Utrecht University
Faculty of Science
Graduate School of Natural Science

Master's thesis

Language of Thought Models for the Learning of Multiple Concepts: to which Degree is Pragmatic Reasoning Involved?

Lorenzo Pavan
l.pavan@students.uu.nl
Student number: 9791981

Master's program in Artificial Intelligence
May 2024

First supervisor: Prof. Dr. Jakub Dotlačil
Department of Languages, Literature and Communication
Utrecht University

Second supervisor: Prof. Dr. Rick Nouwen
Department of Languages, Literature and Communication
Utrecht University

Abstract

This work explores the interaction between pragmatic reasoning and concept learning. Specifically, it examines how reasoning about a speaker's intentions influences concept learning, as well as how learners' beliefs about the meanings of novel words interact when those words are known to have distinct meanings.

The study comprises an experiment where participants learn two concepts simultaneously, involving the elicitation of conversational implicatures. The experimental results are then used to fit cognitive models.

The model for pragmatics being used is the Lexical Uncertainty Model, which is based on the Rational Speech Act framework. This is a Bayesian model, where hypotheses about concept meanings need to be clearly defined in order to evaluate whether experimental observations (such as seeing trials' feedback) support them. This is where Languages of Thought prove useful, as they account for concepts being realized through compositions of symbols, thereby allowing hypotheses about concept meanings to be represented as logical expressions.

The experimental results provide no clear indication that participants engage in conversational implicatures, whereas the cognitive models including pragmatic reasoning do not exhibit a significantly better fit with the experimental data.

Nevertheless, these results should be considered preliminary, as the participant pool was very limited, and individuals capable of the required pragmatic reasoning for this task are uncommon. Furthermore, there is still a vast margin of improvement on the methodologies, given the complexity of the study.

Contents

List of Figures

List of Tables

List of Acronyms

AI Artificial Intelligence. 63, 66

GLMM generalized linear mixed-effects model. 45–50

IQR Iterated Quantal Response. 13, 14, 34

LLM Large Language Model. 66

LOT Language of Thought. 4, 5, 10–12, 21, 23–25, 33, 34, 40, 61, 63, 64

LUM Lexical Uncertainty Model. 18–20, 33, 34, 37–40, 57, 71

M-implicature Manner implicature. 8, 9, 13, 14, 19, 20, 30–32, 34, 39, 40, 50, 57, 61

Q-implicature Quantity implicature. 8, 13, 21, 31, 32, 50, 61

RSA Rational Speech Act. 14–19, 21, 34, 39

1 Introduction

When we encounter an expression in a language that we are learning, we need to understand both the speaker's intended message and the inherent meanings of the words used. Ignorance on word meanings introduces complexity in inferring the intended message, which may diverge from what is expressed by the literal meaning of the sentences involved. Knowing the meaning of words is thus helpful in deciphering the intended message. Conversely, understanding the intended message aids in grasping the meaning of words (Frank et al., 2009). Consider for example the sentence "please, hand me the umbrella". If the meaning of "umbrella" is unknown to the listener, understanding the intended message becomes challenging. However, if the listener is aware that it is currently raining, the speaker is heading outside, and they always use an umbrella in the rain, the listener can infer the intended meaning of the message and use it to conclude what "umbrella" means. Conversely, a listener who knows the meaning of all the words used in the example sentence but is unaware of the speaker's intentions might infer that the speaker is heading outside in the rain.

In this project, I investigate how reasoning on the intentions of the speaker (which is encompassed in pragmatic reasoning) impacts the learning of new concepts, represented as unknown words. Furthermore, I explore how the beliefs that the learners hold about the meanings of novel concepts being learnt interact with each other when it is known that such concepts do not share the same meaning.

To achieve these goals, I ran an experiment with human participants, in which they learn the meaning of two unknown concepts. Subsequently, I ran cognitive models and investigated how they fit the experimental results. This means that I created artificial agents grounded in pragmatics theory and the hypotheses under examination and had them perform the experiment. In this way, I could evaluate their performance against human participants. In this scenario, cognitive models allow for explicitly testing hypotheses and making well-informed predictions. Furthermore, I could investigate the performance of existing models in this particular task, gauging their ability to generalize to this particular case. The models which I used are Bayesian models, which are probabilistic models employing Bayes' theorem to update the probability of hypotheses based on observed data, such as the feedback to previous trials.

In the remainder of this document, background information on the relevant subjects will be provided (see Chapter 2). This will include a concise overview of cognitive modeling (Section 2.1), the Language of Thought (Section 2.2) and pragmatics (Section 2.3). More specific background on the cognitive models relevant for this work is then provided in Chapter 3. This includes a discussion on models for concept learning (Section 3.1) and pragmatics (Section 3.2). Chapter 4.1 will then describe in more details what the aim of the current task is and will give an overview on the experiment (Section 4.1) and the cognitive models implemented for this study (Section 4.2). Chapter 5 reports and comments on the results of the experiment and the cognitive models. Eventually, a final discussion of the results, potential improvements, and the limitations of the current approach is presented in Chapter 6. This chapter also explores the implications of this study in the field of Artificial Intelligence.

The implementations of the experiment, the cognitive models, the code for the data analysis and the input files can be found at the following link: `https://github.com/lpavan98/Master-s-thesis-material`.

2 General Background

This chapter provides an overview of the key general background topics relevant to the current study: cognitive modeling, discussed in Section 2.1; the Language of Thought, covered in Section 2.2; and pragmatic reasoning, with a focus on Gricean pragmatics, presented in Section 2.3.

2.1 Cognitive Modeling

This section briefly introduces cognitive modeling. For a more comprehensive discussion, please refer to (Sun, 2008).

Many fields benefit from the ability to predict human behavior. In economics, understanding the choices made by decision-makers under specific conditions is of great importance. In artificial intelligence, one could be interested in building agents that can either mimic human behavior in specific tasks or predict human actions to provide assistance when they deem that help is needed, such as in the case of tutoring agents. In psychology, researchers may want to explore how people react to various stimuli while driving, but running experiments with human participants would be infeasible as it could result in them being harmed. While these examples are by no means comprehensive, they show already the vital role of predicting human behavior, which is a central focus of cognitive modeling.

Cognitive modeling is in fact the field that aims at predicting human behavior or emulating cognitive processes through the creation of mathematical and computational models. The emphasis on the emulation of cognitive processes underscores the broader scope of cognitive modeling, which extends beyond practical applications and prediction: it also serves theoretical purposes.

Cognitive modeling can in fact be used to compare different theories in cognitive science. Researchers can for instance use cognitive models grounded in theoretical frameworks (top-down) in order to contribute to the validation of theories and hypotheses. The models can be based on different theories and hypotheses, and the researchers could be interested in which of the models better predict human performance. In order to accomplish this, experiments involving human participants are run, or the results of previously conducted experiments are accessed. The researchers then look at how well the models fit the experimental results, hence having a basis to argue which theories best explain them.

2.2 The Language of Thought

The language of thought hypothesis, proposing that thinking occurs in a mental language (often called *Mentalese*), has roots in the medieval era, but became prominent only centuries later, mainly thanks to the work of Fodor (Rescorla, 2023; Fodor, 1975).

The Language of Thought (LOT) accounts for concepts being realized through a composition of symbols, which means that Mentalese is compositional. Let us consider an example based on the one discussed by Piantadosi and Jacobs in "Four problems solved by the probabilistic language of thought", where we define the concept *nephew* (2016, p. 54). In order to define this concept compositionally, we will need to utilize other concepts and operations. A viable option is to use the operations *parent* and *sibling*, the existential quantifier, the logical conjunction and the free variables *x, y, z.* We would then have:

$$NEPHEW(x, y) := \exists z. PARENT(z, x) \land SIBLING(y, z)$$

Each operation can thus be either reduced to others, as seen in the previous example, or it can be a *primitive.* Primitives are innate operations, and assuming their existence is necessary to account for the capability to compute any operation that is not a primitive itself. Given a limited set of Mentalese symbols, along with operations for combining simple expressions into more complex ones, it is possible to repeatedly apply these operations, generating an infinite variety of mental sentences; this aligns with the idea of the productive nature of thought (Rescorla, 2023).

The LOT also accounts for systematicity of thinking. To grasp the main idea as to why this is the case, consider that, according to the LOT, operations are performed over symbols, and the inference pattern that these operations perform can be applied uniformly to any set of premises that possess the correct logical structure (Rescorla, 2023).

The fact that the LOT naturally accounts for the productivity and compositionality of thinking is seen as an argument for the LOT hypothesis (Rescorla, 2023). The LOT allows to model these phenomena while making use of symbols, which is an argument for its suitability of usage in cognitive modeling. In particular, this makes the LOT especially useful to build symbolic models. It is however important to notice that the LOT is not solely confined to purely symbolic models: a probabilistic LOT can be used to build models that exploit both the symbolic and the statistical approaches to cognitive modeling by integrating rule-based representations with Bayesian probabilistic inference (Piantadosi and Jacobs, 2016).

2.3 Pragmatics

In this section, I will first give an overview of the field of pragmatics. Then, I will introduce Gricean pragmatic reasoning (Section 2.3.1). Finally, in Section 2.3.2, I will focus on conversational implicatures, and, specifically, on quantity implicatures and manner implicatures.

Since I delve into a more detailed overview of Grice's theoretical framework, I will continue to utilize Gricean terminology throughout the remainder of this document. However, please note that the ideas which I discuss and the work that I am undertaking do not hinge on the particular formulation of Grice.

For a more in depth discussion on the topics presented in this section, please consult (Korta and Perry, 2020), which is the main source for this segment.

Pragmatics is the field in linguistics that deals with how people communicate and understand what goes beyond the literal meaning of linguistic expressions. This field can be divided in two major categories: near-side and far-side pragmatics.

Near-side pragmatics focuses on facts about the utterances that are important in establishing the content of linguistic expressions. Topics within this domain encompass resolution of ambiguity and vagueness, as well as the reference of proper names. For example, in order to understand to whom a proper name or pronoun is referred, one has to think about the state of the world in which the communication happens. In this way, one goes beyond the literal meaning of linguistic expressions, which is a matter addressed in semantics.

Far-side pragmatics, conversely, deals with the communication that, despite not being expressed at all in linguistic expressions, is conveyed by them. Suppose for example that we are going out together and I tell you "it will be better to bring an umbrella". By this you will not only understand the literal meaning of the sentence which I uttered: you will also understand that it is raining, or that I find likely that it will rain during the time interval which we will spend outside. The way in which this implicated meaning is perceived is a matter addressed in far-side pragmatics, whereas near-side pragmatics, in the specific case of the example above, would deal with the information on who the "I" in the sentence is referred to.

Since far-side pragmatics is the branch of pragmatics that is relevant in my work, I will expand on it. Furthermore, when using the words "pragmatics" or "pragmatic reasoning" in the remainder of this paper, I will be referring to far-side pragmatics.

2.3.1 Introduction to Gricean Pragmatic Reasoning

One of the most important figures for the advancement of pragmatics is Herbert Paul Grice. The influence of Grice's work led in fact to the development of neo-Gricean accounts, such as (Clark, 1996) and (Levinson, 2000), which still hold a prominent position within the field. Grice distinguished between the literal meaning of linguistic expressions and what is implied by them, and focused on communication between cooperative agents. He proposed in fact the Cooperative Principle:

> Make your conversational contribution such as is required, at the stage at which it occurs, by the accepted purpose or direction of the talk exchange in which you are engaged (Grice, 1975, p. 45).

Those who follow the Cooperative Principle are expected to follow rational maxims in order to make the communication more efficient. Such maxims, which appear in the same work where the Cooperative Principle is originally defined (Grice, 1975), are (in their original form) the following:

1. Quantity:
   - Make your contribution as informative as is required (for the current purposes of the exchange).
   - Do not make your contribution more informative than is required.
2. Quality:
   - Supermaxim: try to make your contribution one that is true.
   - Submaxims:
     - Do not say what you believe to be false.

    - Do not say that for which you lack adequate evidence.

3. Relation:
    - Be relevant.
4. Manner:
    - Supermaxim: be perspicuous.
    - Submaxims:
        - Avoid obscurity of expression.
        - Avoid ambiguity.
        - Be brief (avoid unnecessary prolixity).
        - Be orderly.

Understanding the notion of alternatives is of paramount importance in order to comprehend Gricean pragmatic reasoning. Therefore, I will proceed to provide a brief overview of this topic.

Alternatives

There are always more ways to convey the same meaning by using different linguistic expressions, and please note that the word "always" is not excessively strong here, given the productive nature of natural language. In the Gricean framework, it is common to reason about alternative utterances, following the intuition that if a speaker has various ways to communicate a message and they formulate the message in a specific way, it might imply something about what they wanted to communicate. Given the importance of alternatives in the field of linguistics, the emergence of theories of alternatives is not surprising. Such theories are concerned with "where alternatives come from, what constrains which alternatives are considered for a particular sentence, and what operations can be performed on those alternatives" (Gotzner and Romoli, 2022, p. 214).

2.3.2 Conversational Implicatures

When people converse as cooperative agents, they are expected to follow the conversational maxims, i.e., to be relevant, truthful in what they say and so on. When the uttered sentence seems to violate a maxim, the listener or reader is required to make inferences in order to maintain the assumption that the interlocutor follows the Cooperative Principle. Such inferences are called conversational implicatures.

In the remainder of this section, I delve into two types of implicatures. I first explore quantity implicatures (Q-implicatures), which arise when speakers apparently violate the maxim of quantity. Following that, I discuss manner implicatures (M-implicatures), i.e., implicatures in which listeners engage when a speaker seems to violate the maxim of manner.

Quantity Implicatures

Let us consider an example in which the speaker says “I read most of those books”. A speaker that read all the books in question could have chosen to use the word “most” as well of “all” without saying anything false. In fact, saying “most” to mean *all* is not at odds with the semantic meaning of “most”.

Nevertheless, if the speaker read all the books and used the word “most” rather than “all”, they would have picked a less informative alternative. If the speaker wanted the listener to promptly understand the meaning of the utterance, i.e. if the speaker follows the Cooperative Principle, it would make sense to pick the most informative alternative instead, acting according to the maxim of Quantity. Thus, the pragmatic listener hearing the example sentence with “most” would infer that the speaker did not have the option to use the word “all”, as doing so would result in the content of the sentence being false. That is to say, the listener would infer that the speaker did not read all the books in question, thus making a Q-implicature. Specifically, they would make a scalar implicature, i.e., a Q-implicature in which “we can use lexical substitution to generate the relevant alternatives from the sentence uttered” (Geurts, 2010, p. 49). Therefore, when a speaker uses the word “most”, it likely means “most but not all”.

Manner Implicatures

While a lot of work on theories of alternatives for scalar implicatures has been carried, that is not the case for all domains. For example, there are not so many studies on the domain of M-implicatures, i.e., implicatures based on violations of the Gricean maxim of manner.

Among the most studied type of M-implicatures there are instances where the literal meaning is expressed in a convoluted manner when a simpler expression is feasible.

An iconic example of this kind of implicatures originally presented by Horn in "Toward a new taxonomy for pragmatic inference: Q-based and R-based implicature" (1984, p. 27) is the following:

> Black Bart killed the sheriff.
> Black Bart caused the sheriff to die.

Despite both sentences conveying the same literal meaning, they are often interpreted differently: on the one hand, when presented with the second sentence, people tend to perceive that Black Bart caused the sheriff's death in a non-stereotypical way. On the other hand, the first sentence is more likely to be interpreted as Black Bart killing the sheriff in a stereotypical fashion.

In the Gricean framework this can be explained in terms of the second sentence causing an M-implicature: since the message was not expressed as briefly and simply as it could have been, the speaker did not want to convey the same meaning that they could have conveyed by using the first example sentence.

3 Relevant Cognitive Models for this Work

Utilizing computational models in linguistics is a well established practice, exemplified by seminal works on the emergence of language. Computational models can in fact foster explicitness in formulating explanations, show how explanations work and help exploration guiding to the formulation of new theories (Kirby and Hurford, 2002; Christiansen and Kirby, 2003). More recently, probabilistic models have gained significant popularity in the field of linguistics, a trend exemplified by the growing interest in and adoption of Bayesian models for pragmatics and word learning, as evidenced by the influential works of Frank and Goodman (2012) and Tenenbaum and Xu (2002).

In this chapter, I will discuss relevant models in these domains, beginning with concept learning (Section 3.1), and then moving to pragmatics (Section 3.2).

3.1 Modeling Concept Learning

In the realm of cognitive modeling, concept learning stands out as a domain where the advantage of the Language of Thought (LOT) providing without additional cost compositionality and productivity is highly appealing, as conceptual systems surpass collections of examples, thus necessitating a compositional approach (Piantadosi and Jacobs, 2016; Piantadosi et al., 2016). Moreover, some concepts are defined by logical rules. I will provide a few illustrative examples, drawn from the work of Piantadosi et al., where this is discussed more in details (2016, p. 395). Two trivial examples of domains where concepts are rigorously defined by logical rules are mathematics and taxonomies. However, in natural language there are also plenty of examples: there can be words used to express logical relations, such as, for instance, quantifiers or conjunctions.

Modeling concept also benefits from the adoption of a probabilistic framework, as illustrated in the influential study by Tenenbaum and Xu titled “Word learning as Bayesian inference” (2002). In this work, the authors report how Bayesian models for the learning of single words could fit experimental data obtained by both children and adult participants. The mathematical model at the basis of the computational cognitive model is quite simple. Among its basic elements we have an unknown concept C, a hypotheses space H with elements $h \in H$ that are candidate possible concepts for C and a set of observed examples X. Given the examples X, the model systematically assesses each potential hypothesis h regarding candidate word meanings. This evaluation is carried out through the application of Bayes’ theorem. What is computed is thus the posterior probability for a hypothesis h ($p(h|X)$), as:

$$p(h|X) = \frac{p(X|h) \cdot p(h)}{p(X)}$$

In order to employ this Bayesian approach, the authors needed to delineate the hypotheses. While one could manually define the hypothesis space, such an approach would introduce free parameters within the model. The authors thus opted for a structured hypothesis space consisting in a taxonomy of nested categories, constructed through hierarchical clustering on similarity rankings provided by human participants. Hypotheses could then be simply mapped to clusters.

The preceding discussion makes it trivial for the discerning reader to understand how LOT is well suited to be used within a Bayesian framework. Hypotheses can in fact be represented as LOT expressions. This will be further discussed in the next section (3.1.1), where a specific model of importance for this project is talked about. Models that amalgamate Bayesian induction with compositional representation systems, such as Languages of Thought (LOTs), can harness the strengths of both approaches, enabling them to account for the fuzziness of human concepts and for the human simplicity preference in categorization when learning concepts (Piantadosi and Jacobs, 2016; Piantadosi et al., 2016).

3.1.1 Concept Learning Study by Piantadosi et al.

In “The logical primitives of thought: Empirical foundations for compositional cognitive models”, Piantadosi et al. show how they modeled the learning of a concept novel to participants through Bayesian cognitive modeling based on LOTs and propose that any set of primitives and rules for combination can be seen as a scientific hypothesis (2016). Each LOT utilized in the study is in fact viewed as a hypothesis.

The LOTs investigated include for instance a language allowing only expressions in conjunctive normal form, one which they called *SimpleBoolean*, using *and*, *or* and *not*, and *FullBoolean*, which extends *SimpleBoolean* by adding logical implication and biconditional.

The study takes the human bias for representational simplicity as a foundational principle. Since different LOTs assess simplicity in distinct manners, the simplicity bias allows to make predictions about the expectations of each LOT. Simulations are then run for each LOT, and the results thereof are compared with the results of an experiment run with human participants. It can thus be observed how well the models fit the experimental data and see which LOT better predicts the experimental results.

In the experiment, participants were provided with the task of discovering the meaning of "wudsy", a word in an alien language. In each trial, three items were presented to participants. Such items were squares, circles or triangles of one of three sizes and colors each. In each trial, participants had to select all items which they thought were wudsy. The correct answer to the trial was unveiled after the participants made their selection, and what was discovered about the extension of wudsy remained on screen until the end of the task.

The choice of a sequential learning paradigm allowed for the definition of learning curves and patterns of mistakes, which were correctly predicted by the models. In fact, the models could also emulate participants by reaching a point in their learning after which they were accurate for the rest of the task.

The probabilities of primitives and operations for each LOT were inferred from the data. The LOT that could fit human data the best when no quantifier is used in definying wudsy is *FullBoolean*. It is noteworthy that *FullBoolean* has two low probability operations, as Piantadosi et al. inferred from the data: the logical entailment and the biconditional. Thanks to these two operations, *FullBoolean* performs better than *SimpleBoolean*, which employs the same operations apart for these two. This speaks against excluding low probability operations from grammars.

3.2 Modeling Pragmatics

Efforts to model Gricean pragmatic reasoning have manifested in various approaches. In this section, two of the most prominent frameworks in this realm are discussed: the Rational Speech Act and game theoretic approaches. The reason why I focus on these approaches is that they can be extended to not only account for the extensively researched Q-implicatures, but also for M-implicatures in the form discussed in Section 2.3, even in one-shot trials without any learning required.

In the remainder of this section, I will start by giving a brief overview of the game theoretic framework (Section 3.2.1). Then, I will discuss the Rational Speech Act framework (Section 3.2.2) and, finally, the Lexical Uncertainty Model (Section 3.2.3).

3.2.1 Game Theoretic Framework

Game theory provides tools to mathematically model how rational agents take decisions that seek to maximize their utility, based on the decisions that other agents could take. Since pragmatic reasoners are rational agents, game theory can be applied in pragmatics to model the behavior of conversational agents. In such scenarios, the agents involved in the game could be speakers and listeners, with the speakers that need to choose among possible utterances to send and the listeners that reason about how to interpret the received utterances. When modeling Gricean reasoning, it is necessary to assume that the agents, i.e., the interlocutors, are cooperative. Therefore, they will both aim for the listener to arrive at the interpretation intended by the speaker. The advantage of utilizing game theory in modeling pragmatic reasoning is that it provides a framework to represent, analyze, and find solutions for communication problems (Benz and Stevens, 2018).

In the rest of this section, I will briefly introduce one such model that employs game theory to capture Gricean pragmatic reasoning, the Iterated Quantal Response model (Franke and Jäger, 2014). I will not delve into the field of game theory or elaborate on its applications in other pragmatics models, as it is not pertinent to the current work. For an introduction to game theory, you can refer to (Shoham and Leyton-Brown, 2008) or to the more classical work in the field represented by the book “Games and Decisions: Introduction and Critical Survey” (Luce and Raiffa, 1957). For more information on the applications in pragmatics, see (Benz and Stevens, 2018).

The Iterated Quantal Response Model

Equilibria in the linguistic exchange are found by The Iterated Quantal Response (IQR) by modeling agents that iteratively reason about each other. This model does

not predict that interlocutors always take the best action, as it is probabilistic. This contributes in making the IQR model more plausible, as it better explains behavioral data from psycholinguistic experiments (Franke and Jäger, 2014).
The reason why the IQR model holds significance for my work is that it can be extended to account for M-implicatures of the type discussed in Section 2.3.1, with Horn's example in that section serving as an illustration. In order for agents to be more likely to engage in M-implicatures than not, it must however be assumed that if they do not have at their disposal any action leading to a high reward, they are more likely to take actions with lower payoffs, thus engaging in exploratory behavior (Franke and Jäger, 2014).

3.2.2 Rational Speech Act Framework

The Rational Speech Act (RSA) framework utilizes Bayesian update to model pragmatic reasoning as a process of recursive social reasoning (Frank and Goodman, 2012). In the RSA, in fact, listeners take into account the speakers' reasoning about the choice of a particular message among alternatives. Similarly, speakers can base their decisions on their assumptions about the reasoning of listeners. For a more detailed explanation of recursive social reasoning, please refer to the next section, which is exclusively devoted to this topic.

The RSA assumes that language production and interpretation are probabilistic processes, allowing it to better account for the empirically observed variability in these processes (Degen, 2023). Within the RSA frameworks, listeners consider the probability of being in a certain state of the world, conditioned on the message conveyed by the speaker. Conversely, speakers consider the probability of sending one specific message among alternative messages, conditioned on the observed state of the world. In this context, states of the world are different scenarios that may be true. Speakers observe the state of the world and communicate it to listeners. Suppose for example that there is an agent (the speaker) that has an apple in a bag. Further suppose that another agent (the listener) considers it possible that the speaker has an apple in the bag. However, the listener also considers the possibility that the speaker forgot to bring apples. The listener thus considers two states of the world to be possible, one where there is an apple in the bag and one were there is none. The listener might also have different prior probabilities for these states. For example, knowing that the speaker always forgets to bring apples, the listener might consider much more likely that the state of the world is the one in which the speaker has no apples in the bag. Thanks to its Bayesian nature, the RSA framework allows for the specification of such prior beliefs. When the speaker tells the listener that there is an apple in the bag, the listener updates their beliefs of the world being in each possible state.

Being able to account for the variability in language production and interpretation is not the only advantage of the RSA framework. In fact, the vanilla RSA can be extended to account for various factors, including politeness, the speaker's uncertainty about the world state, or lexical uncertainty (the latter will be discussed in Section 3.2.3) (Scontras and Tessler, 2017). All these reasons contribute to make the RSA framework stand as the most influential probabilistic approach within the field of pragmatics (Degen, 2023).

In the rest of this section, I will first dive into the nature of the social reasoning involved in this framework. Subsequently, I will discuss the basic components of the RSA framework, along with their respective mathematical formulations. Lastly, I will show how the Gricean maxims of Quantity, Quality and Manner are encoded. Please note that, although the RSA is a simple model, a deep understanding of its intricacies might not be gained by reading my explanation only. Therefore, I suggest to consult (Scontras et al., 2021) for a more comprehensive discussion and (Scontras and Tessler, 2017) for an interactive implementation with which you can experiment online.

Recursive Social Reasoning

The base case for the recursion in the social reasoning assumed by the RSA framework is the literal listener, indicated as L_0. The literal listener considers the speaker's utterance based on its semantics only. This means that only the literal meaning is taken into account, hence the appellation "literal listener". In fact, L_0 does not engage in pragmatic reasoning. In order to have a pragmatic listener, we need to go one step further with the recursive reasoning and consider L_1 as the listener. L_1 reasons about a speaker that chooses the message to send based on the assumption that the listener is not going to reason pragmatically. This baseline speaker is S_1. The next step in the recursion is to take S_2 as speaker and L_2 as listener, where S_2 considers L_1 as listener and L_2 considers S_2 as speaker.

In general, L_n grounds its reasoning on the assumption that the speaker is S_n, whereas S_n bases its reasoning on the assumption that the listener is L_{n-1}. The discerning reader may now be struck by the fact that if a listener L_n correctly assumes the pragmatic level of a speaker to be S_n, the speaker cannot be assuming the correct level of the listener, in that S_n assumes the listener to be L_{n-1}. The reverse is also true: if the speaker holds a correct assumption about the listener's pragmatic level, the listener cannot be correctly assuming the pragmatic level of the speaker. This however does not constitute a problem. In fact, speakers and listeners only make assumptions about the pragmatic level of their interlocutor, they do not need to know it. It can for example also be the case that the speaker is S_2 and the listener is L_0.

Recursive social reasoning can extend infinitely in theory. Therefore, there would in principle be no issue in having, for example, L_5 listeners. In order to model human behavior, it is however neither necessary nor helpful to engage in such deep recursion. Franke and Degen explored the pragmatic levels at which humans are better modeled (2016). They showed that most participants in their study could be better described as S_1 speakers (with a minority as S_2 speakers). Moreover, participants could be modeled as both L_0, L_1, or L_2 listeners, with L_1 being the most likely.

Basic components

As already explained in a discursive manner, listeners have *interpretation rules* to compute the conditional probability of a state s given an utterance u, whereas speakers have *production rules* to calculate the conditional probability of an utterance u given a state s.

In this section, I will give an overview of the fundamental components of the RSA framework: L_0, L_1 and S_1, giving an overview of their (interpretation or production) rules.

The interpretation rule P_{L_0} of the literal listener L_0 is

$$P_{L_0}(s \mid u) \propto P(s) \cdot [\![u]\!](s)$$

Here, $P(s)$ represents the prior belief of L_0 about s being the actual state of the world, while $[\![u]\!]$ is a function that maps states to 0 if u is incompatible with such states and to 1 otherwise. $[\![u]\!](s)$ thus expresses whether the utterance u is true or false in the state s. In this way, the truthfulness based on semantics is accounted for.

As discussed previously, S_1 considers the perspective of L_0 when choosing the message to use. In practice, this means that the pragmatic speaker accesses the interpretation rule of the literal listener, i.e., P_{L_0} appears in the production rule of S_1, which is

$$P_{S_1}(u \mid s) \propto exp(\alpha \cdot (logP_{L_0}(s \mid u) - C(u)))$$

The term $logP_{L_0}(s \mid u) - C(u)$ is referred as $U(u; s)$, the utility of S_1 with respect to utterance u and state s. The parameter α then controls how rationally the speaker is in choosing utterances. This parameter is defined to be greater or equal than 0. When it is set to 0, utterances are chosen at random. The greater α is, the more optimal the agent's choice is. As α approaches infinity, the speaker deterministically chooses the most optimal utterance. $C(u)$ instead encodes the cost of the utterance u. A greater cost represents a more expensive utterance, which could for example be the result of a longer message. For an explanation of the derivation of the exponential and logarithmic functions in the formula, I recommend referring to the appendix of (Scontras and Tessler, 2017). I will not discuss that here, since it goes beyond the scope of what is necessary for understanding the functioning of the RSA.

The last basic element of the RSA is the pragmatic listener L_1, whose interpretation rule is

$$P_{L_1}(s \mid u) \propto P_{S_1}(u \mid s) \cdot P(s)$$

L_1 thus uses Bayes' rule to infer the most likely state of the world given the received message, and it achieves this by considering the reasoning of S_1.

Encoding Gricean Maxims

We are modeling Gricean pragmatic reasoning, which is based on the Cooperative Principle (see Section 2.3.1). Thus, agents within the RSA framework are cooperative. For this reason, S_1 aims to maximize the probability that L_0 reaches the conclusion that the state of the world is the correct one, i.e., the one that the speaker observed. This is captured by the term $logP_{L_0}$ in the utility function. This term represents in fact the informativeness, and makes it possible to encode the Maxim of Quantity. The maxim of Quality is then encoded thanks to the function $[\![u]\!]$, which allows S_1 to only choose true messages. Knowing that the state of the world is s, S_1 would calculate $[\![u]\!](s)$ for each possible utterance $u \in U$, where U is the set of utterances which are considered. If $[\![u]\!](s) = 0$ for an utterance u, S_1 does not use that utterance, as doing so would result in $P_{L_0}(s|u)$ being 0 and, in turn $P_{S_1}(u|s)$ being 0. Lastly, encoding the maxim of Manner would not be possible without $C(u)$, which incentives the choice of cheaper messages. The cost is however not sufficient to encode the maxim of Manner in the vanilla RSA, as discussed in Section 3.2.3.

3.2.3 Lexical Uncertainty Model

The Lexical Uncertainty Model (LUM) is an extension of the vanilla Rational Speech Act developed by Bergen et al.(2016).
The main idea behind this model is to add lexica to vanilla RSA. Therefore, this section will start with a discussion on the utilization of lexica in this model. Subsequently, the strengths and limitations inherent in the LUM will be outlined.

Adding Lexica to the RSA

The LUM allows the conversational agents to reason about the lexicon in use. Here, the lexicon is defined as a function that assigns truth values to each utterance-world pair. Specifically, $\mathcal{L}(u, w)$ returns 1 if the utterance u and the world w are compatible, and 0 (or a value approaching 0) otherwise. Bergen et al. propose in fact four approaches to define $\mathcal{L}(u, w)$ (2016, pp. 25, 26). I will not provide details on all these approaches and why they were proposed, as that is not relevant for this work. Therefore, I invite the reader to look at the original study for a more detailed discussion. Let me however clarify what approach I am following: I define $\mathcal{L}(u, w)$ as returning 1 if the utterance u and the world w are compatible, and 10^{-9} otherwise. One of the approaches suggested by Bergen et al. consists in fact in defining $\mathcal{L}(u, w)$ as equal or lower to 10^{-6} when u and w are not compatible.

Through the usage of lexica, a conversational agent can address its uncertainty regarding the meanings of messages. Let us examine the example of scalar implicatures. Specifically, consider the running example discussed in Section 2.3.1, "I read most of those books". After being exposed to this sentence, the listener will not be sure about the meaning of "most". It could in fact be taken to signify what it literally means (*most or all*), or the alternative interpretation *most but not all*. In the LUM, this example is encoded by having the listener contemplate both lexicons where "most" corresponds to one option and the other.
With the inclusion of the lexica in the LUM, the mathematical expressions for P_{L_0},

P_{S_1} and P_{L_1} become

$$P_{L_0}(s \mid u, \mathcal{L}) \propto P(s) \cdot \mathcal{L}(u, s),$$
$$P_{S_1}(u \mid s, \mathcal{L}) \propto exp(\alpha \cdot (logP_{L_0}(s \mid u, \mathcal{L}) - C(u))),$$
$$P_{L_1}(s \mid u) \propto P(s) \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot P_{S_1}(u \mid s, \mathcal{L}).$$

Here, Λ is the set of lexica, $\mathcal{L} \in \Lambda$ is a lexicon and the other elements are as defined in vanilla RSA. The factor multiplying α in P_{S_1}, also as in vanilla RSA, is referred to as the utility of S_1 and can be written as $U_1(u \mid s, \mathcal{L})$.
Given that the recursion level in the current study is not confined to the L_1 listener, I will here also present the production and interpretation rules for recursive social reasoning for S_n and L_n:

$$P_{S_n}(u \mid s) \propto exp(\alpha \cdot (logP_{L_n-1}(s \mid u) - C(u))),$$
$$P_{L_n}(s \mid u) \propto P(s)P_{S_n}(u \mid s).$$

Please note that the formulations provided in this section are simplified expressions, due to the assumption that it is common knowledge that the speaker is aware of the relevant world state. The LUM also accounts for speakers that lack knowledge of the relevant world state. However, this is not pertinent to my work. Therefore, I will always keep this assumption in the current work.

Advantages and Limitations

Bergen et al. demonstrated that, for M-implicatures of the type discussed in Section 2.3.1 (exemplified by Horn's example in that section), the basic RSA interprets two utterances conveying the same literal meaning uniformly across all levels of the speaker-hearer recursion. Conversely,the LUM allows for modeling such implicatures (2016). Furthermore, the LUM allows for modeling non-convex disjunctive implicatures. I will however not spend more words on this, as I will stick to what is most relevant for the current study. If interested, please refer to the work of Bergen et al. (2016). Something that is instead noteworthy in the context of the specific application under scrutiny in this study is that the lexica in the LUM can also be used to account for possible meanings of novel words, such as words in a novel language.

Providing here a practical example the LUM's work in modeling M-implicatures is unfeasible, as it would involve pages of calculations. Part of the reason for this is that, in order for the LUM to account for M-implicatures, higher recursive levels are needed, such as a listener being at least L_2. L_1 listeners would in fact be less likely to engage in the M-implicature than to do it. This is an obvious limitation of the LUM, since modeling human communication cannot require interlocutors to engage in so deep social reasoning. In fact, humans do not seem to engage in deep social reasoning, as discussed in Section 3.2.2.

For readers interested in experimenting with an online demonstration of the LUM, please visit the eighth chapter of the webbook by Scontras and Tessler (2017). Moreover, in the Appendix you can find (most of) the code implementing the LUM in the cognitive models that I built as part of this work, which serves as a practical example of an implementation of the LUM.

4 The Current Study

Examining the impact of pragmatic reasoning on concept learning is not a novel idea and has already been investigated through a cognitive modeling approach. In particular, Frank and Goodman have already delved into this area using cognitive models within the RSA framework (Frank and Goodman, 2014). Their study deals with Q-implicatures, a common focus in computational pragmatics. Additionally, they account for concepts being learnt in a single round rather than gradually over time. In fact, even if it has been attempted to extend the RSA framework to accommodate multiple turns in conversation (Anderson, 2021), its vanilla version only contemplates single conversation turns.

In my work, I am using conversational implicatures to investigate how pragmatic reasoning and concept learning interact. This becomes particularly relevant when learning two unknown concepts simultaneously, as discussed in Section 4.1. My work also allows for concepts to be learnt gradually through a series of trials. This approach lets me observe whether the beliefs on the meaning of one concept are updated even when only information on the other concept is acquired, based on the fact that the two concepts are known to have different extensions. Since I am already modeling pragmatic reasoning through a Bayesian model, it makes sense to model the learning similarly. This consideration is reinforced by the favorable outcomes of Bayesian modeling in the context of concept learning (Tenenbaum and Xu, 2002; Piantadosi et al., 2016).

In order to employ Bayesian modeling, hypotheses on the meaning of the concepts to use as priors must be defined. As discussed in Section 3.1, it makes sense to define such hypotheses in a structured way. I will thus employ the Language of Thought. This enables the tracking of probabilities of hypotheses on the meaning of the concepts throughout the task, as demonstrated in (Piantadosi et al., 2016).

4.1 Experiment

The experiment is presented to participants as a concept learning experiment, taking the form of a cooperative communication task. The involvement of pragmatic reasoning is not mentioned, in order not to bias the participants' approach to the task.

Making sure that participants view the experimental task as a conversational task is of paramount importance. In fact, it is needed to elicit pragmatic reasoning. Participants are misled into thinking that they will either send or receive messages from other participants, taking the role of *senders* or *receivers*, respectively. As senders, participants will be informed about the meaning of two words and they will be given the task to help other participants learn such words. They will be presented with three objects (one of which will be marked) and they will have to press the key corresponding to the word which they would like to *send* to the participant who they think will receive their message. Section 4.1.6 explains why the letter and not the word is seen by participants on the receiving side. Conversely, receivers will be shown a set of three items and a word and they will need to select one of the items based on the word.

Importantly, when participants act as senders, no pragmatic reasoning will be involved: the highlighted object will always be in the extension of exactly one concept. This works as a precaution against biasing the participants towards engaging in pragmatic reasoning. Such an approach has already been successfully employed in other works in the field, such as in (Franke and Degen, 2016).

In the remainder of this section, I will explain the experiment and how conversational implicatures are elicited in it (Section 4.1.7).

4.1.1 Implementation

The experiment was run online, as was done in similar works in concept learning and pragmatics (Piantadosi et al., 2016; Buccola et al., 2018; Frank and Goodman, 2014; Franke and Degen, 2016; Bergen and Goodman, 2012). I implemented it as a web page using JavaScript and HTML, so that it could run on participants' browsers. This allowed me to employ the JavaScript framework *jsPsych*, which is specifically designed for the development of behavioral experiments (de Leeuw et al., 2023)[1].

[1] A less complex approach would have been to use tools based on graphical user interfaces or simple languages designed for experiment building, which do not even require downloading

For the back-end, I used *cognition.run*, an environment for running experiments online, designed specifically for hosting *jsPsych* experiments.

4.1.2 Participants

Participants were recruited from my personal social contacts and from students in a class taught by my first supervisor, Prof. Dr. Jakub Dotlačil, resulting in a total of 19 participants [2].

4.1.3 Materials

The items used in the experiment have the same features of those used in (Piantadosi et al., 2016), which is the main reference for the concept learning part of my project. The reason why I did so is that the fewer changes are carried in the experiment design, the more likely it is that the result are similar, i.e., that LOT models for learning are effective. The objects presented to participants thus have three main features: shape, color and size. The possible shapes are triangle, circle and square; the color of an item is red, green or blue; the item's size is small, medium or big.

any additional software (the final version of the experiment implemented in JavaScript using *jsPsych* comprises instead approximately 1000 lines of code). Therefore, I began using simpler tools. The ones I considered first were *PsyToolkit* (Stoet, 2010, 2017), which I had already used to build an experiment as part of a university project, *lab.js* (Henninger et al., 2019), which can be paired with *Qualtrics* (which comes with the advantage of being a software deemed safe by Utrecht University), and *PCIbex Farm* (Zehr and Schwarz, 2023), which was suggested by my supervisor.

Due to the complexity of the experiment and the consequent need for a great flexibility in the implementation process, I however had to abandon these early attempts. I therefore started implementing a web page using JavaScript with the help of *_magpie* (`https://magpie-ea.github.io/magpie-site/`), an architecture for making online experiments which I had already used during my Bachelor's as part of a university project. The documentation of *_magpie* was however not always up to date and I encountered compatibility issues. Henceforth, I searched for alternative tools until I eventually learnt about *jsPsych*. Recognizing its active community, I invested time in learning how to use it and refining my JavaScript skills, which were quite rudimentary at the time, in order to implement my experiment.

[2] Recruiting participants through Prolific, as initially planned, would have allowed for a larger participant pool. However, using Prolific proved problematic, as shown by the data from the first participants recruited through that platform. These participants made in fact numerous errors in both the sender's task and the control trials, suggesting that they were not fully engaged in the task. This observation is further supported by their rapid completion times, significantly shorter than those of the new participants (the experiment generally takes around half an hour to complete). It thus seems that using Prolific to recruit participants for an experiment of this length, where participants have the capability to rush through in order to receive monetary compensation, is not ideal.

Concepts used throughout the experiment describe sets of items. Such sets are defined using the *SimpleBoolean* LOT (see Section 3.1.1) to combine items' features.

The words for the concepts that are utilized are not shown. Instead, only the letters p and q will be displayed, as described in Section 4.1.6.

4.1.4 Design

All participants act first as senders and then as receivers. The sender phase serves the purpose of eliciting pragmatic reasoning (for the experimental group) and understanding the structure of the tasks.

In the remainder of this section, I will first delineate the distinction between the control group and the experimental group. Following this, I will motivate the choice of materials. Lastly, I will explain the randomization procedures involved in the experiment.

Control Group and Experimental Group

The phases in which participants act as receivers are the critical phases. These phases are the ones which differ among the control and experimental groups.

The experimental group is the one for which pragmatic reasoning is involved, whereas the control group is structured in such a way that no pragmatic reasoning should be employed. In the experimental group, on the one hand, two concepts are learnt at the same time for each receiver block. On the other hand, in control groups, concepts are learnt individually. This removes the conversational implicatures of the form discussed in Section 4.1.7. It also avoids the potential update of beliefs regarding the meaning of one concept, given what is learnt about the other concept. For a more detailed description of the receiver blocks, see Section 4.1.5.

Choice of Materials

In each block, the concepts involved are different. The items shown to participants throughout the experiment will always have the same features (color, shape, size). Thus, after learning that in the extension of a concept there are objects with certain features, it might be that participants will assume that having an item with the same features in another phase of the experiment will be more unlikely, even if it will be specified that different concepts in different phases of the experiment are completely unrelated.

To avoid this potential issue, it might be beneficial to use items with different features. For example, instead of color, size and shape, the second time in which participants are receivers the objects' features might be numerosity, colors and shapes. Introducing new features such as numerosity might however change the results in unexpected ways. In fact, manipulating too many variables from the experiment by Piantadosi et al. (2016) may not be advisable if I aim to obtain similar results with the LOT models. Consequently, I am maintaining consistency with the original study by using the same colors, shapes, and sizes.

Even modifying just the shapes is not without risks, as complex shapes might be grouped by participants in unforeseen patterns. For example, pyramids and cones might be perceived as similar and grouped together with a high likelihood. Conversely, simpler shapes like stars, lines, and dots might not be ideal for utilizing existing features such as size. For this reason, I am also keeping the same colors, shapes and sizes as in the original study.

Randomization

The order in which the three items in each trial appear is randomized, as well as the the order of the listener phases. For the control group, which concept is first learnt in each receiver block is also selected randomly.

Conversely, the structural formulation of concepts in each phase of the experiment remains consistent. However, the specific features incorporated into these formulations are not held constant among participants. For example, a concept a will always be formulated as $feature_1 \wedge feature_2$, but this can translate into $triangle \wedge blue$ for a participant and $circle \wedge big$ for another participant. This is done to address the concern posed by salience differences between features.

4.1.5 Procedure

This section outlines the sequence of events that participants experience during the experiment, following the chronological order of the experiment itself. Therefore, I will start describing the initial stages of the experiment. Subsequently, I will cover the sender block and, finally, the receiver blocks.

Starting the Experiment

Participants are first given a general description of the experiment, asked to give their informed consent for the treating of their data, and asked to tick a box if they are color-blind. Afterward, if they are using a mobile device, they see a message telling them that they need to use a laptop or desktop computer, which ends the experiment. If they are not using a mobile device, they are instead provided with some general instructions.

Participants are told that the focus of the study is to learn words in an alien language, that the study involves communication among participants and that they can either be given the meaning of two words and asked to help other participants discover the meaning of those words, or they can be unaware of the meaning of two words and be asked to figure out their meanings with the help of another participant. It is specified that the communication among participants is asynchronous and entire blocks of messages are sent at once and that each alien word is associated to a key on the keyboard (p or q).

The introductory part is at this point done and the experiment enters full-screen mode.

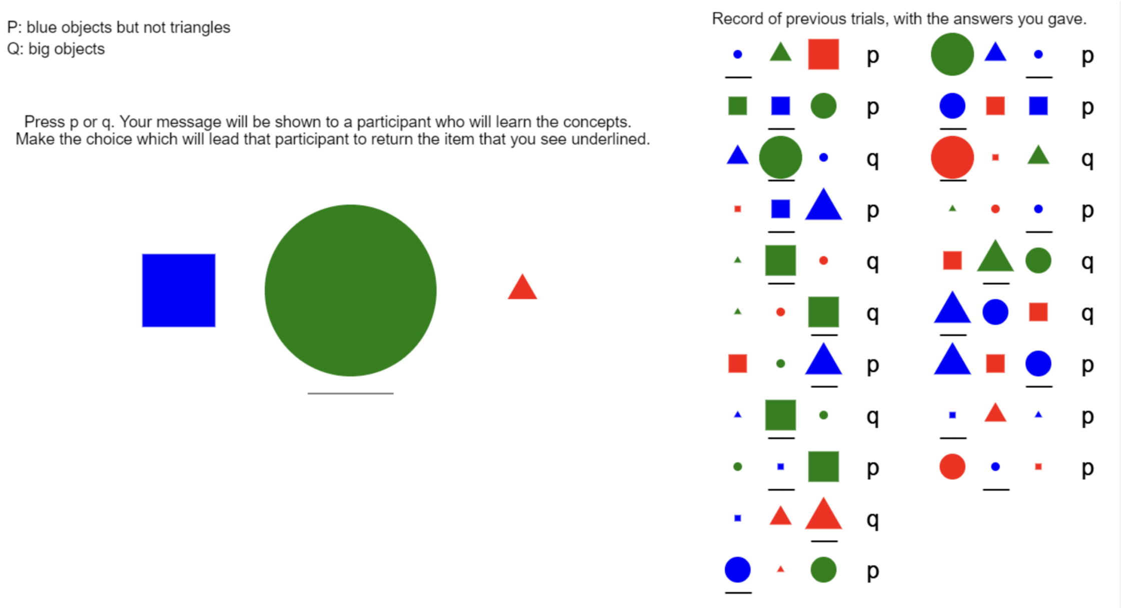
Sender Block

Before the beginning of the task, participants are told what their task as senders is (in short, they need to communicate either p or q to the other participant). Furthermore, it is explained to them that the receiver who will see the word which they send will see it associated with the same set of three objects.

In the actual task, participants are shown the keyboard keys p and q associated to each of the two concepts, together with the meaning of the concepts, as shown in Figure 4.1. Such concepts are displayed on the top of the screen and are the same for all participants. In fact, the randomization in the meaning of the formulations of the concepts only applies to the receiver tasks.

As shown in Figure 4.1, in the left part of the screen the current trial is displayed. The trials consist in fact of three figures, out of which one is underlined, as it is the one that senders need to make sure that the hypothetical receiver picks, by sending the message which they deem to be the best to achieve that goal.

On the right part of the screen, participant see instead the record of the previous trials, i.e., all trials completed up to that point, and the key which they pressed



P: blue objects but not triangles
Q: big objects

Press p or q. Your message will be shown to a participant who will learn the concepts.
Make the choice which will lead that participant to return the item that you see underlined.

Record of previous trials, with the answers you gave.

Figure 4.1: Screenshot of the sender task taken at the 21st trial

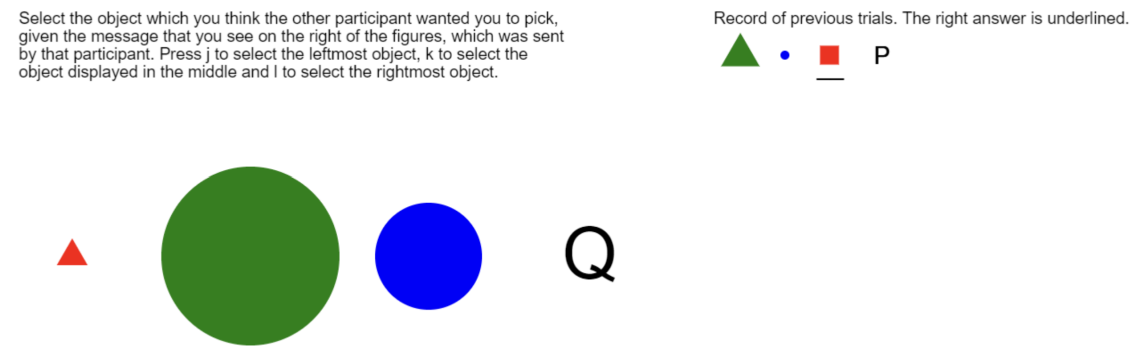
(one among *p* and *q*). Showing the record of previous trials in the sender's task is not useful during that task, but it will be once the participants reach the receivers' tasks, as explained in the next section.

Receiver Blocks

After the first block, in which the participants act as senders, there are four blocks in which they are receivers. Each of these four blocks is split in 2 parts for the control group, as one concept is learnt on each sub-part of the block.
In this section, I will first summarize the information on the task that is given to participants. Then, I will go into how the actual task works.

Before beginning each block as receivers, participants are informed that they are going to learn two novel concepts, and each of those concepts will be associated with one letter among *p* and *q*. The reason for that association, they are told, is that another participant who acted as sender sent them a message (either *p* or *q*) for each set of objects (at each trial). Participants are also told that the aim of senders is to have receivers pick a certain object out of the three displayed ones, for each set of objects. Furthermore, they are instructed that their task as receivers is not only to learn the meaning of the two concepts, but also to select the object that the other participant wanted them to return.

Similarly to what happened in the sender block, the current trial is presented at the left part of the screen, while the right part of the screen is used to display the record of previous trials. However, as you can observe from Figure 4.2, there are four main differences from the sender block. Firstly, trials come with a letter (either *p* or *q*) in the receiver blocks. It is explained to participants that such letters are the messages received by the senders. Secondly, the current trial does not contain any underlined figure, as receivers need to pick a figure based on the message and their belief about its meaning. Thirdly, after each trial the participants receive some feedback, telling them whether their answer was correct or not, where correct answers are the ones that the hypothetical senders want the receivers to pick. Lastly, upon completion of each block, participants are asked for the meaning of the two concepts. Responses are solicited in the form of free-text.



Select the object which you think the other participant wanted you to pick, given the message that you see on the right of the figures, which was sent by that participant. Press j to select the leftmost object, k to select the object displayed in the middle and l to select the rightmost object.

Record of previous trials. The right answer is underlined.

Figure 4.2: Screenshot of a receiver task taken at the 2nd trial

In the control group, the meaning of a concept is asked in the middle of the block, after the section of the block utilizing that concept has concluded. The meaning of the other concept is only asked for at the end of the block, once the second section of the block, which utilizes that concept, has been completed.

Displaying the record of trials is of paramount importance to elicit conversational implicatures. On the record, participants can in fact find trials where the three objects are repeated, but the message received is different. They might thus think that, if the sender changed the message with the same three objects, the object to be picked is not the same as before (see Section 4.1.7).

The reason why it was necessary to see the record of previous trials during the sender task should now be clear: participants are supposed to be more likely to reason in the way that allows them to engage in conversational implicatures if they know that the senders can also reason about previous trials.

4.1.6 Messages Associated to the Concepts

The words for the concepts learnt in the experiment are not written anywhere. Instead, they are always associated with the letters *p* and *q*. The aim of this approach is to avoid the bouba/kiki effect (Köhler, 1967), which predicts that there is a non-random association between speech sounds and object features, in particular shapes. The reason why the words are not written is that if participants cannot read the words which are presented to them, they cannot associate the sounds in the words to the features of the experimental items.

Let me now briefly explain why I chose to pick specifically the letters *p* and *q*.
If participants act as a senders, they have to press the key corresponding to the concept that they want to communicate, and that key is one between *p* and *q*, as they are letters on different sides of the keyboard. The sender is told what the words mean, i.e., the concepts they stand for. Therefore, there should be no bias generated by the location of the keys on the keyboard.
If participants act as receivers, they see instead a letter between *p* and *q* next to each trial. In this way, receivers are led to associate the concepts to the corresponding letters. Participants are told that such letters are the messages received from other participants who acted as senders. Since the receiver blocks come after the sender block and in the sender block the messages to be sent are *p* and *q*, it should make sense for participants acting as receivers why the messages are *p* and *q*. In fact, all participants complete the sender block before completing the receiver block.

4.1.7 Elicitation of Conversational Implicatures

In alignment with the Gricean paradigm, speakers likely select the utterance that maximizes the ease of understanding for the listener. In the case of the current experiment, this translates in the sender choosing the message that -as far as the speaker knows- has the highest probability of leading the receiver to select the intended object.

Elicitation of M-implicatures

In agreement to the maxim of Manner, a speaker should be perspicuous and avoid ambiguity (see Section 2.3). If the receiver expects the sender to choose the message that facilitates them in the task at hand, we can anticipate the receiver to reason pragmatically in situations similar to the one I will present in the following.

Suppose that, out of the three objects shown to the receiver in a specific trial, one -let us suppose without loss of generality, the green square- has a higher likelihood to be the target object of the message chosen by the sender for both messages, even though that object was not observed to be in the extension of either concept by the receiver. Further suppose that the sender sends the message p, and the receiver, returning an object, sees the feedback indicating that the target object was the green square. Assume that the same three objects are then presented again to the receiver. This time, however, the sender chooses the other message. Since the object selected beforehand (the green square) also has a higher probability to be the target object if the message is q, a receiver who is not using pragmatics at all would choose to return the same object again. Conversely, if the receiver engages in pragmatic reasoning, they might think that if the sender wanted the green square to be returned again, they would have chosen the message p, as before. In the cognitive models this translates in the fact that the probability of the green square being the target object of the message corresponding to the p key will be greater than the probability of the same item being the target of the message corresponding to the q key, given the positive feedback of the preceding trial, and, possibly, given the cost of changing the message. Remember that the sender is supposed to be trying to facilitate the receiver. It would therefore not make sense for the sender to send a different message to have the same item selected by receivers. Reasoning in this way, a pragmatic receiver could select an object different from the green square. For a representation of the objects discussed in this example, see Figure 4.3.

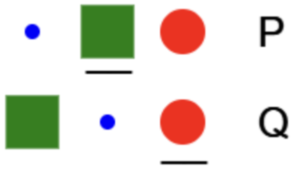


Figure 4.3: Example experimental stimuli with elicitation of a conversational implicature

M-implicatures are thus elicited in such scenarios where there is a repetition of the same three objects as in an earlier trial, but with a different associated message, and where it is not the case that two objects are known to be in the extension of the concept associated to the message that came with the trial in question. Please note that, while in most cases such repetition occurs in consecutive trials, it does not have to be the case. In fact, some such repetitions are not formed by trials located next to each other, but just appearing in the proximity of each other.

Elicitation of Q-implicatures

Q-implicatures are elicited in a similar way as M-implicatures, in the context of this experiment: they are elicited when there is a repetition of the same three objects as in an earlier trial, but with a different associated message, and where two of the three objects are known to be in the extension of the concept associated to the message that came with the trial in question.
For example, in Figure 4.3, it could be the case that, through feedback of previous trials, the receiver knows that both the red circle and the square are in the extension of the concept associated with the message q. In this case, the message q likely indicates the square and the red circle, while the message p likely indicates the square, but no other object among those displayed in the trial, as we suppose that the circles in figure were never revealed to be in the extension of the concept associated with p. This means that, if the target object were the square and sender were to use the most informative message, they would send p. Since the sender sent q instead, a pragmatic reasoner would infer that the target object is the red circle.

4.1.8 Types of Trials

The previous section describes trials created to elicit conversational implicatures, which I will from now on call "pragmatic trials". In the current section, I will first discuss pragmatic trials in greater detail. Subsequently, I will dive into the structure and usefulness of control trials.

Pragmatic Trials

Pragmatic trials always involve the repetition of a set of three objects, but with a different message. There can also be, on top of trials involving exact repetitions of the displayed items, trials that only repeat certain items. The aim of this approach is to lead participants to more easily notice the presence of repetitions.
Regardless, pragmatic trials are presented in two variations, M-implicatures and Q-implicatures, as discussed in the previous section.

Among all blocks, there is a total of 12 trials that utilize the elicitation of M-implicatures to review the belief of a participant about the meaning of the target concept.

The set of trials involving Q-implicatures consists of 4 trials. Differently from what happens in the trials involving M-implicatures, trials that encourage the elicitation of Q-implicatures are not meant to induce participants to review their belief regarding the meaning of the target concept. Instead, these trials only test whether participants employ pragmatic reasoning, without it being bonded with concept learning.

Control Trials

In this section, I will begin by discussing control trials in the sender block. Following that, I will delve into control trials in the receiver blocks, with particular attention to control trials involving the repetition of all figures.

Given the straightforwardness of the sender task, in which there is only one possible answer for each set of three objects, the trials in that task can be used as controls.

Controls in the receiver blocks take the form of trials in which an item that was already observed to be in the extension of the concept associated with a message returns with the same message. This is exemplified in Figure 4.4.

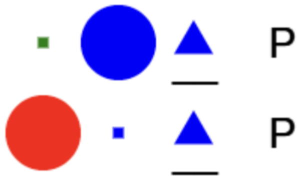


Figure 4.4: Example of a control trial

Trials in which the set of three displayed objects as well as the message are repeated were also inserted in many cases. These control trials have the benefit of discouraging participants from consistently choosing a different object each time in which they see a repetition in the sets of objects, which is a strategy that participants could learn by being exposed to pragmatic trials. Repeated trials are also useful to make participants understand that the sender is rational. Furthermore, if such trials precede trials in which the set of three objects is repeated but the message is not (pragmatic trials), participants will see the repeated set of three objects three times. In this way, it will be easier for them to realize that a repetition occurred.

4.2 Cognitive Models Used in this Study

In this section, I first give an overview on my general approach to modeling concept learning and pragmatics (Section 4.2.1). Subsequently, Section 4.2.2 outlines the models used in this work. In Section 4.2.3, I explain how concept learning is modeled, including the learning of both concepts at the same time. Finally, in Section 4.2.4 I clarify the way in which the LUM is adapted to allow modeling responses to trials.

Please note that all cognitive models used in this work operate at Marr's computational level, as they formalize the problems that participants solve (Marr, 1982).

4.2.1 Modeling Approach

This section briefly describes my general approach to modeling concept learning and pragmatic reasoning, in this order. A more detailed discussion on the topics presented here will follow in the remainder of the chapter.

Concept Learning

The strategy that I followed to model concept learning is based on LOTs and, specifically, on the approach utilized by Piantadosi et al. in the study reported in (2016).

In contrast to the study conducted by Piantadosi et al., my experiment was completed by a smaller number of participants. Therefore, I do not have enough data to fit parameters based on experimental data.

A possible approach would have thus been to use a LOT from the abovementioned study while keeping the probabilities that were found in that study for each logical operation. However, I opted against this approach, as the probabilities of the logical operations found by Piantadosi et al. might be heavily impacted by the specific experimental setup.

Therefore, I assume the probabilities of each logical operation to be equal. The operations are thus equally salient: there is no preference for using an operation instead of another, based on the type of operation only. For this reason, I do not utilize *FullBoolean*, which has low probability operations (see Section 3.1.1). In fact, even though *FullBoolean* is the best performing LOT when no quantifier is needed, equalizing the probabilities of operations could have a great impact, given the presence of low probability operations. This is why I used as LOT *SimpleBoolean*, which contains all the operations in *FullBoolean* apart for the low probability ones and and performed similarly in Piantadosi et al.'s study (Piantadosi et al., 2016).

The features of items are also assumed to be equally salient: picking an item on the basis of its color, size or shape is considered equally likely, unless the prior beliefs of the agent, derived from the results of previous trials, suggest otherwise. Moreover, all subfeatures within a feature, such as for example different colors, are assumed to be equally salient.

Pragmatics

The pragmatic reasoning part of my cognitive models is responsible for the update of the beliefs relative to both concepts at the same time and for the choice of which object is selected at each trial. For the latter task, I considered two possibilities: the LUM and the IQR model. On the one hand, the LUM has the disadvantage of poorly accounting for M-implicatures at lower recursive levels (which are more likely reached by humans), as discussed in Section 3.2.3. The IQR model, on the other hand, can account for M-implicatures only under strong assumptions on the agents' behavior (see Section 3.2.1).

Out of these two models, I decided to implement the LUM. The reason behind this decision is that, in my experiment, M-implicatures are likely elicited only for listeners who reason about a speaker that is at least S_2, who, in turn, reasons about the listener not being literal but L_1 (see Section 3.2.2). The LUM is expected to model L_2 as more likely than not to engage in M-implicatures (Bergen et al., 2016). Moreover, given the popularity of the RSA framework, implementing the LUM provides an opportunity to test its efficacy with the M-implicatures used in my experiment, where the probabilities of lexica are given by the probabilities of hypotheses in the LOT, as described later in Section 4.2.3.

Regarding the recursion level in recursive social reasoning, I will limit it to L_2 listeners. The reason for this is that humans typically do not engage in deeper recursive reasoning (see Section 3.2.2), as the findings of Franke and Degen suggest (2016).

4.2.2 All Models' Variations

In this section I will give an overview on the three models used in this study: the Baseline Model, which does not incorporate pragmatic reasoning, the *Double Update*

Model, which extends the Baseline Model by updating both concepts at the same time, and the *L_2 Model*, which extends the Baseline Model by accounting for a pragmatic reasoning at the L_2 level, when responding to pragmatic trials.

Baseline Model

The Baseline Model does not take pragmatic reasoning into account and, after each trial, only the beliefs regarding the hypotheses for the concept relative to that trial are updated. This model serves as a reference point to evaluate the effectiveness of other models and determine if they provide significant improvements. The Baseline Model should perform best in the task for the control group. In fact, participants in the control group are not supposed to engage in pragmatic reasoning.

Double Update Model

The Double Update Model builds on top of the Baseline Model, inheriting all features present in the latter. The only difference lies in the update process of the agent's beliefs following the observation of feedback indicating the correct object choice for a trial.

In the Double Update Model, both hypotheses regarding the concept related to the current trial and the hypotheses regarding the alternative concept are updated. However, the update magnitude for the alternative hypotheses is smaller compared to that of the hypotheses for the concept at hand.

L_2 Model

The L_2 Model extends the Baseline Model, inheriting all its features. The distinction is that, in this model, pragmatic trials are processed by an L_2 listener instead of an L_1 listener.

Please note that not non-pragmatic trials are still not processed by an L_2 listener. There is in fact no compelling reason to assume that an individual who can engage in pragmatic reasoning at the recursion level of L_2 would always do so: it could well be the case that those capable of reasoning as L_2 listeners would reserve this deeper recursive reasoning for situations where they believe that it is necessary. In the context of this study, this would mean that only when the same objects from a previous trial are presented again with a different message (as discussed in Section 4.1.7), a listener would be prompted to engage in L_2 reasoning.

4.2.3 Concept Learning and Incorporation of Feedback

This section starts with the modeling of concept learning, also addressing the simultaneous learning of both concepts.

The discussion will start with the definition of hypotheses for the meaning of concepts and the creation of the space of hypotheses. Subsequently, the focus shifts to lexica, which integrate a hypothesis for each concept and are essential for determining which object to select in each trial, i.e., how to respond to trials. Finally, the update of beliefs on the meaning of the hypotheses following feedback is addressed.

Creating the Hypotheses

The initial step in modeling concept learning consists in determining the belief of the agent for each hypothesis, i.e., for each possible meaning of each concept. Hypotheses are formulated as propositional logic expressions, making use of the logical negation, conjunction, and disjunction. The literals utilized in such expressions are the colors, sizes, and shapes that define the experimental items. There are therefore nine literals, as for each feature (size, color, shape) there are three subfeatures. The color can in fact be either blue, green or red, the size can be either small, medium or big, and the shape can be either triangle, square or circle.

There exists an infinite number of hypotheses generated from these literals and operations. However, when creating the space of hypotheses, I confined the hypotheses to those involving one or two literals, since no concept in the experiment utilizes more than two literals. Participants could still express concepts using more than two literals. This would for example be the case if they represented a negation as a disjunction (e.g., "not blue" would become "green or red"). In practice, however, no more than two literals are necessary.

I also excluded double negations, as participants, as shown by Piantadosi et al., do not seem to employ them (2016). Furthermore, I omitted concepts signifying *none* or *all* (e.g., "red and not red" or "red or not red"), as well as those where the logical conjunction binds two subfeatures of the same feature (e.g., "triangle and square"), and those where it binds two subfeatures of the same concept, even when one or both are negated (e.g., "triangle and not square"), as these concepts are redundant (e.g., "triangle and not square" can be simplified to just "triangle").

After its creation, each hypothesis is associated with a probability value. To achieve this, I first gave a cost to logical operations and literals. This allowed to assign a cost for each hypothesis. From the costs, I derived the probabilities of hypotheses: higher costs were associated with lower probabilities. These probabilities were then normalized to complete the process.

Creating Lexica

Lexica are constructs necessary for the LUM (see Section 3.2.3). However, I also use them in the Baseline Model, to help estabilish which object to pick at each trial. For more details as to how that is done and why, see Section 4.2.4.

Lexica are created by using the hypotheses defined in the previous section. In fact, each lexicon incorporates a hypothesis for each concept. Therefore, lexica comprehend a hypothesis for the concept associated with the message p and one for the concept associated with q (see Section 4.1.6 for a discussion on the messages utilized). For example, according to a lexicon L_i, the concept associated with p could be $\neg circle$ and the concept associated with q could be $triangle \wedge blue$.

The probability of a lexicon is the product of the probabilities of the two hypotheses corresponding to p and q that are in that lexicon.

Updating Beliefs after Feedback

After each completed trial, feedback is provided (see Section 4.1.5). This feedback allows the listeners to update their beliefs on the meaning of the concepts that they are learning. In the models, the feedback thus prompts an update in the probabilities of hypotheses related to the message presented in the trial. Specifically, the probabilities of these hypotheses increase when they express a concept that includes the feedback object within its extension and the probabilities of other hypotheses related to the same message decrease instead.

The Double Update Model allows for the simultaneous update of hypotheses regarding the meanings of both concepts whenever feedback is provided, irrespective of the message conveyed. In this model, both the beliefs held by the agent about the meaning of the concept whose corresponding message was shown (i.e., the target concept, which I will call t) and the other concept (o) are thus updated after each trial. The beliefs about concept t will see a bigger update than those about o, for which a smaller and inverse update will occur.

For example, if the listener receives the information that *triangle* is in the extension of concept a, the probability of the hypotheses in which a means *triangle* will increase. Similarly, the probability of related hypotheses, like the ones in which

a means *triangle or blue*, will increase. Conversely, the probability of hypotheses where *triangle* or *triangle or blue* are in the extension of concept t will decrease.

In a standard Bayesian update, the probabilities of hypotheses relative to the target concept that do not describe the feedback object would be multiplied by 0. However, I opted against this approach, since singular observations should not lead to the exclusion of hypotheses. The model should instead accommodate instances where feedback is forgotten. Moreover, it should allow for participants to entertain the possibility that the sender made mistakes or acted irrationally on certain occasions.

I therefore decided that, after feedback is presented, the probabilities of hypotheses for the target concept that describes the feedback object are multiplied by 0.9 (I will from now on call this factor "likelihood modifier"), whereas the other hypotheses for the same concept are multiplied by 0.1[3]. If the model in question is the Double Update Model, the probabilities of the hypotheses relative to the other concept are also updated. If such hypotheses do not describe the feedback object, they could be multiplied, for example, by a factor λ that could take values around 0.55. Consequently, if they describe the feedback object, they would be multiplied by $1 - \lambda$. Probabilities are of course normalized afterward.

From now on, I will call λ the multiplicative factor introduced in the Double Update Model.

After updating the hypotheses' probabilities, the probability of each lexicon is calculated again as the product of the probabilities of the two hypotheses corresponding to p and q that are in that lexicon.

4.2.4 Adapting the LUM for Trial Response Determination

In this section, I will explain how I adapted the LUM to the specific case of this study.

Firstly, I illustrate how the interpretation and production rules for L_0, L_1 and S_1 are adapted to the setting of the current study and indicate how the responses to trials are modeled in the Baseline Model. Subsequently, I give an overview as to why including the costs for L_1 and S_1 is not relevant in this context. Lastly, I will

[3] An alternative to this approach is the one discussed by Anderson in "Tell Me Everything You Know: A Conversation Update System for the Rational Speech Acts Framework" (2021). Such an approach is however designed to deal with more complex situations and seems overly complex for the simple scenario of my experiment.

show the interpretation and production rules for L_2 and S_2, which are central to the L_2 *Model.*

Simplifying the Production and Interpretation Rules

In the RSA framework, both the probability of the speaker making an observation and the probability of a particular object being the intended referent for the listener are taken into account. In our scenario, this involves considering the probability of an object being displayed and the probability of the speaker intending for the listener to select that object. These probabilities are distinct from each other. Typically, RSA models assume that all objects have equal probability of being observed by the speaker, unless they deal with M-implicatures where one message is assumed to be more likely to occur than another. In our case, there is no reason to assume that some objects are observed more likely than others. Likewise, there is no reason to assume that the two messages (p and q) are not equiprobable. Neither there is any reason to assign them a different cost. This leads to the possibility of simplifying the production and interpretation rules of the LUM. To better understand the simplifications, let us first look again at the formulas from Section 3.2.3:

$$\begin{gathered}
P_{L_0}(s \mid u, \mathcal{L}) \propto P(s) \cdot \mathcal{L}(u, s), \\
P_{S_1}(u \mid s, \mathcal{L}) \propto exp(\alpha \cdot (log P_{L_0}(s \mid u, \mathcal{L}) - C(u))), \\
P_{L_1}(s \mid u) \propto P(s) \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot P_{S_1}(u \mid s, \mathcal{L}).
\end{gathered}$$

As discussed above, the probability of observing each object can be considered the same in our scenario. Therefore, $P(s)$ is a constant. Thus, it can be accounted for through the coefficient of proportionality. The expressions for P_{L_0} and P_{L_1} can therefore be simplified. Furthermore, the utterance cost, $C(u)$, is the same for the two utterances. Therefore, its value can be set to the same value for both of them. To simplify the expressions, that value can be picked as 0. The resulting production and interpretation rules are:

$$\begin{gathered}
P_{L_0}(s \mid u, \mathcal{L}) \propto \mathcal{L}(u, s), \\
P_{S_1}(u \mid s, \mathcal{L}) \propto exp(\alpha \cdot (log P_{L_0}(s \mid u, \mathcal{L})), \\
P_{L_1}(s \mid u) \propto \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot P_{S_1}(u \mid s, \mathcal{L}).
\end{gathered}$$

Now, notice that $exp(\alpha \cdot (log P_{L_0}(s \mid u, \mathcal{L})) = e^{log P_{L_0}(s|u,\mathcal{L})^\alpha} = P_{L_0}(s \mid u, \mathcal{L})^\alpha$. Given $P_{L_0}(s \mid u, \mathcal{L}) \propto \mathcal{L}(u, s)$ and the transitive nature of proportionality, it holds that $P_{S_1}(u \mid s, \mathcal{L}) \propto \mathcal{L}(u, s)^\alpha$. Similarly, we obtain $P_{L_1}(s \mid u) \propto \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot \mathcal{L}(u, s)^\alpha$. The resulting expressions are:

$$\begin{aligned} P_{L_0}(s \mid u, \mathcal{L}) &\propto \mathcal{L}(u, s), \\ P_{S_1}(u \mid s, \mathcal{L}) &\propto \mathcal{L}(u, s)^\alpha, \\ P_{L_1}(s \mid u) &\propto \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot \mathcal{L}(u, s)^\alpha. \end{aligned}$$

The coefficient of rationality α is a non-negative number. A greater α value corresponds to a more rational speaker. If α is 0, utterances are chosen randomly (see Section 3.2.2). The value of α is in fact usually set higher than 1. For example, when deriving M-implicatures within the LUM, Bergen et al. set the value of α to 4 (2016). I will also not let α take values lower than 1.

In my work, $\mathcal{L}(u, s)$ takes as value either 1 or 10^{-9} (see Section 3.2.3). If $\mathcal{L}(u, s)$ is 1, $\mathcal{L}(u, s)^\alpha = \mathcal{L}(u, s)$, given that $\alpha \geq 1$. If $\mathcal{L}(u, s)$ is 10^{-9}, $\mathcal{L}(u, s)^\alpha < \mathcal{L}(u, s)$. However, notice that I chose arbitrarily 10^{-9}, as a value lower than 10^{-6} and approaching 0. Both 10^{-9} and $(10^{-9})^\alpha$ are values approaching 0 that I could use in my models, and they are small enough in order for their difference not to be that relevant in the results of calculations, permitting their interchangeable use. If this were not the case, $\mathcal{L}(u, s)$ could be readily adjusted to a smaller value.

All of these considerations lead to the conclusion that α does not influence the production and interpretation rules derived in this section. Consequently, these rules can be further simplified as:

$$\begin{aligned} P_{L_0}(s \mid u, \mathcal{L}) &\propto \mathcal{L}(u, s), \\ P_{S_1}(u \mid s, \mathcal{L}) &\propto \mathcal{L}(u, s), \\ P_{L_1}(s \mid u) &\propto \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot \mathcal{L}(u, s). \end{aligned}$$

This leads to the conclusion that L_1 picks objects only based on the probabilities of the lexica supporting that choice. Given that the lexica are created using LOT-based hypotheses as explained in the previous section, a model involving only L_1 that uses the production and interpretation rules described here only uses the LOT-based hypotheses to choose which objects to pick, without other parameters being involved in the process. Therefore, the expressions derived here for L_1 are used for the implementation of the Baseline Model that only accounts for concept learning, and not for pragmatics. Specifically, these formulas are used to calculate the probabilities of selecting each of the objects presented in each trial, thereby determining the response to the trials.

Reintroducing Costs

The cost does not necessarily need to be based on a property of the messages. Instead, it can represent the cost of changing a message. This reflects the intuition that a sender is more likely to send again the message that they previously sent, unless there is a compelling reason to change.
Including the cost in the previous expressions results in the following:

$$\begin{gathered}
P_{L_0}(s \mid u, \mathcal{L}) \propto \mathcal{L}(u, s), \\
P_{S_1}(u \mid s, \mathcal{L}) \propto exp(\alpha \cdot (logP_{L_0}(s \mid u, \mathcal{L}) - C(u))), \\
P_{L_1}(s \mid u) \propto \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot P_{S_1}(u \mid s, \mathcal{L}).
\end{gathered}$$

The production rule $P_{S_1}(u \mid s, \mathcal{L}) \propto exp(\alpha \cdot logP_{L_0}(s \mid u, \mathcal{L}) - \alpha \cdot C(u))$ can be rewritten as $P_{S_1}(u \mid s, \mathcal{L}) \propto \frac{exp(\alpha \cdot logP_{L_0}(s|u,\mathcal{L}))}{exp(\alpha \cdot C(u))}$, which can be simplified as $P_{S_1}(u \mid s, \mathcal{L}) \propto \frac{P_{L_0}(s|u,\mathcal{L})^\alpha}{exp(\alpha \cdot C(u))}$. Therefore, we obtain:

$$\begin{gathered}
P_{L_0}(s \mid u, \mathcal{L}) \propto \mathcal{L}(u, s), \\
P_{S_1}(u \mid s, \mathcal{L}) \propto \frac{\mathcal{L}(u, s)^\alpha}{exp(\alpha \cdot C(u))}, \\
P_{L_1}(s \mid u) \propto \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot \frac{\mathcal{L}(u, s)^\alpha}{exp(\alpha \cdot C(u))}.
\end{gathered}$$

Costs cannot play a role yet, however. This is apparent when examining the calculation of the coefficient of proportionality for $P_{L_1}(s \mid u)$. Such calculations are shown in the following Python code snippet. Here, the message in question is indicated as *message*, the experimental stimulus considered is indicated by *item*, and the variable *nl* is used to store the value of the unnormalized $P_{L_1}(s \mid u)$.

```
for item in all_items:
    nl = 0
    # Iterate over all lexica
    for i in range(len(lexica[message])):
        # If the hypothesis for the current message is true for the current
            item, then L(u,s) is 1.
        if lexica[message][i] in true_hypotheses:
            nl += lexica['probabilities'][i] / (np.exp(cost) ** alpha)
        # If the hypothesis for the current message is false for the current
            item, then L(u,s) is close to 0.
        else:
            nl += 10 ** (-9) / (np.exp(cost) ** alpha)
    nl_list.append(nl)
# k is the coefficient of proportionality.
# k * nl_list[0] + k * nl_list[1] + ... = 1. Therefore:
k = 1 / sum(nl_list)
```

If the costs did not matter, the calculations of *nl* would become:

```
    if lexica[message][i] in true_hypotheses:
        nl += lexica['probabilities'][i]
    else:
        nl += 10 ** (-9)
```

Thus, if costs did not matter, the coefficient of proportionality k in the second code snippet would differ from the one in the first code snippet by a constant factor: to obtain k for the second code snippet, we should divide the coefficient of proportionality from the first code snippet by $(exp(cost))^{\alpha}$.

In cases where costs matter, the non-normalized probability *nl* remains the same as if costs did not matter, except that it is divided by $(exp(cost))^{\alpha}$. Since $P_{L_1}(s \mid u)$ is calculated by multiplying the respective *nl* by the proportionality constant k, it follows that in scenarios where costs are included, *nl*, which incorporates the divisor $(exp(cost))^{\alpha}$, is multiplied by k, which incorporates the multiplier $(exp(cost))^{\alpha}$, leading to the same results that would be obtained if costs were not included. Thus, altering the cost has no effect on $P_{L_1}(s \mid u)$.

The code snippets used to illustrate this are part of the cognitive model's implementation. In the model, I have allowed for costs different from zero being used. In this way, it can be verified that, in the context of this experiment, the cost does not play a role when the receiver is L_1 (and even when it is L_2, as discussed next). Please note that this conclusion holds under the assumption that all observations are equiprobable, i.e., that $P(s)$ is constant.

Allowing for L_2 listeners

As discussed earlier in this section, I will allow for L_2 listeners in the L$_2$ Model. Thus, I will explain how the interpretation and production rules for L_2 and S_2 are derived. Let us start from the general expressions for S_n and L_n, provided in Section

3.2.3:

$$P_{S_n}(u \mid s) \propto exp(\alpha \cdot (logP_{L_n-1}(s \mid u) - C(u))),$$
$$P_{L_n}(s \mid u) \propto P(s)P_{S_n}(u \mid s).$$

These expressions serve as the foundation for deriving the rules for L_2 and S_2. In fact, by substituting n with 2 in these expressions and by removing $P(s)$, which is going to become part of the normalization coefficient, as explained earlier in this section, we obtain:

$$P_{S_2}(u \mid s) \propto exp(\alpha \cdot (logP_{L_1}(s \mid u) - C(u))),$$
$$P_{L_2}(s \mid u) \propto P_{S_2}(u \mid s).$$

Therefore, $P_{L_2}(s \mid u) \propto exp(\alpha \cdot (logP_{L_1}(s \mid u) - C(u)))$. Taking into account the results obtained earlier in this section, the interpretation rule for L_2 becomes $P_{L_2}(s \mid u) \propto exp(\alpha \cdot (log(\sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot \mathcal{L}(u, s))) - C(u)))$, which can be also written as follows:

$$P_{L_2}(s \mid u) \propto \frac{\alpha \cdot \sum_{\mathcal{L} \in \Lambda} P(\mathcal{L}) \cdot \mathcal{L}(u, s)}{exp(C(u))}$$

At this point, it is evident that, as before, the cost can be accounted for in the coefficient of proportionality. Therefore, as the results of the cognitive models confirm, the cost value does not matter even for the listener L_2. The only parameter that plays a role in the latter expression is in fact α.

5 Results

In this chapter, I will present and analyze the results of the experiment, in Section 5.1, and the cognitive models, in Section 5.2. Subsequently, I will discuss these results in Section 5.3.

5.1 Experimental Results

This section first establishes the criteria for data exclusion (see Section 5.1.1). Subsequently, in Section 5.1.2, it addresses the results of the models used to analyze response accuracy. Finally, in Section 5.1.3, it discusses the outcomes of concept learning for participants.

Please, note that throughout the remainder of this section, the responses to the trials corresponding to the first time a message was shown are not considered. In fact, responses to such trials' tend to be random, as participants start the receiver tasks with no prior knowledge on the meaning of the concepts.

5.1.1 Data Exclusion

No participants were excluded for color blindness. Thus, data exclusion was solely based on the accuracy of responses in the sender task and in the control trials from the receiver tasks.

In the following, I will start by briefly discussing the results of the sender task. Afterward, I will focus on the control trials from the receiver task, first showing that they were indeed perceived as easier than the trials that are neither control trials nor pragmatic trials, and then deciding on the utilization of control trials for the exclusion of participants who exhibited lower performance on them.

Sender Task

As mentioned in Section 4.1.8, the trials in the sender task could be used as control trials, due to the task's straightforwardness. Therefore, let us first examine the accuracies of responses in this task.

The average accuracy for the sender task is around 95%, with the lowest accuracy achieved by any participant being approximately 86%. Consequently, no participant's results were excluded based on their performance in this task.

Demonstrating the Simplicity of Control Trials

Before excluding participants based on their performance on control trials, it is important to ensure that such trials were perceived as easier by participants. For this reason, I ran a generalized linear mixed-effects model (GLMM) that uses logistic regression to predict binary outcomes to examin the difference in accuracy between the control trials and the other non-pragmatic trials.

As the dependent variable, I used the correctness of responses. The predictor, referred to as *Trial Type*, indicates whether the trials are control trials or trials that are neither control trials nor pragmatic trials. The fixed effect is thus *Trial Type*, whereas the random effects are *run id* (the participant ID), which allows to take account of individual differences across participants, *block number*, accounting for differences between blocks (for example, where trials in one block may be more challenging than in another) and *task*, which accounts for potential differences in task difficulty between the experimental group and the control group.
The reference level is the condition where *Trial Type* comprehends only control trials.
The R code for the GLMM is the following:

```
normal_vs_controls <- glmer(
    correct ~ Trial_type + (1 | run_id) + (1 | block_number) + (1 | task),
    data = mm5noexclusion, family = binomial(link = "logit"))
```

The intercept has an estimate of 1.7628. This represents the baseline log-odds of a correct response when *Trial Type* is at its reference level. Converting the log-odds to probabilities, we obtain that the probability of a correct response in the control trials is approximately 0.85.

The coefficient of *Trial Type* is -0.6409 with a p-value of 2.17e-05, indicating a statistically significant negative effect: the non-pragmatic, non-control trials have lower odds of a correct response compared to the control trials. Specifically, the probability to respond correctly in the non-pragmatic, non-control trials is approximately 0.75.

Regarding the random effects, for the *run id*, the variance is 0.186847, for the *block number* it is 0.004669 and for the *task* it is 0.095100. This indicates that there is low variability among blocks, and moderate variability among participants and among groups, intended as experimental group and control group.

Data Exclusion Based on Control Trials

It has been shown that control trials are indeed easier to respond to correctly compared to trials that are neither control trials nor pragmatic trials. It thus makes sense to exclude participants based on their accuracy on control trials.
Let us thus look at table 5.1, which displays the accuracies of participants in such trials. Participants with average accuracies below 0.70 seem to be outliers. Therefore, From the control group, the data for the participant with Run ID 71 was excluded, while from the experimental group, the data for participants with Run IDs 57 and 70 were excluded. This results in 7 participants remaining in the control group and 9 participants in the experimental group.

5.1.2 Models for Analyzing Response Accuracy

In this section, I will employ generalized linear mixed-effects models (GLMMs) to investigate differences in accuracy across the following comparisons: between the control group and the experimental group, between pragmatic trials and corresponding trials in the control group, between pragmatic trials and other non-control trials in the experimental group, and between M-implicature trials and Q-implicature trials.
All models used in this section are GLMMs that employ logistic regression to predict binary outcomes (accuracy). Therefore, I will not specify the model type for each specific instance throughout the rest of this section.
Please note that in the rest of this section, when I report the average accuracy values, I will round them to two decimal places.

Control Group and Experimental Group

The average accuracy for the control group is 0.78, whereas the average accuracy for the experimental group is 0.69.
To examine the difference in accuracy between these two groups, I utilized the

Table 5.1: Accuracy in control trials by group

Group	Average Accuracy
Control Group	0.89
Experimental Group	0.80

Run ID	Accuracy (Control Group)
14	0.92
46	0.85
50	0.88
63	0.96
67	0.88
71	0.65
72	0.96
75	1.00

Run ID	Accuracy (Experimental Group)
8	0.81
19	0.81
32	0.81
39	0.85
49	0.73
57	0.69
66	0.81
68	0.88
70	0.65
78	0.92
81	0.81

correctness of responses as dependent variable for the GLMM, with the group type (control or experimental) as the predictor. From now on, I will call this predictor *Group*

The fixed effect is *Group*, whereas the random effects are *run id* (the participant ID), which allows to take account of individual differences across participants and *block number*, accounting for differences between blocks (for example, where trials in one block may be more challenging than in another). The reference level is the control group.

The R code for the GLMM is the following:

```
exp_vs_control <- glmer(
        correct ~ Group + (1 | run_id) + (1 | block_number),
        data = mm1, family = binomial(link = "logit"))
```

The intercept has an estimate of 1.2521, representing the baseline log-odds of a correct response in the control group. Converting the log-odds to probabilities, we can determine that the probability of a correct response in the control group is approximately 0.78.

The coefficient of the *Group* variable is -0.4335 with a p-value of 0.00277, indicating a statistically significant negative effect: the experimental group has lower odds of a correct response compared to the control group. Specifically, the probability to respond correctly in the experimental group is 0.69.

Regarding the random effects, for *run id* the variance is 0.01602, indicating some variability among participants. Similarly, the variance for *block number* is 0.04219, indicating moderate variability across block numbers.

In conclusion, *Group* has a significant impact on the correctness. Specifically, the experimental group has lower odds of a correct response compared to the control group.

Pragmatic Trials and Corresponding Trials in the Control Group

The average accuracy for pragmatic trials in the experimental group is 0.43, whereas the average accuracy for corresponding trials in the control group is 0.41.

To examine the difference in accuracy between these types of trials, I utilized the correctness of responses as dependent variable for the GLMM and the trial type as the predictor. From now on, I will call this predictor *Trial Type*.

The fixed effect is *Trial Type* and the random effects are *run id* and *block number*.

The reference level is set to the trials in the control group that correspond to the pragmatic trials in the experimental group.

The R code for the GLMM is the following:

```
    pragmatics_vs_would_be_pragmatics <- glmer(
        correct ~ Trial_type + (1 | run_id) + (1 | block_number),
        data = mm2, family = binomial(link = "logit"))
```

The intercept has an estimate of -0.37482, representing the baseline log-odds of a correct response in the control group. Converting the log-odds to probabilities, we can determine that the probability of a correct response in the control group is approximately 0.41.

The coefficient of the *Trial Type* variable is 0.08421. This indicates that the trials corresponding to pragmatic trials in the control group have lower odds of a correct response compared pragmatic trials from the experimental group. Specifically, the probability to respond correctly in the experimental group is approximately 0.43. However, the high p-value for the coefficient of *Trial Type* (0.746) suggests that this result is not statistically significant.

Regarding the random effects, for *run id* the variance is 1.01e-08 and for *block number* it is 1.46e-01. This indicates very low variability among blocks and, especially, among participants.

In conclusion, there is no proof that *Trial Type* impacts the accuracy of responses.

Pragmatic Trials and Other Non-Control Trials in the Experimental Group

The average accuracy for pragmatic trials is 0.43, whereas the average accuracy for corresponding the other non-control trials in the experimental group is 0.71.
To examine the difference in accuracy between these types of trials, I utilized the correctness of responses as dependent variable for the GLMM and the trial type as the predictor. From now on, I will call this predictor *Trial Type*.
The fixed effect is *Trial Type* and the random effects are *run id* and *block number*.
The reference level is set to the pragmatic trials.
The R code for the GLMM is the following:

```
pragmatics_vs_normal_experimental <- glmer(
    correct ~ Trial_type + (1 | run_id) + (1 | block_number),
    data = mm3, family = binomial(link = "logit"))
```

The intercept has an estimate of -0.2897, representing the baseline log-odds of a correct response in the pragmatic trials. Converting the log-odds to probabilities, we can determine that the probability of a correct response in pragmatic trials is approximately 0.43.

The coefficient of the *Trial Type* variable is 1.1876, with a p-value of 1.77e-08, indicating a statistically significant effect: pragmatic trials have lower odds of a correct response compared to the the other non-control trials in the experimental group. Specifically, the probability to respond to non-pragmatic, non-control trials in the experimental group is approximately 0.71.

Regarding the random effects, for *run id* the variance is 0.06500 and for *block number* it is 0.09925. This indicates moderate variability among blocks and participants.

In conclusion, *Trial Type* has a significant impact on the correctness. Specifically, the pragmatic trials have lower odds of a correct response compared to the other non-control trials in the experimental group.

M-implicatures and Q-implicatures

The average accuracy for pragmatic trials corresponding to M-implicatures is 0.38, whereas the average accuracy for pragmatic trials corresponding to Q-implicatures is 0.58.
To examine the difference in accuracy between these types of trials, I utilized the correctness of responses as dependent variable for the GLMM and the trial type (*Trial Type*) as the predictor.
The fixed effect is *Trial Type* and the random effect is *block number*. The reference level is set to the Q-implicatures trials.
The R code for the GLMM is the following:

```
m_vs_q <- glmer(
     correct ~ Trial_type + (1 | block_number),
     data = mm4, family = binomial(link = "logit"))
```

The intercept has an estimate of 0.09357, representing the baseline log-odds of a correct response in the Q-implicatures trials. Converting the log-odds to probabilities, we can determine that the probability of a correct response in Q-implicatures trials is approximately 0.52.

The coefficient of *Trial Type* is -0.51983. This indicates that the M-implicatures trials have lower odds of a correct response compared Q-implicatures trials. Specifically, the probability to respond correctly to M-implicatures trials is approximately 0.40. However, the high p-value for the coefficient of *Trial Type* (0.237) suggests that this result is not statistically significant.

The random effect *block number* has variance 0.2722. This indicates moderate variability among blocks.

In conclusion, there is no proof that *Trial Type* impacts the accuracy of responses.

5.1.3 Concept Learning Outcomes

This section first examines the accuracy in concept learning to show that concepts were effectively learnt. Following this, it discusses the participant-reported learning outcomes.

Accuracy in Concept Learning

Table 5.2 presents the accuracies in the receiver task, broken down by control and experimental groups. The table also includes overall accuracies for each group. Please, note that these results were derived by analyzing only the trials that are neither control trials nor pragmatic trials, to emphasize concept learning.

Table 5.2: Average and per block accuracies for control and experimental groups, considering only non-pragmatic, non-control trials

Group	Average Accuracy
Control	0.827
Experimental	0.711

Block Number	Accuracy (Control Group)	Accuracy (Experimental Group)
1	0.857	0.789
2	0.743	0.689
3	0.844	0.758
4	0.878	0.556

From these results, it is evident that the accuracies are higher than what would be expected by chance, indicating that participants learnt the concepts. In fact, if responses were random, the expected accuracy would be $\frac{1}{3}$, given that each trial involves selecting one object from a set of three.

Participant-Reported Learnt Concepts

It is often challenging to express one's thought processes and, in this context, to describe what is being learnt. This point is also emphasized by Piantadosi et al. (2016). Therefore, even though participants described what they believed the concepts meant at the end of the listener tasks, it might be more accurate to evaluate their responses to the trials to determine whether they understood the concepts. Furthermore, the focus of this study is not on whether participants learnt the concepts, but rather on the learning process itself and the extent to which they employed pragmatic reasoning during that process.

Nevertheless, asking participants to describe the concepts at the end of the tasks served the purpose of reinforcing the impression that the study was only focused on concept learning. Furthermore, the results still provide valuable insights.

First, in some incorrect responses, participants only provided simplified versions of the correct concepts, suggesting that they nearly grasped the right meanings. Second, it appears that participants tended to avoid using logical negations when describing the concepts. Some employed quantifiers, and in many cases, object priorities were established. For example, they would indicate that the target object is the one with a specific feature, and if that's not present, the object with another specific feature becomes the target. Fuzzy inclusions were also observed, such as describing a concept as preferably having a certain feature. Additionally, in many cases participants used more than two literals to express the meaning of concepts.

Based on my estimations, around $\frac{1}{8}$ of the concepts are incorrectly labeled due to an error in the code of the experiment. This means that, while participants' responses to what they think the concepts mean are stored successfully, the real meaning of the concepts is not always determined successfully. This error results in a slight underestimation of the participants' accuracy in determining the right concepts. It is still unclear why that error occurs. However, please note that the learnt concepts are not used in data analysis and they do not play a role in the cognitive models.

5.2 Cognitive Models' Results

In this section, I will first explore the Baseline Model's results in concept learning, in Section 5.2.1. Subsequently, in Section 5.2.2, I will focus on the learning patterns exhibited by the cognitive models, showing learning curves for all models.

5.2.1 Exploring the Model's Result in Concept Learning

As a first step in analyzing the cognitive models' results, let us examine whether the Baseline Model provides correct responses to the experimental trials and what meaning it assigns to the learnt concepts. This step is needed because, if the Baseline Model does not produce correct responses, there might be flaws in the cognitive models or in the experimental setup.

A simple example illustrating that the learning outcomes in the Baseline Model align with expectations is also shown in Appendix 6.2. Now, let us explore whether the more complex Baseline Model behaves as expected.

Assessing Model Accuracy

Out of the 74 non-pragmatic trials in the experiment, the Baseline Model responds incorrectly to 16 of them (pragmatic trials are expected to be incorrect in this model and are therefore not considered). Table 5.3 focuses on these trials: for each trial with an incorrect response, the table indicates the block from which it originates and the difference between the probability of the model's selected response and the probability of the correct response. The Baseline Model does not provide correct responses in block 1 with probabilities lower than those of selecting other objects. Blocks 2 and 3 each contain 4 trials where the model is more likely to give an incorrect response, whereas block 4 has 8 such trials.

Table 5.3: Trials responded to incorrectly by the Baseline Model, with block of origin and difference between the probability of the model selecting the most probable response and its probability of selecting the correct response.

Block	Difference
2	0.00013
2	0.07616
2	0.14004
2	0.02494
3	0.14360
3	0.00594
3	0.00165
3	0.00040
4	0.00684
4	5.79187e-06
4	1.42495e-05
4	0.11039
4	0.01887
4	0.02930
4	1.09625e-05
4	0.00237

It is worth noting that in many of these trials, the probability of the most likely response provided by the model is often close to the probability of the model providing the correct response. Nevertheless, this is not ideal, as it would be preferable for the probability of the correct response to be significantly higher than the probabilities of other responses.

Concepts Learnt by the Baseline Model

Concepts to be learnt are randomized among features and subfeatures for each run of the experiment and each block, as discussed in Section 4.1.4. When discussing in this section the concepts to be learnt, I will nevertheless use a specific instantiation of them, in order to make the discussion clearer.

The right concepts to be learnt and the concepts learnt by participants are reported in table 5.4. In block 1, which is the block in which the model was most accurate, one concept is learnt correctly, whereas the other, associated with the message *p*, is partially learnt, reflecting a pattern often observed in human participants. In block 2, one concept is correctly learnt, but the other remains unclear. In block 3, both concepts are unclear, while in block 4, one concept remains unclear and the other is learnt incorrectly. The reduced accuracy in block 4 aligns with earlier results discussed in this section.

Table 5.4: Comparison of learnt concepts with correct concepts. Learnt concepts are indicated as "unclear" when more hypotheses for such concepts were equiprobable.

Block	Learnt Concept	Correct Concept
1 (p)	Triangle	Triangle and not small
1 (q)	Small or red	Small or red
2 (p)	Green	Green
2 (q)	Unclear	Not (blue or big)
3 (p)	Unclear	Not small
3 (q)	Unclear	Red and not circle
4 (p)	Not (small and blue)	Circle and red
4 (q)	Unclear	Big or blue

5.2.2 Learning Patterns

This section explores the learning patterns of the cognitive models. It begins with the Baseline Model, followed by the Double Update Model, and concludes with the L_2 Model.

For each model, learning curves are provided. Please, note that the learning curves do not use the same scales. In fact, the probabilities of correct responses derived by the models are generally less extreme than the participants' average probabilities of correct responses to each trial. Moreover, there is variability across the different models concerning the occurrence of trials with more extreme probabilities of correct responses.

Baselilne Model

In the following, I will start by comparing the results of the Baseline Model with those of the control group, then move on to compare them with those of the experimental group.

Initially, a Spearman correlation analysis is conducted to examine the relationship between participants' average probability of responding correctly to each trial and the Baseline Model's corresponding probability of correct responses.
For the control group, the analysis revealed a weak correlation, with a Spearman correlation coefficient of approximately 0.19. The associated p-value, approximately 0.07, suggests that this correlation is not statistically significant.
In contrast, for the experimental group, the Spearman correlation coefficient between the Baseline Model's results and the experimental data was approximately 0.37. The associated p-value, 0.0003, indicates a statistically significant moderate correlation.

A more close look the results through the visual inspection of learning curves from figures 5.1 and 5.2 makes however clear that the Baseline Model's results align with the experimental results only for the first experimental block.
Such visual inspection also indicates that the probabilities of the Baseline Model providing a correct response are not as extreme as the participants' probabilities of doing so. In fact, the Baseline Model provides a wrong response only to 16 non-pragmatic trials, half of which were in the fourth experimental block (see Section 5.2.1). However, the learning curves show that for most trials, the probability of responding correctly is low, indicating that the responses of the model have similar probabilities in most cases.

Double Update Model

I ran the Double Update Model with different values of λ (see Section 4.2.3 for a description of λ). The λ values tested, along with the corresponding cumulative absolute differences between the model predictions and the experimental results are shown in table 5.5.

The only λ value for which the Double Update Model yields a lower cumulative absolute difference compared to the Baseline Model (which corresponds to the Dou-

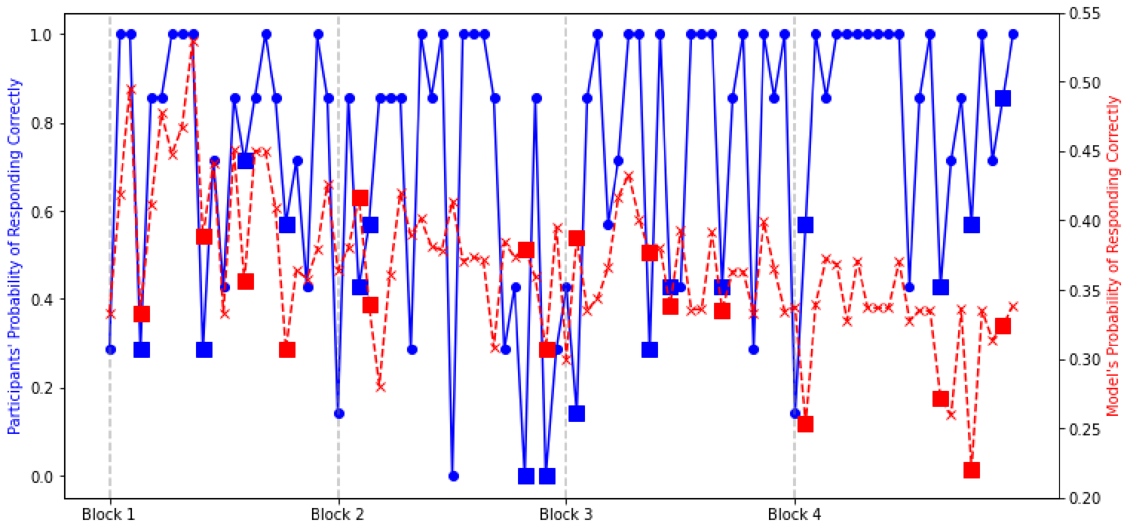


Figure 5.1: Learning curve of the Baseline Model (in red) compared to experimental results from the control group (in blue). Squares indicate pragmatic trials. In each block, trials with the message p precede those with the message q.

ble Update Model with $\lambda = 0.5$) is 0.525. The Spearman correlation coefficient between the Double Update Model's results (with $\lambda = 0.525$) and the experimental data is approximately 0.37, matching the correlation observed with the Baseline Model. The associated p-value of 0.0004 indicates a statistically significant moderate correlation.

In order to provide a visual comparison between the cognitive model's predictions and the experimental results, the learning curve of the Double Update Model with $\lambda = 0.525$ alongside the experimental data is displayed in figure 5.3.

L_2 Model

I tested the Double Update Model with different values of the rationality coefficient α (see Section 3.2.2 for a description of α). The α values tested, along with the corresponding cumulative absolute differences between the model predictions and the experimental results are shown in table 5.6.

Based on the cumulative absolute differences, none of the tested α values lead the L_2 Model to better fit the experimental data than the Baseline Model.

A visual examination of the learning curve of the L_2 Model with $\alpha = 4$ alongside the experimental data shows that, in the L_2 Model, the probabilities to correctly respond to pragmatic trials become more extreme in value with respect to the Baseline Model,

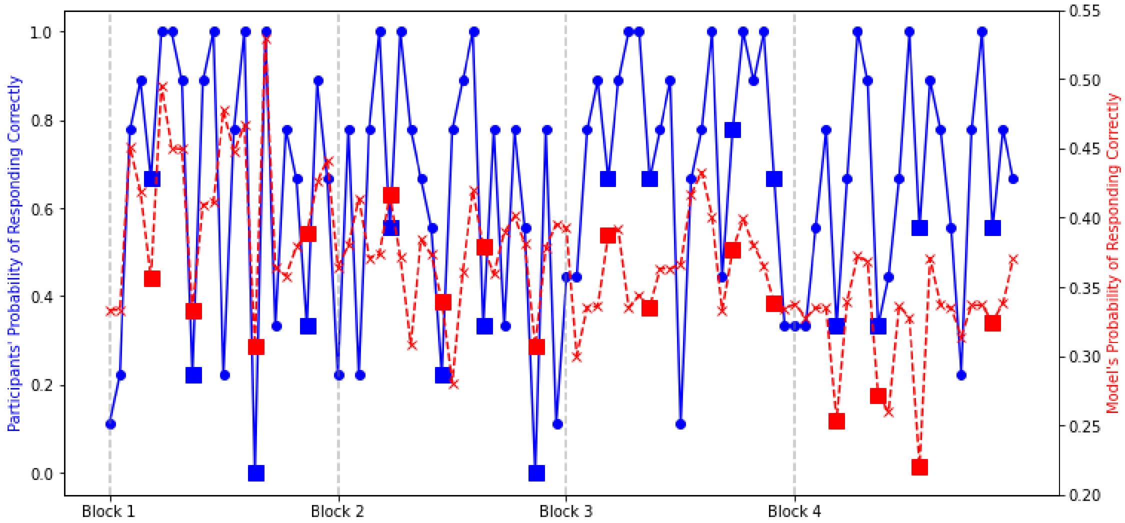


Figure 5.2: Learning curve of the Baseline Model (in red) compared to experimental results (in blue). Squares indicate pragmatic trials.

usually increasing.

The learning curves for the α values tested look similar. I decided to show in figure 5.4 the one with $\alpha = 4$ for the simple reason that that is the same α value used by Bergen et al. in the example with M-implicatures in the LUM (2016).

Table 5.5: Cumulative absolute differences for different λ values

λ	Cumulative absolute differences
0.50	30.4993
0.525	30.2526
0.55	31.9047
0.575	31.9420
0.60	31.7799
0.625	31.8248
0.65	31.6369
0.70	30.8767
0.75	30.8905

Table 5.6: Cumulative absolute differences for different α values

α	Cumulative absolute differences
1	30.9732
2	31.1120
3	31.2380
4	31.3491
5	31.4451

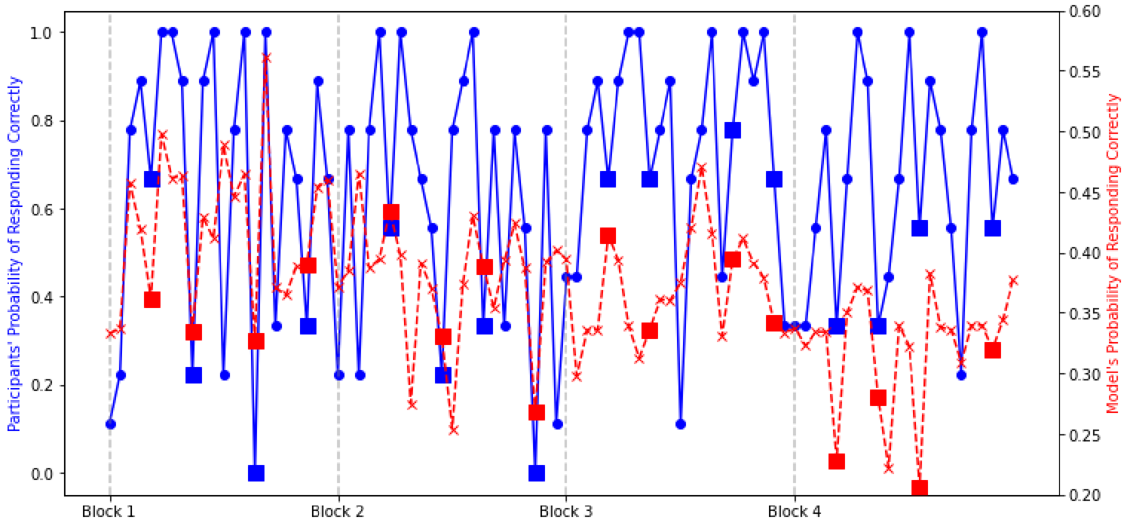


Figure 5.3: Learning curve of the Double Update Model with $\lambda = 0.525$ (in red) compared to experimental results (in blue). Squares indicate pragmatic trials.

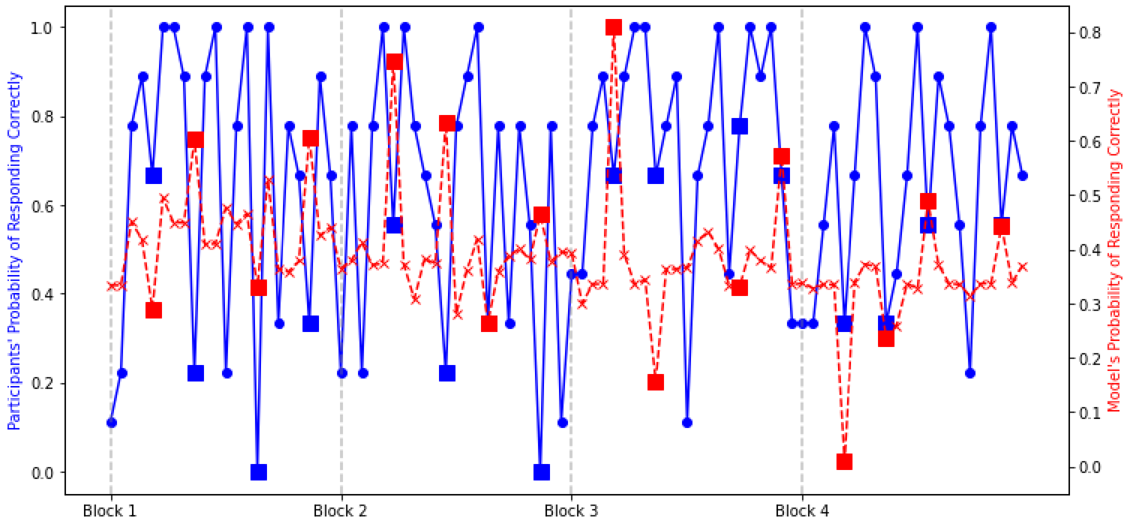


Figure 5.4: Learning curve of the L_2 Model with $\alpha = 4$ (in red) compared to experimental results from the experimental group (in blue). Squares indicate pragmatic trials.

5.3 Discussion

In this section, I will first discuss the results regarding concept learning, in section 5.3.1. Subsequently, in Section 5.3.2 I will focus on the results relative to pragmatic reasoning.

5.3.1 Concept Learning Results

This section begins with a discussion of the learning results achieved by participants, followed by an examination of the learning outcomes of the cognitive models.

Participants' Learning Results

The participants' accuracies in the receiver tasks suggest that learning occurred (see Section 5.4). The concepts learnt by participants sometimes exhibited a bias towards simplicity (simplified versions of the concepts were sometimes reported as the right concepts), a bias that is also implemented in the the cognitive models used in this work. In fact, please notice that the mistake in the learnt concepts for block 1 in the Baseline Model is just a simplification of the correct concept, as shown in Section 5.2.1.

Participants generally avoided using logical negations when explaining the concepts and sometimes they used quantifiers. Thus, it might be beneficial to include quantifiers in the cognitive models and account for varying saliency in operations. Additionally, participants frequently employed more than two literals in their descriptions, suggesting that this may be a useful feature to incorporate into the cognitive models. Regarding participants' establishment of object priorities and the use of fuzzy inclusions (as discussed in Section 5.1.3), these tendencies could be indicative of uncertainty about the concepts' meanings. Thus, this might not be a significant concern. Nevertheless, it will not be a concern at least until more important adjustments are made.

Cognitive Models' Learning Results

In all cognitive models employed in this work, the likelihood modifier was always set to 0.9 (for a description of the likelihood modifier, see how feedback leads to updates in beliefs in the cognitive models implemented as part of this work in Section 4.2.3). This parameter is not the only one which was kept constant and was not learnt,

as the same holds for the costs of literals and logical operations in the LOT that I utilized, *SimpleBoolean* (see Section 4.2.1).
These constraints may have contributed to the cognitive models' limited performance in concept learning. A visual inspection of learning curves suggests that only for the first block learning could have occurred. This could be indicative of the fact that the implementation of the other blocks was less effective. In fact, only in the first block the Baseline Model never produced incorrect responses with lower probabilities than those of selecting other objects, in non-pragmatic trials (see Section 5.2.1).

5.3.2 Pragmatic Reasoning Results

This section discusses the outcomes related to pragmatic reasoning, starting with the experimental results and then shifting to the results obtained from the cognitive models.

Experimental Results

Pragmatic trials in the experimental group exhibited significantly lower odds of a correct response compared to other non-control trials. This outcome aligns with expectations, as pragmatic reasoning presents added complexity and not many people are supposed to engage in it in cases similar to this.
Another statistically significant finding is that the experimental group had lower odds of providing correct responses compared to the control group. Thus, despite having more information available to them (the record with previous trials for both concepts was displayed, allowing for pragmatic reasoning), participants in the experimental group seem to struggle with effectively utilizing that information. This could be due to several factors. Firstly, there is no evidence indicating that participants engage in conversational implicatures. In fact, there is no evidence suggesting that whether trials are pragmatic trials or their corresponding control group trials affects the response accuracy. Secondly, the additional information displayed on the screen, combined with longer task durations, may lead to participants becoming less inclined to use that information. This could be due to fatigue, forgetting, or simply choosing not to review older information when that would require a long time.

Finally, there is no evidence suggesting that whether pragmatic trials are M-implicatures or Q-implicatures impacts response accuracy. This is not unexpected, given the limited number of participants and the small number of data points for these specific trials.

Cognitive Models' Results

Switching from the Baseline Model to the Double Update Model yielded a better fit with experimental results only for $\lambda = 0.525$, among the tested λ values (see Section 5.2.2). Furthermore, the improvement was not very significant.

Similarly, using the L_2 Model instead of the Baseline Model did not lead to a better fit with the experimental results for any of the tested α values. This finding is unsurprising, as it was found no evidence that participants engaged in conversational implicatures.

6 Conclusion

This chapter begins with a brief overview of the results of the study, including potential explanations for the outcomes, as well as a discussion of limitations and suggested improvements, in Section 6.1. Subsequently, in Section 6.2, the relevance of the current study in the field of Artificial Intelligence is discussed.

6.1 Outcomes, Limitations and Improvements

This section offers a brief overview of the results, discusses possible explanations for the observed outcomes, and proposes improvements to address identified issues and enhance the study.

The section opens with a discussion of the concept learning results, in Section 6.1.1. Subsequently, Section 6.1.2 focuses on the outcomes of learning two concepts simultaneously. Finally, Section 6.1.3 examines the results concerning conversational implicatures.

6.1.1 Concept Learning

Participants appear to learn the concepts' meaning. Conversely, the cognitive models seem not to learn for any experimental block other than, possibly, the first. This could be due to flaws in the design of the other blocks or to the lexica (the combinations of the hypotheses for the two concepts) having probabilities too close to each other, resulting in similar probabilities for each response.

Possible improvements include re-evaluating the design of the experimental blocks and confirm that a cognitive model focusing solely on concept learning can accurately predict correct responses to non-pragmatic trials. This would help verify that learning through hypotheses expressible as logical formulas in the utilized LOT is a viable approach and that the concepts intended to be learnt in the tasks are the simplest to learn in the corresponding LOT.

Moreover, an improved concept learning model could be created. A viable option would be to use λ calculus, adopting a methodology similar to that employed by Piantadosi et al. (2016). This approach might help overcome the problem of storing many hypotheses with similar probabilities. Furthermore, it would trivially allow the inclusion of more than two literals per hypothesis, which might be an important step for improving the cognitive model.

Lastly, it could be beneficial to include quantifiers in the LOT and let the model learn the likelihood modifier (see Section 4.2.3), along with the costs associated to the features in the experimental stimuli and the operations in the LOT. In order to do this, a larger participant pool would be required.

6.1.2 Learning both Concept Simultaneously

If learners' beliefs about concept meanings interacted and this interaction enhanced learning, the experimental group, which learns two concepts at once, should have outperformed the control group, which learns only one concept at a time. However, the opposite occurred, suggesting that this interaction, if present, may hinder learning. A more plausible explanation, however, is that the additional information in the experimental group, along with longer tasks, could lead to fatigue, forgetfulness, or reluctance to review earlier trials because it takes a long time.

Regarding the results from the cognitive models, the Double Update Model only marginally improves upon the Baseline Model at a specific λ value of 0.525, among the tested λ values (see Section 5.2.2). This improvement is however marginal, and in order to better investigate the model's predictions, it would be best to first implement the improvements outlined in the previous section.

Please, note that due to the the extensive computation time required to run the cognitive models and the lack of access to high-performance computing resources, only few λ values were tested.

An improvement in the data analysis would be to embed the cognitive models in a Bayesian model to determine the best combination of parameters to fit the data.

Simplifying the tasks, such as using less complex concepts, would help mitigate the issues mentioned in this section. Shortening the experiment by reducing the number of trials per block or the total number of blocks could also serve that purpose.

Reducing the amount of blocks might however not be ideal. In fact, having more receiver blocks can lead to a learning effect, which I think not to be harmful in this experiment and is most relevant for the experimental group. In fact, initially participants might not recognize that the task involves conversational implicatures. However, they might consider that an option as they progress with the tasks.

6.1.3 Engaging in Conversational Implicatures

Participants do not seem to engage in conversational implicatures. This could however be due to the limited number of participants and the rarity of pragmatic listeners at the L_2 level, who are needed to engage in the pragmatic implicatures present in the current study. In fact, after applying data exclusion criteria, only 9 participants remained in the experimental group, which is the one where pragmatic reasoining is involved.

To make sure to have participants who reason as L_2 listeners, it is thus crucial to recruit a larger number of participants.

The results of the L_2 Model should be compared to experimental results from participants who reason as L_2 listeners. Since participants in the current study seemingly did not engage in conversational implicature, it is less likely that they acted as L_2 listeners. Therefore, a discussion about the L_2 Model implemented as part of this study would not be meaningful.

An improvement to the analysis of responses to pragmatic trials is the implementation of a label to indicate instances when participants did not engage in conversational implicatures. This improvement, which is now part of the experimental implementation, is discussed below.

In pragmatic trials, among the three objects displayed, one is the target and another was the correct choice from when these same objects were shown in a previous trial with a different message (see Section 4.1.7); let us call this the “false object”. If participants are engaging in conversational implicatures, they would avoid choosing the false object, as the implicature implies that the correct choice is any object other than the false one. Therefore, by labeling the false objects in pragmatic trials, it is possible to determine when participants did not engage in conversational implicatures, assuming their selection of the false object was not due to distraction.

6.2 Relevance of this Study in the Field of Artificial Intelligence

As outlined in Section 2.1, cognitive models can be useful when building agents that emulate human behavior in specific tasks or anticipate human actions to offer assistance when necessary. In this section, I will briefly explore how this applies to cognitive models on concept learning and pragmatics, such as those discussed in this work.

Being able to model concept learning would allow artificial agents to achieve a better level of human-like communication. In fact, they could learn new concepts from the users in a way that resembles the one in which humans learn concepts themselves. Similarly, integrating pragmatic reasoning into AI systems can enhance their ability to communicate in a way that better resembles human communication, since humans heavily rely on this kind of reasoning. This would allow for artificial agents to better understand users and generate natural language that is more similar to that generated by humans, leading to significant improvements in virtual assistants and chatbots, among other systems.

Research into regulating pragmatic reasoning in Large Language Models (LLMs) is currently underway. For instance, Miehling et al., have discussed adaptations of the Gricean maxims of conversation (see Section 2.3.1) for communication between humans and LLMs. Furthermore, they have proposed new maxims specifically tailored for interactions with LLMs (2024).

As I have just outlined, both concept learning and pragmatics are important aspects in human language, and being able to model them can result in AI systems performing better. Since the learning of novel concepts and the communication relying on pragmatic reasoning must not occur indipendendtly of each other, it makes sense to try modeling them together and explore their interactions, as done in this study, and integrate models that take into account such interactions in artificial agents.

7 Bibliography

Carolyn Jane Anderson. Tell me everything you know: A conversation update system for the Rational Speech Acts framework. In *Proceedings of the Society for Computation in Linguistics 2021*, pages 244–253, Online, February 2021. Association for Computational Linguistics. URL `https://aclanthology.org/2021.scil-1.22`.

Anton Benz and Jon Stevens. Game-theoretic approaches to pragmatics. *Annual Review of Linguistics*, 4(1):173–191, 2018. doi: 10.1146/annurev-linguistics-011817-045641. URL `https://doi.org/10.1146/annurev-linguistics-011817-045641`.

Leon Bergen and Noah Goodman. That's what she (could have) said: How alternative utterances affect language use. *Proceedings of the Thirty-fourth Annual Conference of the Cognitive Science Society*, 01 2012.

Leon Bergen, Roger Levy, and Noah D. Goodman. Pragmatic reasoning through semantic inference. *Semantics and Pragmatics*, Vol 9 (2016), 2016. ISSN ISSN: 1937-8912. doi: http://dx.doi.org/10.3765/sp.9.20. URL `http://semprag.org/article/view/sp.9.20`.

Brian Buccola, Isabelle Dautriche, and Emmanuel Chemla. Competition and symmetry in an artificial word learning task. *Frontiers in Psychology*, 9, 2018. ISSN 1664-1078. doi: 10.3389/fpsyg.2018.02176. URL `https://www.frontiersin.org/articles/10.3389/fpsyg.2018.02176`.

Morten H. Christiansen and Simon Kirby. Language evolution: consensus and controversies. *Trends in Cognitive Sciences*, 7(7):300–307, 2003. ISSN 1364-6613. doi: https://doi.org/10.1016/S1364-6613(03)00136-0. URL `https://www.sciencedirect.com/science/article/pii/S1364661303001360`.

Herbert H Clark. *Using language.* Cambridge university press, 1996.

Joshua R. de Leeuw, Rebecca A. Gilbert, and Björn Luchterhandt. jspsych: Enabling an open-source collaborative ecosystem of behavioral experiments. *Journal of Open Source Software*, 8(85):5351, 2023. doi: 10.21105/joss.05351. URL https://doi.org/10.21105/joss.05351.

Judith Degen. The rational speech act framework. *Annual Review of Linguistics*, 9 (1):519–540, 2023. doi: 10.1146/annurev-linguistics-031220-010811. URL https://doi.org/10.1146/annurev-linguistics-031220-010811.

Jerry A. Fodor. *The Language of Thought.* Harvard University Press, 1975.

Michael C. Frank and Noah D. Goodman. Predicting pragmatic reasoning in language games. *Science*, 336(6084):998–998, 2012. doi: 10.1126/science.1218633. URL https://www.science.org/doi/abs/10.1126/science.1218633.

Michael C. Frank and Noah D. Goodman. Inferring word meanings by assuming that speakers are informative. *Cognitive Psychology*, 75:80–96, 2014. ISSN 0010-0285. doi: https://doi.org/10.1016/j.cogpsych.2014.08.002. URL https://www.sciencedirect.com/science/article/pii/S0010028514000589.

Michael C. Frank, Noah D. Goodman, and Joshua B. Tenenbaum. Using speakers' referential intentions to model early cross-situational word learning. *Psychological Science*, 20(5):578–585, 2009. doi: 10.1111/j.1467-9280.2009.02335.x. URL https://doi.org/10.1111/j.1467-9280.2009.02335.x. PMID: 19389131.

Michael Franke and Judith Degen. Reasoning in reference games: Individual- vs. population-level probabilistic modeling. *PLoS ONE*, 11, 2016. URL https://api.semanticscholar.org/CorpusID:721358.

Michael Franke and Gerhard Jäger. Pragmatic back-and-forth reasoning. In Pragmatics, Semantics and the Case of Scalar Implicatures, 2014. Pages 170-200.

Bart Geurts. *Quantity Implicatures.* Cambridge University Press, 2010. doi: 10.1017/CBO9780511975158.

Nicole Gotzner and Jacopo Romoli. Meaning and alternatives. *Annual Review of Linguistics*, 8(1):213–234, 2022. doi: 10.1146/annurev-linguistics-031220-012013. URL https://doi.org/10.1146/annurev-linguistics-031220-012013.

H. P. Grice. Logic and conversation. In Donald Davidson and Gilbert Harman, editors, *The Logic of Grammar*, pages 64–75. Encino, CA, 1975.

Felix Henninger, Yury Shevchenko, Ulf Mertens, Pascal Kieslich, and Benjamin Hilbig. lab.js: A free, open, online study builder, 01 2019.

Laurence R. Horn. *Toward a new taxonomy for pragmatic inference: Q-based and R-based implicature.* Georgetown University Press, 1984. URL `https://www.degruyter.com/database/COGBIB/entry/cogbib.166/html`.

Simon Kirby and James R. Hurford. The emergence of linguistic structure: An overview of the iterated learning model. In A. Cangelosi and D. Parisi, editors, *Simulating the Evolution of Language*, pages 121–147. Springer Verlag, 2002.

Wolfgang Köhler. Gestalt psychology. *Psychologische Forschung*, 31(1):XVIII–XXX, 1967.

Kepa Korta and John Perry. Pragmatics. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy.* Metaphysics Research Lab, Stanford University, Spring 2020 edition, 2020.

Stephen C Levinson. *Presumptive meanings: The theory of generalized conversational implicature.* MIT press, 2000.

Robert Duncan Luce and Howard Raiffa. *Games and Decisions: Introduction and Critical Survey.* Wiley, New York, 1957.

David Marr. *Vision: A computational approach.* MIT Press, 1982.

Erik Miehling, Manish Nagireddy, Prasanna Sattigeri, Elizabeth M. Daly, David Piorkowski, and John T. Richards. Language models in dialogue: Conversational maxims for human-ai interactions, 2024.

Steven Piantadosi and Robert Jacobs. Four problems solved by the probabilistic language of thought. *Current Directions in Psychological Science*, 25:54–59, 02 2016. doi: 10.1177/0963721415609581.

Steven T. Piantadosi, Joshua B. Tenenbaum, and Noah D. Goodman. The logical primitives of thought: Empirical foundations for compositional cognitive models. *Psychological review*, 123 4:392–424, 2016. URL `https://api.semanticscholar.org/CorpusID:12777355`.

Michael Rescorla. The Language of Thought Hypothesis. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Winter 2023 edition, 2023.

G. Scontras and M. H. Tessler. Probabilistic language understanding: An introduction to the rational speech act framework, 2017. URL https://michael-franke.github.io/probLang/. Accessed on 2023-11-7.

Gregory Scontras, Michael Henry Tessler, and Michael Franke. A practical introduction to the rational speech act modeling framework, 2021.

Yoav Shoham and Kevin Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, USA, 2008. ISBN 0521899435.

Gijsbert Stoet. Psytoolkit: a software package for programming psychological experiments using linux. *Behavior research methods*, 42(4):1096—1104, November 2010. ISSN 1554-351X. doi: 10.3758/brm.42.4.1096. URL https://link.springer.com/content/pdf/10.3758/BRM.42.4.1096.pdf.

Gijsbert Stoet. Psytoolkit: A novel web-based method for running online questionnaires and reaction-time experiments. *Teaching of Psychology*, 44(1):24–31, 2017. doi: 10.1177/0098628316677643. URL https://doi.org/10.1177/0098628316677643.

Ron Sun. *The Cambridge Handbook of Computational Psychology*. Cambridge Handbooks in Psychology. Cambridge University Press, 2008. doi: 10.1017/CBO9780511816772.

Joshua Tenenbaum and Fei Xu. Word learning as bayesian inference. *Cognitive Sciences*, 10, 12 2002.

Jérémy Zehr and Florian Schwarz. Penncontroller for internet based experiments (ibex), Jan 2023. URL osf.io/md832.

Appendix

Implementing the Lexical Uncertainty Model

The following Python code is (most of) the implementation of the LUM in the cognitive models that I built as part of this work. It includes detailed comments to enhance readability. This code is utilized to determine the probability to pick each object appearing in a trial. Such probabilities are stored in the variables *p_obj1*, *p_obj2* and *p_obj3*.

```python
nl_list = []
l_one_list = []
p_obj1 = p_obj2 = p_obj3 = 0

# Calculate the coefficient of proportionality
# Iterate over all items
for item in all_items:
    # List all hypotheses for the message in question (called ’message’) that are true for the experimental stimulus ’item’.
    nl = 0
    # Iterate over all lexica and apply the interpretation rule (the formula) for L_1.
    for i in range(len(lexica[message])):
        # If the hypothesis for the current message is true for the current item, then L(u,s) is 1. Alpha is the rationality coefficient. Cost is the cost of the message.
        if lexica[message][i] in true_hypotheses:
            nl += lexica[’probabilities’][i] / (np.exp(cost) ** alpha)
        # If the hypothesis for the current message is true for the current item, then L(u,s) is close to 0.
        else:
            nl += 10 ** (-9) / (np.exp(cost) ** alpha)

    # unnormalized probabilities of picking the objects displayed in the trial
    if item == trial[0]:
        p_obj1 = nl
    elif item == trial[1]:
        p_obj2 = nl
    elif item == trial[2]:
        p_obj3 = nl
    # If we are working with the L_2 listener, the unnormalized probabilities for L_1 must be saved
```

```python
        if L2:
            l_one_list.append(nl)

        # True hypotheses will be recreated for the next item at the next iteration of the loop.
        del true_hypotheses
        nl_list.append(nl)

    # k is the coefficient of proportionality.
    # k * nl_list[0] + k * nl_list[1] + ... = 1. Therefore:
    k = 1 / sum(nl_list)

    # If the listener is L_1:
    if not L2:
        # Update the probability of selecting the items in the trial by using k
        p_obj1 *= k
        p_obj2 *= k
        p_obj3 *= k
        # The probabilities must sum up to 1
        total_probability = p_obj1 + p_obj2 + p_obj3
        if total_probability != 0:
            p_obj1 /= total_probability
            p_obj2 /= total_probability
            p_obj3 /= total_probability
        else:
            p_obj1 = p_obj2 = p_obj3 = 1 / 3

    # If the listener is L_2:
    else:
        # Keep track of the displayed items' probabilities
        p_obj1 = l_one_list.index(p_obj1)
        p_obj2 = l_one_list.index(p_obj2)
        p_obj3 = l_one_list.index(p_obj3)

        # Transform unnormalized probabilities for L_1 to probabilities by using k.
        l_one_list = list(map(lambda x: x * k, l_one_list))
        totprob = sum(l_one_list)
        l_one_list = [prob / totprob for prob in l_one_list]

        nl = 0
        del nl_list
        nl_list = []
        # Calculate L2's normalization coefficient. This is very similar to what was done earlier for L_1. You can see the recursion, as in the interpretation rule for L_2 appear L_1
        for listener_one in l_one_list:
            nl += (listener_one ** alpha) / np.exp(alpha * cost)
            nl_list.append(nl)
        k = 1 / sum(nl_list)

        p_obj1 = nl_list[p_obj1] * k
        p_obj2 = nl_list[p_obj2] * k
        p_obj3 = nl_list[p_obj3] * k
        total_probability = p_obj1 + p_obj2 + p_obj3
        if total_probability != 0:
            p_obj1 /= total_probability
            p_obj2 /= total_probability
            p_obj3 /= total_probability
        else:
            p_obj1 = p_obj2 = p_obj3 = 1 / 3
```

Example Showing Learning Outcomes in Baseline Model Implementation

The following Python code snippet serves as an example of how concept learning works in the cognitive models that I built for this study. The output that is produced by the code snippet is reported underneath it.

To run the code, it is sufficient to substitute it to the code that is located right under the declarations of functions in the *cognitive_models.py* file that contains the cognitive models for this work, which can be found at the link `https://github.com/lpavan98/Master-s-thesis-material`.

```
    # Create some simple hypotheses for the concept associated with the message p.
    # Each literal has cost 1, as well as each operation (only the logical
        conjunction is used as operation in this example).
    p_hypotheses = {
        "hypotheses": [’s1’, ’s2’, ’s2 and c2’, ’c1’, ’c2’, ’s1 and c2’, ’s2 and
            c1’, ’s1 and c1’],
        "costs": [-1,-1,-3,-1,-1,-3,-3,-3],
        "probabilities": []
    }
    # Costs are converted to probabilities.
    p_hypotheses[’probabilities’] = cost_to_probability(p_hypotheses[’costs’])
    # In the beginning, the hypotheses for the concept associated with the message
        q have the same probabilities as those for the concept associated with p.
    q_hypotheses = deepcopy(p_hypotheses)
    lexica = create_lexica(p_hypotheses, q_hypotheses)

    # Feedback allows here to observe that s1c1h1 (the object with size 1, color 1
        and shape 1) is in the extension of the concept associated with p.
    lexica = update_beliefs_based_on_feedback(message=’p’, feedback=’s1c1h1’,
        cuncurrent_reasoning=False, likelihood_modifier_concurrent=0.5)
    # s2c2h1 (the object with size 2, color 2 and shape 1) is in the extension of
        the concept associated with q.
    lexica = update_beliefs_based_on_feedback(message=’q’, feedback=’s2c2h1’,
        cuncurrent_reasoning=False, likelihood_modifier_concurrent=0.5)

    # Print the best 10 lexica. For each lexicon, the meaning of the concepts
        associated with the two messages are displayed, along with the lexicon’s
        probability.
    sorted_lexica = sorted(zip(lexica["p"], lexica["q"], lexica["probabilities"]),
        key=lambda x: x[2], reverse=True)
    for i in range(min(10, len(sorted_lexica))):
        p, q, prob = sorted_lexica[i]
        print(f"Lexicon {i+1}: p={p}, q={q}, Probability={prob}")
```

The output produced by the code provided above is the following:

```
Lexicon 1: p=s1, q=s2, Probability=0.17322557580768658
Lexicon 2: p=s1, q=c2, Probability=0.17322557580768658
Lexicon 3: p=c1, q=s2, Probability=0.17322557580768658
Lexicon 4: p=c1, q=c2, Probability=0.17322557580768658
Lexicon 5: p=s1, q=s2 and c2, Probability=0.023443532365758586
Lexicon 6: p=c1, q=s2 and c2, Probability=0.023443532365758586
Lexicon 7: p=s1 and c1, q=s2, Probability=0.023443532365758586
Lexicon 8: p=s1 and c1, q=c2, Probability=0.023443532365758586
Lexicon 9: p=s1, q=s1, Probability=0.019247286200854058
Lexicon 10: p=s1, q=c1, Probability=0.019247286200854058
```