Reference Resolution for Pedestrian Wayfinding Systems

Jana Götze and Johan Boye

Abstract References to objects in our physical environment are common especially in language about wayfinding. Advanced wayfinding systems that interact with the pedestrian by means of (spoken) natural language therefore need to be able to *resolve* references given by pedestrians (i.e. understand what entity the pedestrian is referring to). The contribution of this paper is a probabilistic approach to reference resolution in a large-scale, real city environment, where the context changes constantly as the pedestrians are moving. The geographic situation, including information about objects' location and type, is represented using OpenStreetMap data.

Key words: pedestrian navigation, wayfinding, data-driven methods, reference resolution, natural language processing, OpenStreetMap, probabilistic approach

1 Introduction

When humans give wayfinding instructions to each other, they are extensively using referring expressions, phrases that are referring to objects and actions about which they want to convey information. The hearer needs to link the words to representations of these entities, making several choices along the way, and taking different sources of information into account: Is the speaker talking about a landmark in the immediate vicinity? Is he referring to something that has recently been mentioned? Which of the objects match his descriptions and which one is most likely to be the correct target?

If the conversation involves solving a task such as finding the way in an unknown area, it is not enough to understand the meaning of the word "bakery" in an instruc-

Jana Götze
KTH Royal Institute of Technology, e-mail: `jagoetze@kth.se`

Johan Boye
KTH Royal Institute of Technology, e-mail: `jboye@kth.se`

tion like “Turn left at the bakery with the blue sign” in a general way. Not only does the hearer need to know what a bakery is in a general sense, he also needs to identify the *target* object, in this case the particular bakery in his environment in order to carry out the task successfully. That means he needs to ground the meaning of the word ‘bakery’ in the real world (or some representation of it). Methods that automate the understanding and grounding of referring expressions in the physical environment are required for a number of applications in which robots need to carry out actions of various kinds, such as grasping particular objects (Matuszek et al. 2014) or following route directions (MacMahon et al. 2006).

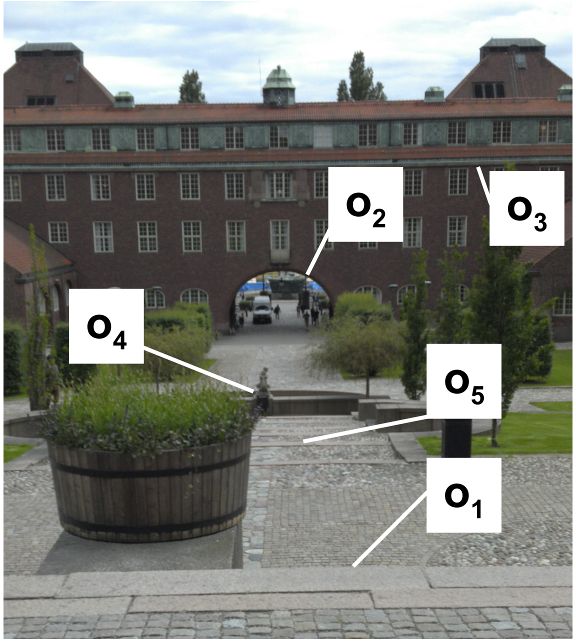
In this work, we focus on reference resolution for pedestrian wayfinding. Wayfinding instructions typically involve many references to landmarks (Denis 1997), i.e. to objects in the environment of the pedestrian. At each point along a route in a city environment, there are many geographical objects of different types, such as buildings and streets, that a pedestrian can refer to. Automatically understanding exactly which objects someone is referring to is an important part of interactive wayfinding systems. The pedestrian might ask clarification questions such as “Do you mean the red building to the right?” or signal problems of understanding such as “I cannot see any church, but I can see a shop straight ahead.” Situations in which the system refers to a landmark that the pedestrian cannot identify are unavoidable, and lengthy sub-dialogs where the system “tries” the next-best landmarks can instead be replaced by letting the user choose a landmark, e.g. by asking an open-ended question like “What can you see?”.

The contribution of this paper is to show how a probabilistic approach to reference resolution can successfully be applied to difficult real-life situations. We base our research on a corpus of route instructions, that are given by pedestrians while they are walking along a path (Götze & Boye 2016*b*). In this setting, the environment is rich in geographical objects of various kinds that a pedestrian could possibly refer to, and it changes continuously as they are moving. We show how the words-as-classifiers method applied by (Kennington & Schlangen 2015) to a toy domain can be applied to our data on the basis of the OpenStreetMap representation of the pedestrians’ environment. We then explain how we extend the original method to deal with frequently occurring phenomena in this context: that the referring expression has no or several target objects.

2 Reference Resolution Method

Whenever the pedestrian uses a referring expression (RE), we want to identify the *target object(s)*, the object or objects that the pedestrian intended to refer to when saying the words in the RE. In a probabilistic framework, we want to find the o that maximizes $P(o|r)$ for some set of objects, i.e. the object o that most likely was referred to by the set of words r in the RE.

In the domain we consider, the pedestrians are walking along a route and are primarily referring to objects in their immediate environment. As they are moving



"I continue in a southwesterly direction *down the steps* [RE$_1$] *towards the arch at the bottom* [RE$_2$]"

u_1 : 'I continue in a southwesterly direction down the steps towards the arch at the bottom'
RE$_1$: "down the steps", $o_{t_1} = \{o_1, o_5\}$
RE$_2$: "towards the arch at the bottom", $o_{t_2} = \{o_2\}$
candidate set cs_1 : $\{o_1 \ldots o_k\}$

Fig. 1 Example utterance containing 2 REs.

along the path, objects are appearing and disappearing from their view: the set of objects that are possible referents for their descriptions – *the candidate set* – changes constantly. This means that in addition to the words, we need to consider the pedestrian's position p. Furthermore, we assume that dialog context information c about what the pedestrian has previously referred to during his walk also plays a role. Therefore, what we really want to model are the probabilities $P(o|r, p, c)$. Technically, however, we will encode the position p and the dialog context c as part of the object properties (see Section 3.3). Thus, we want to estimate $P(o|r)$, where o is a geographical object as seen from position p, and referred to in a context c.

Figure 1 shows an example utterance u from the data we use. The pedestrian uses two REs to refer to three objects. RE$_1$ ("down the stairs") refers to two objects, RE$_2$ ("towards the arch at the bottom") refers to one object. When applying the classifiers to a new RE, each word determines whether the expression can refer to an object in the new candidate set. Usually, objects are described by noun phrases. However, we expect that more information than just the noun phrase will contribute to the correct resolution of an RE. For example, the classifier for the preposition *along* will learn to associate itself with objects of type `street` or `building`, but not with type `shop`. Therefore, we define an RE rather loosely as any substring from the utterance that contains information about an object. Specifically, we included spatial prepositions like *along* and *through* and transitive motion verbs like *cross*. Relevant REs are annotated manually. In the particular situation in Figure 1, there are n objects o the pedestrian could refer to. Every object o_i is represented as a vector of features, encoding information about what kind of object it is, how it is positioned with respect to the pedestrian, and whether it has been mentioned before.

The task is then, given each of the REs and the set of candidate objects, to find the target set of objects o that the words in r are most likely referring to. We approach this task following Kennington & Schlangen (2015), who addressed the problem of reference resolution in a small-scale puzzle piece scenario.

The objects are represented as vectors of numerical features that encode, for example, their type (see Section 3.3). We train individual word classifiers c that, when applied to the vector representation of a geographical entity $o_i = (x_1, \dots, x_n)$, compute the probability that the word r_j refers to o_i. That is, $c_{r_j}(o_i) = P(o_i|r_j)$, where c_{r_j} is the classifier for the word r_j. Each c_{r_j} is a logistic regression classifier.

In general, for a referring expression r consisting of several words $r_1 \dots r_m$, we compute the probability that r refers to each of the objects in the candidate set as a function of the probabilities for each word:

$$P(o_i|r = r_{1\dots m}) = f(c_{r_1}(o_i) \dots c_{r_m}(o_i)) \tag{1}$$

Following Kennington & Schlangen, we let f be the arithmetic average of all $c_{r_j}(o_i)$. Then, objects with a higher probability value are more likely to be the intended targets of the RE.

Using the data we describe in Section 3, we train these logistic regression classifiers that compute an object's suitability as a referent based on the object's features. For every RE, the target object o is a positive example for each word in the RE. As negative examples for each of the words, we randomly choose another object from the candidate set. If an RE has more than one target in a candidate set, one positive example (and one negative example) is added for each target. During training, the information about how many targets an RE refers to is not represented explicitly.

Intuitively, the classifier for the word 'building' will learn to associate high probability to objects that represent buildings (because they appeared as positive examples), and lower probabilities to other objects, such as streets. The classifier for the word 'the' on the other hand is likely to associate equal probabilities to buildings and streets, because speakers use it with both kinds of objects.

Table 1 shows a small example of how the word classifiers are used. In this scene, the candidate set contains 5 objects. The pedestrian utters the words "towards the arch at the bottom". The method then takes each word r of the utterance, computes the probability $P(o|r)$ for each object o in the candidate set (Step a), and then computes a final probability score for each object o by averaging over all word probabilities for that object (Step b), as given by Equation 1. If no classifier is available for a word (because it has not appeared in the training data), the word is ignored.

In Step c, the method picks the object with the highest probability as answer, i.e. it assumes that this object is the target object. In the example, it returns object o_2.

Table 1 Example application of word classifiers. If no classifier is available, the word is ignored.

(a) Word-to-object classification $P(o_i \mid r_j)$					
		Candidate objects $o_{1...n}$			
Words $r_{1...m}$		o_1	o_2	o_3	o_4
r_1=*towards*	→	0.11	0.99	0.89	0.26
r_2=*the*	→	0.59	0.90	0.76	0.76
r_3=*arch*	→	0.19	0.88	0.95	0.19
r_4=*at*	→	0.29	0.89	0.87	0.90
r_5=*the*	→	0.59	0.90	0.76	0.76
r_6=*bottom*	→	–	–	–	–
(b) Composition		↓	↓	↓	↓
$\frac{\Sigma_{j=1}^{m} P(o_i \mid r_j)}{m}$		0.35	0.91	0.85	0.57
(c) Selecting the target			↓		
argmax $P(o \mid r)$			o_2		

In practice, not every RE refers to exactly one object. It is possible that an RE refers to two or more objects (as in the example in Figure 1), or that it refers to no object in the candidate set. As mentioned in Section 1, we assume that an RE refers to an object in the pedestrian's environment. It is however possible that the target object is not part of the candidate set. The target object may not exist in the database, or it was not considered as a candidate in the given situation, e.g. because it was not considered visible given the pedestrian's position. Furthermore (as will be described in more detail in Section 3), number information in the RE itself, i.e. whether it is a plural or a singular RE, does not necessarily correspond to the size of the expected target object set.

We therefore extend the method presented above to incorporate these additional cases. In the following section, we explain the data and features we use. We then first look at the case of simple references in which one RE corresponds to one target object and show how our data achieves results comparable to Kennington & Schlangen (2015). We then suggest an extension of the method, capable of dealing with the cases when there are 0, 1, or more target objects for a given RE, and present results from experiments on our data.

3 Data

3.1 *The* SPACEREF *data*

The data that we are studying and that is described in (Götze & Boye 2015, 2016*b*) contains transcriptions of pedestrians describing their environment while walking along a given path. Referring expressions are annotated with the identifier(s) of their target referent as represented in the map (see Section 3.2). Positional information in the form of GPS coordinates was automatically logged.

The corpus contains a total of $1,303$ referring expressions that are annotated with one or more target referents or tagged as having no referent object in the map. 559 (42.9%) REs have exactly one target object, 218 (16.7%) have more than one target (3 on average), and 526 (40.4%) of the REs having no target referent in their respective candidate set. The candidate set contains on average 33 objects.

3.2 *The geographic representation*

To represent the city environment in which the pedestrians are moving, we choose OpenStreetMap (Haklay & Weber 2008). OpenStreetMap represents objects such as buildings, streets, and shops in a way that is suitable for this task: all objects have information about their position associated with them in the form of GPS latitude/longitude coordinates and the map covers about 96% of the objects mentioned by the pedestrians. This makes it possible to automatically compute a candidate set on the basis of the pedestrian's position.

As described in (Götze & Boye 2015), the way that OpenStreetMap segments space into objects does not always correspond to how a pedestrian views his environment. In OpenStreetMap, streets are cut up into many small segments, each with their own specification of speed limits or access restrictions. Likewise, plurals do not necessarily have more than one target object, e.g. a block of buildings can be represented as one object, but perceived and referred to as "the buildings". We address how to make the choice of how many objects to return as referents in Section 5. We modify the selection step in a way that it returns all the relevant objects as correct target referents.

3.3 *Features*

In order to train the word classifiers, we need to represent the objects in each candidate set using suitable features that capture a part of the word's meaning. Most REs contain descriptions of the objects' type. Therefore, an object's features should contain a notion of their type, i.e. whether the object is a street, a building, or a bench. As mentioned earlier, we use OpenStreetMap to compute the candidate set on the basis of the pedestrian's position: all objects that are in view at the time of using each word are computed as described by Boye et al. (2014). In addition to positional information, OpenStreetMap provides semantic tags for each object, specifying information like names, types, opening hours or other access information.

OpenStreetMap is a crowd-sourced database. It defines the usage of many tags and their values,[1] but contributors are in no way restricted in what tags they assign to an entity. As in (Götze & Boye 2016*a*), we derive 427 binary type features from the OpenStreetMap annotation. If an entity is of a certain type, it has value 1 for this feature, otherwise 0. We base the derivation of features only on such tags that are defined in OpenStreetMap's wiki. That means that other, user-defined tags that also carry non-relevant information, are not excluded from the feature set and introduce a fair amount of noise to our object representations. We will show in Section 5 that these features, that we obtain with only little processing of the original data, perform well when computing word meanings.

In addition to the type features (called OSM in Table 2), we derive positional information (POS) for each object: the distance and angle relate each object to the pedestrian's position. The feature set CONTEXT contains context information on whether an object has been mentioned before and how recent this mention was in terms of time or traveled distance. This context feature set is an extension of features used in (Iida et al. 2011) and is intended to capture and incorporate the meaning of function words, such as the determiners 'a' and 'the', 'that', 'this' etc. For example, referring expressions of the form 'a *x*' are likely to refer to a new object while mentioning 'this *x*' is an indication of a previously mentioned object. Table 2 shows the full list of features.

Feature	Values
OSM	
type	0/1 The object is of that type (427 features)
POS	
dist	2-log distance from the pedestrian's position to the object
angle	Angle between the walking direction and the object direction, measured from the pedestrian's position
CONTEXT	
mrRE	0/1 The object is referred to by the most recent RE
m10	0/1 The time distance to the last mention of this object is $\leq$ 10s
m20	0/1 The time distance to the last mention of this object is $\leq$ 20s and $>$10s
m20+	0/1 The time distance to the last mention of this object is $>$ 20s
never	0/1 The object has never been referred to
t50	0/1 The distance to the last mention of this object is $\leq$ 50m
t100	0/1 The distance to the last mention of this object is $\leq$ 100m and $>$50m
t100+	0/1 The distance to the last mention of this object is $>$100m

Table 2 The features that describe each candidate object. The first five CONTEXT features correspond to L1-5 in (Iida et al. 2011).

[1] http://wiki.openstreetmap.org/map_features

4 Experiments

4.1 One-to-one references

In this setting, we select from the data only those instances with target set size 1, i.e. where each RE corresponds to exactly one object identifier in the map. This is the case in 559 instances. We are training the word classifiers on different combinations of features, in the way as described in Section 2. Testing is done using 10-fold cross-validation. Since negative examples are chosen randomly at training time, this process is repeated 10 times and we report averages in Table 3. We report the First Hit Success Rate (FHS), i.e. in how many cases the target object was correctly ranked highest (and thus selected) by the method, and the Mean Reciprocal Rank (MRR), indicating how high the correct object was ranked on average.[2] For comparison, in the puzzle piece setting in (Kennington & Schlangen 2015), the classifiers found 42% (FHS) of the targets and reached a MRR of 0.61. Table 3 shows that already when using only the information contained in the OpenStreetMap tags, the FHS Rate reaches 55% with a MRR of 0.66. Including positional and context information, we obtain a FHS in 59% of the cases and a MRR of 0.72.

4.2 One-to-many references

The assumption that there is exactly one object that is the correct target referent does not hold for more than half of the referring expressions in our data. In 40% the correct target is not among the candidate referents (cf. Section 3), and in another 17% the target corresponds to a set of more than one object in the database.

Using the original method and choosing the most likely object will result in a wrong answer when there is no correct target, and an insufficient answer when there is more than one. Instead, when there are several targets, a reference resolution method should preferably return all these targets, and when there is no target, the method should return the empty set.

A possible solution is to define a threshold value t, where only the object or objects that have a probability of at least t will be considered as referents. If the highest ranked object is below the threshold, no object will be returned.

Table 3 Evaluation for one-to-one references.

	FHS	MRR
OSM	54.64	0.66
OSM+POS	58.09	0.70
OSM+POS+CONTEXT	59.17	0.72

[2] The Reciprocal Rank measure calculates the reciprocal of the rank. It is 1 if the correct object is ranked highest, 0.5 if the correct object is ranked second, etc. The Mean Reciprocal Rank (MRR) is the average across many such calculations.

In the next step, we split our data into a training set of 80% (1,025 instances) and a development and test set of 10% each (132 and 146 instances, respectively[3]). The training set is used for training the word classifiers as described previously, whereas the development set is used for determining a suitable threshold value. The test set is used for evaluation. Unless otherwise stated, all training and testing is carried out in 10 iterations, and we report averages (negative examples for the word classifiers are chosen at random and differ for each training run).

For evaluation, we now look at how many objects were (in)correctly classified for each RE. Recall that for each RE, there is an average of 33 candidate objects. Each object is assigned a probability that it is the correct referent of the referring expression in question. In finding the threshold value, we use all three feature sets, and vary the threshold over a range of [.5;.95] in steps of 0.05. All objects that obtain a probability of at least the threshold value will be returned by the method.

We computed accuracy, precision, recall, and F-measure for each threshold value. Every target referent is a positive, all other objects are negatives. For an RE without a target referent in the candidate set, all objects should be classified as negative. For each RE, there are many more objects that should be classified as negative, i.e. as not being the correct referent. When classifying all objects as negative, i.e. never returning a referent, we would obtain an accuracy of 0.97. With the original method of choosing the object(s) with highest probability, the accuracy on the development set is also 0.97. Starting at a threshold value of 0.8 the accuracy improves over the original setting.

Looking at the F-measure, a threshold of around 0.80 is best in terms of both positives and negatives (F=0.46). For the positive class (objects chosen as referents), this threshold means a recall of 0.52 and a precision of 0.41. For the negative class (objects rejected as referents), both precision and recall are close to 1.0 (cf. Figure 2). In a particular application, it may be desirable to prefer higher precision over higher recall (being sure that what was found is a correct referent), or vice

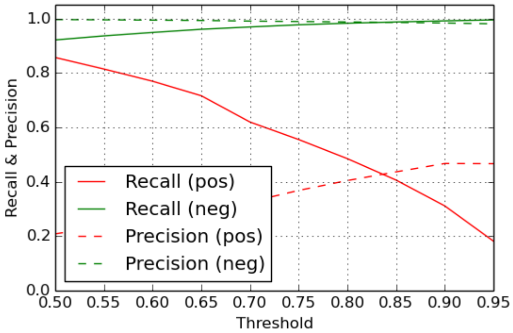


Fig. 2 Evaluation for varying thresholds

[3] The data is split on the utterance level, where each utterance contains one or more referring expressions.

versa (finding as many targets as possible at the expense of including false positives). Here, we are not making such a choice and set the threshold value at 0.80. At this threshold, the method also works well in terms of how many targets it finds for the different conditions: It predicts on average 1 object for the case where there is only 1 or no target referent and slightly over 2 in the case where there are more.

4.3 Testing on the held-out test set

Table 4 shows the results for applying the learned models on the remaining 10% of the data (146 instances) with a threshold of 0.80 for selecting referent objects. The results on this test set are similar to the results on the development set.

Table 4 Evaluation results for the held-out test data when selecting objects that have a probability of at least 0.8

	Test Set		Dev Set	
Measure	Pos	Neg	Pos	Neg
Accuracy	0.97		0.97	
Precision	0.40	0.98	0.40	0.99
Recall	0.45	0.98	0.48	0.98
F-measure	0.42	0.98	0.44	0.98

4.3.1 Evaluation per Referring Expression

The evaluation measures in Table 4 show what happens within each candidate set. Table 5 shows how many of the referring expressions the method resolves correctly. In the strictest setting (in which the method returns all targets and no false positives), the method resolves 44.3% of the referring expressions correctly.

When there is no target referent, it answers correctly with the empty set in more than half of the cases. When there is one referent, it answers correctly in one third of the cases, when there is more than one referent, in one fourth of the cases. Allowing also false positives in the answer set, it answers correctly about half of the time. For all cases, the target set of objects obtains a rank of 1.5 (i.e. a MRR of 0.66) on average. When there are several targets, all of them are ranked high, with an average MRR of 0.73, i.e. about rank 1.4.

4.4 Results

The results in Section 4.1 show that the basic approach of training word classifiers and applying them to features derived from OpenStreetMap representations of objects works well. Choosing the most likely object resolves simple one-to-one references in almost 60% of the REs. In assessing the success of the method recall that

Table 5 Evaluation per RE on the test set (threshold=0.8). TP:True Pos., FP:False Pos.

Target Set Size s			MRR
s=0	Correct (TP=0, FP=0)	59.0%	0.59
s=1	Correct (TP=1, FP=0)	37.0%	0.71
	Partly correct (TP=1)	48.4%	
s>1	Correct (TP=s , FP=0)	25.9%	0.73
	Partly correct (TP=s)	26.8%	
	Partly correct (TP≥2)	51.4%	
	Partly correct (TP≥1)	67.3%	
Total	TP=s, FP=0	44.3%	0.66
	TP=s	49.4%	

this reference resolution problem is a difficult one – the candidate set contains 33 candidate referents on average.

For the general case – where there might be 0, 1, or more correct referents – the extended method using thresholds resolves 44.3% of the referring expressions correctly, meaning that it selected exactly the right set of referents, so this is an even more difficult problem than the one-to-one case. Not surprisingly, the result is not as good as the one-to-one case, but still higher than the 42% for the one-to-one references in Kennington & Schlangen's puzzle piece setting.

When there is one target referent, the extended method produces a completely correct answer in 33.3% of the REs, and a partly correct answer in 49.2%. The basic method of choosing the most likely object was correct in 59%. However, we can now also resolve the other cases without explicitly representing information about the target set size.

5 Discussion

Given the sparseness of the language data and the crowd-sourced nature of the geographical data, we consider the results a good step towards incorporating spatial reference resolution into a real-time system.

Since OpenStreetMap tags most often are plain English words, an obvious alternative idea is to simply look for those words in the input (i.e. if the user mentions a "building", this would translate to OpenStreetMap entities having the tag `building`). This is in fact what Götze & Boye (2015) have tried before. However, that straightforward approach has drawbacks: It is language-dependent, it requires manual intervention and translation-rule writing (since some words like "street" have OpenStreetMap tag counterparts that no user would ever say: `primary`, `secondary` etc.), and it presupposes that every reference refers to exactly one entity. The probabilistic approach presented in this paper has none of these drawbacks.

Recall that the geographical representation is imperfect in two ways. First, we cannot be sure that all information in OpenStreetMap is complete and correct. Second, the GPS signal of the pedestrian's position is only an approximation of his real position. This situation is however realistic in this domain (Modsching et al. 2006) and we have therefore not manipulated neither the map representation nor the GPS signal. The features that represent positional information are noisy, and a closer look at the classifiers of the words *left* and *right* reveals that they have not learned any association with these features, their associated weights are close to 0, i.e. they do not influence the object rank. We expect that a more accurate GPS signal will improve the results (cf. Misu et al. 2014).

On the other hand, semantic information about an object's type or appearance correlates well with type features that we would expect. For example, the classifier for the noun *building* associates the highest weight with the feature `building_yes` and one of the lowest weights to the feature `highway`.

Table 6 shows the highest and lowest weighted features for the nouns *road* and *building*.

Table 6 Semantic (OSM) features correlate well with types: extract of the word classifiers for *road* and *building*

c_{road}		$c_{building}$	
feature	weight	feature	weight
highway	1.8894	building=yes	1.3778
name	1.6214	website	1.1827
secondary	0.8873	t50	0.5660
bus_stop	−0.5856	waste_basket	−0.2552
distal	−0.7588	highway	−0.9854
never	−0.8692	distal	−1.6728

Context information about whether an object has been mentioned before, within a certain time span, or a certain distance traveled, also shows the expected correlations. For the modifier *same*, the corresponding classifier learns high weights for features indicating that the object has been recently mentioned and the lowest weights to features that indicate that an object has not recently been mentioned as shown for the words *same* and *this* in Table 7.

Table 7 Context features in function words: extract of the word classifiers for *this* and *same*. (cf. Table 2)

c_{this}		c_{same}	
feature	weight	feature	weight
highway	1.0784	t100	1.1097
mrRE	0.7123	m10	0.9109
secondary	0.6251	secondary	0.6382
track	−0.2252	never	−0.1744
angle	−0.3001	distal	−0.5068
distal	−1.5258	mrRE	−0.7048

The size of the vocabulary that the pedestrians use to describe their environment is relatively small. From our training set, we obtain about 280 word classifiers, most of them have seen only few examples. In the complete data set of (313 distinct tokens), the average number of examples per classifier is 63.3, the median is 31.5. Only 73 words occur at least 10 times, 45 occur at least 20 times. With the training set that we have used, at least 90% of the words in an RE had classifiers and

60% have classifiers that were trained on at least 20 examples. Figure 3 shows the classifier coverage for different training set sizes.

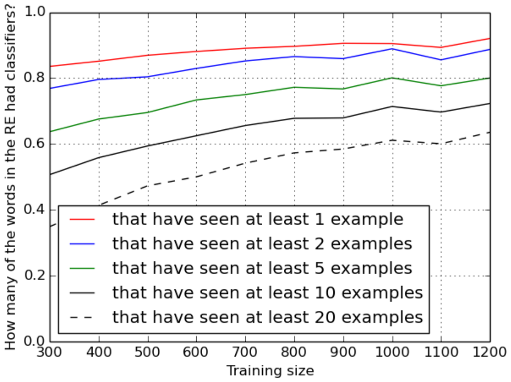


Fig. 3 Words per RE that have classifiers

6 Related Work

How the meaning of words can be grounded in perceptive information has recently become an active area of research (Roy 2005, Mooney 2008). When dealing with the problem of Reference Resolution, mainly visual information is considered to model the meaning of words and phrases (Kruijff et al. 2006, Matuszek et al. 2014, Kennington & Schlangen 2015).

In our domain of pedestrian wayfinding, direct visual input is hard to obtain. Some studies rely on photographs (Baltaretu et al. 2015), but in a working real-time wayfinding system, photographs will be insufficient as pedestrians are not restricted in their movement and can quickly turn around to face another direction.

Works that do not rely in visual information, but on a direct representation of the physical environment typically work on a domain that is considerably smaller than ours, with 10 candidate objects or fewer and where objects are of only few distinctive types (Gorniak & Roy 2005, Schütte et al. 2010, Funakoshi et al. 2012). In (Kennington & Schlangen 2015), the set of objects is comparable in size to ours. They process the set of puzzle pieces using computer vision methods. However, all objects are clearly distinct from each other and all are of the same type. Instead of direct perceptual input, we rely on a crowd-sourced map representation of the environment that covers the study area well and use features from the semantic tags associated to the objects and that are also crowd-sourced. The Pursuit corpus (Blaylock 2011), in which car drivers are also describing while they are moving along a path, is in principle suitable for this task as well, but the area is not very well covered in OpenStreetMap at this point and the object annotation contains references to several different databases, and no information on other candidate objects.

Misu et al. (2014) have attempted to resolve spatial references with good success in a similar setting. Instead of descriptions, car drivers pose queries about Points of Interest (POI). Like in the data we use, they use information about the speaker position and the POI positions and types. Additionally, they have access to the drivers' head pose when speaking and an analysis of the data showed that directional information (left/right) aligned well with the speakers' mentions of directions, even though they also report errors in the GPS information. In this work however, knowledge about the referent candidates is assembled manually. They also explicitly exclude context information such as dialog history.

How to segment the context into objects is an active area of research. Context segmentation is typically done independently for each modality and the information then fused. Kruijff et al. (2006) have proposed a framework to incorporate this step into a rule-based reference resolution algorithm, and Bruni et al. (2014) fuse information from the linguistic and visual context to obtain an integrated representation of meaning. For context representations based on visual input, computer vision algorithms are applied with good results for small domains and where the objects are clearly distinct from each other (Matuszek et al. 2012, Kennington & Schlangen 2015). Krishnamurthy & Kollar (2013) and Malinowski & Fritz (2014) perform this segmentation on photographs that depict rather everyday scenes and Malinowski & Fritz (2014) account for uncertainty in the image segmentation by utilizing the associated confidence scores. All of these approaches do however assume that every object (or segment) corresponds to a referent (unless the RE is a plural).

In OpenStreetMap, objects do have clear boundaries, but as we have described in Section 3.2, this segmentation does not align with the objects that the pedestrians in the data refer to. We handle this discrepancy by resolving REs to sets of objects based on the probability distribution returned by the word classifiers. An alternative approach is to structure the context representation beforehand, i.e. decide which sets of entities are available for reference and modify the candidate set accordingly. Funakoshi et al. (2012) use Reference Domain Theory (Salmon-Alt & Romary 2009), grouping tangram pieces based on proximity to determine which reference domains, i.e. sets of objects, can be referred to. There is good evidence for how humans perceptually group objects (Thórisson 1994), e.g. based on proximity. However, in our domain and with the geographic representation at hand, it remains unclear how to represent a set of objects based on the features of the individual objects. This is a known issue in research using OpenStreetMap (Ballatore et al. 2013) and we leave this as a question for future research.

Finally, in an interactive wayfinding system, references to landmarks are also an essential part of the generation process. There are at least two steps involved in this process. The first one decides which landmark to choose, usually on the basis of the current routing situation and some calculation of which objects are most salient (e.g. Raubal & Winter 2002, Götze & Boye 2016*a*). The second step decides how to translate the object representation into a suitable referring expression (e.g. Garoufi & Koller 2011, Paraboni & van Deemter 2014).

7 Conclusion

We have presented a method for situated reference resolution in a large-scale environment where the context changes with the speaker's movement. Using an existing, crowd-sourced geographic database that represents objects at different granularities than the speakers refer to them. We have shown a way to extend current methods to allow for cases where the correct set of target objects is empty or contains more than one object.

References

Ballatore, A., Bertolotto, M. & Wilson, D. C. (2013), 'Geographic knowledge extraction and semantic similarity in OpenStreetMap', *Knowledge Information Systems* **37**(1), 61–81.

Baltaretu, A., Krahmer, E. & Maes, A. (2015), 'Improving route directions: The role of intersection type and visual clutter for spatial reference', *Applied Cognitive Psychology* **29**(5), 647–660.

Blaylock, N. (2011), 'Semantic annotation of street-level geospatial entities', *Proceedings of the IEEE ICSC Workshop on Semantic Annotation for Computational Linguistic Resources* .

Boye, J., Fredriksson, M., Götze, J., Gustafson, J. & Königsmann, J. (2014), 'Walk this way: Spatial grounding for city exploration', *Natural Interaction with Robots, Knowbots and Smartphones* pp. 59–67.

Bruni, E., Tran, N.-K. & Baroni, M. (2014), 'Multimodal distributional semantics', *Journal of Artificial Intelligence Research (JAIR)* **49**, 1–47.

Denis, M. (1997), 'The description of routes: A cognitive approach to the production of spatial discourse', *Current Psychology of Cognition* **16**(4), 409–458.

Funakoshi, K., Nakano, M., Tokunaga, T. & Iida, R. (2012), 'A unified probabilistic approach to referring expressions', *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue* pp. 237–246.

Garoufi, K. & Koller, A. (2011), 'The Potsdam NLG systems at the GIVE-2.5 Challenge', *Proceedings of the 13th European Workshop on Natural Language Generation (ENLG)* pp. 307–311.

Gorniak, P. & Roy, D. (2005), 'Probabilistic grounding of situated speech using plan recognition and reference resolution', *Proceedings of the 7th International Conference on Multimodal Interfaces* pp. 138–143.

Götze, J. & Boye, J. (2015), ‘Resolving spatial references using crowdsourced geographical data’, *Proceedings of the 20th Nordic Conference of Computational Linguistics, NODALIDA* pp. 61–68.

Götze, J. & Boye, J. (2016*a*), ‘Learning Landmark Salience Models from Users’ Route Instructions’, *Journal of Location Based Services* **10**(1), 47–63.

Götze, J. & Boye, J. (2016*b*), ‘SPACEREF: A corpus of street-level geographic descriptions’, *Proceedings of LREC* .

Haklay, M. & Weber, P. (2008), ‘Openstreetmap: User-generated street maps’, *Pervasive Computing, IEEE* **7**(4), 12–18.

Iida, R., Yasuhara, M. & Tokunaga, T. (2011), ‘Multi-modal reference resolution in situated dialogue by integrating linguistic and extra-linguistic clues’, *The 5th International Joint Conference on Natural Language Processing* pp. 84–92.

Kennington, C. & Schlangen, D. (2015), ‘Simple Learning and Compositional Application of Perceptually Grounded Word Meanings for Incremental Reference Resolution’, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* pp. 292–301.

Krishnamurthy, J. & Kollar, T. (2013), ‘Jointly learning to parse and perceive: Connecting natural language to the physical world’, *TACL* **1**, 193–206.

Kruijff, G.-J., Kelleher, J. & Hawes, N. (2006), ‘Information Fusion for Visual Reference Resolution in Dynamic Situated Dialogue’, *Perception and Interactive Technologies* **4021**, 117–128.

MacMahon, M., Stankiewicz, B. & Kuipers, B. (2006), ‘Walk the Talk: Connecting Language, Knowledge, and Action in Route Instructions’, *Proceedings of the 21st National Conference on Artificial Intelligence* pp. 1475–1482.

Malinowski, M. & Fritz, M. (2014), ‘A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input’, *NIPS* pp. 1682–1690.

Matuszek, C., Bo, L., Zettlemoyer, L. & Fox, D. (2014), ‘Learning from Unscripted Deictic Gesture and Language for Human-Robot Interactions’, *Proceedings of AAAI* pp. 2556–2563.

Matuszek, C., FitzGerald, N., Zettlemoyer, L. S., Bo, L. & Fox, D. (2012), ‘A joint model of language and perception for grounded attribute learning’, *ICML* .

Misu, T., Raux, A., Gupta, R. & Lane, I. (2014), Situated language understanding at 25 miles per hour, *in* ‘Proceedings of the 15th SIGdial Workshop on Discourse and Dialogue’.

Modsching, M., Kramer, R. & ten Hagen, K. (2006), Field trial on gps accuracy in a medium size city: The influence of built-up, *in* ‘3rd workshop on positioning, navigation and communication’, pp. 209–218.

Mooney, R. J. (2008), Learning to Connect Language and Perception, *in* ‘Proceedings of AAAI’, pp. 1598–1601.

Paraboni, I. & van Deemter, K. (2014), ‘Reference and the facilitation of search in spatial domains’, *Language, Cognition and Neuroscience* **29**(8), 1002–1017.

Raubal, M. & Winter, S. (2002), Enriching Wayfinding Instructions with Local Landmarks, *in* ‘Geographic Information Science’, Vol. 2478 of *Lecture Notes in Computer Science*, pp. 243–259.

Roy, D. (2005), ‘Grounding words in perception and action: computational insights’, *Trends in Cognitive Sciences* **9**(8), 389–396.

Salmon-Alt, S. & Romary, L. (2009), Reference Resolution within the Framework of Cognitive Grammar, *in* ‘International Colloquium on Cognitive Science’, pp. 284–299.

Schütte, N., Kelleher, J. & Mac Namee, B. (2010), Visual Salience and Reference Resolution in Situated Dialogues: A Corpus-based Evaluation, *in* ‘AAAI Symposium on Dialog with Robots’, pp. 109–114.

Thórisson, K. R. (1994), Simulated perceptual grouping: An application to human computer interaction, *in* ‘In Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society CSS94’, pp. 876–881.