Visual Reference Resolution: A Machine Learning Approach

Natalia Smirnova

Thesis submitted for the degree of
Master in Informatics: Language and Communication
60 credits

Department of Informatics
Faculty of mathematics and natural sciences

UNIVERSITY OF OSLO

Spring 2017

Visual Reference Resolution: A Machine Learning Approach

Natalia Smirnova

Visual Reference Resolution:
A Machine Learning Approach

http://www.duo.uio.no/

Printed: Reprosentralen, University of Oslo

Abstract

The goal of this thesis is to model the resolution of referring expressions (e.g., *the red ball*) to visual entities in real world. This task is known as *visual reference resolution*. In order to address it, two types of information have to be combined: the visual aspects of the objects in the world and the linguistic information provided by the speaker. In this thesis, we use a machine learning approach to construct a model that incorporates both types of information. For each object in the world and each referring expression, we calculate the probability of resolving this referring expression to each object given this referring expression and the visual aspects of the world. A binary logistic regression classifier using a combination of visual and linguistic features is trained to resolve such references. Both simple references (*the red ball*) and relational references (*the red ball under the green cube*) are handled. The model has been evaluated on two datasets using both virtual and real-world scenes. The evaluation shows that the model performs well, in several cases outperforming existing baselines. It is also shown to be robust to visual uncertainty in the world and to noisy speech input. The model can be extended to incorporate other modalities.

Acknowledgments

As the May snow is falling down, the work on this thesis has almost come to an end. It has been a long journey and it would never be completed without support of many people.

First of all, I want to express my gratitude to my supervisor Pierre Lison for his tremendous help, encouragement and motivation. Starting as a student with no knowledge on machine learning, statistics or dialogue systems, I had to learn everything from scratch. Thank you for teaching me, guiding me on my way and believing in me.

I am also grateful to Casey Kennington and Julian Hough for giving me an opportunity to work with the PentoRef corpus and providing me with the source code for the WAC model.

Thanks also go to Language Technology Group at the University of Oslo, for the nice atmosphere on the 7th floor, interesting seminars and fascinating and useful courses in language technology throughout my bachelor's and master's degrees.

A special thanks goes to Jan Tore Lønning for agreeing to be my formal supervisor, for all the fine courses at Ifi and for always listening to me and my fellow students during the meetings of the Program Committee. It is nice to be heard!

I am incredibly grateful to my family and friends for their never-ending "It is going to be fine! You can do it!" and for still inviting me to different events, in spite of all my rejections during the last year.

And finally, the most important person in my life — Jørgen. Thank you for the countless discussions on machine learning, python, statistics and latex. Thank you for proof-reading this thesis. Thank you for your endless love and support. Thank you for always being there, no matter what.

Natalia Smirnova
Oslo, 15th May 2017

Contents

List of Figures **vii**

List of Tables **xi**

1 Introduction **1**

1.1 Visual Reference Resolution 1

1.1.1 Challenges 3

1.2 Motivation 5

1.3 Goal & Proposed Solution 5

1.4 Thesis structure 6

2 Background **7**

2.1 Overview 7

2.1.1 Rule-based approaches 8

2.1.2 Probabilistic approaches 9

2.2 Givenness Hierarchy 11

2.2.1 GH-based algorithms and their challenges 12

2.2.2 GH-POWER 14

2.2.3 Limitations 15

2.3 Simple Incremental Update Model 15

2.3.1 Model 15

2.3.2 Submodels 18

2.3.3 Evaluation 19

2.3.4 Limitations 20

2.4 Words-as-Classifiers 20

2.4.1 Model 20

2.4.2 Evaluation 22

2.4.3 Limitations 24

3 Approach **25**

3.1 Motivation 25

3.2 Model 26
3.2.1 Simple References 27
3.2.2 Relational References 28
3.3 Possible extensions 29

4 Data 33
4.1 TAKE 33
4.2 TAKE-CV 36

5 Evaluation 41
5.1 Experimental design 41
5.1.1 Visual features 41
5.1.2 Linguistic features 42
5.1.3 Feature selection 43
5.1.4 Classifier 44
5.1.5 Other parameters 44
5.2 Results 45
5.2.1 Simple references 45
5.2.2 Relational references 56
5.3 Discussion 59
5.3.1 Feature selection and most informative features 59
5.3.2 Error analysis 61

6 Conclusion & Future Work 65
6.1 Future Work 66

References 69

List of Figures

1.1 Example of a situated environment 2
1.2 Example scene. The target is highlighted in black for presentation. The scene is borrowed from the TAKE-CV corpus, described in chapter 4 3
1.3 Example of landmark use, from Danescu-Niculescu-Mizil and Lee (2011) 4
1.4 Example of ellipsis, from Danescu-Niculescu-Mizil and Lee (2011) 4
1.5 Example of corrections, from Danescu-Niculescu-Mizil and Lee (2011) 5

2.1 Example of visual environment from Schutte et al. (2010) . . . 8
2.2 Example scene from Funakoshi et al. (2012) 10
2.3 Example scene from Matuszek et al. (2012) 11
2.4 Givenness Hierarchy, from Gundel et al. (1993) 11
2.5 Example participant interface from Kehler (2000) 13
2.6 Comparison of update-incremental and restart-incremental models 17

3.1 Example scene for construction of feature combinations. Selected piece is highlighted in white 28
3.2 Example of creation of feature combinations 28

4.1 12 Pentomino pieces, from R. A. Nonenmacher (2017) 33
4.2 Scene from TAKE 34
4.3 Scene from TAKE, distorted, from Kennington et al. (2015a) 34
4.4 Examples of utterances in TAKE corpus 35
4.5 Extract from scene information, TAKE 35
4.6 Scene from TAKE-CV, target is highlighted in green (a black arrow is added for presentation) 36

4.7 Scene from TAKE-CV, target is highlighted in green and landmark in blue (a black and a red arrow are added for presentation) . . . 36
4.8 Extract from scene information, TAKE-CV . . . 37
4.9 Examples of utterances in TAKE-CV corpus . . . 39

5.1 Results of different setups for TAKE corpus without uncertainty. *l1* and *l2* denote the regularisation used, and *uni* and *bi* indicate whether the bigrams were included into the feature set. The use of stemmer is also specified. . . . 47
5.2 Evaluation results on the TAKE corpus, comparing the accuracy of two models, namely SIUM and the single classifier. The results are obtained using certain (predefined) visual features and linguistic features based on hand transcription . . . 48
5.3 Evaluation results on the TAKE corpus, comparing the accuracy of three models, namely SIUM, WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on hand transcription . . . 49
5.4 Evaluation results on the TAKE corpus, comparing the accuracy of three models, namely SIUM, WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on output from ASR . . . 50
5.5 Evaluation results for simple references on the TAKE-CV corpus, comparing the accuracy of two models, namely WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on output from ASR. All words in each utterance are used 51
5.6 Evaluation results for simple references on the TAKE-CV corpus, comparing the accuracy of two models, namely WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on output from ASR. Only words tagged with `t` from each utterance are used . . . 52
5.7 Example scene from TAKE. Selected tile highlighted in white. Tile numbers are added for presentation. . . . 53
5.8 Probability distributions for incremental RR . . . 56
5.9 Incremental results: accuracy (TAKE, no uncertainty introduced) . . . 57

5.10 Evaluation results for simple and relational references on the TAKE-CV corpus, comparing the accuracy of two models, namely WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on output from ASR 58
5.11 Learning curve for the resolution of relational references 58
5.12 Average number of words and standard deviation for TAKE corpus 62
5.13 Typical ambiguous scenes from TAKE, where the model often makes mistakes. The target (gold) and the guess (predicted) are highlighted in white for presentation 64

List of Tables

2.1 Results of SIUM 19
2.2 Results of WAC 23

5.1 Overview over visual features 43
5.2 Averaged accuracy over setups, TAKE corpus 46
5.3 Most informative feature combinations extracted from a setup where colour features are denoted using one-hot encoding . . . 60
5.4 Most informative feature combinations extracted from a setup where colour features are encoded as Euclidean distance between colours 61

Chapter 1

Introduction

1.1 Visual Reference Resolution

Visual reference resolution (RR) is the task of finding the target object of a given referring expression (RE) in a situated setting. Formally, this task consists of several obligatory steps (Kennington and Schlangen, 2017):

- The speaker (see figure 1.1) perceives an object with specific visual features
- She forms the intention of referring to this object by:
  - uttering a descriptive referring expression (e.g. *the small cube to the left of the ball*)
  - using a demonstrative phrase (e.g., *that*, while pointing)
  - combining these two strategies (e.g., *that small cube* and pointing)
- The listener perceives the objects and hears the utterance
- The listener combines her knowledge about the visual features of the objects and the information received from the utterance and tries to identify the intended object

This intended object is usually called *target*, or *referent*. In our toy example (figure 1.1), the leftmost cube is the target. The ball, which is mentioned in the RE, is the *landmark*. All other objects in the scene are *distractors*. The whole utterance in this example is the RE itself, but in many cases the RE is only the part of the utterance (cf. *Please take the small cube and put it on the big cube*, where one of the two REs is *the small cube*). The RE can contain information about the target's colour, shape, size, spatial position, etc.

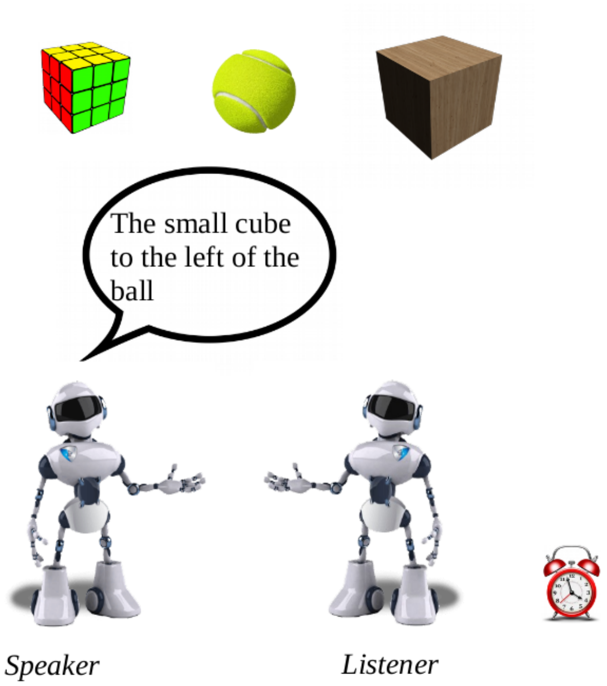


Figure 1.1: Example of a situated environment

The described setting is an example of a *situated dialogue*. Dialogue is defined as a "joint process of communication, which involves sharing of information (data, symbols, context) between two or more parties" (Kruijff et al., 2007). In a situated dialogue, the participants in addition share a common environment. They can perceive the same objects and events, so space is shared. Time is shared as well since the listener starts resolving the utterance as soon as the speaker starts talking (Kennington and Schlangen, 2017).

In a situated dialogue, *language grounding* must occur. It means that the representations of the meanings of natural language have to be tied to the physical world (Matuszek et al., 2012), or, to put it briefly, words have to be connected to perception.

Visual RR is quite difficult even in human-human interaction because of a lot of uncertainty in the perception of the world and to ambiguity in natural language. In human-robot interaction (HRI), it is an extremely complex task due to several challenges which are described below.

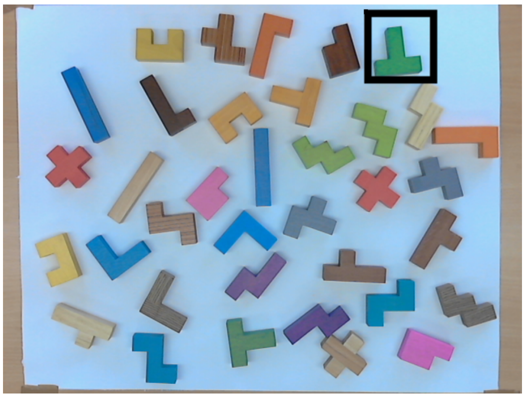


Figure 1.2: Example scene. The target is highlighted in black for presentation. The scene is borrowed from the TAKE-CV corpus, described in chapter 4

1.1.1 Challenges

First, the environment is only partially observable. It means that the agent does not have a perfect and complete perception of the state of the environment. The observations are noisy and provide incomplete information (Kaelbling et al., 1996).

The environment can also be dynamic and change over time. The participants in the dialogue usually can move, the objects in the scene can move, appear or disappear. A simple action of picking up the object changes both the state of the agent and the environment.

Moreover, visual processing is a very difficult task. Colours, for example, are quite hard to determine, and the colour "blue“ perceived by the computer can be rather different from the "perfect“ blue colour `[0,0,255]`. For instance, the piece in a black frame (figure 1.2) seems almost unambiguously green for a human eye, but computer vision perceives RGB values `[52,144,105]` and estimates the colour as blue.

Finally, in many cases the uttered references are not simple. When landmarks are used to describe the intended object, the RE are called *relational*. For such expressions, the agent has to not only find the target, but also all landmarks used and all the relations between possible target and landmarks. In a short dialogue (1.3), the referent of the RE in line 3 is *the battery room*, but also the landmark *the water tank* and the relation *next to* have to be resolved.

Visual RR in spontaneous situated dialogue is even more challenging. First, utterances in human-human interaction are generated and processed incrementally. We do not wait for the dialogue partner to finish the sentence

```
PICARD: Where’s the battery room for the hospital?
[...]
DOCTOR: Outside, around back. Next to the water tank.
```

Figure 1.3: Example of landmark use, from Danescu-Niculescu-Mizil and Lee (2011)

```
1   MISS GULCH: What’s she done? I’m all but lame from the bite
2   on my leg!
3   UNCLE HENRY: You mean she bit you?
4   MISS GULCH: No, her dog!
5   UNCLE HENRY: Oh, she bit her dog, eh?
6   MISS GULCH: NO!
```

Figure 1.4: Example of ellipsis, from Danescu-Niculescu-Mizil and Lee (2011)

or the referring expression to start resolving the reference.

Furthermore, there can be a lot of elliptical constructions in human speech. In example dialogue 1.4, the confusion arises i.a. because of the elliptical construction in line 4.

Humans also tend to make mistakes while speaking, and then corrections are inevitable (see dialogue 1.5). These corrections can be quite difficult to resolve. The module for RR needs somehow to understand which words should not be a part of the actual RE. It should not only handle negations (e.g., *red... no, green ball*), but also corrections that are much less explicit and can occur not directly after the last uttered word. For instance, in line 5, dialogue 1.4, Uncle Henry assumes that ”her dog“ is the correction for the last uttered word, ”you“, in line 3, which leads to misunderstanding.

RR in dialogue is inherently interactive, so another challenge is taking into consideration all forms of interaction feedback provided — different types of confirmation, interest and so on. Non-verbal information, like gaze, gestures, nodding, shaking head is also an essential part of the dialogue which can enrich or specify the meaning of a given utterance, so it would be beneficial to process and apply it as well.

All things considered, visual RR is a complex and challenging task which opens up for a lot of interesting research.

```
JOE: You want me to leave?
TOWNY: No, yes. No, I mean yes please go. Help me to be good.
Come back tomorrow. Promise.
```

Figure 1.5: Example of corrections, from Danescu-Niculescu-Mizil and Lee (2011)

1.2 Motivation

Visual RR is not only a comprehensive and difficult task, it is also a very important task. It is one of the essential components of any situated dialogue. Whether it is human-human or human-robot interaction, REs are always used to refer to different kinds of objects. REs can vary from simple noun phrases in everyday life (*the window* in *Look out the window*) to more complex expressions which possibly contain several other objects and relations between them (*the green book near the red ball which is under the big wooden table*). All such expressions have to be resolved in order to have a successful dialogue act. Therefore, it is essential to have a reliable RR module in a dialogue system.

1.3 Goal & Proposed Solution

The goal of this thesis is to create a model for the task of visual RR. Given the objects with some visual aspects and a RE, the model has to return the target object. The model has to be robust and be able to handle uncertain visual features and noisy linguistic input. Both simple and relational references should be handled. The model should provide respectable results even with little data available.

The proposed solution presents a probabilistic model for the task of visual RR. Given the visual features of the objects in the world and the RE, the model returns a probability distribution over candidate objects. The target object is the argmax of this distribution. To train the model, one needs the representation of the objects (one-hot encoded or low-level visual features), the RE and the annotation of the target. A single binary logistic regression classifier is then used. The model can handle uncertainty in the world and noisy input, and provides good results compared to the other models evaluated on the same datasets.

1.4 Thesis structure

Chapter 2 provides an overview of related work on visual RR. It concentrates on three approaches: an approach based on Givenness Hierarchy, Simple Incremental Update Model (SIUM) and Words-as-Classifiers (WAC). Both SIUM and WAC are the baselines for our own approach. Both the method, the data and the results are described.

Chapter 3 presents the developed approach for solving the task of visual RR. We describe the motivation for a chosen solution and present the model itself. Feature creation is explained in detail. Finally, possible extensions to the model are given.

Chapter 4 describes the two corpora of data we have worked on and gives examples of several types of REs.

Chapter 5 outlines our experimental setup. It describes all possible tuning parameters and provides analysis of the results and comparison to previous work on the same corpora.

Chapter 6 is a summary and conclusion of the thesis, and also a discussion of future work.

Chapter 2

Background

In this chapter, an overview of the approaches used for solving the visual RR task, is given. In the first section, we briefly outline several types of models. In the second one, we more thoroughly describe an approach based on Givenness Hierarchy. The third and the fourth sections are devoted to the intuitively similar, but still different approaches, Simple Incremental Update Model (SIUM) and Words-as-Classifiers model (WAC), which were evaluated on partly the same datasets.

2.1 Overview

The problem of RR is well-studied in several different fields such as linguistics (Pineda and Garza, 2000; Abbott, 2010), psychology (Dahan et al., 2002; Staudte and Crocker, 2009), human-human interaction (Iida et al., 2010; Kennington et al., 2015b) and human-robot interaction (Brøndsted, 1999; Chen and Xu, 2006; Funakoshi et al., 2012). Two comprehensive theses by Denis (2007) and Kennington (2016) provide an exhaustive overview over research in the fields of RR and visual RR respectively. Götze (2016) also describes a substantial part of relevant literature in her thesis. In this section, therefore, we do not attempt to give a *complete* overview over existing research, but rather a brief summary of the approaches. We will concentrate only on comprehension of RR, as the related task of generating REs is beyond the scope of this thesis.

Approaches to the task of RR can be roughly divided into two parts: rule-based and probabilistic.

2.1.1 Rule-based approaches

Rule-based approaches are quite often used for anaphora resolution. Anaphoric REs refer to something already mentioned in the text (e.g., *it* in *Find a red ball. Give it to me*). Especially in written discourse the field of anaphora resolution has been actively researched for many years (Dahl, 1986; Williams et al., 1996; Mitkov, 1998; Akker et al., 2002; Lee et al., 2013). In situated dialogues, REs are usually *exophoric*, i.e. denoting external objects which have not been introduced in the linguistic context yet, but which are within the immediate environment of the speaker (Götze, 2016) (e.g., *the red ball* in *Give me the red ball*).

An example of a rule-based approach resolving exophoric REs is presented in the paper by Schutte et al. (2010). The virtual environment consists of a set of rooms that contain cabinets and buttons (see figure 2.1). Cabinets can be opened and closed, and buttons can be activated. Some cabinets contain items. To fulfil the task, the participants had to retrieve certain items and move them to different cabinets. All objects are assigned a score based on their visibility.

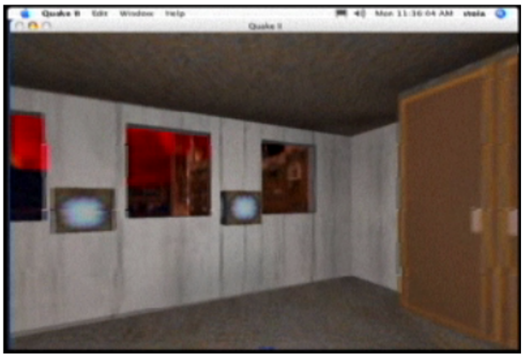


Figure 2.1: Example of visual environment from Schutte et al. (2010)

In order to solve this task, a following set of rules was created:

1. Extract which type of object (door or button) is referred to in the instruction by matching the instruction with the regular expressions *[...]*[1].
2. Collect all objects visible during the time covered by the instruction.
3. Filter out all objects of types incompatible to the instruction.
4. For each remaining object sum the number of ray hits for that object[2].

[1] Regular expressions are predefined by the authors.

[2] In other words, compute a visibility score.

5. Rank the objects using a salience metric.

6. Return the object with the highest salience.

The first step in these rules finds the RE, whereas the second one provides an overview of candidate objects. The RR itself occurs here in steps 3–6.

Another example of a rule-based approach is described in Kruijff et al. (2006). The paper presents the strategies for *intra-modal* and *inter-model* fusion. Intra-modal fusion strategy is used to establish whether different REs denote the same object, whereas the goal of inter-modal fusion is to establish relations between equivalence classes (EC) across different modalities. When the RE is uttered, an equivalence class is created to hold this linguistic representation. Subsequent references are then fused into the same EC. To create these ECs and process all REs, the set of rules is used.

After applying the first model, an inter-modal fusion is carried out, i.e. a linguistic EC is fused with respective ECs from other modalities (e.g., visual properties). The new bindings are created with another set of rules, where each action depends on the number of retrieved inter-modal binding structures.

Other examples of rule-based approaches are models built on the Givenness Hierarchy (Kehler, 2000; Chai et al., 2006; Williams et al., 2016) which are described in section 2.2.

2.1.2 Probabilistic approaches

Probabilistic approaches assign probabilities to each object being the target given a RE. Formally, given a world W and an utterance U, the purpose of RR is to compute a probability distribution over a set of candidate objects. The referred object I is the argmax of this distribution:

$$I^* = \underset{I}{\text{argmax}}\ P(I|U, W) \tag{2.1}$$

An example of a probabilistic approach is presented in Funakoshi et al. (2012). The domain used is a puzzle game Tangram (see figure 2.2). The world W is represented as a set of concepts (shape types, size, etc.) and the utterance U is represented by words in the RE. The data is in Japanese and was collected during human–human interaction. To learn the mapping between W and U, the Bayesian network is used.

The task of RR is formalised in the following way ($\mathcal{W}$[1], X and D represent an observed word, the referent of the RE and the presupposed reference

[1] In the cited paper, the word is denoted by W

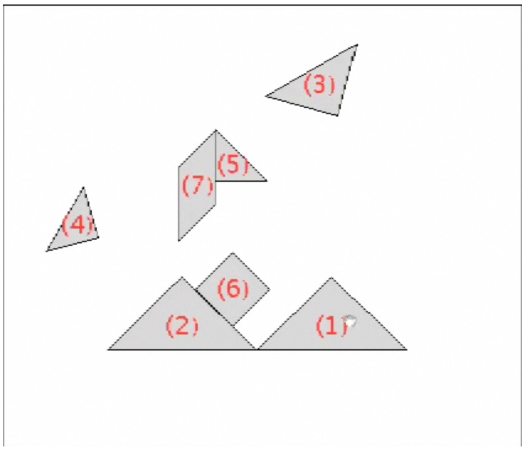


Figure 2.2: Example scene from Funakoshi et al. (2012)

domain):

$$x^{'} = \underset{x \in \mathcal{D}(X)}{\text{argmax}}\ P(X = x|\mathcal{W}_1 = w_1, ..., \mathcal{W}_N = w_N) \tag{2.2}$$

$P(X|\mathcal{W}_1, ..., \mathcal{W}_N)$ is obtained by marginalizing the joint probabilities that are computed with the help of four probability tables.

In order to compute $P(X|\mathcal{W}_1, ..., \mathcal{W}_N)$, four probability tables are needed. The first table is the probability that a hearer observes the word w from the concept c and the referent of the RE x. Formally, it is expressed as $P(\mathcal{W}_i = w|C_i = c, X = x)$. The second one is the probability that concept c is chosen from domain $\mathcal{D}(C_i)$ to indicate the referent x in reference domain d — $P(C_i = c|X = x, D = d)$. The third table is the prediction model: the probability that entity x in reference domain d is referred to $(P(X = x|D = d))$. The final table represents the probability that reference domain d is presupposed at the time the RE is uttered $(P(D = d))$. Since reference domains are implicit, the data cannot be collected to estimate this model. Several a priori approximation functions are used to calculate this probability. By marginalising these four joint probabilities, $P(X|\mathcal{W}_1, ..., \mathcal{W}_N)$ from equation 2.2 is obtained.

This model can handle both definite references, exophoric pronoun references and deictic references. It can also be used for resolving REs with a single target as well as references to two objects.

Another probabilistic approach is described in Matuszek et al. (2012). The authors present the approach for learning three components of the model: (1) visual classifiers that identify the appropriate object properties, (2) representations of the meaning of individual words that incorporate these classifiers, and (3) a model of compositional semantics used to analyse complete sentences. To train visual classifiers (i.e., to represent objects in the world W), colour and shape features are used. To represent an utterance U, a semantic parsing model is used (each utterance then is a Combinatory Categorial Grammar parse). The domain used is a selection of toys, including wooden blocks, plastic food, and building bricks (see figure 2.3). Resolving the reference is computing a joint distribution over the representation of the world and the utterance. The approach is robust to noisy visual input and provides good results (e.g., a precision score of 82%).

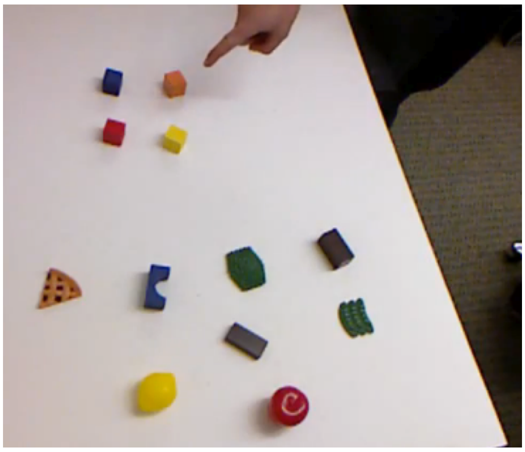


Figure 2.3: Example scene from Matuszek et al. (2012)

Two other probabilistic models are presented further in this chapter, section 2.3 and 2.4.

2.2 Givenness Hierarchy

Givenness hierarchy (GH) is a scale which represents six possible kinds of information status that referring expressions can signal (see figure 2.4). It was developed by J.Gundel in 1993 (Gundel et al., 1993) and is used in several algorithms within HRI. Statuses on the GH are not mutually exclusive: if any piece of information has a certain status, it also attains all lower classes. For

in focus	>	activated	>	familiar	>	uniquely identifiable	>	referential	>	type identifiable
{it}		{*that* *this* *this* N}		{that N}		{the N}		{indef. *this* N}		{*a* N}

Figure 2.4: Givenness Hierarchy, from Gundel et al. (1993)

instance, if some information is in the focus of attention, then it means that it is also activated (in working memory), familiar (in long-term memory), can be uniquely identified (gets a unique mental representation by the end of the NP), can be referred to (is unique by the end of the sentence) and its type can be identified. In the sentence *That red object to the left is a cup*, the determiner *that* implies that the listener has a representation of the object in long-term memory (that it is familiar). But if *that* is replaced by *the*, *The red object to the left is a cup*, the only information encoded is that the addressee is expected to associate a unique representation with the NP, either by retrieving a representation from memory or by constructing a new one (Gundel et al., 2010).

To decide which cognitive status an NP has, Gundel et al. have developed a "coding protocol" which contains different criteria that might be used to determine possible status. Some examples of such criteria are listed below (Gundel, 2010):

> *A referent can be assumed to be in focus if*
>
> 1. the addressee is intently looking at it.
> 2. it was introduced in a syntactically prominent position in the immediately preceding sentence.
>
> *A referent can be assumed to be at least activated if*
>
> 1. it is present in the immediate extralinguistic context.
> 2. it is mentioned in the immediately preceding sentence.

Thus, the GH and the coding protocol provide both data structures for RR and guidelines for how to populate and access them. This information is then used to develop reference resolution algorithms. Williams et al. in their paper describe some of the existing algorithms and present their own solution, GH-POWER (Williams et al., 2016).

2.2.1 GH-based algorithms and their challenges

One of the implementations which is examined is an algorithm by Kehler (2000). It is based on a modified GH where the two last levels (referential and type identifiable information) are omitted. The four following rules are provided for resolving any references:

1. If the object is gestured to, choose that object

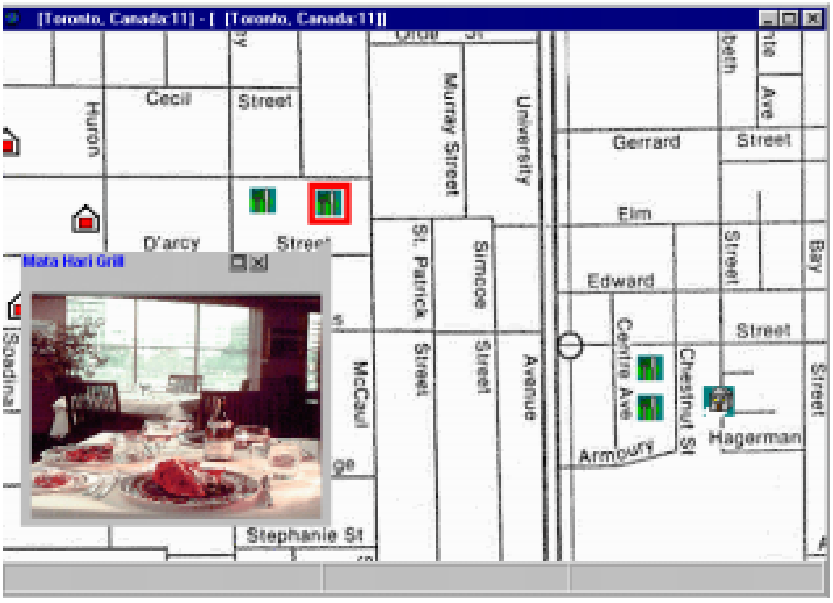


Figure 2.5: Example participant interface from Kehler (2000)

2. Otherwise, if the currently selected object meets all semantic type constraints imposed by the referring expression (i.e., “the museum” requires a museum referent; bare forms such as “it” and “that” are compatible with any object), choose that object.
3. Otherwise, if there is a visible object that is semantically compatible, then choose that object.
4. Otherwise, a full NP (such as a proper name) was used that uniquely identified the referent

The model was evaluated on the data collected by the author. Participants had to plan their holiday in Toronto, given a map of the city and points of interest (figure 2.5). They could ask the wizard questions about the districts (e.g., *What restaurants are there in this area?*), and the corresponding items were highlighted in the scene. The algorithm was able to achieve 100% accuracy, resolving all REs correctly.

The second implementation is made by Chai et al. (2006). This modification of GH includes four other levels: *gesture* (entities gestured towards), *focus* (a combination of “in focus” and “activated” tiers from original GH), *visible* (a combination of “familiar” and “uniquely identifiable”) and *others* (“referential” and “type identifiable”). A greedy algorithm is then used. It first assigns a score between each referential expression X and entity N in a set of vectors (Gesture, Focus, Visible). This score is calculated by multiplying the probability of selecting N from its vector, the probability of selecting that tier given the form of X and the compatibility between X and N. Then the algorithm greedily binds references to entities.

This approach, however, does not capture all aspects of reference resolution in HRI. Williams et al. (2016) concentrate on the five following aspects:

1. Complete certainty of a property is impossible in HRI. An entity can have a certain property with some probability.

2. The algorithm cannot handle not currently visible, hypothetical objects whereas in HRI they are very common and many of the scenarios assume open world.

3. Not physically existing entities (e.g., references referring to events) represent a problem as well.

4. Some references cannot be distinguished since "in focus" and "activated" levels are combined.

5. A greedy algorithm can potentially have difficulties resolving subsequent referential expressions if the first one is incorrectly resolved.

Taking into consideration all these problems, T.Williams et al. propose extended guidelines for GH and a new domain-dependent open-world reference resolution algorithm, GH-POWER.

2.2.2 GH-POWER

The GH-POWER algorithm first parses the utterance and generates a dependency graph which is then converted into a tree. From the tree structure one can extract a set of formulae representing semantics, a set of "status cue" mappings for each referenced entity (e.g., $\{X \to familiar, Y \to infocus\}$) and a type of utterance (e.g., "Statement"). Secondly, GH-POWER populates and sorts four data structures, FOC (in focus), ACT (activated), FAM (familiar) and LTM (long-term memory) using the following rules (only implemented rules are included here):

1. FOC
   - Main clause subject of clause n-1
   - Syntactic Focus of clause n-1
2. ACT

- All other entities referenced in clause n-1

3. FAM
    - All entities referenced in clause n-1
4. LTM
    - All declarative memory

Lastly, the references in a given clause are resolved. If more or less than one hypothesis was found, the set of solutions is returned and the RE is marked as either ambiguous or unresolvable. If only one hypothesis remains, the semantics the RE is resolved.

2.2.3 Limitations

The algorithm developed by Williams et al. provides improvements on all five problematic aspects named earlier. It can handle uncertainty[1], open worlds, references to hypothetical entities, references to unobservable entities and complex noun phrases. There are, however, several areas where more work is needed — resolving plural references (e.g., *the objects*), non-discrete entities (parts or regions of an object), using gesture and eye-gaze for disambiguation and dealing with idiomatic expressions.

2.3 Simple Incremental Update Model

Another approach to solving the task of visual RR is developed by C. Kennington et al. (Kennington et al., 2013, 2014; Kennington, 2016; Kennington and Schlangen, 2017). It is called Simple Incremental Update Model (SIUM) and is a generative RR model.

2.3.1 Model

As mentioned earlier, the goal of visual RR is to determine a referent for a given RE. Formally, RR is a function f_{rr} that, given a representation U of the RE and a representation W of the world, returns I^*, the identifier of the referent (Kennington and Schlangen, 2015). Since the model is stochastic, a

[1] Compared to other GH-based algorithms. Since GH does not specify how to handle uncertainty and how to resolve intra-tier ambiguity, it is a hard task for GH-based approaches. In (Williams et al., 2016) the notion of probability is introduced. They show that if there is 70% of choosing one referent and 40% of choosing another, the RE is resolved to the first object.

probability distribution over candidate objects is computed, and the target object is then the argmax:

$$I^* = \operatorname*{argmax}_I P(I|U, W) \tag{2.3}$$

To make equation 2.3 generative, Bayes' rule is applied:

$$P(I|U, W) = \frac{P(U|I, W)P(I|W)}{P(U|W)} \tag{2.4}$$

From this equation, one can see that it is necessary to maintain a model for all possible intentions and world configuration, and that is not feasible. In order to be able to solve the problem, several assumptions are introduced. To begin with, it is assumed that words in U are uttered precisely to identify the target. Therefore, a mediating variable R is inserted between U and I. R represents more directly what is uttered in U, and also maintains a connection to the target. It represents *properties* that objects have, mapped to words in REs[1].

$$P(I|U, W) = \sum_{r \in R} \frac{P(U|R = r)P(R = r|I, W)P(I|W)}{P(U|W)} \tag{2.5}$$

Then it is also assumed that $P(I|W)$ and $P(U|W)$ can be simplified to $P(I)$ and $P(U)$ respectively, due to conditional independence. They can also be moved out of summation since they do not depend on R. $P(R|I, W)$ can be computed by reading off properties of the objects in W. Equation 2.5 can be then rewritten as following:

$$P(I|U, W) = \frac{1}{P(U)} P(I) \sum_{r \in R} P_w(U|R = r)P(R = r|I) \tag{2.6}$$

Formula 2.6 represents the model working on the whole RE. SIUM, however, is an *incremental* model, assuming that each word in a RE corresponds with one property of an object. It means that the formulation in 2.6 has to be altered, otherwise a different formulation would be required for the REs of different length. Moreover, an *update-incremental* model is preferred to a *restart-incremental* one. An update-incremental model keeps its internal state between incremental update steps, enriching it at each increment with the delta between the current and the previous increment (Kennington,

[1] Properties in the model can be visual properties (colour), shape (e.g., cross or T-shaped) or spatial placement (e.g., `left-of`). The properties can also be connected to additional modalities, for instance an object which a speaker is pointing to and using the word "that" can have a `pointed-at` property.

```
Update-incremental:
(1) the
(2)     red
(3)         ball

Restart-incremental:
(1) the
(2) the red
(3) the red ball
```

Figure 2.6: Comparison of update-incremental and restart-incremental models

2016). A restart-incremental model, on the other hand, the internal state is thrown away between updates and output is always recomputed from scratch using the current input prefix and not just the newest increment of it. Figure 2.6 presents a simple example of both models.

To make the formulation of the model update-incremental, I then is treated as a different variable at each increment, and I in the current step is dependent on all other variables in the current step and the previous step (for a two-word RE):

$$P(I_2|I_1, U_1, U_2, R_1, R_2) = \frac{P(I_1, I_2, U_1, U_2, R_1, R_2)}{P(I_1, U_1, U_2, R_1, R_2)} \tag{2.7}$$

It can be altered in a similar way as 2.6:

$$P(I_2|I_1, U_1, U_2) = P(I_2|I_1)P(I_1) \sum_{r_2 \in R_2} \frac{P(U_2|R_2)(P(R_2|I_2)}{P(U_2)} \sum_{r_1 \in R_1} \frac{P(U_1|R_1)(P(R_1|I_1)}{P(U_1)} \tag{2.8}$$

Several more simplifications are needed to arrive to the final model. First, $P(I_2|I_1)$ is defined as a function that is set to zero when I_1 does not equal I_2. Furthermore, the last summation in 2.8 is the computation from the previous step, which is a distribution over I_1. $P(I_1)$ is then treated as that distribution being made a prior probability that is set to the posterior of the previous step. $P(U_k)$ can be dropped by assuming that all words are equally likely to be uttered. The final formulation then is as following:

$$P(I|U) = P(I) \sum_{r \in R} P(U|R = r)P(R = r|I) \tag{2.9}$$

2.3.2 Submodels

The described model consists of several sub-models, such as the model linking objects and properties together, language and properties and also a prior $P(I)$. These sub-models are briefly explained below.

Objects and properties ($P(R|I)$)

This sub-model connects objects and their properties (colour, shape, position, etc.). It is assumed that with equal probability one of the properties that the object has will be verbalised and as a consequence, zero probability is left to the properties the object does not have. In other words, it is expected for a rational speaker to mention properties that are realised and not all other properties.

If the properties are not clear, this sub-model can also have uncertainty in its representation. In this case, it maintains a *distribution* over properties (the highest probability will then represent the strongest belief that the given object has this property).

$P(R|I)$ can also encode salience information in the distribution over properties. Then $P(R)$ in the derivation is not uniform and should be kept in the model.

It is up to oneself to decide whether to include uncertainty or salience in the model.

Language and properties ($P(U|R)$)

Another sub-model, $P(U|R)$, is responsible for mapping between language and properties. It can be seen as a function from a word (or another linguistic element) to a semantic concept where the set of properties represent the existing semantic concepts. For instance, the word *red* would correspond to the concept *redness* represented by certain properties (e.g., a certain combination of RGB values). $P(U|R)$ is not pre-defined by rules, but learned from data using Maximum Likelihood estimation. For training, it is counted how many times a word co-occurs with a given property, out of all times when the property was represented. This is a kind of grounded semantics.

Contextual prior ($P(I)$)

The third sub-model, $P(I)$, allows to keep track of the distribution over I as the RE incrementally unfolds. At the beginning of the analysis the prior $P(I)$ is set to a uniform distribution. For later steps, it is set to be the posteriori of the previous step.

2.3.3 Evaluation

SIUM was evaluated on the two sub-corpora, TAKE and WOZ, of the PentoRef corpus of spoken references in task-oriented dialogues (Zarrieß et al., 2016). WOZ is a somewhat small corpus which is not used in other experiments, so we will focus on TAKE in this thesis. It is outlined in the next section in comparison with TAKE-CV, another sub-corpus, and presented in detail in chapter 4. We will nevertheless provide a very short description here as well.

TAKE is a German language corpus collected in a Wizard-of-Oz study in Pentomino domain. The participants were shown a Pento board with 15 pieces (figure 4.2), and they had to choose and describe one of the pieces to a wizard. The wizard made a guess, either a confirmation or a rejection was uttered, and the whole process was repeated. Gaze and deixis were also recorded.

For evaluation of SIUM, two kinds of experiments were conducted. For the first one the raw data was used, i.e. the visual properties of the objects were given beforehand. The results include accuracy for the basic model and also for combination with gaze and deixis. We are mostly interested in the speech-only SIUM, so only these numbers are provided below. Other results with additional modalities can be found in the cited papers (Kennington, 2016; Kennington and Schlangen, 2017). In the experiment number two, uncertainty in the perception of the world was introduced. The images were distorted in a particular way (more about it in chapter 4, section 4.1), and the visual properties of the objects were read from these pictures.

The relevant results are presented in the table below (table 2.1).

Corpus		Accuracy, %
TAKE	random	7
	hand transcription	76.7
	ASR output	69.5
TAKE, uncertainty	random	7
	hand transcription	61
	ASR output	43.2

Table 2.1: Results of SIUM

As seen from the table, SIUM seems to be a well-performing model, robust to noisy visual input and to uncertainty in speech recognition. However, combining both types of uncertainty (the last line in the table) provides a quite major drop in accuracy which could advantageously be improved.

2.3.4 Limitations

Despite good results, the described model has some limitations. The most important one is that it can handle only simple references (e.g., *the red ball*), whereas more complex, *relational* references are not taken into consideration (e.g., *the red ball near the green cup*). Negative REs (e.g., *not the red ball*) are not modelled either. It would be also interesting to see whether the model manages to reach equally good results with more objects in the scene.

2.4 Words-as-Classifiers

A similar approach to solving the task of visual RR is presented in several papers by C. Kennington et al. (Kennington and Schlangen, 2015; Kennington et al., 2015a,c). It is called Words-as-Classifiers (WAC) and is a stochastic discriminative model which, given a representation of the RE and a representation of the world, returns a probability distribution over a specified set of potential referents. The target is the argmax of this distribution. In this section, we will have a closer look at this model and the conducted experiments.

2.4.1 Model

This model is based on the same function described in previous section; a function that given a representation U of the RE and a representation W of the world, returns I^*, the identifier of the referent, and argmax is the referent itself:

$$I^* = \underset{I}{\operatorname{argmax}}\ P(I|U, W) \tag{2.10}$$

The task of computing the distribution is divided into two main subtasks: modelling the word meaning for each word and then application and composition of these word meanings.

To model a word meaning, a function from perceptual features of a given object to a judgement about how well this object and this word fit together, is created. This corresponds to the intension, or meaning, of the word. Two different types of words are modelled: those describing properties of a given object (e.g., *red* in *the red ball*) and those picking out *relations* of two objects (e.g., *next to* in *the red ball next to the brown cube*).

Subsequently, the composition of the relevant word meanings is applied. It gives the probability distribution over candidate objects. Here, two types of references are being modelled, *simple references* and *relational references.*

Word meanings

Both types of words are modelled in a similar way. For simple references, for each word w, a binary logistic regression classifier is trained. The classifier takes a representation of a candidate object in the form of visual features x and returns a probability p_w for the object being a good fit to the word:

$$p_w(x) = \sigma(w^T x + b) \tag{2.11}$$

In the formula, w is the weight vector that is learned and σ is the logistic function.

Using the mentioned earlier correspondence, the intension of a word can be seen as the classifier itself, a function from an object to a probability:

$$[\![w]\!]_{obj} = \lambda x.p_w(x) \tag{2.12}$$

In this equation, $[\![w]\!]$ is the meaning of w, x is of the type of feature given by f_{obj}, the function which computes a feature representation for a given object. The classifiers are trained using a corpus of RE, visual representations of the objects in the world and annotations of the referent in each scene. For positive samples, each word in a RE is paired with the features of the target object. For negative samples, a randomly picked object in the same scene (but not the referent) is used.

Training classifiers for relational references is done in a similar way. However, instead of visual features of one object, features of a *pair* of objects are used (for instance, Euclidean distance between two objects, vertical and horizontal differences, left/right and higher/lower than relationships).

Composition

The model for word meanings indicates how well the object and the word fit together. However, RE is seldom represented by only one word, it is usually a combination of several words or sometimes even sentences. It means that all of these words have to be taken into consideration and somehow combined.

As mentioned earlier, two types of references are modelled, simple references and relational references. Simple references are approximately the same as simple NPs (e.g., *the green book*). To get a distribution for a single word, the word classifier is applied to all candidate objects, and then the distribution is normalized. Afterwards, the evidence from all the words in a given RE has to be composed. In order to do that, the contributions of constituent words are averaged, assuming that each word contributes equally.

The averaging function is defined as following (W is the given universe):

$$avg([\![w_1]\!]^{\mathrm{W}}, ..., [\![w_n]\!]^{\mathrm{W}}) = P_{avg}(I|w_n, w_n) \tag{2.13}$$

where

$$P_{avg}(I|w_1, ..., w_n) = \frac{1}{n}(P(I = i|w_1) + ... + P(I = i|w_n)) \text{ for } i \in I \tag{2.14}$$

This function is incremental, meaning that $avg(a, b, c) = avg(avg(a, b), c)$, and can be extended "on the right".

Relational references (e.g., *the green book near the red ball*) have a more complex structure. They consist of, in this case, two simple references (one for target and one for landmark) and a relation between them. For each relation, a "word" classifier is trained (relations like "on the left" are treated as a single token). So, the meaning of the phrase is the function of the meaning of the constituent parts. Assuming that the target constituent contributes $P(I_t|w_1, ..., w_k)$, the landmark constituent $P(I_l|w_1^{'}, ..., w_m^{'})$, and the relation expression $P(R_1, R_2|r)$, the combination of evidence is calculated using multiplication and is as following:

$$\begin{aligned} &P(R_1|w_1, ..., w_k, r, w_1^{'}, ..., w_m^{'}) = \\ &\quad \sum_{R_2}\sum_{I_l}\sum_{I_t} P(R_1, R_2|r)P(I_l|w_1^{'}, ..., w_m^{'})\times \\ &\quad P(I_t|w_1, ..., w_k)P(R_1|I_t)P(R_2|I_l) \end{aligned} \tag{2.15}$$

The last two factors force the pairs being evaluated by the relation expression consist of objects evaluated by target and landmark expression, respectively (Kennington and Schlangen, 2015).

2.4.2 Evaluation

The described model was also evaluated on the two parts of the PentoRef corpus. One of them, TAKE, was introduced in the previous section, and both of them are presented in detail in chapter 4, as they were also used in our experiments. Here we provide a brief description of both corpora, so that it would be easier to see the differences between TAKE and TAKE-CV.

The domain of the corpora is Pentomino puzzle game. The two used subcorpora, TAKE and TAKE-CV, are Wizard-of-Oz studies conducted in the German language. The participants had to describe Pentomino pieces, either selected by themselves (in TAKE) or randomly selected by a system (TAKE-CV) to a wizard whose task was to choose the referred object. Wizard showed their choice to the participants, and after a confirmation (or a rejection if the selected piece was wrong), the new episode started (either a new scene appeared on the screen (TAKE) or new objects to describe were chosen (TAKE-CV)). TAKE corpus also provides us with evidence from gaze and deixis. Both of them are incorporated in the model and improve the results compared to speech only (Kennington et al., 2015a). However, since the other sub-corpus does not provide us with this data, we choose to analyse only the results from the speech model.

Before considering the results, we also need to describe the features used. In TAKE corpus, each object was represented via colour features (RGB (red, green, blue) values, HSV (hue saturation value)), shape (number of edges), position (centroid, orientation) and skewness[1](horizontal and vertical). Almost the same features were used for pieces in TAKE-CV corpus (RGB, HSV, x and y coordinates of the centroids, Euclidean distance from the centre, number of edges). For the relation classifiers, features relating two objects were used (Euclidean distance between objects, vertical and horizontal distances, two binary features denoting higher than/lower than and left/right (Kennington and Schlangen, 2015)).

The results (accuracy scores) are presented in table 2.2. As seen from the table, the model performs better with less noisy input (hand transcription vs. automatic speech recognition). Especially for TAKE-CV, the results are impressive: out of 32 recognized objects, with very noisy ASR input, without extra modalities such as gaze or deixis, the model provides accuracy of 65.3%. The papers (Kennington et al., 2015a; Kennington and Schlangen, 2015) also provide results of incremental processing and some analysis of selected word classifiers.

Corpus		**Accuracy, %**
TAKE	random	7
	hand transcription	63.9
	ASR output	49.9
TAKE-CV	random	
	hand transcription	*no data*
	ASR output	65.3

Table 2.2: Results of WAC

[1] Skewness, uncertainty and distorted images are described in sec.4.1

2.4.3 Limitations

The described approach is the basis for a robust and well-performing model. However, one can argue that the model has a weakness as well: all words in the RE contribute equally to the final result. In a RE *the red ball*, all the three words are considered to be equally important when RR takes place, although this is not true. One solution to this problem would be a stop-list, but such lists as a rule contain a lot of prepositions, for instance, and they are very important for resolving references. There are also some content words that have a weak referential content (e.g., the word *piece* in the described corpora), but such words are never in the predefined stop-lists. Such stop-lists are also different for every language. The other possible solution is letting the machine learning algorithm decide which words are more important. Then it could decide which words in the RE have to be taken into consideration. This second solution is used in our approach and described in chapter 3.

Other possible improvements which partly depend on enough relevant training data are handling negation in different forms (e.g., *not the red ball* and *the not so red ball*) and generalised quantifiers (e.g., *all red balls*).

Chapter 3

Approach

In this chapter, we describe the developed approach for solving the task of visual RR. In the first section, the motivation for creating a new model is presented. The model itself is outlined in section 2. In the same section, creation of feature combinations is explained, and a simple example with a toy vocabulary and scene is given. In section 3, possible extensions to the model are presented.

3.1 Motivation

In the previous chapter we described two types of approaches used for resolving references — rule-based and probabilistic. The model we will present in this chapter is a representative of the second type — a probabilistic model which computes a probability distribution over candidate objects given a RE and the world. As mentioned earlier, our model is strongly influenced and inspired by SIUM (section 2.3) and WAC (section 2.4), but differs in key areas. We try to offer a solution for some of the described limitations or disadvantages of the previous models.

To begin with, WAC assumes that all words contribute equally to the meaning of the RE. From a linguistic point of view, this is not true as some words have a weaker referential content than other. In our model, we try to take this difference into consideration. The classifier is trained in such a way that more informative features and, as a consequence, words, receive more weight. This approach is more principled and supposedly, should lead to better results

Moreover, WAC relies on the estimation of a separate classifier for each word. In contrast, our approach requires only one classifier to train. Although the feature vector is very large, it is also very sparse, so the computation is not more expensive.

Compared to SIUM, our model can handle not only simple references, but also more complex relational references. To be able to do that, one extra classifier is created. The results from classification from simple classifiers and relational classifier are then combined, and target object is the argmax of the probability distribution. We discuss relational references in more detail further below.

Finally, the model is easily extensible. In the experiments that we conducted only some of the possible extensions were used, for instance, restart-incremental model for classification part and introduction of some more complex structures than single words into features. Possible extensions to the model are more thoroughly described in section 3.3, and all the experiments and the results are presented in chapter 5.

3.2 Model

The model we developed is a probabilistic model. Its goal is to return a probability distribution over candidate objects given a RE and visual features of all objects. In other words, we want to calculate a probability of resolving a referring expression RE as object o given this referring expression and the world W. The target object is the argmax of the probability distribution. Formally, we can express this in a following equation:

$$o_{\mathrm{T}} = \underset{o}{\operatorname{argmax}}\ P(resolution(RE) = o|RE, W) \tag{3.1}$$

To compute this probability distribution, we need to calculate a *fit function* for each candidate object/RE–pair and normalise:

$$P(resolution(RE) = o|RE, W) = \frac{P(fit(o, RE))}{\sum_{o'} P(fit(o', RE))} \tag{3.2}$$

In order to calculate a fit function, we train a binary logistic regression classifier which takes a representation of a candidate object via a combination of visual and linguistic features and returns the probability for each object in the scene being the target object given the referring expression. We repeat here the formula from the previous chapter:

$$p_{re}(x) = \sigma(w^T x + b) \tag{3.3}$$

In this equation, x is the combination of visual and linguistic features, w^T are the weights learned, b is the intercept and σ is the logistic function. $p_{re}(x)$ is the probability of the object being a target object given the RE.

To train a classifier, we use a corpus of REs (annotated or not annotated), visual representations of the scenes and annotations of the referent in each scene. In our corpora, there was only one target object per scene. In order to collect positive samples, feature combinations for the target object are used. For negative samples, an arbitrary number of other (random) objects in the scene are used.

3.2.1 Simple References

For simple references, a single classifier is trained. Training data is collected using the visual properties of the objects paired with linguistic information in the RE. In other words, *feature combinations* are created — features that contain both visual and linguistic information.

In order to create them, we need a vocabulary and a list of predefined visual features. Such visual features can be, for example, colours, shapes, position, etc. Feature names are arbitrary, but for simplicity's sake we also combine linguistic and visual information in each feature name. For instance, feature `red_kreuz` contains information about a colour propery (*red*) and a word used (*kreuz*). The cardinality of the created feature set is then the number of visual features times the number of words in the vocabulary.

After the feature set is created, we have to create a training set. For a given object and a respective RE, each feature gets a positive value if both visual and linguistic information is true (i.e., the object has both the given visual feature and the given word is observed in the RE), and 0 otherwise. For each scene, the one selected piece is a positive sample. Number of negative samples is a tuning parameter and can be freely chosen between 1 and number of all pieces -1.

A simple example of feature set creation and population is given below.

Example of creation of feature combinations

Assume that the predefined visual features[1] are listed on line 1 (figure 3.2), the vocabulary is given on line 2, the RE and its translation to English are provided on lines 3 and 4 respectively. The scene is depicted on figure 3.1, the chosen piece is highlighted. Raw visual features, read directly from the scene or given beforehand, are provided on lines 5 and 6, figure 3.2. The set of new features is then the Cartesian product of visual features and vocabulary, and its cardinality is 6 visual features $\times 8$ words $= 48$. The subset of this set is

[1] Visual features `c` and `l` denote shape of a Pentomino piece; more information about the domain and the corpora is provided in chapter 4

presented on lines 7 and 8. Finally, a part of the populated feature set for the given RE and given piece is presented on lines 9 and 10.

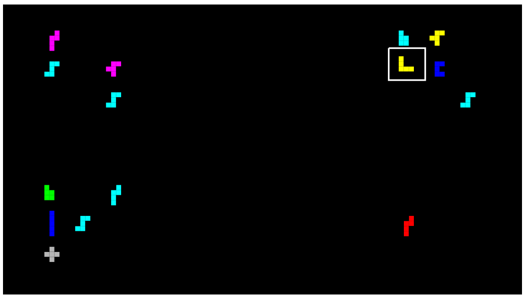


Figure 3.1: Example scene for construction of feature combinations. Selected piece is highlighted in white

```
 1  Visual features: red, yellow, left, right, c, l
 2  Vocabulary: das, rote, gelbe, l, unten, oben, links, rechts
 3  Referring expression: das gelbe L oben rechts
 4  Translation: the yellow L on the top right
 5  Raw visual features: {red:0, yellow:1, left:0, right:1,
 6                        c:0, l:1}
 7  Feature set: (red_das, red_rote, red_gelbe, l, ...,
 8                yellow_das, yellow_rote, yellow_gelbe, ...)
 9  Features: {red_das:0, red_rote:0, red_gelbe:0, ...,
10            yellow_das:1, yellow_rote:0, yellow_gelbe:1, ...}
```

Figure 3.2: Example of creation of feature combinations

3.2.2 Relational References

Relational references are references which contain a relation between a simple reference to a target and a simple reference to a landmark. To resolve such references, we do not only need to calculate the fit between a given RE for the target and a RE for the landmark, but also a relation between them.

Formally, it can be expressed in a following way:

$$
\begin{aligned}
P(fit(o_{\mathrm{T}}, RE)) = P(fit(o_{\mathrm{T}}, RE_{\mathrm{T}}))\times \\
\sum_{o_{\mathrm{L}}} P(fit(o_{\mathrm{L}}, RE_{\mathrm{L}}))\times \\
P(fit(relation(o_{\mathrm{T}}, o_{\mathrm{L}}))
\end{aligned}
\tag{3.4}
$$

To calculate the first two probabilities, we train two classifiers — the first one, *simple classifier*, is exactly the same as described in the previous subsection. It is used for all simple references in the utterance. For instance, in the RE named in chapter 1, *the small cube to the left of the ball*, there are two simple references, *the small cube* and *the ball.* In order to identify these objects, a simple classifier is used. The rest of the utterance, however, is a relation (*to the left of*). To handle such relations another classifier is created — a relational one. It is also built on feature combinations, but the features used for this classifier contain information *relating* two objects. All of them are positional — Euclidean distance between objects, vertical and horizontal differences and also binary features denoting the relationships above/under and left/right.

To combine the evidence from simple classifiers and the relational classifier, we multiply[1] the provided probabilities.

3.3 Possible extensions

The described model can resolve two kinds of REs, simple and relational. It was evaluated on two corpora and the results are provided in chapter 5. The model, however, can also serve as a basis for more complex models which incorporate several modalities and take, for instance, syntactic structure into consideration. Some of these extensions are analysed and evaluated in the next chapter, whereas other are more theoretical due to the limitations of the available corpora.

Incrementality

The model can be altered to be *restart-incremental.* During human-human spontaneous dialogue, we do not need to wait for the end of the utterance to start resolving it. The same approach can be used in our model: when perceiving an utterance, it attempts to identify a target object from the very

[1] We sum the provided probabilities given that we work with log-probabilities

first word. Formally, it would mean that we have to restart the resolving process after each new word in the utterance. This extension was implemented for our dataset.

Complex linguistic features

For creating feature combinations, our model, as described in this chapter, takes into consideration visual and linguistic features. For the linguistic features, the use of separate words seems to be sufficient. However, more complex structures can complement the features set. One example of such structures is n-grams. During training, all or x most common n-grams can be extracted and then used in the same way as a simple feature combination. For instance, feature `red_rote_kreuz` will receive a positive value if the object is red and the RE describing this object contains a bigram `rote kreuz`. Integrating more complex features with bigrams is implemented and evaluated for our dataset. Another complex structures that could also be used in the task are collocations and idioms.

Cardinality

The model can potentially resolve references to several objects. In all the previous examples we assumed that there is always only one referent for the given RE. It is a very common case in reality as well, but it is not the only case. It is also possible to refer to several objects. For instance, the RE *two red balls* refers to two objects and has therefore a cardinality of two. Our model can take cardinality into consideration and return several referents if that is the case. Formally, it can be expressed in the following way (given that O is a set of potential referent objects and C is a random variable corresponding to the cardinality of the RE):

$$\begin{aligned} P(resolution(RE) = O|RE, W) \\ &= \sum_{c=1,2,...} P(resolution(RE) = O, C = c|RE, W) \\ &= \sum_{c} P(resolution(RE) = O|RE, W, C = c) \times \\ &P(C = c|RE, W) \end{aligned} \tag{3.5}$$

This equation could also include cardinality of the world as an important variable since number of the referents can not exceed the number of the objects in the world. The formalisation would be then as following:

$$\begin{aligned}
P(resolution(RE) &= O|RE, W, |W|) \\
&= \sum_{c=1,2,\ldots,|W|} P(resolution(RE) = O, C = c|RE, W) \\
&= \sum_{c} P(resolution(RE) = O|RE, W, C = c) \times \\
& P(C = c|RE, W)
\end{aligned} \tag{3.6}$$

Our dataset did not include any REs with several referents, so this extension was not implemented or evaluated.

Salience

The model can also be extended by including the information about *salience*. Salience is defined as the property of being distinct, particular, discriminating, remarkable, or prominent in a certain context (Götze, 2016). In other words, some objects can be more salient than the others because they have more distinguishable features. Knowing those features can help to identify the referent. Formally, salience can be incorporated in our model as the prior probability in the following way (assuming that $sal(o)$ is salience of the given object):

$$P(resolution(RE) = o|RE, W) = \frac{P(fit(o, RE))}{\sum_{o'} P(fit(o', RE))} \times sal(o) \tag{3.7}$$

Salience was not available in our corpora, so it was not evaluated in our experiments.

Gaze and deixis

Finally, the model can be extended by using several modalities, for instance evidence from gaze and deixis. It can be done in the same way as described in Kennington and Schlangen (2017). For each speaker, a reference point on the scene has to be calculated. For gaze, it can be the fixated point provided by an eyetracker, and for deixis, the point on the scene that was pointed at based on a vector calculated from the shoulder to the hand, provided by a motion controller. Then the centroids of all objects can be compared to the reference point and yield a probability of that object being ”referred“

to by a given modality (gaze or deixis) by introducing a Gaussian window over the location of the point. Gaze and deixis can be then incorporated using the linear interpolation. In our thesis, we did not concentrate on other modalities, so neither gaze nor deixis are integrated into the model.

Chapter 4

Data

To perform evaluation of our model, we chose to use data from PentoRef (Zarrieß et al., 2016). It is a corpus of spoken references in task-oriented dialogues collected in systematically manipulated settings. The domain is a puzzle game Pentomino, based on 12 different combinations of five squares (figure 4.1). The corpus consists of several sub-corpora which were collected for different goals and in different settings. For our evaluation, we used two of the sub-corpora, TAKE and TAKE-CV which are thoroughly described below.

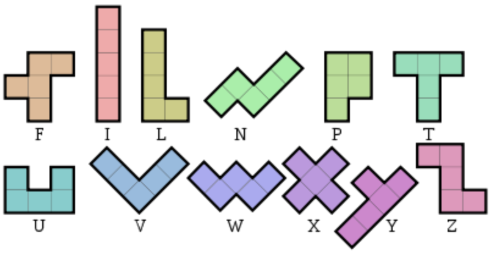


Figure 4.1: 12 Pentomino pieces, from R. A. Nonenmacher (2017)

4.1 TAKE

The TAKE corpus is a result of a Wizard-of-Oz study conducted in German language. The participants were presented a game board on a computer screen, with 15 randomly selected Pentomino pieces. The pieces were grouped in the corners of the screen (see figure 4.2). The participants then had to choose silently a random piece and describe it to the wizard, using words and gestures. When a piece was selected, the participants had either to utter

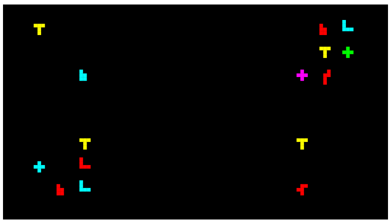


Figure 4.2: Scene from TAKE

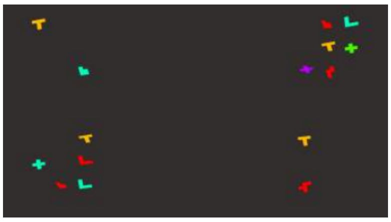


Figure 4.3: Scene from TAKE, distorted, from Kennington et al. (2015a)

a confirmation or give a negative feedback. If the wizard misunderstood anything or there was a technical problem, the episode could be flagged by the wizard. Finally, a new board was created and the process repeated. One such ”round“ we call an *episode*.

The utterances, arm and eye movements and board states were recorded. All utterances are provided in TextGrid files, with timestamps and comments (e.g., ”girl used the wrong form (blaues Kreuz) -> <m=""blaues""> <v=""blaues""> “). Each scene is described via an xml file, where for each piece id, colour, shape and position are available (figure 4.5). All pictures of the scenes are provided in png format.

For parts of our experiments we needed to test if the model can handle uncertainty and noisy input. The pictures were then manipulated exactly in the same way as it is described in the earlier mentioned papers (Kennington et al., 2015a; Kennington, 2016). Briefly, the colours and shapes were distorted, and the resulting images (see figure 4.3) were processed using Canny Edge Detector to segment the objects. As a result, features that were ”closer“ to the real world were extracted. These features will be discussed further in section 5.1.1.

In total, there were eight participants, all university students, aged 18–30. Seven of them were native German speakers, the non-native speaker had a very good command of German. 1214 episodes were recorded, and 165 of them were flagged, so altogether 1049 episodes are used for evaluation.

The participants are different, so the REs they uttered are also quite different. Some examples of the utterances with English translation are presented in figure 4.4[1].

[1] Each German utterance corresponds to one separate episode. Some of the participants failed to utter a confirmation (or it was provided non-verbally)

```
1. das rosa symbol rechts oben
   the pink symbol right  on the top


2. dann aus der gruppierung da unten    links einmal
   then from the group    down there to the left again

   das lila l   das auf kopf steht ... ja, richtig
   the purple l which on head stands ... yes, correct


3. unten   links    das grüne ... okay
   down to the left the green ... okay


4. und dazu dann wir haben ja diese fünf zeichen da oben
   and    then   we  have     these five symbols up there

   und ich möchte genau   das      in der mitte haben ... richtig
   and i    want exactly the one in the middle      ... correct
```

Figure 4.4: Examples of utterances in TAKE corpus

```
<piece type="P" id="tile-3" label="" color="blue">
    <posture rotation="0" isMirrored="false"/>
    <start-field>grid4.1-0</start-field>
    <goal select="false" delete="false">
        <posture rotation="0" isMirrored="false"/>
    </goal>
</piece>
```

Figure 4.5: Extract from scene information, TAKE

4.2 TAKE-CV

The other used corpus, TAKE-CV, is also a Wizard-of-Oz study conducted in German, however, there are several important differences to mention. In this setting, the participant was placed in front of the table with 36 Pentomino bricks randomly spread across the table. Above the table there was a camera, filming it; one object (or one pair of objects) was chosen randomly and shown to the participant on the display in front of herself (see figure 4.6). The experiment consisted of two phases.

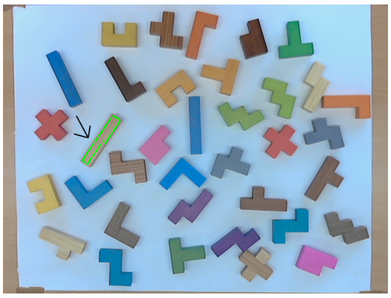


Figure 4.6: Scene from TAKE-CV, target is highlighted in green (a black arrow is added for presentation)

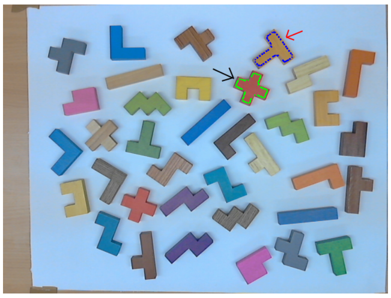


Figure 4.7: Scene from TAKE-CV, target is highlighted in green and landmark in blue (a black and a red arrow are added for presentation)

In phase one, only one object was chosen. The participant had to describe the object to the wizard using only speech. The wizard had an identical screen in front of herself. She clicked on what she thought was the target object, and if it was correct, a tone sounded and a new episode began. If not, another tone sounded, and the episode was flagged. Several times the participant was instructed to shuffle the Pentomino pieces.

In phase two, the pair of objects were chosen and presented to the participant. They were highlighted in different colours (see figure 4.7), and one of them was supposed to be a target and the other one a landmark. The participant had to describe the target object using the landmark object and a suitable relational expression.

The utterances and board states were recorded. All utterances are provided in TextGrid files as mentioned above. Each scene is described via an xml file, with a similar to TAKE, but somewhat different structure (figure 4.8). All pictures of the scenes are provided in png format.

In total, nine participants, aged 17 to 31, took part in the experiment.

```
<object id="0" isLandmark="False" isTarget="False">
  <position global="right bottom" x="394" y="437"/>
  <shape BestResponse="P">
    <distribution F="0.2" I="0.0" L="0.0" N="0.0" P="0.8"
                  T="0.0" U="0.0" V="0.0" W="0.0" X="0.0"
                  Y="0.0" Z="0.0"/>
    <orientation value="14.1549884093"/>
    <skewness horizontal="left-skewed" vertical="symmetric"/>
    <nbEdges value="6"/>
  </shape>
  <colour BestResponse="Pink">
    <distribution Blue="3.47419472069e-33" Brown="1.516339287e-20"
                  Grey="0.00167243290562" Green="4.81099258083e-36"
                  Orange="4.94079260013e-12" Pink="0.782038731836"
                  Purple="0.209469777687" Red="0.00681905756658"
                  Yellow="1.9143319164e-24"/>
    <hsvValue H="152.03081914" S="112.01703163"
              V="184.217356042"/>
    <rgbValue B="178.728304947" G="104.124087591"
              R="182.701540957"/>
  </colour>
</object>
```

Figure 4.8: Extract from scene information, TAKE-CV

All but one were native speakers of German. Phase 1 for one participant and phase 2 for another participant were damaged due to technical difficulty and misunderstanding and were not used. Altogether, 870 not flagged episodes were recorded, thereof 410 episodes in phase 1 and 460 episodes in phase 2. In each scene there were 36 Pentomino pieces, and 32 of them were recognized on average by computer vision.To obtain the speech, Google Web Speech was used, with a word error rate of 0.65.

Each utterance was annotated using a simple tagging scheme[1], where different tags were used for words describing a target object (`t, td, tdc, tds, tdf, tdo, tp`[2]), landmark object (in the same way, but starting with (`l`), relational expressions (`r`) and other linguistic material (`o`). If multiple landmarks were used, the tags and the respective relation words were marked with numbers. If the landmark was described relative to the target, the `r`-tag was used for the relation word. Some examples of tagged utterances are presented in figure 4.9

[1] The tagging scheme can be found at `https://uni-bielefeld.sciebo.de/index.php/s/FDBhTOb1539nWQZ/download?path=%2FTake_CV_PENTOREF%2Fdocumentation%2Fpost&files=TAKE-CVAnnotationScheme.pdf`. Accessed May 3, 2017.

[2] In these tags, `d` stands for description, `c` – for colour, `s` – for shape, `f` – for field, `o` – for other (length of the object, how it is turned), `p` – for pronoun

```
1. äh das objekt ist ein rotes kreuz ganz links
    o  t    t     t   t   tdc   tds   tdf  tdf
   ah the object is  a    red  cross on the left

   neben dem blauen t
     r    l   ldc  lds
   next   to  blue  t


2. neben das ziel    objekt liegt einem blauen t
    r-    t   t        t      t     l    ldc  lds
   next to the target object   is    a    blue  t


3. das rote kreuz ist neben einem blauen t
    t   tdc  tds   t   r1    l1    l1dc  l1ds
   the  red cross  is next to a    blue   t

   und liegt unter einem grünen objekt
    t    t    r2     l2   l2dc   l2
   and   is  under   a   green  object
```

Figure 4.9: Examples of utterances in TAKE-CV corpus

Chapter 5

Evaluation

In this chapter, we start with describing the experimental design we used for our experiments — the feature set creation, feature selection, choice of classifier and different parameters. In section 2, the results are presented and compared to the baselines. Finally, some discussion is provided.

5.1 Experimental design

Given the datasets, our task was to implement the model described in chapter 3. The model is intuitively simple and was implemented in Python with the help of several libraries (`scikit-learn, NumPy, TextBlob, colormath` and some others). In this section, we will present design solutions for all parts of the model.

5.1.1 Visual features

First, the visual features of the objects in each scene have to be collected. For the two corpora, these features are somewhat different. All features for both corpora are presented in table 5.1 and more thoroughly described below.

TAKE

As mentioned earlier, the data from the TAKE corpus can be used as either gold standard annotations or features extracted from the actual images through computer vision. If the data is read directly from the provided xml-files, the scenes, objects and features are *certain*. The features collected are represented with the help of one-hot encoding. Altogether, there are seven colours (`red, green, blue, magenta, yellow, grey, cyan`), 12 shapes (all possible Pentomino bricks as described in chapter 4) and four position variables (`left, right, top, bottom`), whereas two of them get value 1. For some experiments, we replaced the one-hot encoding for colours with the measure of difference between colours. There are two ways to do that: to calculate the Euclidean distance between RGB values or to use a special metric for colour distance Delta E (ΔE^*). Since only "perfect" colours are used (for instance, if we have to calculate the distance between the blue and the red colours, we use the "perfect" blue colour with the RGB value of `[0,0,255]` and the "perfect" red colour `[255,0,0]`), no uncertainty is introduced, so we include these settings under the "certainty" label.

On the other hand, if the visual features are read from the distorted images, the experiments incorporate *uncertainty.* For the colour features then the RGB and HSV[1] values are used, the shape is represented by the number of edges[2], the position is defined by `x` and `y` coordinates of the centroids and also skewness is presented. Skewness is encoded using a one-hot vector, with variables `left, right` and `symmetric` to denote horizontal skewness, and `top, bottom` and `symmetric` for vertical skewness.

TAKE-CV

TAKE-CV is a corpus with real-life objects, so there is no certainty in the settings. Visual features are almost the same as for the setting with uncertainty in TAKE corpus. The skewness is not used, and the position features include also a distance from centre (`cdiff`). TAKE-CV, however, contains not only simple references, but also relational references, so visual features for the relational classifier are also collected. These features include the Euclidean distance between centroids of the objects, the vertical and horizontal differences and four features to encode relative position of the target (`higher than, lower than, to the left of, to the right of`).

5.1.2 Linguistic features

To construct necessary feature combinations, linguistic features are also needed. They are extracted in the same way for both TAKE and TAKE-CV. The most basic setup is just extraction of the words and case folding. This gives us the vocabulary size of 382 for hand transcription and 1048 for ASR for TAKE corpus and 516 and 1306 words respectively for TAKE-CV.

Words can also be processed. Two possible extensions here are the use of a lemmatiser and a stemmer. The lemmatiser used is provided by the

[1] HSV stands for hue, saturation, value colour space, which is basically a representation of RGB values in a cylindrical coordinate system (Hanbury, 2002)

[2] Several Pentomino pieces can have the same number of edges

Corpus		Visual features	
TAKE	certainty	colour	one-hot encoding colour distance
		shape	one-hot encoding
		position	grid rules, one-hot encoding
	uncertainty	colour	RGB, HSV values
		shape	number of edges
		position	x,y coordinates
		skewness	horizontal and vertical
TAKE-CV	uncertainty	colour	RGB, HSV values
		shape	number of edges
		position	x,y coordinates, cdiff

Table 5.1: Overview over visual features

German extension of the `TextBlob` library, `textblob-de`[1]. To stem words, the German version of the `SnowballStemmer`[2] from `NLTK` was used (Bird et al., 2009). Both the lemmatiser and the stemmer reduce the vocabulary size and can potentially improve the results.

Linguistic features can also be more complex. They can contain not only separate words, but n-grams or collocations. In our experiments, we also used 10 most common (in the given corpus) bigrams. The cardinality of the feature set is then increased with 10× number of visual features.

5.1.3 Feature selection

Since the feature sets are large, it is worth using a feature extraction module to reduce the size of the used features and to analyse the most informative features. In our experiments, we used a module from `scikit-learn`, `SelectFromModel`[3]. It is a meta-transformer for selecting features based on importance weights. A float to describe a threshold which tells whether the given feature is informative enough to keep it can be provided for the module. The threshold is ”mean“ by default.

[1] The package can be found at `https://pypi.python.org/pypi/textblob-de/`. Accessed May 3, 2017.

[2] Source code of the stemmer can be found at `http://www.nltk.org/_modules/nltk/stem/snowball.html`. Accessed May 3, 2017.

[3] The module can be found at `http://scikitlearn.org/stable/modules/generated/sklearn.feature_selection.SelectFromModel.html`. Accessed May 3, 2017.

The results of using the feature selection module depend also on the chosen classifier, its solver and regularisation penalty. The classifier and its parameters are presented in the next subsection.

5.1.4 Classifier

The classifier used in our experiments is a binary logistic regression classifier. Logistic regression is a linear model for classification which provides as an output a probability that the given point belongs to a certain class. The main assumption of such classifier is that the data can be separated using a linear boundary (Lemeshow and Hosmer, 2008).

For our model, we use a logistic regression classifier as implemented in `scikit-learn`[1]. This classifier has several tuning parameters, but we mostly concentrated on two of them.

To begin with, it is possible to specify the type of *regularisation*. There are two types of regularisation, L1 and L2. The first one uses a penalty term which encourages the *sum of the absolute values* of the parameters to be small. The second, L2 regularisation, encourages the *sum of the squares* of the parameters to be small (Ng, 2004). L1 regularisation in many models causes a lot of parameters to equal zero so that the parameter vector is sparse. This makes it a natural candidate in feature selection settings, where potentially many features should be ignored. We conducted our experiments with both types of regularisation and found out that only in some settings the results differ for L1 and L2.

Another important parameter is *solver* which can be set to be `newton-cg`, `lbfgs`, `liblinear` and `sag`. Default solver is `liblinear`. It is more suitable for smaller datasets, and it can be used with both L1 and L2 penalty, so it was a natural choice in our experiments. However, it should be mentioned that the `sag` solver was also used and it led to better results for the TAKE-CV corpus. For this solver, maximum number of iterations to converge has to be increased till 250 in order to get the reported results.

5.1.5 Other parameters

The remaining parameters for the setup that were not mentioned earlier, is the choice of the transcription and the number of negative samples to use. The transcription is provided in two versions, a record by human transcribers (HAND) and an output from Google automatic speech recogniser (ASR). The

[1] The classifier and its parameters can be found at `http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html`. Accessed May 3, 2017.

second choice leads to more uncertainty and allows us to check whether the model is robust to noisy speech input.

Finally, the number of negative samples has to be taken into consideration. In theory, any number between 1 and (number of objects in the scene – 1) can be chosen. For our experiments, we chose to set this parameter to be equal 14 for the TAKE corpus (i.e., we include all other objects in the scene except for the target as negative samples) and 25 for the TAKE-CV corpus (in each scene, approximately 32 objects are recognised, so almost all of them are used as negative samples). In general, the experiments show that the more negative samples are chosen, the better results are achieved.

5.2 Results

All experiments were conducted using 10-fold cross validation. The results are averaged over 10 runs. We provide accuracy (in percentage) as it allows us to compare our results with the baseline from (Kennington, 2016).

5.2.1 Simple references

Both TAKE and TAKE-CV corpora contain simple references. As we described in the previous section, there are several parameters we can choose from when running RR. However, our experiments show that there is not much difference between various setups. Figure 5.1 presents the results from 16 different setups and runs. All of these experiments were conducted on TAKE corpus, with no uncertainty involved. In eight runs the colours are denoted using one hot encoding, in the other eight runs the Euclidean distance between RGB values of colours is used. We have also tried different types of regularisation (L1 and L2), use of ten most common bigrams in addition to all unigrams in feature sets and use of `SnowballStemmer` from NLTK. Accuracy for these runs varies from 88% to 92%. The best result is not surprisingly achieved in the setup with both bigram feature combinations and stemmer, and one-hot encoding for colours. In the table 5.2, we present the averaged results for each pair of eight runs. For instance, we can divide all runs after the regularisation used; then, we can calculate the mean accuracy for all the runs with L1 and L2 regularisation respectively. The presented numbers show that features are the most important setting to tune since the difference between the setups with one-hot encoding and with Euclidean distance is clearly the largest (1.2%).

Taking these results into consideration, we run the rest of the experiments with the minimal settings — no bigrams or stemmer is used. The regulari-

Setups	**Accuracy, %**
One-hot encoding	90.8
Euclidean distance	89.6
L1 regularisation	90.4
L2 regularisation	90
Unigrams only	90.1
Unigrams and bigrams	90.4
No stemmer	89.9
SnowballStemmer	90.5

Table 5.2: Averaged accuracy over setups, TAKE corpus

sation is chosen to be L1, as it provides better results. It also makes it easier to compare our results to the baseline, since the settings are alike.

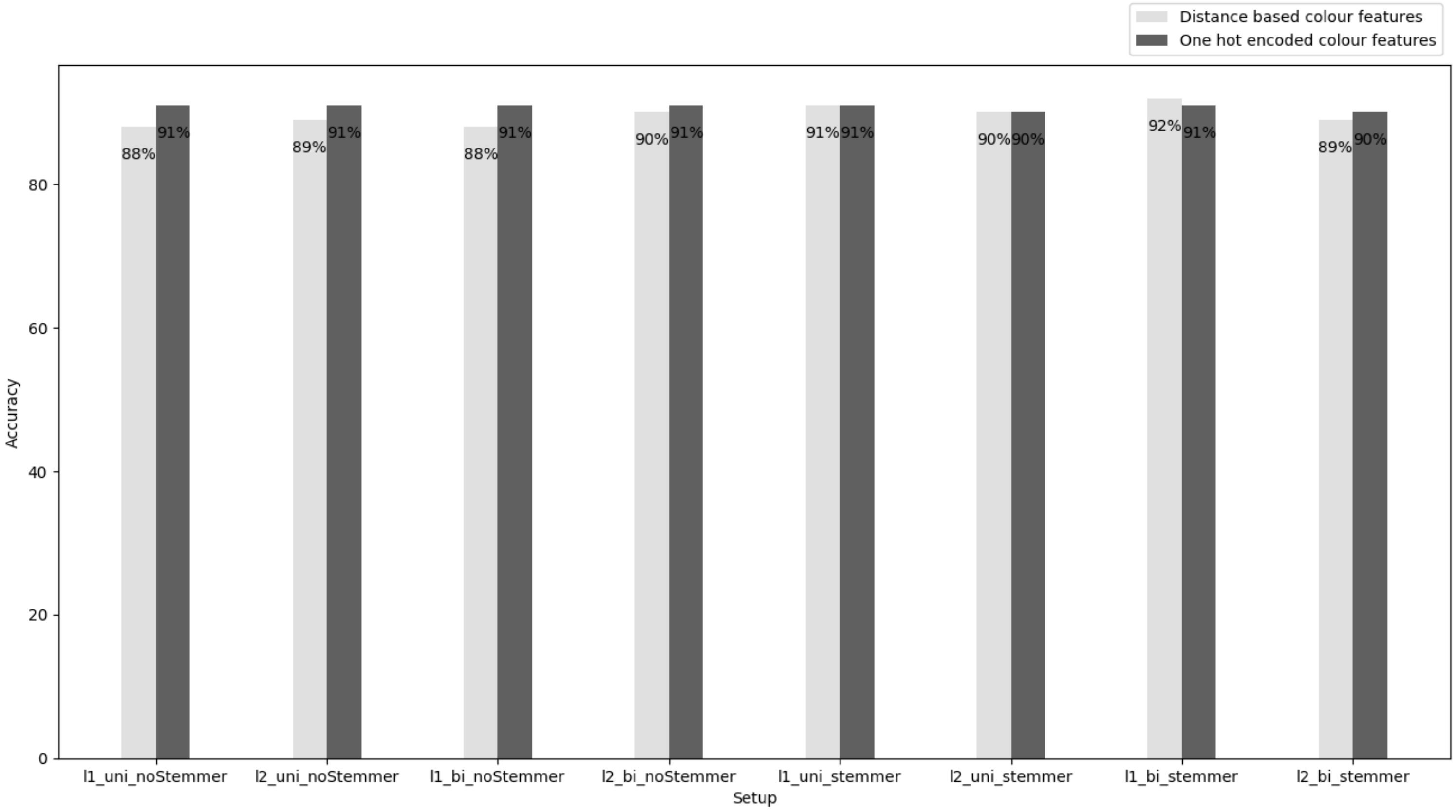


Figure 5.1: Results of different setups for TAKE corpus without uncertainty. *l1* and *l2* denote the regularisation used, and *uni* and *bi* indicate whether the bigrams were included into the feature set. The use of stemmer is also specified.

TAKE

As mentioned earlier, in the TAKE corpus both certainty and uncertainty can be used. The results from the setup with certainty are presented in figure 5.2[1]. For this evaluation, following features were used: one-hot encoded colours, shapes and position (one-hot encoded quadrant). The data is recorded by human transcribers. The random baseline of 7% is not shown.

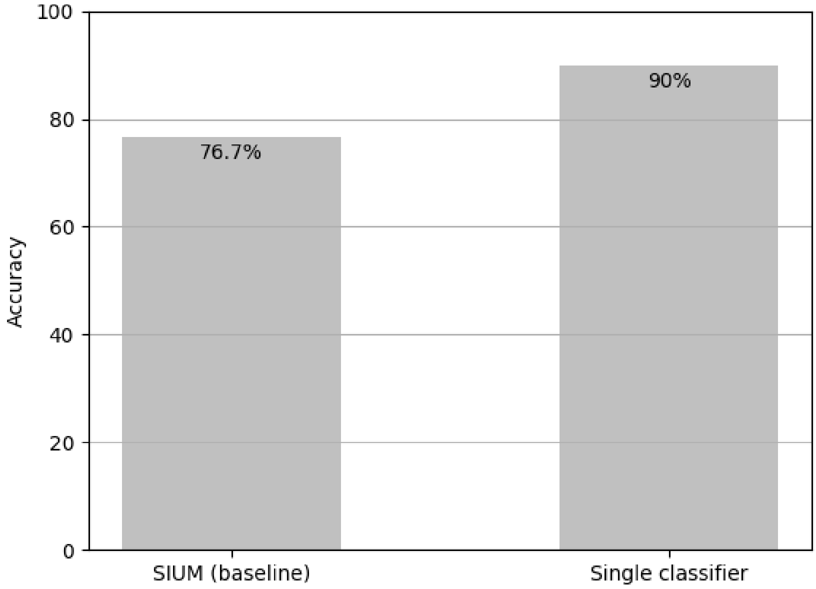


Figure 5.2: Evaluation results on the TAKE corpus, comparing the accuracy of two models, namely SIUM and the single classifier. The results are obtained using certain (predefined) visual features and linguistic features based on hand transcription

The uncertainty can also be introduced, as described in section 4.1. It can be incorporated in visual features only or the linguistic input can be noisy as well. The results for both setups are shown in figures 5.3 and 5.4 respectively. Since TAKE corpus with incorporated uncertainty was evaluated for both SIUM and WAC models, those results are also included as a baseline. The random baseline of 7% is not shown here either. The features used are RGB and HSV representations of colours read from the distorted pictures; number of edges to denote shape; x and y coordinates of the centroids to represent

[1] Hereinafter the results of our model are denoted as "Single classifier“

position; and the one-hot encoded information about skewness.

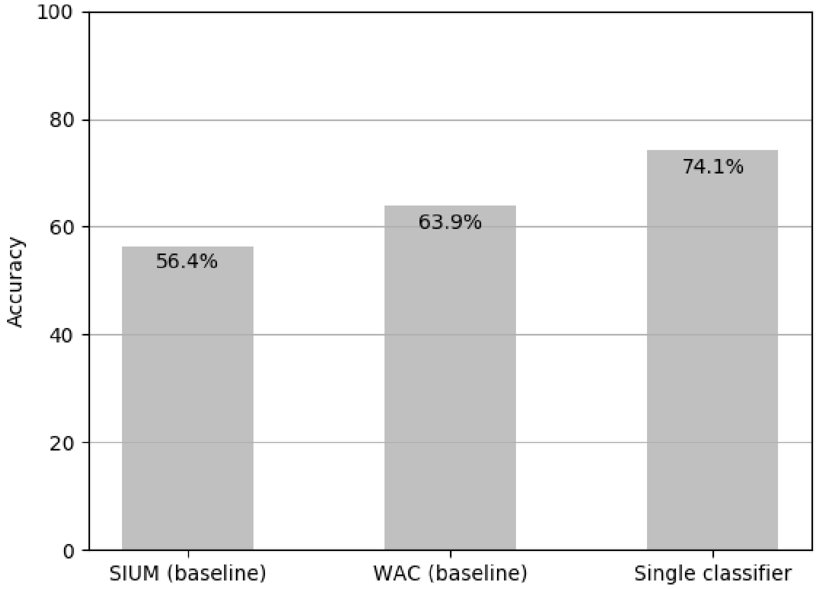


Figure 5.3: Evaluation results on the TAKE corpus, comparing the accuracy of three models, namely SIUM, WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on hand transcription

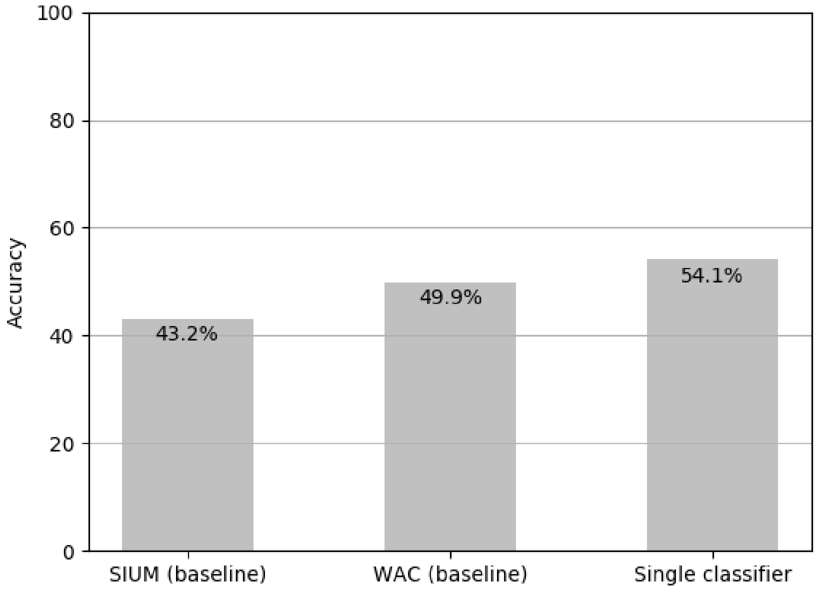


Figure 5.4: Evaluation results on the TAKE corpus, comparing the accuracy of three models, namely SIUM, WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on output from ASR

As seen from the plots, our model improves the results achieved by both SIUM and WAC. The noisy input affects our model as well, in a greater degree than the baseline models.

TAKE-CV

Using TAKE-CV, we conducted two experiments on simple references. Since the corpus is tagged, we have an opportunity to check whether the model takes into consideration "some notion of syntactic structure“ (Kennington, 2016). Figure 5.5 therefore shows the results of the model when all words in the utterance were used for RR, whereas in figure 5.6 the results of using only the words tagged with `t` – target tag – are presented. The random baseline of 3% is not shown. The features used are as described in table 5.1: RGB and HSV values for colour, number of edges for shape and x, y coordinates and Euclidean distance from the centre for position. Output from automatic speech recognition was used.

As seen from the table, taking into consideration some syntactic structure

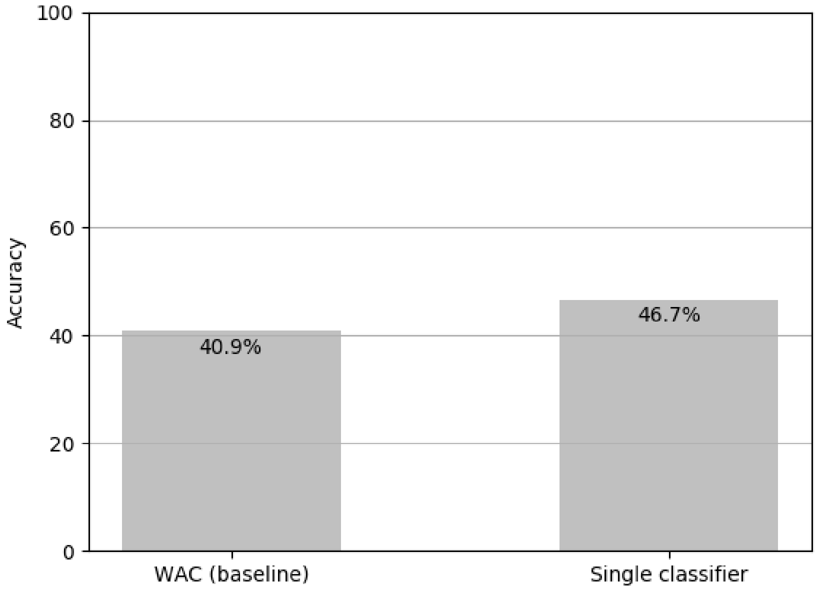


Figure 5.5: Evaluation results for simple references on the TAKE-CV corpus, comparing the accuracy of two models, namely WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on output from ASR. All words in each utterance are used

slightly improves the results. When using all words, our model performs better than WAC, but with the tagged words the results are somewhat worse. The possible explanation is using the single classifier: our model can determine itself which words are more important, so we do not need to choose only the words that are a part of the REs. We consider it to be a positive property, since the tagging can be skipped without much impact on the results. The results are still much worse than the results from the TAKE corpus. Two possible reasons for this is a smaller dataset for training (cf. 1000 episodes in TAKE and 870 in TAKE-CV), less precise data from computer vision and the much bigger number of the objects in the scene (15 in TAKE and 36, with 32 recognised in TAKE-CV).

Restart-incremental results

As mentioned in section 3.3, the model can also be run as restart-incremental. It means that the resolving process is restarted after each word in the given

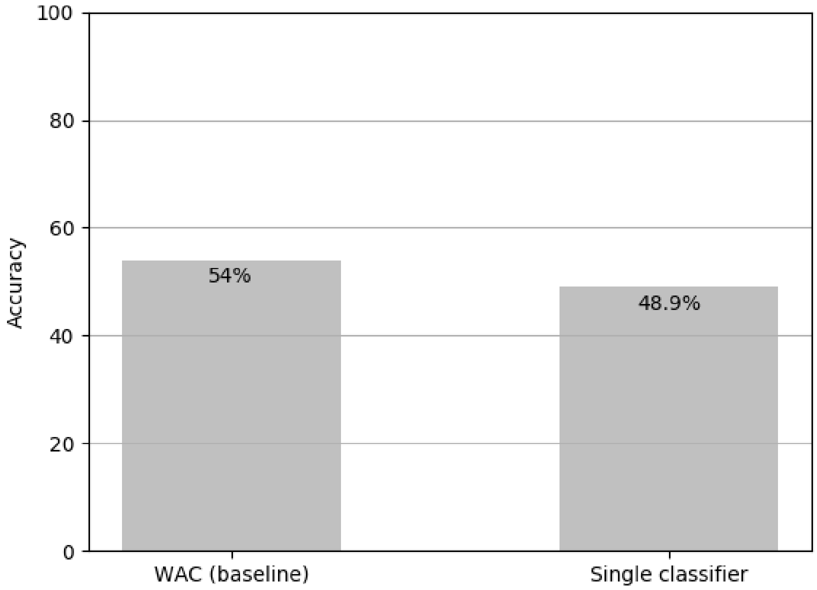


Figure 5.6: Evaluation results for simple references on the TAKE-CV corpus, comparing the accuracy of two models, namely WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on output from ASR. Only words tagged with `t` from each utterance are used

utterance. An example of incremental RR is presented in figures 5.7 and 5.8. The first one represents the scene itself. The target tile is highlighted in white, and the tile numbers are added for presentation. Probability distributions after each new word in the utterance are presented in figure 5.8. As we can see, the probability distribution after the first word *das* (the definite article) is almost uniform: it varies from 0.036 till 0.125. After the second word, *gelbe* (*yellow*), all yellow pieces receive considerably higher probability — tiles 0, 1, 2, 3 and 12. The third word, `T` (denotes the shape of a brick), reduces the probabilities of the other pieces even more. Finally, the fourth word *unten* (*bottom*) clearly enhances the only yellow T (tile 2) out of four identical pieces. The rest of the words confirm this distribution, and the probability of tile 2 being the referent given the RE *das gelbe T unten rechts in der Ecke* (*the yellow T at the bottom right in the corner*) is increased from 0.59 (after the fourth word) till 0.95 (after the last word).

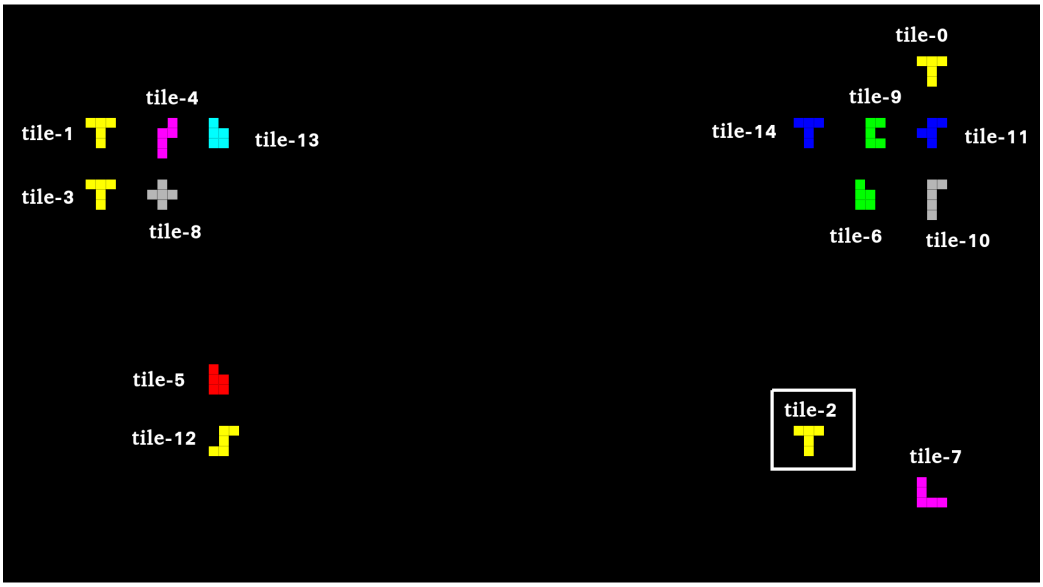
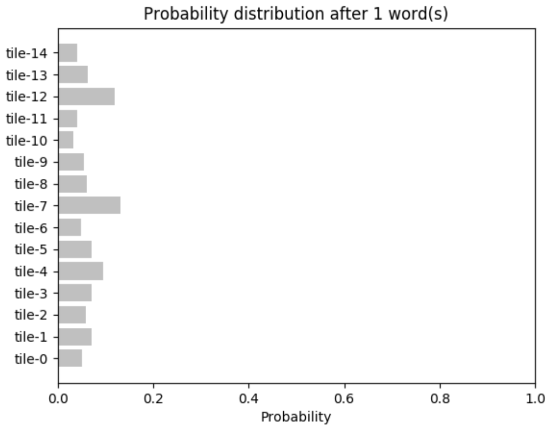
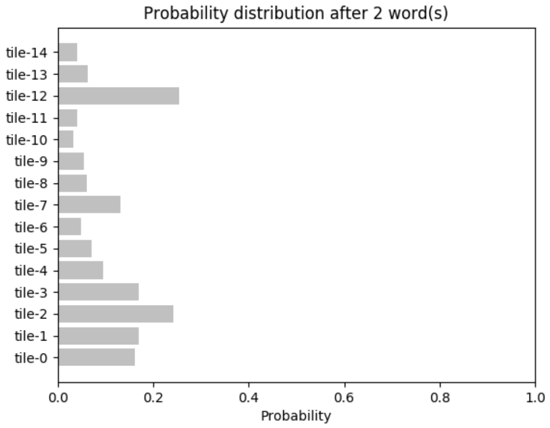


Figure 5.7: Example scene from TAKE. Selected tile highlighted in white. Tile numbers are added for presentation.

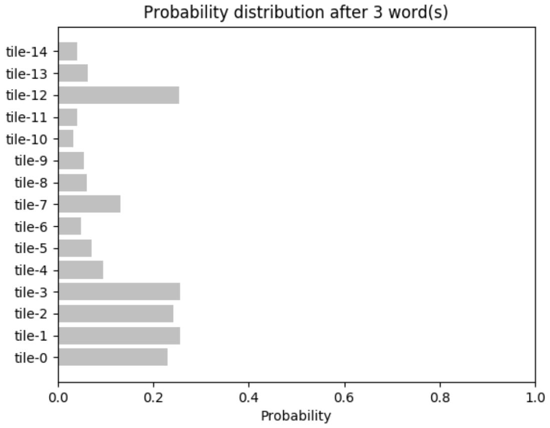
das



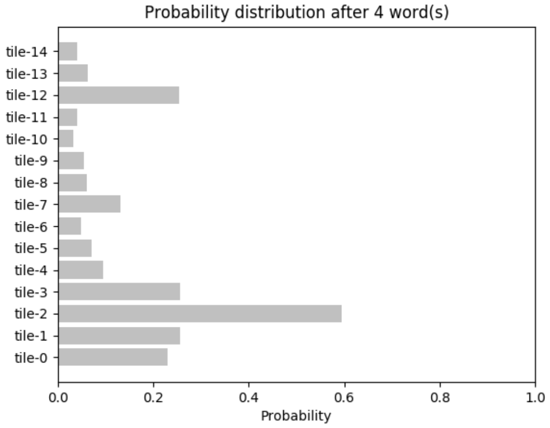
das gelbe



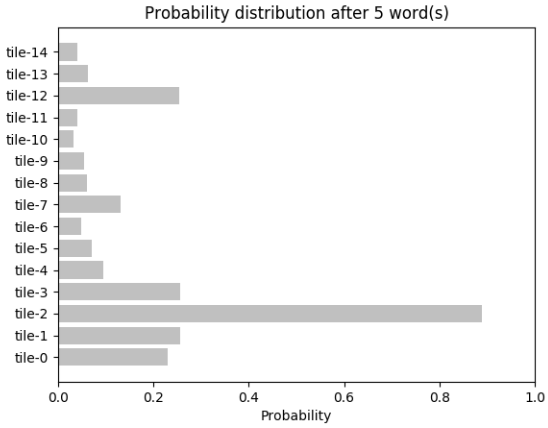
das gelbe T



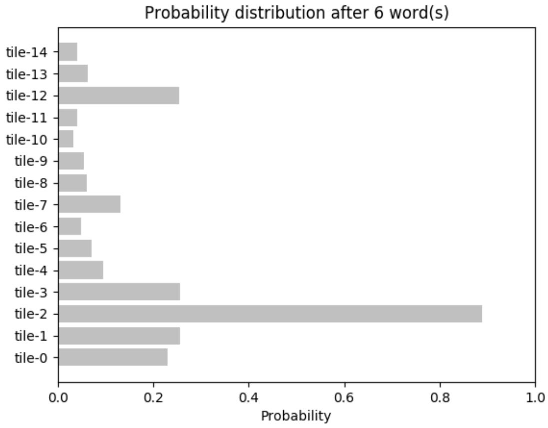
das gelbe T unten



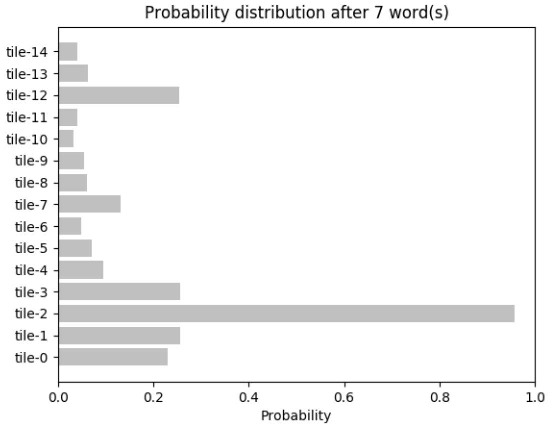
das gelbe T unten rechts



das gelbe T unten rechts in



das gelbe T unten rechts in der



das gelbe T unten rechts in der Ecke

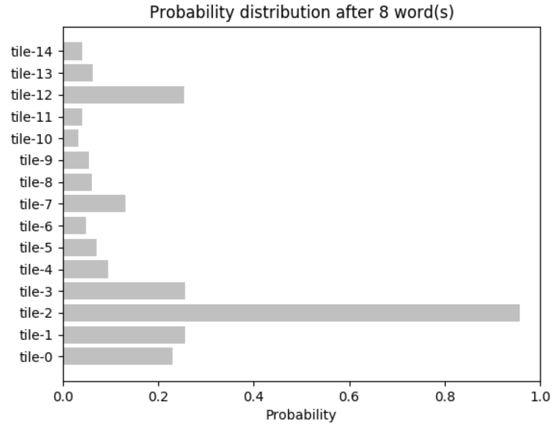


Figure 5.8: Probability distributions for incremental RR

In this example, despite quite a lot of ambiguity in the scene, the model manages to identify the target object quite fast. However, this is not always the case. For instance, the unique characteristics of the object can be mentioned only in the end of the utterance, and the argmax of the distribution can be changed only after the last word.

We have evaluated the restart-incremental model on the TAKE corpus without any uncertainty introduced. Figure 5.9 shows incremental results for the corpus: we can see that accuracy is quite good already after the half of the RE. However, the difference between the accuracy for the whole RE and for the half of the RE is still quite large and significant, so processing the whole RE is important.

5.2.2 Relational references

Relational references are represented only in TAKE-CV corpus. To resolve a relational reference, we need to take into consideration not only the RE for the target, but also the RE for the landmark and relational expression(s). In order to do that, we train two classifiers — the first to handle REs and the second one to classify relational expressions. For the first classifier, the same previously mentioned features are used (RGB and HSV for colour, number of edges for shape, and x, y coordinates and Euclidean distance from the centre for position). For the relational classifier, the features denoting a *relation* between two objects are used — Euclidean distance between the centroids of the objects, vertical and horizontal differences and also binary

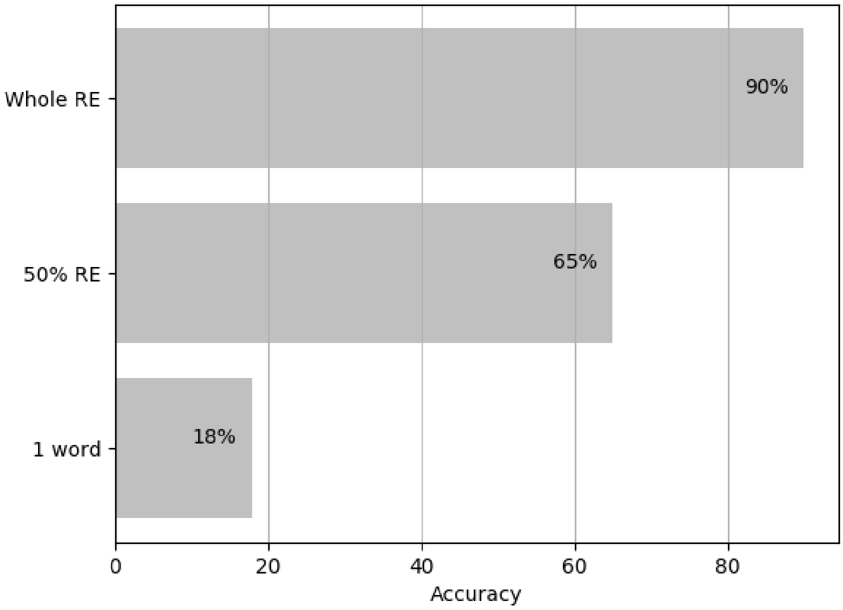


Figure 5.9: Incremental results: accuracy (TAKE, no uncertainty introduced)

features expressing the relationships above/under and left/right. Output from automatic speech recognition is used.

The results of the classification of relational references are presented in figure 5.10. As seen from the plot, our model does not produce the state-of-the-art results for this task and this part of the corpus. Our intuitive explanation was that the quite small size of the corpus (only 460 episodes with relational expressions) is the reason for that, but it seems that the learning curve (figure 5.11) reaches the plateau, so with the bigger corpus we won't get much better results. The most probable reason for these results then is the very noisy input, both visually and linguistically. It would be interesting to collect some data where more certain visual input is possible (like in TAKE) and check how well our model performs in that case. We will leave that for future work.

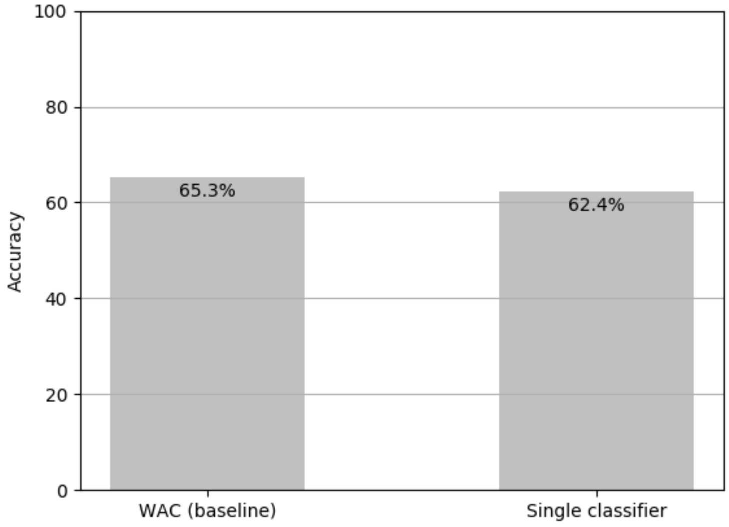


Figure 5.10: Evaluation results for simple and relational references on the TAKE-CV corpus, comparing the accuracy of two models, namely WAC and the single classifier. The results are obtained using visual features extracted from computer vision and linguistic features based on output from ASR

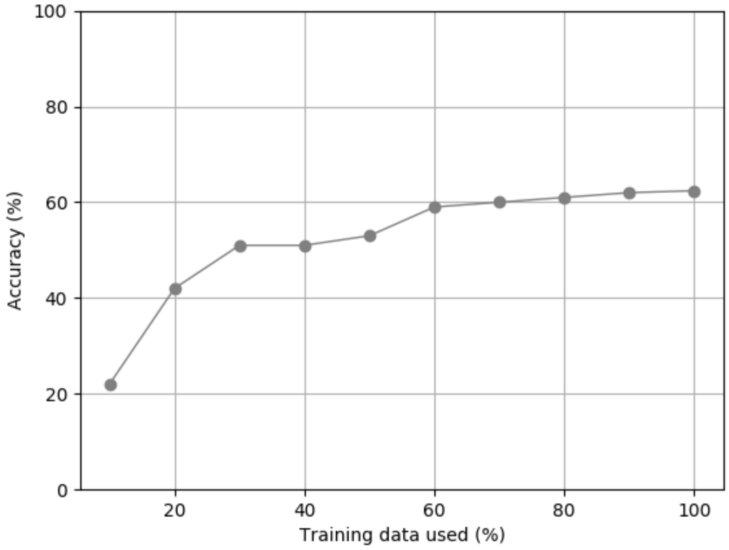


Figure 5.11: Learning curve for the resolution of relational references

5.3 Discussion

We have tested our model in several experiments. The results provided in the previous section show that our model performs well compared to the other models evaluated on the same datasets. For simple references, our model provides good results for the TAKE corpus, outperforming both SIUM and WAC.

The results for TAKE-CV are significantly better when using all words in the utterance and significantly worse when using only the tagged words. We performed a paired t-test for both models and both setups, where we paired the accuracy from each of the 10 folds. For the setup where all words are used, the result of the analysis gives us an absolute value of t-statistic of 2.24 and a p-value of 0.03. By assuming a significance level of 5% we can, based on this analysis, reject the null-hypothesis and conclude that our model performs significantly better. For another setup, where only tagged words are used, an absolute value of a calculated t-statistic is 2.24 and p-value is 0.04. It means that in this case our model performs significantly worse than WAC.

For relational references, the observed difference between empirical results is not statistically significant. We performed the same test as described before, and got a t-statistic of 1.49 and a p-value of 0.16. By assuming a significance level of 5% we can not reject the null-hypothesis. Therefore, the difference between the two models is not statistically significant. It would be interesting to compare the models on a different, bigger, dataset.

Analysing the numbers, we can observe that the model is relatively robust to noisy inputs, both visual and linguistic. Various visual features can be used, both read directly from a symbolic scene representation and from a real-world scene. Both manual transcription and ASR output can be used. The model performs respectably, although not always providing state-of-the-art results.

The model can operate incrementally, and for a dataset without uncertainty good results are achieved. It seems, however, that our model is most suitable for processing the whole REs.

Below we present some further analysis of some parts of the model.

5.3.1 Feature selection and most informative features

The most informative features vary for each setup and for each corpus. In setups with uncertainty in all visual features, the set of the most informative features contains both features that describe colours (r_rote)[1] and shapes (numedges_strich). Difference between the extracted features occurs when only some uncertainty is introduced. For instance, for one of the setups for TAKE corpus, when every visual feature is represented via one-hot encoding, the most informative features include mainly feature combinations with colours (see table 5.3). Only two of the most informative features denote shape — U_c and X_kreuz. The second feature combination is quite obvious: *kreuz* in German means *cross* in English, and the Pentomino piece *X* looks like a cross. The first one seems somewhat illogical, but the reason for this feature to be extracted is that the Pentomino piece with the shape *U*, as shown in figure 4.1, always occurred in the scenes rotated 90° clockwise and looked therefore like a *C*.

Feature name	**Feature weight**
yellow_gelbe	6.7504
red_rote	6.3048
gray_graue	6.1109
green_grüne	5.7953
U_c	5.3160
X_kreuz	5.3015
blue_blaue	5.0405
red_rot	4.9947
yellow_gelb	4.7144
blue_dunkelblaue	4.6776

Table 5.3: Most informative feature combinations extracted from a setup where colour features are denoted using one-hot encoding

When the setting is less certain, for instance, if the one-hot encoding of the colours is replaced with the Euclidean distance between colours, the set of the most informative features looks differently. More weight is given to the features that denote shape (see table 5.4). For some of these features, the link between the visual part and the linguistic part is obvious (e.g., German *strich* (*line*) and Pentomino piece *I*, *winkel* (*angle*) and pieces *V* and *L*), whereas for others it is not that clear. For instance, the binding between the piece *Y* and the word *halbe* (*half*) occurs because one participant in the experiment chose several times a *Y* tile and described it as "half of T" (*halbe T*):

[1] r stands for "red" in RGB

- *und dazu das halbe T oben links ... richtig*
  and then the half of T on the top left ... correct

- *das halbe T unten links ... das kleine halbe T*
  the half of T on the bottom left ... the small half of T

- *dann einmal das zweite halbe gelbe T oben links*
  then the yellow half of T on the top left

The *Y* tile was always rotated 45° counterclockwise in the scenes (compared to the tile in figure 4.1), so it reminded the participant of the *T* that was missing the half of the horizontal line.

Another feature `Z_s` is extracted because the *Z* tile was always mirrored and looked more like an *S*. The feature `V_l` is also informative as the participant were thinking about a capital *L* when describing a *V* tile.

Feature name	**Feature weight**
X_kreuz	8.7183
I_strich	7.2495
U_c	6.4033
T_t	5.1134
Y_halbe	4.5728
V_l	4.4399
V_winkel	3.8897
I_balken	3.4089
Z_s	2.4080
L_winkel	2.4080

Table 5.4: Most informative feature combinations extracted from a setup where colour features are encoded as Euclidean distance between colours

5.3.2 Error analysis

There are several types of errors made by our model. The main reason for errors is, first and foremost, noisy ASR input. If at lest half of the words in the utterance are not recognised correctly, the model has difficulties building a good reliable classifier and achieving high accuracy.

Moreover, the longer utterances tend to be more confusing for the model. Average length of the utterances in TAKE corpus, for instance, is 13.9 words, with standard deviation of 7.3. Average number of words in correctly resolved utterances is 11.9 (standard deviation 5.0). When the model is wrong, the average number of words is higher, 16.1, with standard deviation of 8.9 (see figure 5.12). The same pattern is observed in Kennington (2016). Some examples of such long utterances are presented below (English translation word for word is also given; the punctuation is provided for easier understanding):

- *ähm das grüne in der rechts unteren ecke ... soll ich jetzt einfach weiterreden oder ... ja richtig*
  uhm the green in the right lower corner ... I should now simply continue or ... yes correct

- *okay ähm also ich hab mir ein kreuz ausgesucht ... äh unten rechts das blaue kreuz ... das sieht jetzt gut aus ... soll ich das irgendwie bestätigen oder ... ja ist richtig*
  okay uhm so I have a cross selected ... uh down right the blue cross ... that looks good now ... I should confirm somehow or ... yes is correct

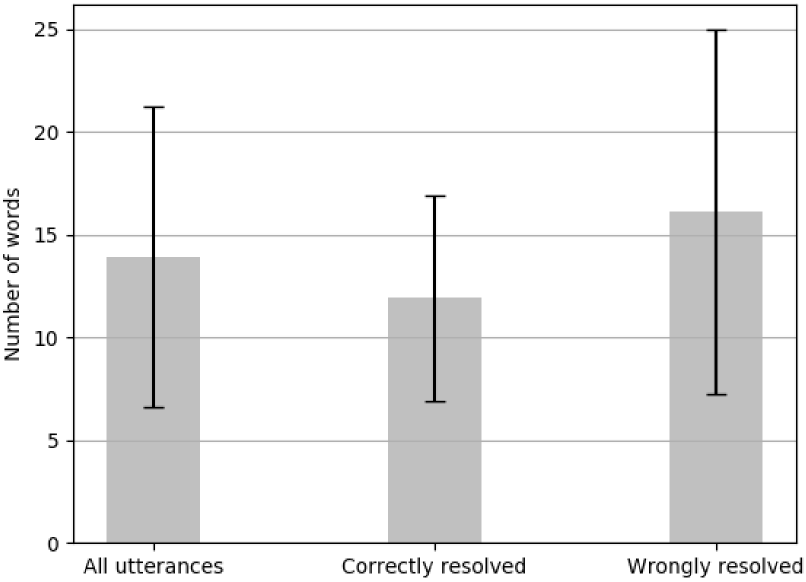


Figure 5.12: Average number of words and standard deviation for TAKE corpus

The model has also difficulties resolving the references when there is too much ambiguity in the scene. For instance, if the same quadrant contains two identical Pentomino pieces and the RE does not reflect that sufficiently, the errors can occur. The sample utterances that lead to such errors are presented below and the corresponding scenes are shown in figure 5.13:

- *die graue form rechts oben in der ecke ... die graue form rechts oben in der ecke ... der graue winkel rechts oben in der ecke ... richtig*
  the grey form right up in the corner ... the grey form right up in the corner ... the grey angle right up in the corner ... correct

- *das obere grüne c ... das oberste grüne c ... rechts oben in der ecke ... richtig*
  the upper green c ... the top green c ... right up in the corner ... correct

Some other errors occur because of corrections (especially if the speaker uses both words *left* and *right* in the same utterance, the model can be confused) and too short REs that mention only one of the visual features (for instance, only colour is mentioned and there are several distractors in the scene). The model can sometimes mix the colours that are quite similar (for instance, blue and cyan, or red and magenta pieces can be confused).

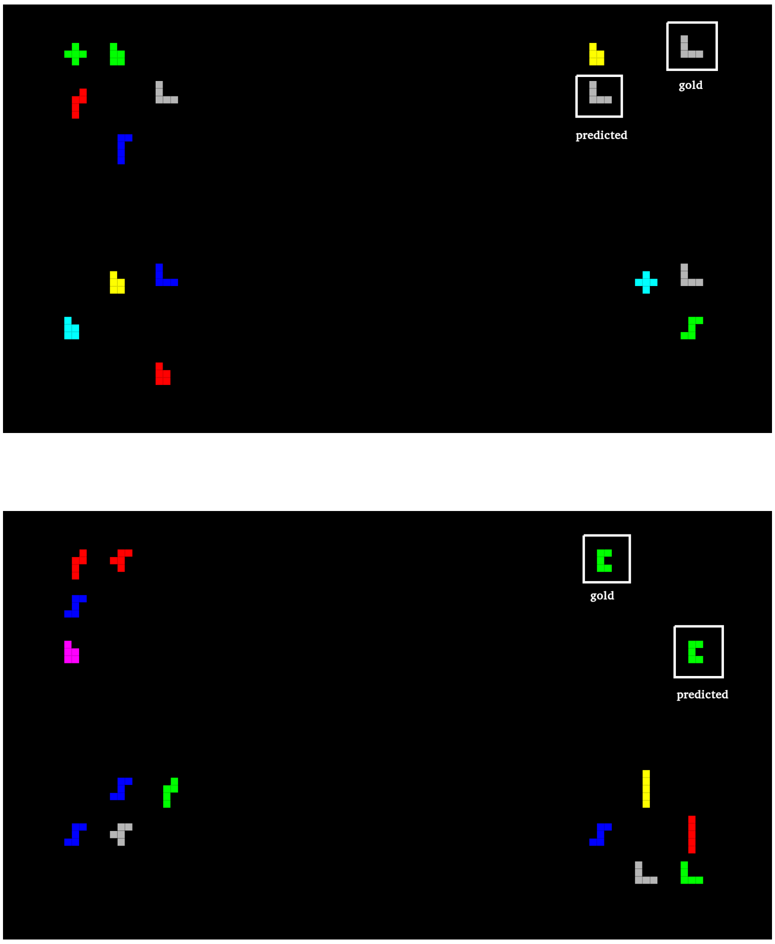


Figure 5.13: Typical ambiguous scenes from TAKE, where the model often makes mistakes. The target (gold) and the guess (predicted) are highlighted in white for presentation

Chapter 6

Conclusion & Future Work

In this thesis, we have explored the task of visual RR. We started with describing the task itself and the related challenges. We showed that visual RR is difficult because of dynamic partially-observable environment, noisy visual input and complex linguistic constructions. The task is also very important: visual RR is an essential part of any situated dialogue. REs are used both in everyday life and in task-oriented dialogues, and a reliable dialogue system should incorporate a RR component.

After giving an overview over related work in chapter 2, focusing on one rule-based model (GH-POWER) and two probabilistic models (generative SIUM and discriminative WAC), we presented our approach in chapter 3. The model we developed is a probabilistic model for the task of visual RR. Given the visual features of the objects and the RE, the model determines the target by providing a probability distribution over all candidate objects. Each object is represented by combination of visual and linguistic features which are described in section 3.2. With the help of these features, a single binary logistic regression classifier is built. If the REs are relational, one extra classifier is needed. It encodes the *relation* between two objects. The results of the classifiers are then combined.

This model has been evaluated on the two datasets described in chapter 4. Using the domain of Pentomino puzzle game and several different setups (virtual and real-world scenes with and without uncertainty in both visual and linguistic input), two corpora were created — TAKE and TAKE-CV. Our experiments were conducted on both corpora, and the results were compared to the baseline SIUM and WAC models.

The experimental design and the results are described in chapter 5. We used accuracy to compare the performance of our model and the baseline models. The analysis of the results shows that the model performs respectably in all settings. For TAKE corpus, the model provides better results than both SIUM and WAC. For TAKE-CV, the results for tagged simple REs are somewhat worse for our model, whereas the results for relational REs are not significantly different.

The model satisfies all the requirements we set in chapter 1:

- The model can handle uncertain visual features
- The model can handle noisy linguistic input
- The model can handle simple and relational references
- The model is robust even with little data available

The biggest challenge for the model is noisy linguistic input. The difference in the performance of the model on the same corpus with hand transcriptions compared to ASR output is significant (20% in accuracy for the TAKE corpus). It seems that an improved automatic speech recognition system would provide much better results of our model.

We concluded the thesis with error analysis of our results, where we described the most common situations when the model gets confused.

There are many approaches to visual RR. In this thesis, we developed one more probabilistic model which provides good results, in some cases outperforming the available baselines.

6.1 Future Work

There are several aspects of our model that could be improved and developed further. In section 3.3, we described some of the possible extensions to our model — including cardinality, salience information, using more sophisticated linguistic features, incorporating other modalities, such as gaze and gesture.

Another challenge is modelling negations. Although the fully negated REs are unusual (e.g., it is much more natural to refer to an object as *a red ball* than *not so green ball*), in the relational references the negations can be quite common (e.g., *a red ball under the brown table but not near a green ball*). Since determining the scope of negation is a very complicated task, the RR model could be combined with a negation handling system, for instance (Lapponi, 2012).

Moreover, it would be useful and interesting to evaluate our system on several more corpora. It could help to explain the observed differences in performance between TAKE and TAKE-CV.

The model developed during the work on this thesis is a Python script that works independently. Since RR is essential in situated dialogues, it would be interesting to implement the model as a component of a spoken dialogue system framework.

Finally, RR is closely connected to a subtask of natural language generation — generation of REs. It could be possible to use the developed classifier for this task.

References

Barbara Abbott. *Reference.* Oxford Surveys in Semantics and Pragmatics, 2010.

Rieks op den Akker, Marjan Hospers, Erna Kroezen, Anton Nijholt, and Danny Lie. A rule-based reference resolution method for dutch discourse analysis. In *Proceedings of Symposium on Reference Resolution in Natural Language Processing*, pages 59–66, 2002.

Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* O'Reilly Media, Inc., 2009.

Tom Brøndsted. Reference problems in chameleon. In *ESCA Tutorial and Research Workshop on Interactive Dialogue in Multi-Modal Systems*, pages 133–136, 1999.

Joyce Yue Chai, Zahar Prasov, and Shaolin Qu. Cognitive principles in robust multimodal interpretation. *Journal of Artificial Intelligene Research*, 27: 55–83, 2006.

Xiaowu Chen and Nan Xu. A multimodal reference resolution approach in virtual environment. In *International Conference on Virtual Systems and Multimedia*, pages 11–20, 2006.

Delphine Dahan, Michael K Tanenhaus, and Craig G Chambers. Accent and reference resolution in spoken-language comprehension. *Journal of Memory and Language*, 47(2):292–314, 2002.

Deborah A Dahl. Focusing and reference resolution in pundit. In *Proceedings of the workshop on Strategic computing natural language*, pages 114–126, 1986.

Cristian Danescu-Niculescu-Mizil and Lillian Lee. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, 2011.

Pascal Denis. *New learning models for robust reference resolution.* PhD thesis, The University of Texas at Austin, 2007.

Kotaro Funakoshi, Mikio Nakano, Takenobu Tokunaga, and Ryu Iida. A unified probabilistic approach to referring expressions. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 237–246, 2012.

Jana Götze. *Talk the walk: Empirical studies and data-driven methods for geographical natural language applications.* PhD thesis, KTH Royal Institute of Technology, 2016.

Jeanette K Gundel. Reference and accessibility from a Givenness Hierarchy perspective. *International Review of Pragmatics*, 2(2):148–168, 2010.

Jeanette K Gundel, Nancy Hedberg, and Ron Zacharski. Cognitive status and the form of referring expressions in discourse. *Language*, 69:274–307, 1993.

Jeanette K Gundel, Mamadou Bassene, Bryan Gordon, Linda Humnick, and Amel Khalfaoui. Testing predictions of the Givenness Hierarchy framework: A crosslinguistic investigation. *Journal of Pragmatics*, 42(7):1770–1785, 2010.

Allan Hanbury. The taming of the hue, saturation and brightness colour space. In *Proceedings of the 7th Computer Vision Winter Workshop*, 2002.

Ryu Iida, Shumpei Kobayashi, and Takenobu Tokunaga. Incorporating extralinguistic information into reference resolution in collaborative task dialogue. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1259–1267, 2010.

Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligene Research*, 4: 237–285, 1996.

Andrew Kehler. Cognitive status and form of reference in multimodal human-computer interaction. In *Association for the Advancement of Artificial Intelligence*, pages 685–690, 2000.

Casey Kennington. *Incrementally resolving references in order to identify visually present objects in a situated dialogue setting*. PhD thesis, Bielefeld University, 2016.

Casey Kennington and David Schlangen. Simple learning and compositional application of perceptually grounded word meanings for incremental reference resolution. In *Proceedings of the Conference for the Association for Computational Linguistics*, pages 292–301, 2015.

Casey Kennington and David Schlangen. A simple generative model of incremental reference resolution for situated dialogue. *Computer Speech & Language*, 41:43–67, 2017.

Casey Kennington, Spyridon Kousidis, and David Schlangen. Interpreting situated dialogue utterances: an update model that uses speech, gaze, and gesture information. pages 173–182, 2013.

Casey Kennington, Spyros Kousidis, and David Schlangen. Situated incremental natural language understanding using a multimodal, linguistically-driven update model. In *COLING*, pages 1803–1812, 2014.

Casey Kennington, Livia Dia, and David Schlangen. A discriminative model for perceptually-grounded incremental reference resolution. In *Proceedings of the 11th International Conference on Computational Semantics*, 2015a.

Casey Kennington, Ryu Iida, Takunobu Tokunaga, and David Schlangen. Incrementally tracking reference in human/human dialogue using linguistic and extra-linguistic information. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics–Human Language Technologies*, 2015b.

Casey Kennington, Maria Soledad Lopez Gambino, and David Schlangen. Real-world reference game using the words-as-classifiers model of reference resolution. In *Proceedings of the 19th Workshop on the Semantics and Pragmatics of Dialogue*, 2015c.

Geert-Jan M Kruijff, John D Kelleher, and Nick Hawes. Information fusion for visual reference resolution in dynamic situated dialogue. In *International Tutorial and Research Workshop on Perception and Interactive Technologies for Speech-Based Systems*, pages 117–128, 2006.

Geert-Jan M Kruijff, Hendrik Zender, Patric Jensfelt, and Henrik I Christensen. Situated dialogue and spatial organization: What, where… and why? *International Journal of Advanced Robotic Systems*, 4(1):125–138, 2007.

Emanuele Lapponi. Why not! sequence labeling the scope of negation using dependency features. master thesis, University of Oslo, 2012.

Heeyoung Lee, Angel Chang, Yves Peirsman, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4): 885–916, 2013.

Stanley Lemeshow and David W. Hosmer. Logistic regression. In *Wiley Encyclopedia of Clinical Trials. 1.* John Wiley and Sons, Inc., 2008.

Cynthia Matuszek, Nicholas FitzGerald, Luke Zettlemoyer, Liefeng Bo, and Dieter Fox. A joint model of language and perception for grounded attribute learning. 2012.

Ruslan Mitkov. Robust pronoun resolution with limited knowledge. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 869–875, 1998.

Andrew Y Ng. Feature selection, L1 vs. L2 regularization, and rotational invariance. In *Proceedings of the 21st international conference on Machine learning*, pages 78–85, 2004.

Luis Pineda and Gabriela Garza. A model for multimodal reference resolution. *Computational Linguistics*, 26(2):139–193, 2000.

R. A. Nonenmacher. Pentomino, 2017. URL `https://en.wikipedia.org/wiki/Pentomino#/media/File:Pentomino_Naming_Conventions.svg`. [Online; accessed April 11, 2017].

Niels Schutte, John Kelleher, and Brian Mac Namee. Visual salience and reference resolution in situated dialogues: A corpus-based evaluation. In *Association for the Advancement of Artificial Intelligence Symposium on Dialog with Robots*, pages 109–114, 2010.

Maria Staudte and Matthew W Crocker. Visual attention in spoken human-robot interaction. In *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, pages 77–84, 2009.

Sandra Williams, Mark Harvey, Keith Preston, et al. Rule-based reference resolution for unrestricted text using part-of-speech tagging and noun phrase parsing. In *Proceedings of the International Colloquium on Discourse Anaphora and Anaphora Resolution*, pages 441–456, 1996.

Tom Williams, Saurav Acharya, Stephanie Schreitter, and Matthias Scheutz. Situated open world reference resolution for human-robot dialogue. In *The 11th ACM/IEEE International Conference on Human Robot Interaction*, pages 311–318, 2016.

Sina Zarrieß, Julian Hough, Casey Kennington, Ramesh Manuvinakurike, David DeVault, Raquel Fernández, and David Schlangen. Pentoref: A corpus of spoken references in task-oriented dialogues. In *10th edition of the Language Resources and Evaluation Conference*, pages 125–131, 2016.