Visual Reference Resolution: A Machine Learning Approach

Natalia Smirnova

Department of Informatics
University of Oslo

20th March 2017

UNIVERSITY OF OSLO

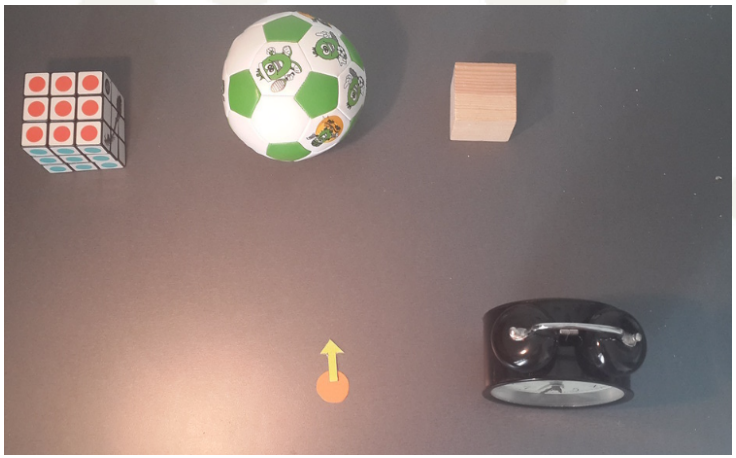
Overview

Visual Reference Resolution

- The task of resolving referring expressions to a referent, the entity to which they are intended to refer[1]
- Determine objects and their visual features
- Process the referring expression
- Analyse non-linguistic information, such as gaze and deixis
- Reconstruct the intended connection between words and given world

Visual Reference Resolution: Some Terms

- Referring expression (RE): “the cube to the left of the ball”
- Referent, or Target: the cube
- Landmark: the ball
- Distractors: all other objects
- Locative expression: “to the left of“



Visual Reference Resolution: Challenges

- The environment is only partially observable
- It is also dynamic and changes over time
- Visual processing is not flawless
- Often we not only need to resolve the reference to one particular object, but also to other objects and relation between them

Visual Reference Resolution and Dialogue: Challenges

- Referring expressions in dialogues are generated and processed incrementally
- There can be a lot of elliptical constructions
- In human speech there are quite a lot of corrections which can be difficult to resolve
- During the dialogue act, a lot of interaction feedback is provided — different types of confirmation, interest and so on
- There is a lot of non-verbal information included, like gaze, deixis, nodding, shaking head

Existing Approaches

- Model of Givenness Hierarchy
- Model of Spatial Prepositions and Spatial References
- Words As Classifiers

Existing approaches: Model of Givenness Hierarchy

- Givenness hierarchy (GH) is a scale which represents six possible kinds of information status that referring expressions can signal

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

Existing approaches: Model of Givenness Hierarchy

- Givenness hierarchy itself does not make claims about how a piece of information might acquire a particular status
- It means that the coding protocol is needed
- Such a coding protocol and the algorithm for resolving RE (called *GH-POWER*) is provided by T.Williams[2]

Example of a coding protocol:

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.

Existing approaches: Model of Givenness Hierarchy

GH-POWER:

- The algorithm first parses the utterance and generates a dependency graph which is then converted into a tree
- A set of "status cue" mappings for each referenced entity (e.g., $\{X \rightarrow \textit{familiar}, Y \rightarrow \textit{infocus}\}$) is extracted from the tree
- Then, *GH-POWER* populates and sorts four data structures, *FOC* (in focus), *ACT* (activated), *FAM* (familiar) and *LTM* (long-term memory) using some simple rules (e.g., *FOC* is populated by main clause subjects of clause $n-1$ and syntactic focus of the same clause)
- Lastly, the references in a given clause are resolved

Existing approaches: Model of Givenness Hierarchy

- All bindings for all variables and tiers in a given clause are stored in a table
- A set of candidate hypotheses is created
- The most likely binding between the tier and the variable is searched
- All hypotheses below a chosen threshold are removed
- Bindings are updated

Existing approaches: Model of Givenness Hierarchy

Benefits:

- The approach can handle open world assumption
- The approach can handle uncertainty
- References to hypothetical entities are also handled ("Imagine a box" resolves to an imaginary box)

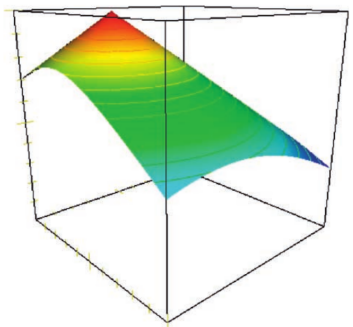
Existing approaches: Model of Givenness Hierarchy

Limitations:

- Resolving plural references is not implemented
- The model has problems with non-discrete entities (parts or regions of an object)
- The model does not incorporate gaze and deixis

Existing approaches: Models of Spatial Prepositions and Spatial References

- The approach is developed by J.Kelleher et al.[3]
- It models two type of prepositions — topological (e.g., *at, in, on*) and projective (e.g., *to the right of, to the left of*) — with the help of potential fields
- Basic idea: for each preposition, a potential field can be created (different prepositions — different equations to create such a field)
- However, these fields do not account for the influence of other objects in the scene



Existing approaches: Models of Spatial Prepositions and Spatial References

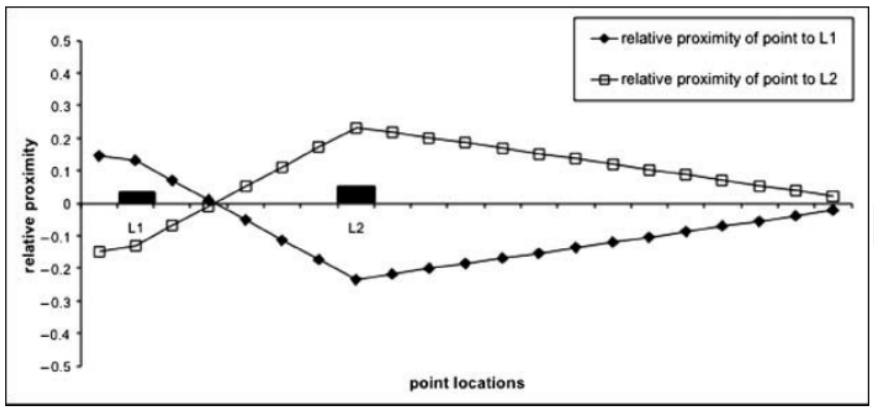
- So, if absolute potential fields are overlayed, relative potential fields which represent the semantics of the prepositions can be created
- At first, absolute proximity between each point and each landmark in a scene is computed
- Then, for each landmark for each point in the scene the absolute proximity of this landmark is compared to the absolute proximity of all other landmarks.
- The relative proximity field for that landmark at that point then is its absolute proximity field minus the highest absolute proximity field for any other landmark at that point.
- Then the overall proximal area for a given landmark is an area where its relative proximity field is above zero.

Existing approaches: Models of Spatial Prepositions and Spatial References

- The idea here is that the other landmark with the highest absolute proximity is acting in competition with the selected landmark.
- If the absolute proximity of the other landmark is higher that of the selected landmark, the relative proximity of the later one will be negative
- But when the absolute proximity of the other landmark is lower, than the selected landmark gets a high relative proximity score

Existing approaches: Models of Spatial Prepositions and Spatial References

- In the figure, any point where the relative proximity for one particular landmark is above the zero line represents a point which is proximal to that landmark, rather than to the other landmark
- We can see that proximal area is dependent on the landmark's position relative to the other landmark and to the boundaries, as well as on the size of the landmark (here, the size represents salience)



Existing approaches: Models of Spatial Prepositions and Spatial References

Limitations:

- The model assumes that the visual features of the objects are already resolved
- Only quite simple prepositions are modeled; more complex static dynamic prepositions, like *between*, *among*, *within*, *beside*, *around* could be addressed

Existing approaches: Words As Classifiers

- Developed by C.Kennington et al. [1], [4], [5]
- The model computes a probability distribution over candidate objects, given a referring expression
- Formally: reference resolution is a funciton f_{rr} that, given a representation U of the RE and a representation W of the world, returns I^*, the identifier of one of the objects in the world that is the referent of the RE
- When using a stochastic model, we in fact get a distribution over a set of candidate solution, so I^* then is the *argmax*

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

Existing approaches: Words As Classifiers

- The model consists of two parts: modelling word meanings from perceptual features and composition of these word meanings
- Two types of words: those picking up properties of a single object ("red") and those picking up relations between two objects ("under")
- The composition gives the probability distribution over candidate objects
- Two types of composition: simple references ("the green book") and relational references ("the green book near the red ball")

Existing approaches: Words As Classifiers

Word meanings:

- For each word from the corpus of REs, a binary logistic regression classifier is trained
- This classifier takes a representation of a candidate object via visual features (x) and returns a probability p_w for it being a good fit to the word (w is the weight vector that is learned and σ is the logistic function)

$$p_w(x) = \sigma(w^T x + b)$$

- So, the meaning of a word can be seen as the classifier itself, a function of an object to a probability ($[\![w]\!]$ is the meaning of w, x is of the type of feature given by f_{obj}, the function computing a feature representation for a given object):

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

Existing approaches: Words As Classifiers

- These classifiers are trained using a corpus of RE, visual representations of the objects in the world and annotations of the referent
- For positive samples, each word in a RE is paired with the features of the referent
- For negative samples, each word is paired with the features of a randomly picked object in the same scene
- If the word describes a relation between two objects, its meaning is presented as a vector of features of a *pair* of objects, for example binary features `lower than` or `to the left of`

Existing approaches: Words As Classifiers

Composition:

- For every word in a given referring expression, a respective word classifier is applied to all candidate objects and then normalised
- Finally, the contributions of constituent words are averaged, assuming that each word contributes equally

Existing approaches: Words As Classifiers

Composition:

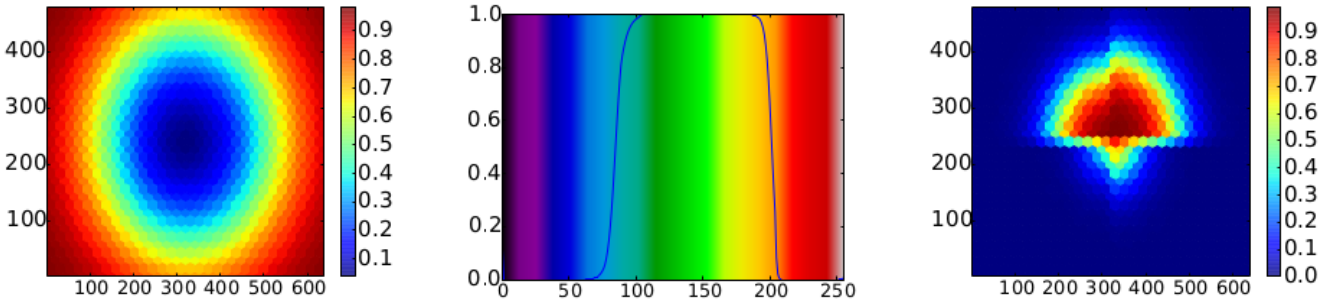
- Relational references are handled in a similar way
- The meaning of the phrase is the function of the meaning of the constituent parts (the simple references for target and landmark, the relation expression and the construction)
- The relation expression is trained as a simple classifier
- The relational construction is the combination of evidence:
  - the target constituent contributes $P(I_t|w_1, ..., w_k)$
  - the landmark constituent contributes $P(I_l|w_1^{'}, ..., w_m^{'})$
  - the relation expression contributes $P(R_1, R_2|r)$

$$P(R_1|w_1, ..., w_k, r, w_1^{'}, ..., w_m^{'}) = \sum_{R_2} \sum_{I_l} \sum_{I_t} P(R_1, R_2|r) *$$
$$P(I_l|w_1^{'}, ..., w_m^{'}) * P(I_t|w_1, ..., w_k) * P(R_1|I_t) * P(R_2|I_l)$$

- The last two factors force the pairs being evaluated by the relation expression consist of objects evaluated by target and landmark expression, respectively

Existing approaches: Words As Classifiers

- Features: RGB representations for the colour and features such as skewness, number of edges etc. for the shapes
- *ecke (corner), grün (green), über (above)*



Existing approaches: Words As Classifiers

Corpus	Features	Accuracy
TAKE (simple references)[5]	RGB, HSV, skewness, number of edges, coordinates	0.63
TAKE (simple references)[4]	colours, shapes, rules-based position	0.76
TAKE-CV (simple and relational references)[1]	RGB, skewness, number of edges, orientation, coordinates	0.65

Existing approaches: Words As Classifiers

Limitations:

- All words contribute equally to the final result
- Negation or generalised quantifiers are not handled
- It is not clear how more descriptive REs can be resolved

Existing approaches: Words As Classifiers

However:

- the model is robust and has good results
- we have access to the same corpus

So,

- this model is used as a starting point and a baseline for our approach

Data

- Corpus of referring expressions in Pentomino puzzle domain recorded at Bielefeld University in 2014 [6]
- Wizard-of-Oz study
- One game board, 15 random selected Pentomino puzzle pieces (12 shapes, 6 colours) grouped in the four corners of the screen

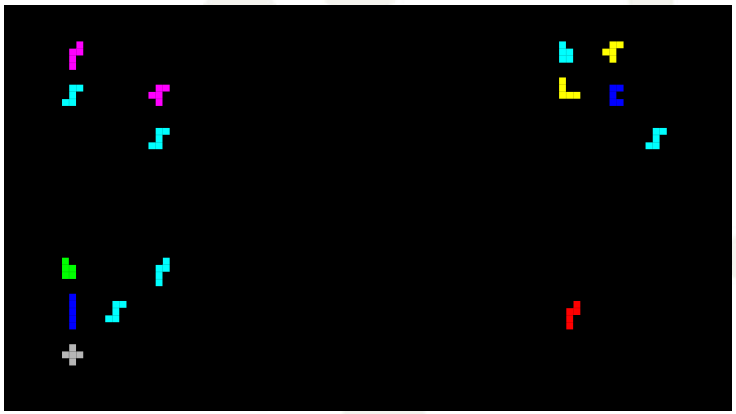


Figure: Example Pentomino board

Data: Task Description

- The participant sees a game board on the screen
- The participant chooses any piece and describes it
- The system (wizard) has to select the piece with a visual outline
- The participant has to confirm or reject the choice
- When the correct piece is selected, a new game board appears

Data: Quick Facts

- Measurements: audio, video, eye and arm movements
- Language: German
- Participants: 8, all university students, all but one native speakers
- Episodes: 1049
- Transcription: expert transcribers

Example

Different people, different types of utterances:

- "Das rosa Symbol rechts oben" — "the pink symbol on the top right"
- "Dann aus der Gruppierung da unten links einmal das lila L das auf kopf steht ... ja, richtig" — "Then from the group down there to the left the only purple L on the head ... yes, correct"
- "Unten links das grüne ... okay" — "the green one to the left ... okay"
- "Und dazu dann wir haben ja diese fünf Zeichen da oben und ich möchte genau das in der Mitte haben ... richtig" — "And then we have these five symbols up there and I want exactly the one in the middle ... correct"

Data: Corpus

So, the data recorded and provided is as following:

- For each participant, we have a TextGrid file with transcription
- For each scene, we have an xml file where each piece has attributes *colour, shape, position*
- We also have a picture of each scene
- For each episode, we have a txt file with a string encoding the selected piece
- We also have audio and video for all experiments

Approach

- We want to calculate a probability of resolving a referring expression *ref* as object *o* given this referring expression and the world *W*
- Formally:

$$P(resolution(ref) = o | ref, W) = \frac{P(fit(o, ref)}{\Sigma_{o'} P(fit(o', ref))}$$

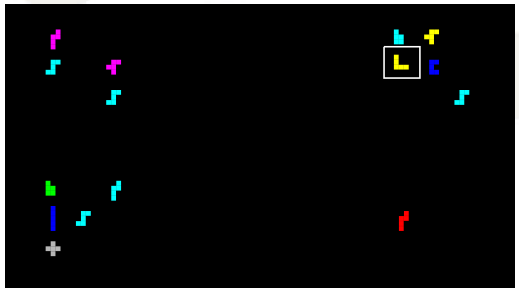
- We need to calculate this fit function
- To do it, we train a classifier on the given data
- The approach is similar to Words As Classifiers, but instead of training multiple classifiers for each word in the dictionary, we train a single classifier which uses the so-called cross-features

Approach

- Cross-features are a combination of visual and linguistic information
- To create them, we need a vocabulary and a list of predefined visual features, in our case colours, shapes and position
- Examples of such features: `red_kreuz, left_symbolen`
- Each feature gets a positive value if both visual and linguistic information is true for a given piece and a given referring expression, and 0 otherwise

Example

- Visual features: red, yellow, left, right, c, l
- Vocabulary: das, rote, gelbe, l, unten, oben, links, rechts
- Referring expression: "das gelbe L oben rechts"
- Actual piece: highlightened piece below
- Feature values: `red_das:0`, `red_rote:0`, `red_gelbe:0`, `...`, `yellow_das:1`, `yellow_rote:0`, `yellow_gelbe:1` and so on, for this example 6 visual features * 8 words = 48 features in one feature vector for one piece



Approach

- All in all, we get 23 visual features * 377 words = 8671 features
- For each scene, the one selected piece is a positive sample, and the rest 14 pieces are a set of negative samples
- We train a logistic regression classifier which gives us as output the probability distribution over each object in the scene being the target object given the referring expression

Why cross-features?

- When using cross-features, we do not need to average the contributions of constituent words of a given RE
- I.e. we do not assume that each word has equal influence on the probability of an object being a god fit for the referring expression; linguistically, that would be wrong (function words)
- Instead, 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, will lead to better results

Why cross-features?

- We can integrate features that capture the combination of visual features with different n-grams, or particular syntactic relations
- We have only one classifier to train; although the feature vector is very big, it is also very sparse, so this is not a problem

Approach: Other tuning parameters

- Several experiments were conducted with the classifier including different tuning parameters
  1. L1 and L2 regularisation were used
     - L1 regularisation uses a penalty term which encourages the *sum* of the absolute values of the parameters to be small
     - L2 regularisation encourages the *sum of the squares* of the parameters to be small[7]
  2. Different colour encoding
     - One hot encoding
     - RGB-values and Euclidean distance between colours
  3. Stemming and lemmatisation
     - Snowball stemmer from NLTK — on and off
     - German TextBlob lemmatiser — on and off
  4. Unigrams and bigrams
     - Only unigrams as linguistic part of all features
     - Unigrams and bigrams combined

Results

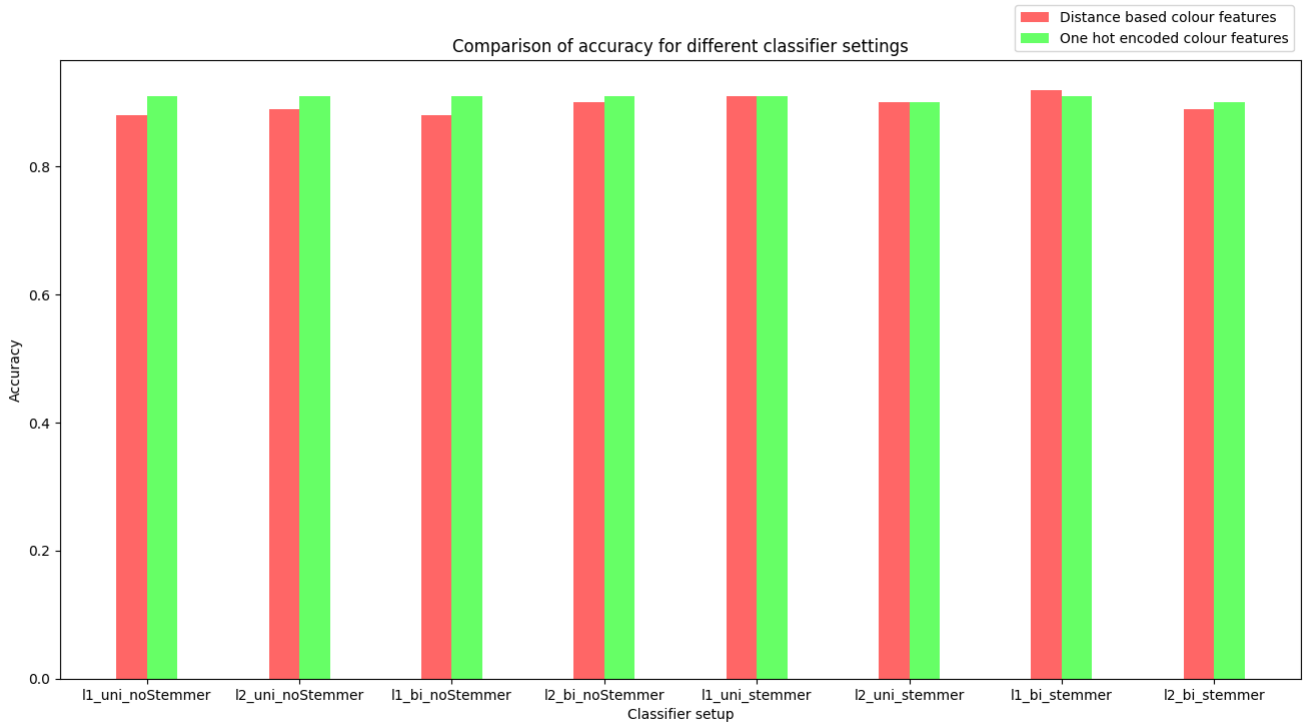
For a basic setup, using L1 regularisation, only unigrams and no stemmer or lemmatiser (10 fold cross-validation):

	One hot encoding	RGB
Precision	0.91	0.89
Recall	0.89	0.87
F1	0.88	0.86
Accuracy	0.91	0.88

Results

Corpus	Features	Accuracy
TAKE (simple references)[5]	RGB, HSV, skewness, number of edges, coordinates	0.63
TAKE (simple references)[4]	colours, shapes, rules-based position	0.76
TAKE-CV (simple and relational references)[1]	RGB, skewness, number of edges, orientation, coordinates	0.65
TAKE (simple references)	colours, shapes, rules-based position	0.91
TAKE (simple references)	RGB, shapes, rules-based position	0.88

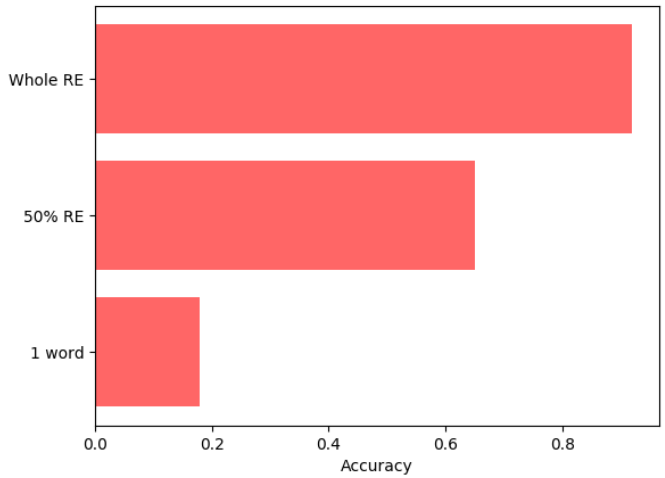
Results



UNIVERSITY OF OSLO

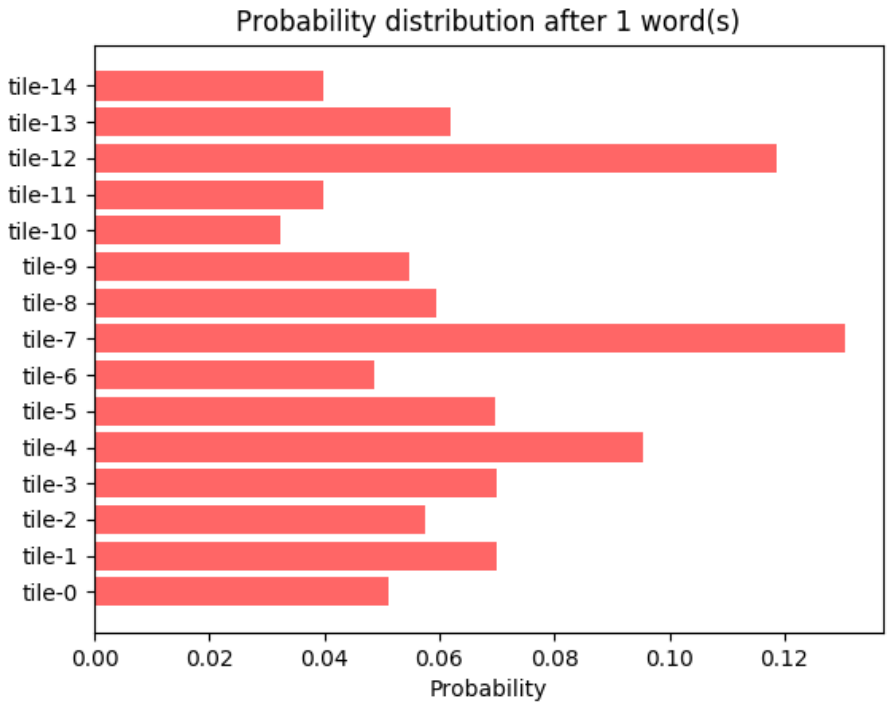
Incremental reference resolution

- One of the advantages of words as classifiers approach is that it is applied incrementally, so we do not need to wait for the whole referring expression to be uttered to start processing and get first results
- Our approach has the same benefits



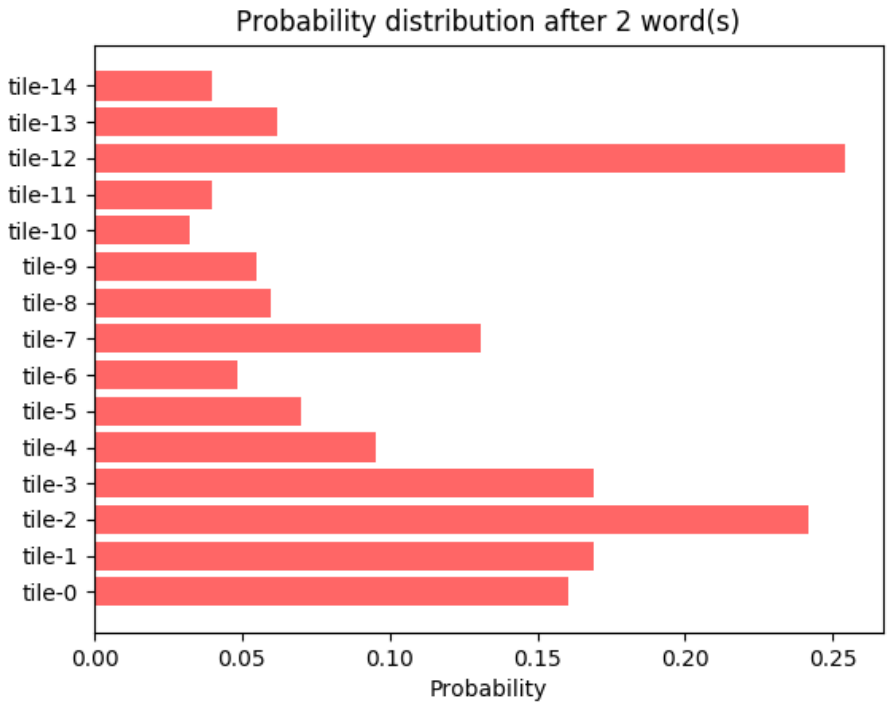
Example of resolving one referring expression

das



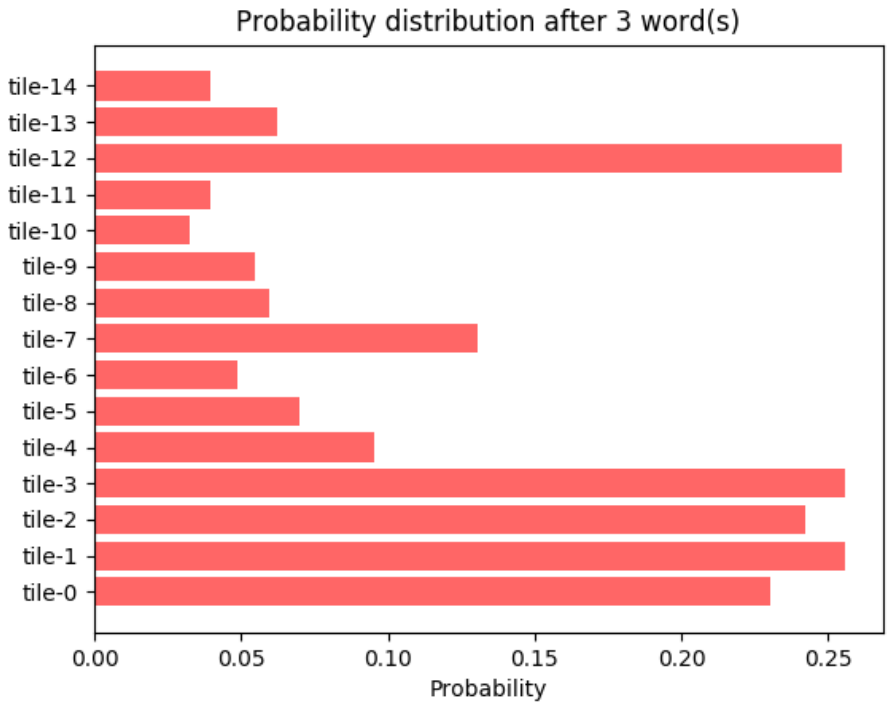
Example of resolving one referring expression

das gelbe



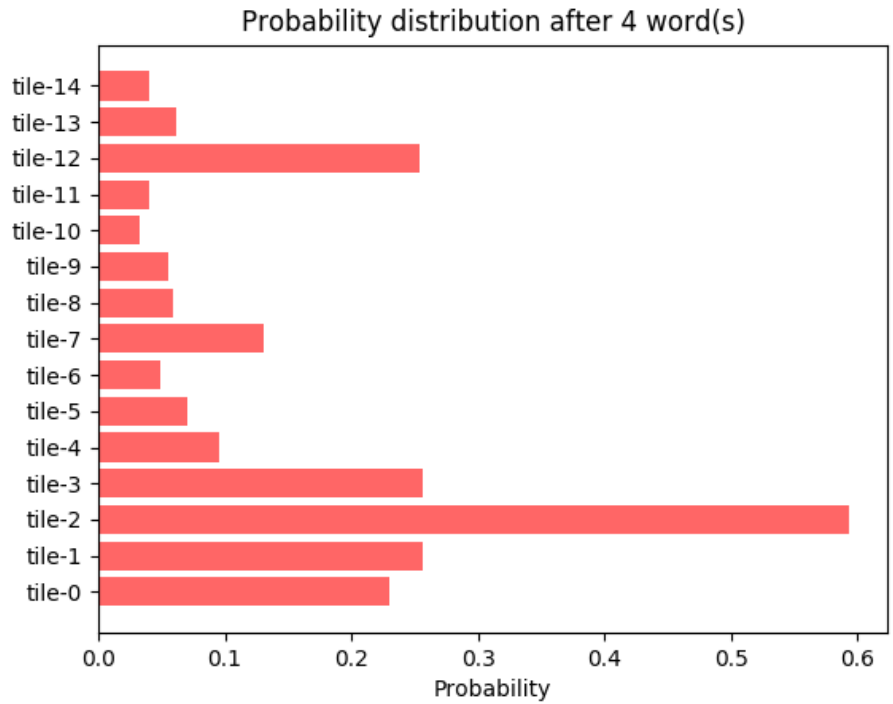
Example of resolving one referring expression

das gelbe T



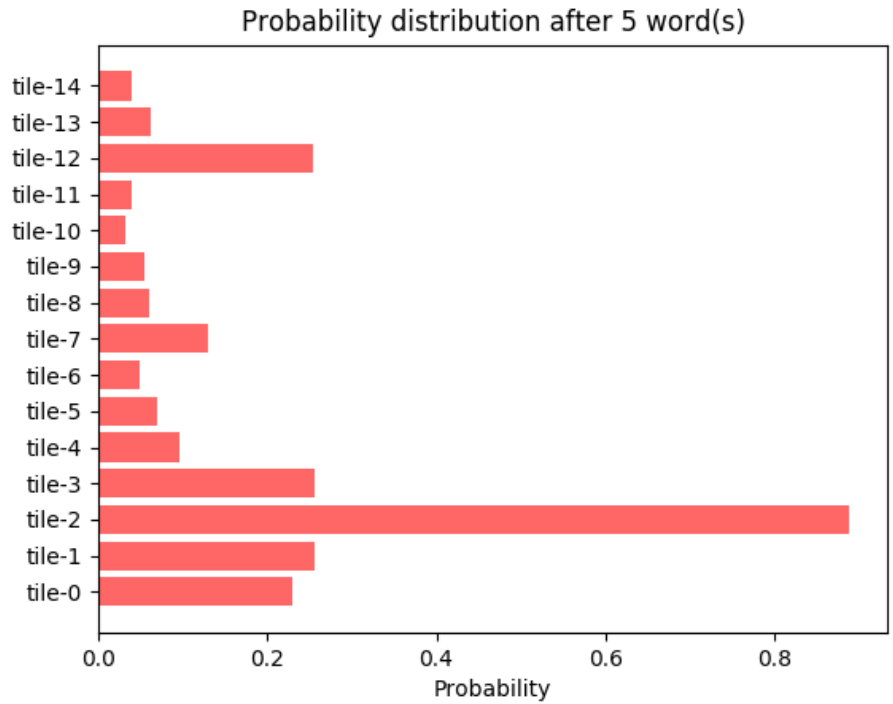
Example of resolving one referring expression

das gelbe T unten



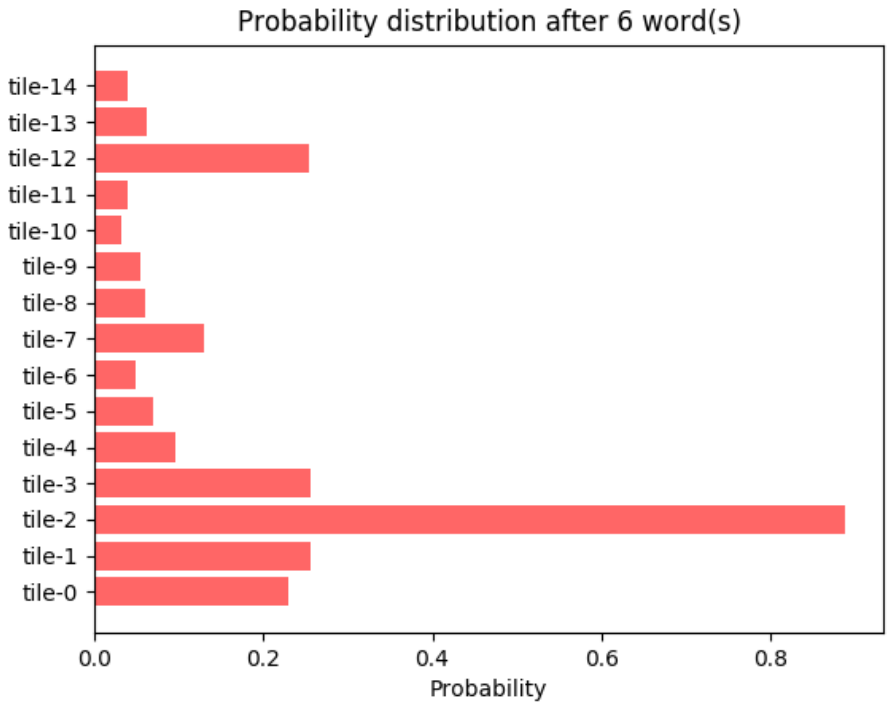
Example of resolving one referring expression

das gelbe T unten rechts



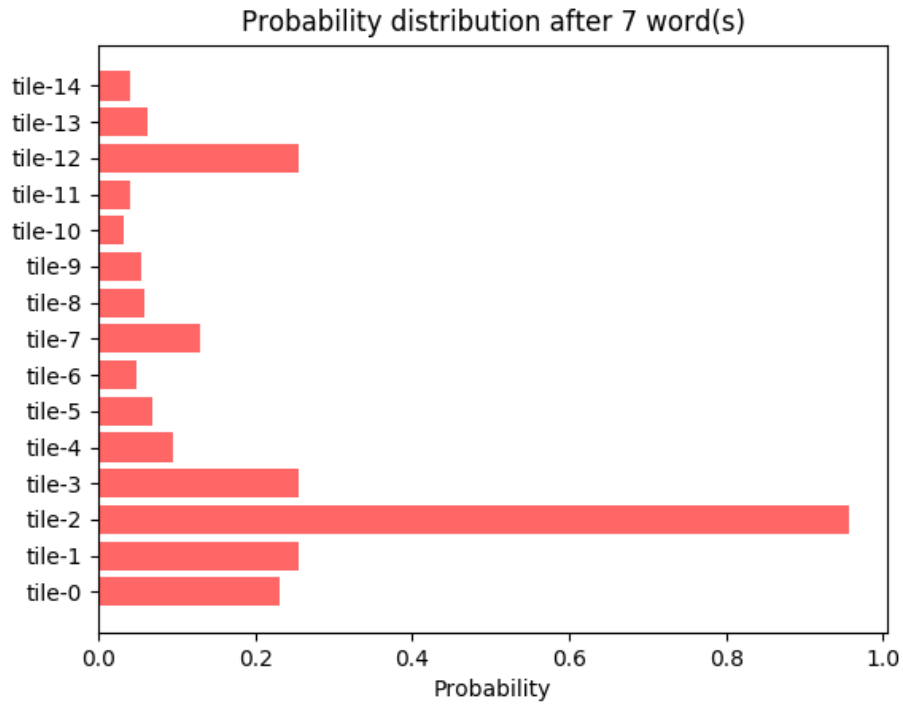
Example of resolving one referring expression

das gelbe T unten rechts in



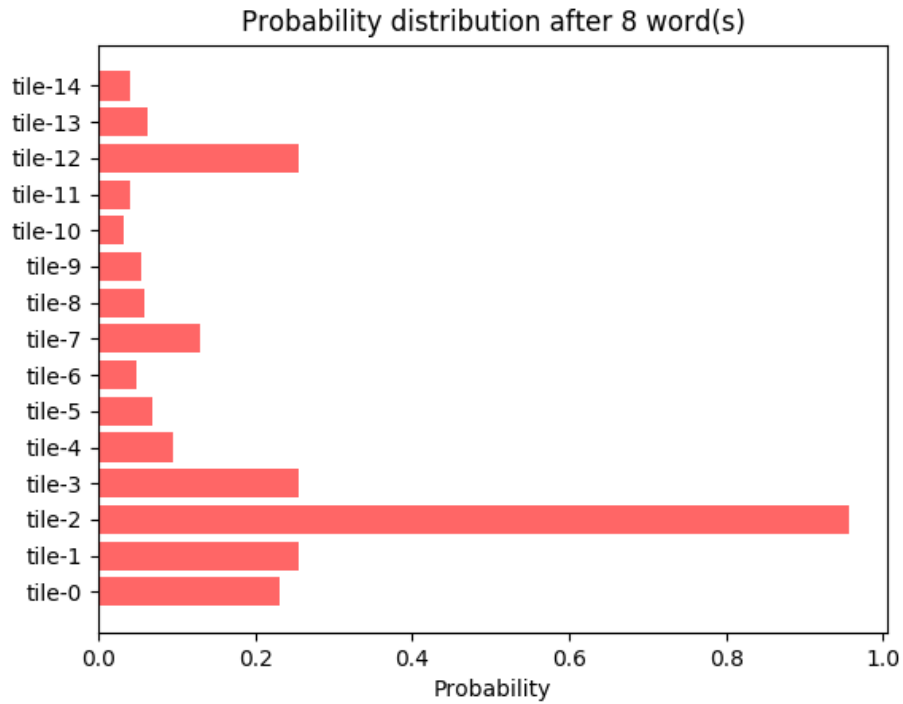
Example of resolving one referring expression

das gelbe T unten rechts in der



Example of resolving one referring expression

das gelbe T unten rechts in der Ecke



Results

And just a couple of words about feature selection:

- We used the module SelectFromModel from scikit-learn to select features
- We ran the classifier with different penalty and different colour encoding to see if feature selection results were different

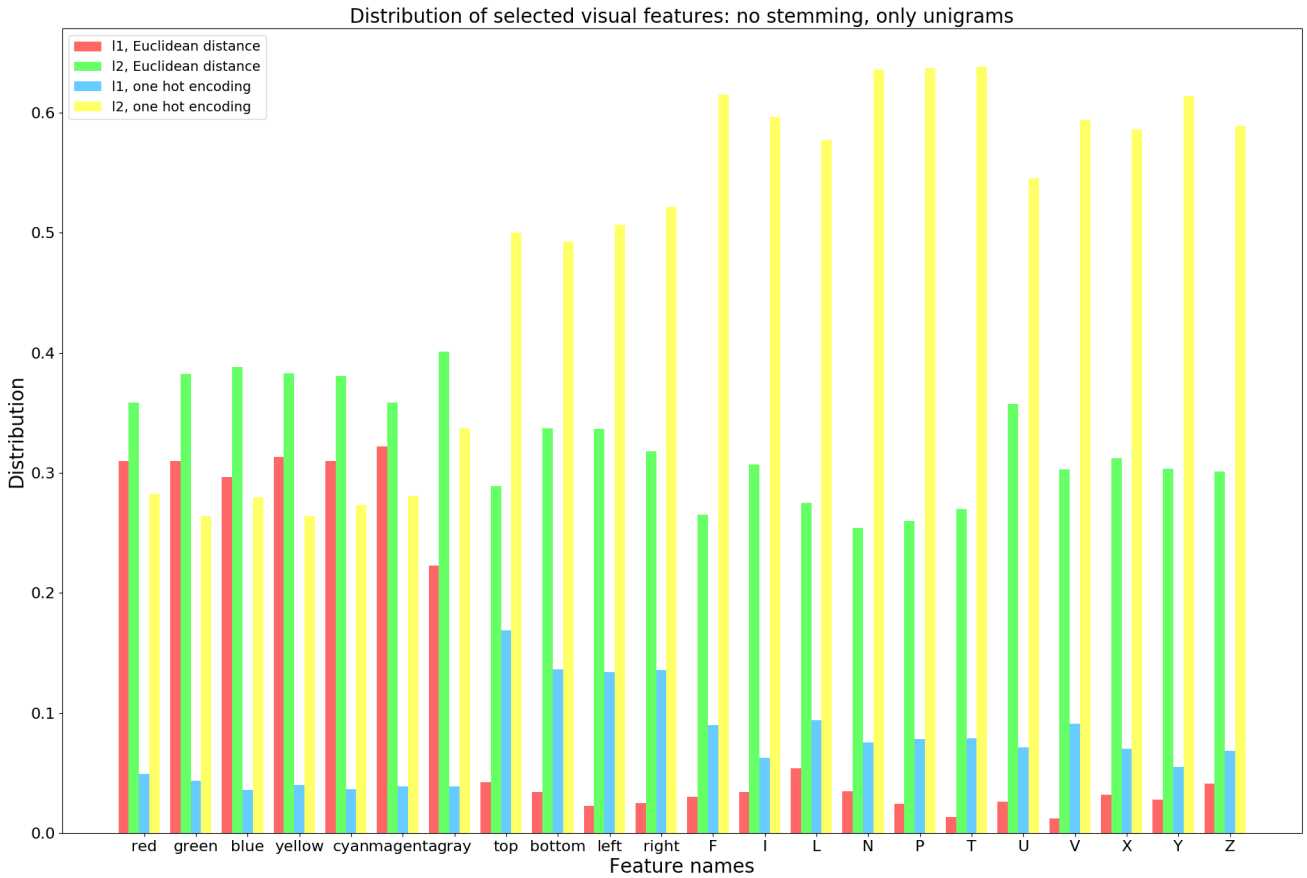
Setup	Number of features
L1, Euclidean distance	920
L2, Euclidean distance	1958
L1, One hot encoding	426
L2, One hot encoding	2486

Table: Number of extracted features

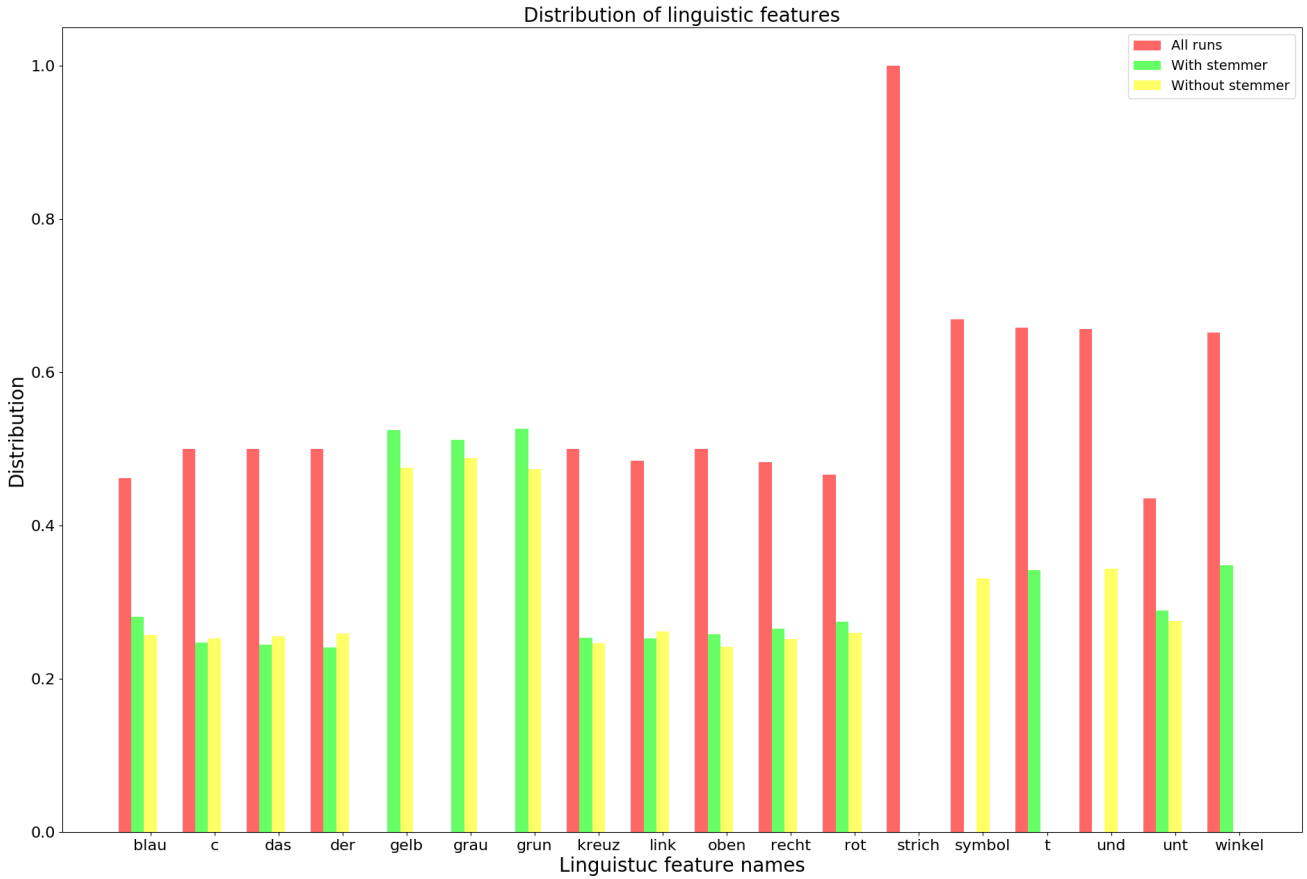
Results

- For visual parts of each feature, we looked at what features were most frequently selected, and what the most informative features were
- The same for linguistic parts, but we concentrated on opposition "runs with stemming – runs without stemming"

Results



Results



Results

For one hot encoding:

L1		L2	
Feature name	Feature weight	Feature name	Feature weight
yellow_gelbe	6.7504	yellow_gelbe	4.6561
red_rote	6.3048	gray_graue	4.3537
gray_graue	6.1109	red_rote	4.2318
green_grüne	5.7953	green_grüne	4.1148
U_c	5.3160	X_kreuz	3.4711
X_kreuz	5.3015	blue_blaue	3.3890
blue_blaue	5.0405	red_rot	3.0634
red_rot	4.9947	U_c	3.0084
yellow_gelb	4.7144	T_t	2.8611
blue_dunkelblaue	4.6776	magenta_lila	2.7579

For Euclidean distance:

L1		L2	
Feature name	Feature weight	Feature name	Feature weight
X_kreuz	8.7183	top_oben	0.7904
I_strich	7.2495	right_rechts	0.7888
U_c	6.4033	bottom_unten	0.7698
T_t	5.1134	left_links	0.7307
Y_halbe	4.5728	X_kreuz	0.3215
V_l	4.4399	U_c	0.2802
V_winkel	3.8897	T_t	0.2650
I_balken	3.4089	I_strich	0.1892
Z_s	2.4080	U_das	0.1752
L_winkel	2.4080	X_das	0.1504

Conclusion

- The presented approach of a single classifier trained on cross-features yields high accuracy and precision scores
- The model can be trained on a quite small amount of data which is easy to annotate: we only need the scene, visual features, referring expressions and the selected piece
- The model will be further extended by adding analysis of relational referring expressions

C. Kennington and D. 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 (ACL)*, pp. 292–301, 2015.

T. Williams, S. Acharya, S. Schreitter, and M. Scheutz, "Situated open world reference resolution for human-robot dialogue," in *The Eleventh ACM/IEEE International Conference on Human Robot Interaction*, pp. 311–318, IEEE Press, 2016.

J. D. Kelleher and F. J. Costello, "Applying computational models of spatial prepositions to visually situated dialog," *Computational Linguistics*, vol. 35, no. 2, pp. 271–306, 2009.

C. Kennington and D. Schlangen, "A simple generative model of incremental reference resolution for situated dialogue," *Computer Speech & Language*, vol. 41, pp. 43–67, 2017.

C. Kennington, L. Dia, and D. Schlangen, "A discriminative model for perceptually-grounded incremental reference resolution," in *Proceedings of the 11th International Conference on Computational Semantics (IWCS) 2015*, 2015.

S. Zarrieß, J. Hough, C. Kennington, R. Manuvinakurike, D. DeVault, R. Fernández, and D. Schlangen, "Pentoref: A corpus of spoken references in task-oriented dialogues," in *10th edition of the Language Resources and Evaluation Conference*, 2016.

A. Y. Ng, "Feature selection, l 1 vs. l 2 regularization, and rotational invariance," in *Proceedings of the twenty-first international conference on Machine learning*, p. 78, ACM, 2004.