CAN OTHERS USE THIS WORK?

Researchers, developers, and practitioners interested in utilizing the PrAACT method and findings presented in this work can follow the guidelines outlined below to enhance communication card prediction in AAC systems:

- **Constructing a Synthetic AAC Corpus**: Researchers can extend the method for constructing the synthetic AAC corpus to create their own corpus. This approach can be applied to different languages or specific target populations. By following the methodology described in Section 5.1.2 and in (PEREIRA et al., 2023a), researchers can adapt the process and gather data relevant to their specific context and objectives.
- **Fine-tuning a Language Model**: The constructed synthetic AAC corpus can be used for fine-tuning transformer-based language models such as BERT or GPT. Researchers can combine the corpus with the methodology presented in Section 4.2 to adapt the language model for communication card prediction. This process allows for personalized message authoring in AAC systems, enhancing the system's relevance and accuracy in generating suggestions.

- **Developing AAC Systems with Communication Card Prediction**: Developers can leverage the proposed method to design AAC systems that perform communication card or word prediction based on the user's vocabulary. To implement this, developers must create a language modeling head comprising the encoded user vocabulary for each user. Incorporating the user's vocabulary into the model can generate a probability distribution over the user's vocabulary items during inference. This distribution can suggest the most appropriate communication cards to complete a sentence, facilitating efficient and effective message composition.

- **Utilizing Existing BERT Models**: As demonstrated in the results chapter, no additional training is required to apply the proposed method. Developers can use the current versions of BERT available on platforms like HuggingFace[1] to implement communication card prediction in AAC systems. By following the proposed method, developers can adapt these existing models without extensive training resources, saving time and effort in the development process.

- **Utilizing Models based on Colourful Semantics**: When using models based on CS, it is necessary to prepare input sequences with the CS roles. For example, if a user starts composing a sentence by selecting the word "I" as the agent and intends to choose the verb of the sentence, the input sequence should be prepared as "<who> I </who> <verb> [MASK] </verb>". Incorporating CS roles ensures the model understands the intended semantic structure and produces accurate predictions based on the user's input.

[1] <https://huggingface.co/models>

7 CONCLUSION

This chapter presents the conclusions of this work. Section 7.1 presents the final considerations of the main topics of this work. Section 7.2 offers the contributions of this work, including the articles published or under review. Section 7.3 presents the limitations of this work. Finally, Section 7.4 shows the next steps.

7.1 FINAL CONSIDERATIONS

This study aims to propose a method for adapting large language models to communication card prediction in Augmentative and Alternative Communication (AAC) systems to facilitate message authoring for individuals with Complex Communication Needs (CCN) who rely on AAC. The proposed method involves adapting a transformer-based Language Model (LM) (e.g., BERT and GPT-2) to the AAC domain by fine-tuning it using a telegraphic sentence corpus or incorporating visual cues. The model is then modified by replacing its LM head with an encoded version of the user's vocabulary. This allows it to produce a probability distribution over the user's vocabulary items during inference. The proposed method takes advantage of the transfer learning ability of transformer-based language models, such as Bidirectional Encoder Representations from Transformers (BERT), to facilitate message authoring in AAC systems in a low-effort setting. We evaluate the proposed method in English and Brazilian Portuguese and demonstrate that the models produced using this method outperform models pre-trained for the task. Additionally, we demonstrate that incorporating the Colourful Semantics (CS) structure into the fine-tuning process of BERT enhances the accuracy and relevance of the model's predictions.

With these results, we answered the Research Questions (RQs) presented in Section 1.2. RQ-1 ("How can a transformer-based neural network be adapted to improve communication card prediction?") is answered by the method described in Chapter 4, which presents how to adapt transformer-based LMs to perform communication card prediction. RQ-2 ("What is the performance of adapted transformer-based neural network models for communication card prediction in high-tech AAC systems in a zero-shot or few-shot setting?") is answered by the experimental results presented in Section 6.1, which demonstrates that a BERT-based adapted model fine-tuned only with 7K sentences outperforms a model pre-trained for the task with a

955K sentences corpus. While RQ-3 ("*Can the use of Colourful Semantics (CS) improve the accuracy of communication cards prediction models for AAC systems?*") is answered by the experimental results presented in Section 6.2, which demonstrate that using CS roles enhances the accuracy and relevance of the model's predictions.

The main findings of this work highlight the effectiveness of adapting transformer-based neural networks for communication card prediction in AAC systems. The proposed method, which involves fine-tuning a transformer-based LM using a telegraphic sentence corpus or incorporating visual cues, successfully enhances the accuracy and relevance of the model's predictions. By adapting transformer-based models like BERT and GPT-2, we demonstrate that it is possible to improve communication card prediction without the need for extensive training data. Even with a relatively small dataset of 7K sentences, the fine-tuned models outperform models pre-trained on much larger corpora (955K sentences). This suggests that transfer learning with adaptation to the AAC domain is an effective approach for low-effort message authoring in high-tech AAC systems.

Furthermore, the incorporation of CS structure into the fine-tuning process of BERT improves the model's accuracy by considering the roles of *who*, *what doing* and *what* the model produces more accurate and contextually relevant predictions. This demonstrates the potential of incorporating linguistic structure in training models for AAC systems.

Overall, this study contributes to the field of AAC by providing a method that empowers individuals with CCN to author messages more effectively using communication cards. The findings also highlight the importance of leveraging transformer-based neural networks and linguistic structures to enhance the performance of communication card prediction models.

7.2 CONTRIBUTIONS

This work presents two main contributions:

1. A method that harnesses the transfer learning ability of transformers-based language models to facilitate message authoring in AAC systems in a low-effort setting.
2. An strategy for incorporating CS into communication card prediction models for AAC systems.

Besides these main contributions, we also advance the state of the art because we:

- Make a systematic mapping study of methods used for prediction in AAC systems, which allow developers and researchers to embassies the decisions regarding pictogram prediction in AAC systems;
- Construct a corpus of AAC-like sentences for Brazilian Portuguese;
- Provide an experiment on how to represent communication cards for prediction models better;
- Provide Deep Learning (DL) models to perform communication card prediction with high quality.

We highlight that some of these contributions have been published or are under review. We also have other publications that contribute to this thesis. All these publications are listed below:

- **Pereira, J. A.**, Pereira, J. A., & Fidalgo, R. do N. (2021). *Caregivers Acceptance of Using Semantic Communication Boards for teaching Children with Complex Communication Needs.* Anais Do XXXII Simpósio Brasileiro de Informática Na Educação (SBIE 2021). This paper presents a study on the acceptance of therapists, parents, and educators of children with CCN using a communication board that performs pictogram prediction as an educational tool.
- **Pereira, J. A.**, Macêdo, D., Zanchettin, C., De Oliveira, A. L., & Fidalgo, R. D. (2022). *Pictobert: Transformers for next pictogram prediction*. Expert Systems with Applications, 202, 117231. This paper presents our experiments on pre-training BERT for pictogram prediction in English.
- **Pereira, J. A.,** Medeiros, S. de, Zanchettin, C., & Fidalgo, R. do N. (2022). *Pictogram Prediction in Alternative Communication Boards: a Mapping Study*. Anais Do XXXIII Simpósio Brasileiro de Informática Na Educação (SBIE 2022). This paper presents a systematic mapping study we performed on the methods used for pictogram prediction in AAC systems.
- **Pereira, J. A.**, Nogueira, R., Zanchettin, C., & Fidalgo, R. do N. *An Augmentative and Alternative Communication Synthetic Corpus for Brazilian Portuguese*. The 23rd IEEE International Conference on Advanced Learning Technologies (ICALT 2023). This paper presents our method to construct the Brazilian Portuguese AAC corpus.

- **Pereira, J. A.**, Nogueira, R., Zanchettin, C., & Fidalgo, R. do N. *Predictive Authoring for Brazilian Portuguese Augmentative and Alternative Communication*. **Under review**. Submitted to the Special Issue on Natural Language Processing Applications for Low-Resource Languages of the Cambridge Natural Language Engineering Journal in December (2022). This paper presents our experiments on communication card prediction for Brazilian Portuguese. Preprint published in Pereira et al. (2023b).

- **Pereira, J. A.**, Pereira, J., Zanchettin, C., & Fidalgo, R. do N. *Praact: Predictive Augmentative and Alternative Communication with Transformers*. **Under review**. Submitted to the Expert Systems with Applications journal in August 2023. This paper presents our experiments on communication card prediction for English. Preprint published in Pereira et al. (2023).

7.3 LIMITATIONS

Despite the relevant results, it is necessary to assume some limitations that can be addressed in future works. We will list them below:

- Portuguese dataset – This work employs a synthetic corpus of sentences constructed using a pre-defined vocabulary generated through an automated process. While this corpus augments sentences typically used by practitioners of AAC, it may not accurately reflect the language used in actual AAC boards.

- Communication card representation evaluation – the experiments for finding the best way to encode pictograms for prediction using the neural network has a sense disambiguation step that was not evaluated in other datasets or scenarios;

- User-centered evaluation – The effectiveness of AAC solutions should be evaluated through user-centered approaches, in which individuals with CCN actively participate in the evaluation process. This approach allows for a more comprehensive assessment of the solution's functionality and usability.

7.4 FUTURE WORK

In future work we intend to:

1. Experiment on the best manner to represent a communication card using as a basis the same corpus used for PictoBERT, which is already annotated with word senses that can be mapped to Aragonese Portal of Augmentative and Alternative Communication (ARASAAC) pictograms;

2. Perform ablation experiments changing a) model size; b) vocabulary encoding aggregation method (e.g., average or sum);

3. Implement a AAC system that performs prediction using the Reaact platform as a basis;

4. Compare the effectiveness of BERTptCS and BERTptAAC when used by humans.

REFERENCES

ALGOET, P. H.; COVER, T. M. A sandwich proof of the shannon-mcmillan-breiman theorem. *The Annals of Probability*, Institute of Mathematical Statistics, v. 16, n. 2, p. 899–909, 1988. ISSN 00911798. Available at: <http://www.jstor.org/stable/2243846>.

American Speech-Language-Hearing Association. *Augmentative and Alternative Communication*. [S.l.]: ASHA, n.d. Retrieved from <https://www.asha.org/practice-portal/professional-issues/augmentative-and-alternative-communication/>. Accessed July 25, 2022.

BAXTER, S.; ENDERBY, P.; EVANS, P.; JUDGE, S. Barriers and facilitators to the use of high-technology augmentative and alternative communication devices: a systematic review and qualitative synthesis. *International Journal of Language & Communication Disorders*, v. 47, n. 2, p. 115–129, 2012.

BEUKELMAN, D. R.; LIGHT, J. C. *Augmentative & Alternative Communication: Supporting Children and Adults with Complex Communication Needs*. Fifth edition. [S.l.]: Paul H. Brookes Baltimore, 2020.

BOLDERSON, S.; DOSANJH, C.; MILLIGAN, C.; PRING, T.; CHIAT, S. Colourful semantics: A clinical investigation. *Child Language Teaching and Therapy*, Sage Publications Sage UK: London, England, v. 27, n. 3, p. 344–353, 2011.

BROWN, T.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J. D.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A.; AGARWAL, S.; HERBERT-VOSS, A.; KRUEGER, G.; HENIGHAN, T.; CHILD, R.; RAMESH, A.; ZIEGLER, D.; WU, J.; WINTER, C.; HESSE, C.; CHEN, M.; SIGLER, E.; LITWIN, M.; GRAY, S.; CHESS, B.; CLARK, J.; BERNER, C.; MCCANDLISH, S.; RADFORD, A.; SUTSKEVER, I.; AMODEI, D. Language models are few-shot learners. In: LAROCHELLE, H.; RANZATO, M.; HADSELL, R.; BALCAN, M.; LIN, H. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2020. v. 33, p. 1877–1901. Available at: <https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf>.

BRYAN, A. Colourful semantics: Thematic role therapy. In: ______. *Language Disorders in Children and Adults*. [S.l.]: John Wiley & Sons, Ltd, 2003. chap. 3.2, p. 143–161. ISBN 9780470699157.

CHEN, L.; BABAR, M. A.; ZHANG, H. Towards an evidence-based understanding of electronic data sources. In: *14th International conference on evaluation and assessment in software engineering (EASE)*. [S.l.: s.n.], 2010. p. 1–4.

CHOWDHERY, A.; NARANG, S.; DEVLIN, J.; BOSMA, M.; MISHRA, G.; ROBERTS, A.; BARHAM, P.; CHUNG, H. W.; SUTTON, C.; GEHRMANN, S.; SCHUH, P.; SHI, K.; TSVYASHCHENKO, S.; MAYNEZ, J.; RAO, A.; BARNES, P.; TAY, Y.; SHAZEER, N.; PRABHAKARAN, V.; REIF, E.; DU, N.; HUTCHINSON, B.; POPE, R.; BRADBURY, J.; AUSTIN, J.; ISARD, M.; GUR-ARI, G.; YIN, P.; DUKE, T.; LEVSKAYA, A.; GHEMAWAT, S.; DEV, S.; MICHALEWSKI, H.; GARCIA, X.; MISRA, V.; ROBINSON, K.; FEDUS, L.; ZHOU, D.; IPPOLITO, D.; LUAN, D.; LIM, H.; ZOPH, B.; SPIRIDONOV, A.; SEPASSI, R.; DOHAN, D.; AGRAWAL, S.; OMERNICK, M.; DAI, A. M.; PILLAI, T. S.; PELLAT, M.; LEWKOWYCZ, A.; MOREIRA, E.; CHILD, R.; POLOZOV, O.; LEE, K.; ZHOU, Z.; WANG, X.; SAETA, B.; DIAZ, M.; FIRAT, O.; CATASTA, M.; WEI, J.; MEIER-HELLSTERN, K.;

ECK, D.; DEAN, J.; PETROV, S.; FIEDEL, N. *PaLM: Scaling Language Modeling with Pathways*. arXiv, 2022. Available at: <https://arxiv.org/abs/2204.02311>.

CHRISTOPOULOU, M.; VONIATI, L.; DROSOS, K.; ARMOSTIS, S. Colorful semantics in cypriot-greek-speaking children with autism spectrum disorder. *Folia Phoniatrica et Logopaedica*, Karger Publishers, v. 73, n. 3, p. 185–194, 2021.

CHUNG, Y.-C.; CARTER, E. W. Promoting peer interactions in inclusive classrooms for students who use speech-generating devices. *Research and Practice for Persons with Severe Disabilities*, v. 38, n. 2, p. 94–109, 2013.

CONIA, S.; BRIGNONE, F.; ZANFARDINO, D.; NAVIGLI, R. InVeRo: Making semantic role labeling accessible with intelligible verbs and roles. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, 2020. p. 77–84.

DAVIS, F. D. *A technology acceptance model for empirically testing new end-user information systems: Theory and results*. Phd Thesis (PhD Thesis) — Massachusetts Institute of Technology, 1985.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 4171–4186.

DONATO, C.; SPENCER, E.; ARTHUR-KELLY, M. A critical synthesis of barriers and facilitators to the use of aac by children with autism spectrum disorder and their communication partners. *Augmentative and Alternative Communication*, Taylor & Francis, v. 34, n. 3, p. 242–253, 2018.

DOSOVITSKIY, A.; BEYER, L.; KOLESNIKOV, A.; WEISSENBORN, D.; ZHAI, X.; UNTERTHINER, T.; DEHGHANI, M.; MINDERER, M.; HEIGOLD, G.; GELLY, S.; USZKOREIT, J.; HOULSBY, N. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. Available at: <https://arxiv.org/abs/2010.11929>.

DUDY, S.; BEDRICK, S. Compositional Language Modeling for Icon-Based Augmentative and Alternative Communication. *Proceedings of the conference. Association for Computational Linguistics. Meeting*, v. 2018, p. 25–32, jul 2018. ISSN 0736-587X. Available at: <https://pubmed.ncbi.nlm.nih.gov/33935351https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8087438/>.

ELMAN, J. L. Finding structure in time. *Cognitive Science*, v. 14, n. 2, p. 179–211, 1990. Available at: <https://onlinelibrary.wiley.com/doi/abs/10.1207/s15516709cog1402_1>.

ELSAHAR, Y.; HU, S.; BOUAZZA-MAROUF, K.; KERR, D.; MANSOR, A. Augmentative and alternative communication (aac) advances: A review of configurations for individuals with a speech disability. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 19, n. 8, p. 1911, 2019.

FABBRI, S.; SILVA, C.; HERNANDES, E.; OCTAVIANO, F.; THOMMAZO, A. D.; BELGAMO, A. Improvements in the start tool to better support the systematic review process. In: *Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering*. New York, NY, USA: Association for Computing Machinery, 2016. (EASE '16). ISBN 9781450336918. Available at: <https://doi.org/10.1145/2915970.2916013>.

FILHO, J. A. W.; WILKENS, R.; IDIART, M.; VILLAVICENCIO, A. The brWaC corpus: A new open resource for Brazilian Portuguese. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA), 2018. Available at: <https://aclanthology.org/L18-1686>.

FITZGERALD, E. *Straight language for the deaf: a system of instruction for deaf children*. [S.l.]: Volta Bureau, 1949.

FRANCO, N.; SILVA, E.; LIMA, R.; FIDALGO, R. Towards a reference architecture for augmentative and alternative communication systems. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2018. v. 29, p. 1073.

GARCIA, L. F.; OLIVEIRA, L. C. de; MATOS, D. M. de. Evaluating pictogram prediction in a location-aware augmentative and alternative communication system. *Assistive Technology*, Taylor & Francis, v. 28, n. 2, p. 83–92, 2016. Available at: <https://doi.org/10.1080/10400435.2015.1092181>.

GARCÍA, P.; LLEIDA, E.; CASTÁN, D.; MARCOS, J. M.; ROMERO, D. Context-Aware Communicator for All. In: ANTONA, M.; STEPHANIDIS, C. (Ed.). *Universal Access in Human-Computer Interaction. Access to Today's Technologies*. Cham: Springer International Publishing, 2015. p. 426–437. ISBN 978-3-319-20678-3.

GOLDBERG, Y.; HIRST, G. *Neural Network Methods in Natural Language Processing*. [S.l.]: Morgan & Claypool Publishers, 2017. ISBN 1627052984.

HERVÁS, R.; BAUTISTA, S.; MÉNDEZ, G.; GALVÁN, P.; GERVÁS, P. Predictive composition of pictogram messages for users with autism. *Journal of Ambient Intelligence and Humanized Computing*, Springer Berlin Heidelberg, v. 11, n. 11, p. 5649–5664, 2020. ISSN 1868-5145.

HETTIARACHCHI, S. The effectiveness of colourful semantics on narrative skills in children with intellectual disabilities in sri lanka. *Journal of intellectual disabilities : JOID*, 06 2015.

HOLYFIELD, C.; LORAH, E. Effects of High-tech Versus Low-tech AAC on Indices of Happiness for School-aged Children with Multiple Disabilities. *Journal of Developmental and Physical Disabilities*, 2022. ISSN 1573-3580. Available at: <https://doi.org/10.1007/s10882-022-09858-5>.

HONNIBAL, M.; MONTANI, I. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear. 2017.

HUGHES, D. M.; VENTO-WILSON, M.; BOYD, L. E. Direct speech-language intervention effects on augmentative and alternative communication system use in adults with developmental disabilities in a naturalistic environment. *American Journal* *of Speech-Language Pathology*, v. 31, n. 4, p. 1621–1636, 2022. Available at: <https://pubs.asha.org/doi/abs/10.1044/2022_AJSLP-21-00242>.

JUDGE, S.; TOWNEND, G. Perceptions of the design of voice output communication aids. *International Journal of Language & Communication Disorders*, v. 48, n. 4, p. 366–381, 2013.

JURAFSKY, D.; MARTIN, J. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 3. ed. [s.n.], 2019. Available at: <https://web.stanford.edu/~jurafsky/slp3/ed3book_sep212021.pdf>.

LAW, J.; LEE, W.; ROULSTONE, S.; WREN, Y.; ZENG, B.; LINDSAY, G. 'what works': interventions for children and young people with speech, language and communication needs. Department for Education, 2012.

LEITE, J. A.; SILVA, D.; BONTCHEVA, K.; SCARTON, C. Toxic language detection in social media for Brazilian Portuguese: New dataset and multilingual analysis. In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*. Suzhou, China: Association for Computational Linguistics, 2020. p. 914–924. Available at: <https://aclanthology.org/2020.aacl-main.91>.

LIGHT, J.; MCNAUGHTON, D. The changing face of augmentative and alternative communication: Past, present, and future challenges. *Augmentative and Alternative Communication*, Taylor & Francis, v. 28, n. 4, p. 197–204, 2012. PMID: 23256853. Available at: <https://doi.org/10.3109/07434618.2012.737024>.

LORAH, E. R.; HOLYFIELD, C.; MILLER, J.; GRIFFEN, B.; LINDBLOOM, C. A Systematic Review of Research Comparing Mobile Technology Speech-Generating Devices to Other AAC Modes with Individuals with Autism Spectrum Disorder. *Journal of Developmental and Physical Disabilities*, v. 34, n. 2, p. 187–210, 2022. ISSN 1573-3580. Available at: <https://doi.org/10.1007/s10882-021-09803-y>.

LORAH, E. R.; PARNELL, A.; WHITBY, P. S.; HANTULA, D. A systematic review of tablet computers and portable media players as speech generating devices for individuals with autism spectrum disorder. *Journal of autism and developmental disorders*, Springer, v. 45, p. 3792–3804, 2015.

LORAH, E. R.; TINCANI, M.; PARNELL, A. Current trends in the use of handheld technology as a speech-generating device for children with autism. *Behavior Analysis: Research and Practice*, Educational Publishing Foundation, Lorah, Elizabeth R.: Department of Curriculum and Instruction, University of Arkansas, 410 N. Arkansas Ave., Fayetteville, AR, US, 72701, lorah@uark.edu, v. 18, n. 3, p. 317–327, 2018. ISSN 2372-9414(Electronic).

LU, J.; BEHBOOD, V.; HAO, P.; ZUO, H.; XUE, S.; ZHANG, G. Transfer learning using computational intelligence: A survey. *Knowledge-Based Systems*, v. 80, p. 14–23, 2015. ISSN 0950-7051.

MACWHINNEY, B. *The CHILDES project: Tools for analyzing talk, Volume II: The database*. 3. ed. [S.l.]: Psychology Press, 2014.

MARTíNEZ-SANTIAGO, F.; DíAZ-GALIANO, M.; UREñA-LóPEZ, L.; MITKOV, R. A semantic grammar for beginning communicators. *Knowledge-Based Systems*, v. 86, p. 158–172, 2015. ISSN 0950-7051.

MCDONALD, E. T.; SCHULTZ, A. R. Communication boards for cerebral-palsied children. *Journal of Speech and Hearing Disorders*, v. 38, n. 1, p. 73–88, 1973. Available at: <https://pubs.asha.org/doi/abs/10.1044/jshd.3801.73>.

MCNAUGHTON, D.; LIGHT, J.; BEUKELMAN, D. R.; KLEIN, C.; NIEDER, D.; NAZARETH, G. Building capacity in AAC: A person-centred approach to supporting participation by people with complex communication needs. *Augmentative and Alternative Communication*, Taylor & Francis, v. 35, n. 1, p. 56–68, 2019.

MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. *Efficient Estimation of Word Representations in Vector Space*. arXiv, 2013. Available at: <https://arxiv.org/abs/1301.3781>.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G. S.; DEAN, J. Distributed representations of words and phrases and their compositionality. In: BURGES, C.; BOTTOU, L.; WELLING, M.; GHAHRAMANI, Z.; WEINBERGER, K. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2013. v. 26. Available at: <https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf>.

MIKOLOV, T.; YIH, W.-t.; ZWEIG, G. Linguistic regularities in continuous space word representations. In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Atlanta, Georgia: Association for Computational Linguistics, 2013. p. 746–751. Available at: <https://aclanthology.org/N13-1090>.

MILLER, G. A. Wordnet: a lexical database for english. *Communications of the ACM*, Association for Computing Machinery, v. 38, n. 11, p. 39–41, 1995.

MORIN, F.; BENGIO, Y. Hierarchical probabilistic neural network language model. In: PMLR. *International workshop on artificial intelligence and statistics*. [S.l.], 2005. p. 246–252.

OPENAI. *GPT-4 Technical Report*. 2023.

PALAO, S. *ARASAAC: Aragonese Portal of Augmentative and Alternative Communication*. 2019. Available at: <http://www.arasaac.org/>.

PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, v. 22, n. 10, p. 1345–1359, 2010.

PAPANDREA, S.; RAGANATO, A.; BOVI, C. D. Supwsd: A flexible toolkit for supervised word sense disambiguation. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Copenhagen, Denmark: Association for Computational Linguistics, 2017. p. 103–108.

PEREIRA, J.; FRANCO, N.; FIDALGO, R. A Semantic Grammar for Augmentative and Alternative Communication Systems. In: SOJKA, P.; KOPEČEK, I.; PALA, K.; HORÁK, A. (Ed.). *Text, Speech, and Dialogue*. Cham: Springer International Publishing, 2020. p. 257–264. ISBN 978-3-030-58323-1.

PEREIRA, J.; MEDEIROS, S.; ZANCHETTIN, C.; FIDALGO, R. Pictogram prediction in alternative communication boards: a mapping study. In: *Anais do XXXIII Simpósio Brasileiro de Informática na Educação*. Porto Alegre, RS, Brasil: SBC, 2022. p. 705–717. ISSN 0000-0000. Available at: <https://sol.sbc.org.br/index.php/sbie/article/view/22452>.

PEREIRA, J.; NOGUEIRA, R.; ZANCHETTIN, C.; FIDALGO, R. An augmentative and alternative communication synthetic corpus for brazilian portuguese. In: *The 23rd IEEE International Conference on Advanced Learning Technologies (ICALT)*. [S.l.: s.n.], 2023.

PEREIRA, J.; NOGUEIRA, R.; ZANCHETTIN, C.; FIDALGO, R. *Predictive Authoring for Brazilian Portuguese Augmentative and Alternative Communication*. 2023. Available at: <https://arxiv.org/abs/2308.09497>.

PEREIRA, J.; PENA, C.; MELO, M. de; CARTAXO, B.; SOARES, S.; FIDALGO, R. Facilitators and barriers to using alternative and augmentative communication systems by aphasic: Therapists perceptions. In: *2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)*. [S.l.: s.n.], 2019. p. 349–354.

PEREIRA, J.; PEREIRA, J.; FIDALGO, R. Caregivers acceptance of using semantic communication boards for teaching children with complex communication needs. In: *Anais do XXXII Simpósio Brasileiro de Informática na Educação*. Porto Alegre, RS, Brasil: SBC, 2021. p. 642–654. ISSN 0000-0000. Available at: <https://sol.sbc.org.br/index.php/sbie/article/view/18094>.

PEREIRA, J.; PEREIRA, J.; ZANCHETTIN, C.; FIDALGO, R. *Praact: Predictive Augmentative and Alternative Communication with Transformers*. 2023. Available at: <https://dx.doi.org/10.2139/ssrn.4544152>.

PEREIRA, J. A.; MACêDO, D.; ZANCHETTIN, C.; de Oliveira, A. L. I.; FIDALGO, R. do N. Pictobert: Transformers for next pictogram prediction. *Expert Systems with Applications*, v. 202, p. 117231, 2022. ISSN 0957-4174. Available at: <https://www.sciencedirect.com/science/article/pii/S095741742200611X>.

PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. Systematic mapping studies in software engineering. In: *12th International Conference on Evaluation and Assessment in Software Engineering (EASE) 12*. [S.l.: s.n.], 2008. p. 1–10.

PETERSEN, K.; VAKKALANKA, S.; KUZNIARZ, L. Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, v. 64, p. 1–18, 2015. ISSN 0950-5849. Available at: <https://www.sciencedirect.com/science/article/pii/S0950584915000646>.

PORTER, G. *Pragmatic Organisation Dynamic Display Communication Books Direct Access Templates*. [S.l.]: Cerebral Palsy Education Centre, 2007.

QI, P.; ZHANG, Y.; ZHANG, Y.; BOLTON, J.; MANNING, C. D. Stanza: A Python natural language processing toolkit for many human languages. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. [s.n.], 2020. Available at: <https://nlp.stanford.edu/pubs/qi2020stanza.pdf>.

RADFORD, A.; NARASIMHAN, K.; SALIMANS, T.; SUTSKEVER, I. et al. Improving language understanding by generative pre-training. 2018.

RADFORD, A.; WU, J.; CHILD, R.; LUAN, D.; AMODEI, D.; SUTSKEVER, I. Language models are unsupervised multitask learners. 2019.

RAE, J. W.; BORGEAUD, S.; CAI, T.; MILLICAN, K.; HOFFMANN, J.; SONG, F.; ASLANIDES, J.; HENDERSON, S.; RING, R.; YOUNG, S.; RUTHERFORD, E.; HENNIGAN, T.; MENICK, J.; CASSIRER, A.; POWELL, R.; DRIESSCHE, G. v. d.; HENDRICKS, L. A.; RAUH, M.; HUANG, P.-S.; GLAESE, A.; WELBL, J.; DATHATHRI, S.; HUANG, S.; UESATO, J.; MELLOR, J.; HIGGINS, I.; CRESWELL, A.; MCALEESE, N.; WU, A.; ELSEN, E.; JAYAKUMAR, S.; BUCHATSKAYA, E.; BUDDEN, D.; SUTHERLAND, E.; SIMONYAN, K.; PAGANINI, M.; SIFRE, L.; MARTENS, L.; LI, X. L.; KUNCORO, A.; NEMATZADEH, A.; GRIBOVSKAYA, E.; DONATO, D.; LAZARIDOU, A.; MENSCH, A.; LESPIAU, J.-B.; TSIMPOUKELLI, M.; GRIGOREV, N.; FRITZ, D.; SOTTIAUX, T.; PAJARSKAS, M.; POHLEN, T.; GONG, Z.; TOYAMA, D.; D'AUTUME, C. d. M.; LI, Y.; TERZI, T.; MIKULIK, V.; BABUSCHKIN, I.; CLARK, A.; CASAS, D. d. L.; GUY, A.; JONES, C.; BRADBURY, J.; JOHNSON, M.; HECHTMAN, B.; WEIDINGER, L.; GABRIEL, I.; ISAAC, W.; LOCKHART, E.; OSINDERO, S.; RIMELL, L.; DYER, C.; VINYALS, O.; AYOUB, K.; STANWAY, J.; BENNETT, L.; HASSABIS, D.; KAVUKCUOGLU, K.; IRVING, G. *Scaling Language Models: Methods, Analysis & Insights from Training Gopher*. arXiv, 2021. Available at: <https://arxiv.org/abs/2112.11446>.

RENFREW, C. E. *The Renfrew language scales: Action picture test*. [S.l.]: Speechmark, 2016.

SATURNO, C. E.; RAMIREZ, A. R. G.; CONTE, M. J.; FARHAT, M.; PIUCCO, E. C. An augmentative and alternative communication tool for children and adolescents with cerebral palsy. *Behaviour & Information Technology*, Taylor & Francis, v. 34, n. 6, p. 632–645, 2015. Available at: <https://doi.org/10.1080/0144929X.2015.1019567>.

SCARLINI, B.; PASINI, T.; NAVIGLI, R. With More Contexts Comes Better Performance: Contextualized Sense Embeddings for All-Round Word Sense Disambiguation. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. [S.l.]: Association for Computational Linguistics, 2020. p. "3528 – 3539".

SCHWAB, D.; TRIAL, P.; VASCHALDE, C.; VIAL, L.; ESPERANÇA-RODIER, E.; LECOUTEUX, B. Providing semantic knowledge to a set of pictograms for people with disabilities: a set of links between wordnet and arasaac: Arasaac-wn. In: *LREC*. [s.n.], 2020. p. 166–171. Available at: <https://hal.archives-ouvertes.fr/hal-02888279/>.

SENNOTT, S. C.; AKAGI, L.; LEE, M.; RHODES, A. AAC and Artificial Intelligence (AI). *Topics in language disorders*, v. 39, n. 4, p. 389–403, 2019. ISSN 0271-8294. Available at: <https://pubmed.ncbi.nlm.nih.gov/34012187https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8130588/>.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. Bertimbau: Pretrained bert models for brazilian portuguese. In: CERRI, R.; PRATI, R. C. (Ed.). *Intelligent Systems*. Cham: Springer International Publishing, 2020. p. 403–417. ISBN 978-3-030-61377-8.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L. u.; POLOSUKHIN, I. Attention is all you need. In: GUYON, I.; LUXBURG, U. V.; BENGIO, S.; WALLACH, H.; FERGUS, R.; VISHWANATHAN, S.; GARNETT, R. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. v. 30. Available at: <https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>.

VERTANEN, K.; KRISTENSSON, P. O. The imagination of crowds: Conversational aac language modeling using crowdsourcing and large data sources. In: *Proceedings of the*

Conference on Empirical Methods in Natural Language Processing (EMNLP). [S.l.]: ACL, 2011. p. 700–711.

WU, Y.; SCHUSTER, M.; CHEN, Z.; LE, Q. V.; NOROUZI, M.; MACHEREY, W.; KRIKUN, M.; CAO, Y.; GAO, Q.; MACHEREY, K.; KLINGNER, J.; SHAH, A.; JOHNSON, M.; LIU, X.; KAISER Łukasz; GOUWS, S.; KATO, Y.; KUDO, T.; KAZAWA, H.; STEVENS, K.; KURIAN, G.; PATIL, N.; WANG, W.; YOUNG, C.; SMITH, J.; RIESA, J.; RUDNICK, A.; VINYALS, O.; CORRADO, G.; HUGHES, M.; DEAN, J. *Google's neural machine translation system: Bridging the gap between human and machine translation*. 2016.

WU, Y.; SCHUSTER, M.; CHEN, Z.; LE, Q. V.; NOROUZI, M.; MACHEREY, W.; KRIKUN, M.; CAO, Y.; GAO, Q.; MACHEREY, K.; KLINGNER, J.; SHAH, A.; JOHNSON, M.; LIU, X.; KAISER, l.; GOUWS, S.; KATO, Y.; KUDO, T.; KAZAWA, H.; STEVENS, K.; KURIAN, G.; PATIL, N.; WANG, W.; YOUNG, C.; SMITH, J.; RIESA, J.; RUDNICK, A.; VINYALS, O.; CORRADO, G.; HUGHES, M.; DEAN, J. *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation*. arXiv, 2016. Available at: <https://arxiv.org/abs/1609.08144>.

XIAN, Y.; SCHIELE, B.; AKATA, Z. *Zero-Shot Learning – The Good, the Bad and the Ugly*. 2020.