The Impact of Listener Gaze on Predicting Reference Resolution

Nikolina Koleva[1] **Martín Villalba**[2] **Maria Staudte**[1] **Alexander Koller**[2]

[1]Embodied Spoken Interaction Group, Saarland University, Saarbrücken, Germany
[2] Department of Linguistics, University of Potsdam, Potsdam, Germany
{nikkol | masta}@coli.uni-saarland.de
{martin.villalba | alexander.koller}@uni-potsdam.de

Abstract

We investigate the impact of listener's gaze on predicting reference resolution in situated interactions. We extend an existing model that predicts to which entity in the environment listeners will resolve a referring expression (RE). Our model makes use of features that capture which objects were looked at and for how long, reflecting listeners' visual behavior. We improve a probabilistic model that considers a basic set of features for monitoring listeners' movements in a virtual environment. Particularly, in complex referential scenes, where more objects next to the target are possible referents, gaze turns out to be beneficial and helps deciphering listeners' intention. We evaluate performance at several prediction times before the listener performs an action, obtaining a highly significant accuracy gain.

1 Introduction

Speakers tend to follow the listener's behavior in order to determine whether their communicated message was received and understood. This phenomenon is known as *grounding*, it is well established in the dialogue literature (Clark, 1996), and it plays an important role in collaborative tasks and goal–oriented conversations. Solving a collaborative task in a shared environment is an effective way of studying the alignment of communication channels (Clark and Krych, 2004; Hanna and Brennan, 2007).

In situated spoken conversations ambiguous linguistic expressions are common, where additional modalities are available. While Gargett et al. (2010) studied instruction giving and following in virtual environments, Brennan et al. (2013) examined pedestrian guidance in outdoor real environments. Both studies investigate the interaction of human interlocutors but neither study exploits listeners' eye movements. In contrast, Koller et al. (2012) designed a task in which a natural language generation (NLG) system gives instructions to a human player in virtual environment whose eye movements were tracked. They outperformed similar systems in both successful reference resolution and listener confusion. Engonopoulos et al. (2013) attempted to predict the resolution of an RE, achieving good performance by combining two probabilistic log–linear models: a *semantic* model P_{sem} that analyzes the semantics of a given instruction, and an *observational* model P_{obs} that inspects the player's behavior. However, they did not include listener's gaze. They observed that the accuracy for P_{obs} reaches its highest point at a relatively late stage in an interaction. Similar observations are reported by Kennington and Schlangen (2014): they compare listener gaze and an incremental update model (IUM) as predictors for the resolution of an RE, noting that gaze is more accurate before the onset of an utterance, whereas the model itself is more accurate afterwards.

In this paper we report on the extension of the P_{obs} model to also consider listener's visual behaviour. More precisely we implement features that encode listener's eye movement patterns and evaluate their performance on a multi–modal data collection. We show that such a model as it takes an additional communication channel provides more accurate predictions especially when dealing with complex scenes. We also expand on concepts from the IUM, by applying the conclusions drawn from its behaviour to a dynamic task with a naturalistic interactive scenario.

2 Problem definition

We address the research question of how to automatically predict an RE resolution, i.e., answering the question of which entity in a virtual environment has been understood by the listener af-

ter receiving an instruction. While the linguistic material in instructions carries a lot of information, even completely unambiguous descriptions may be misunderstood. A robust NLG system should be capable of detecting misunderstandings and preventing its users from making mistakes.

Language comprehension is mirrored by interlocutors' non verbal behavior, and this can help when decoding the listener's interpretation. Precise automatic estimates may be crucial when developing a real–time NLG system, as such a mechanism would be more robust and capable at avoiding misunderstandings. As mentioned in section 1, Engonopoulos et al. (2013) propose two statistical models to solve that problem: a semantic model P_{sem} based on the linguistic content, and an observation model P_{obs} based on listener behavior features.

More formally, let's assume a system generates an expression r that aims to identify a target object o_t among a set O of possible objects, i.e. those available in the scene view. Given the state of the world s at time point t, and the observed listener's behavior $\sigma(t)$ of the user at time $t \geq t_b$ (where t_b denotes the end of an interaction), we estimated the conditional probability $p(o_p|r, s, \sigma(t))$ that indicates how probable it is that the listener resolved r to o_p. This probability can be also expressed as follows:

$$P(o_p|r, s, \sigma(t)) \propto \frac{P_{sem}(o_p|r, s)P_{obs}(o_p|\sigma(t))}{P(o_p)}$$

Following Engonopoulos et al. (2013) we make the simplifying assumption that the distribution of the probability among the possible targets is uniform and obtain:

$$P(o_p|r, s, \sigma(t)) \propto P_{sem}(o_p|r, s)P_{obs}(o_p|\sigma(t))$$

We expect an NLG system to compute and output an expression that maximizes the probability of o_p. Due to the dynamic nature of our scenarios, we also require the probability value to be updated at certain time intervals throughout an interaction. Tracking the probability changes over time, an NLG system could proactively react to changes in its environment. Henderson and Smith (2007) show that accounting for both fixation location and duration are key to identify a player's focus of attention.

The technical contribution of this paper is to extend the P_{obs} model of Engonopoulos et al. (2013) with gaze features to account for these variables.

3 Episodes and feature functions

The data for our experiment was obtained from the GIVE Challenge (Koller et al., 2010), an interactive task in a 3D virtual environment in which a human player (instruction follower, IF) is navigated through a maze, locating and pressing buttons in a predefined order aiming to unlock a safe. While pressing the wrong button in the sequences doesn't always have negative effects, it can also lead to restarting or losing the game. The IF receives instructions from either another player or an automated system (instruction giver, IG). The IF's behavior was recorded every 200ms, along with the IG's instructions and the state of the virtual world. The result is an interaction corpus comprising over 2500 games and spanning over 340 hours of interactions. These interactions were mainly collected during the GIVE-2 and the GIVE-2.5 challenges. A laboratory study conducted by Staudte et al. (2012) comprises a data collection that contains eye-tracking records for the IF. Although the corpus contains both successful and unsuccessful games, we have decided to consider only the successful ones.

We define an *episode* over this corpus as a typically short sequence of recorded behavior states, beginning with a manipulation instruction generated by the IG and ending with a button press by the IF (at time point t_b). In order to make sure that the recorded button press is a direct response to the IG's instruction, an episode is defined such that it doesn't contain further utterances after the first one. Both the target intended by the IG (o_t) and the one selected by the IF (o_p) were recorded.

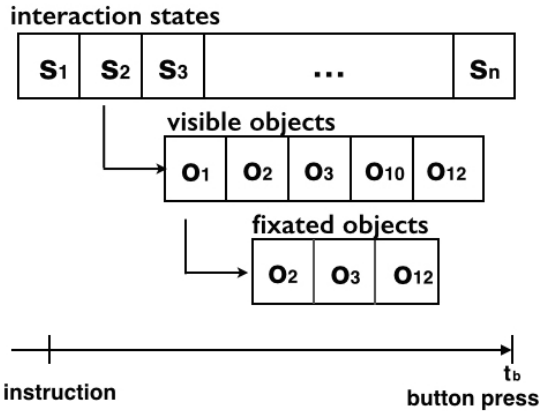

Figure 1: The structure of the interactions.

Figure 1 depicts the structure of an episode when eye-tracking data is available. Each episode

can be seen as a sequence of interaction states $(s_1, \ldots, s_n)$, and each state has a set of visible objects ($\{o_1, o_2, o_3, o_{10}, o_{12}\}$). We then compute the subset of fixated objects ($\{o_2, o_3, o_{12}\}$). We update both sets of visible and fixated objects dynamically in each interaction state with respect to the change in visual scene and the corresponding record of the listener's eye movements.

We developed feature functions over these episodes. Along with the episode's data, each function takes two parameters: an object o_p for which the function is evaluated, and a parameter d seconds that defines how much of the episode's data is the feature allowed to analyze. Each feature looks only at the behavior that happens in the time interval $-d$ to 0. Henceforth we refer to the value of a feature function over this interval as its value at time $-d$. The value of a feature function evaluated on episodes with length less than d seconds is undefined.

4 Prediction models

Given an RE uttered by an IG, the *semantic* model P_{sem} estimates the probability for each possible object in the environment to have been understood as the referent, ranks all candidates and selects the most probable one in a current scene. This probability represents the semantics of the utterance, and is evaluated at a single time point immediately after the instruction (e.g. "press the blue button") has been uttered. The model takes into account features that encode the presence or absence of adjectives carrying information about the spatial or color properties (like the adjective "blue"), along with landmarks appearing as post modifiers of the target noun.

In contrast to the semantic model, the *observational* model P_{obs} evaluates the changes in the visual context and player's behavior after an instruction has been received. The estimated probability is updated constantly before an action, as the listener in our task–oriented interactions is constantly in motion, altering the visual context. The model evaluates the distance of the listener position to a potential target, whether it is visible or not, and also how salient an object is in that particular time window.

As we have seen above, eye movements provide useful information indicating language comprehension, and also how to map a semantic representation to an entity in a shared environment. Interlocutors constantly interact with their surrounding and point to specific entities with gestures. Gaze behaviour is also driven by the current state of an interaction. Thus, we extend the basic set of P_{obs} features and implement eye–tracking features that capture gaze information. We call this the *extended observational* model P_{Eobs} and consider the following additional features:

1. *Looked at:* feature counts the number of interaction states in which an object has been fixated at least once during the current episode.

2. *Longest Sequence:* detects the longest continuous sequence of interaction states in which a particular object has been fixated.

3. *Linear Distance:* returns the euclidean distance $dist$ on screen between the gaze cursor and the center of an object.

4. *Inv-Squared Distance:* returns $\frac{1}{1+dist^2}$.

5. *Update Fixated Objects:* expands the list of fixated objects in order to consider the IF's focus of attention. It successively searches in 10 pixel steps and stops as soon as an object is found (the threshold is 100 pixels). This feature evaluates to 1 if the list of fixated objects is been expanded and 0 otherwise.

When training our model at time $-d_{train}$, we generate a feature matrix. Given a training episode, each possible (located in the same room) object o_p is added as a new row, where each column contains the value of a different feature function for o_p over this episode at time $-d_{train}$. Finally, the row based on the target selected by the IF is marked as a positive example. We then train a log-linear model, where the weights assigned to each feature function are learned via optimization with the L-BFGS algorithm. By training our model to correctly predict a target button based only on data observed up until $-d_{train}$ seconds before the actual action t_b, we expect our model to reliably predict which button the user will select. Analogously, we define accuracy at testing time $-d_{test}$ as the percentage of correctly predicted target objects when predicting over episodes at time $-d_{test}$. This pair of training and test parameters is denoted as the tuple (d_{train}, d_{test}).

5 Dataset

We evaluated the performance of our improved model over data collected by Staudte et al. (2012) using the GIVE Challenge platform. Both training and testing were performed over a subset of the data obtained during a collection task involving worlds created by Gargett et al. (2010), designed to provide the task with varying levels of difficulty. This corpus provides recorded eye-tracking data, collected with a remote faceLAB system. In contrast, the evaluation presented by Engonopoulos et al. (2013) uses only games collected for the GIVE 2 and GIVE 2.5 challenges, for which no eye-tracking data is available. Here, we do not investigate the performance of P_{sem} and concentrate on the direct comparison between P_{obs} and P_{Eobs} in order to find out if and when eye–tracking can improve the prediction of an RE resolution.

We further filtered our corpus in order to remove noisy games following Koller et al. (2012), considering only interactions for which the eye-tracker calibration detected inspection of either the target or another button object in at least 75% of all referential scenes in an interaction. The resulting corpus comprises 75 games, for a combined length of 8 hours. We extracted 761 episodes from this corpus, amounting to 47m 58s of recorded interactions, with an average length per episode of 3.78 seconds ($\sigma = 3.03sec.$). There are 261 episodes shorter than 2 sec., 207 in the 2-4 sec. range, 139 in the 4-6 sec. range, and 154 episodes longer than 6 sec.

6 Evaluation and results

The accuracy of our probabilistic models depends on the parameters (d_{train}, d_{test}). At different stages of an interaction the difficulty to predict an intended target varies as the visual context changes and in particular the number of visible objects. As the weights of the features are optimized at time $-d_{train}$, it would be expected that testing also at time $-d_{test} = -d_{train}$ yields the highest accuracy. However, the difficulty to make a prediction decreases as $t_b - d_{test}$ approaches t_b, i.e. as the player moves towards the intended target. We expect that testing at $-d_{train}$ works best, but we need to be able to update continuously. Thus we also evaluate at other timepoints and test several combinations of the (d_{train}, d_{test}) parameters.

Given the limited amount of eye-tracking data available in our corpus, we replaced the cross-corpora-challenge test setting from the original P_{obs} study with a ten fold cross validation setup. As training and testing were performed over instances of a certain minimum length according to (d_{train}, d_{test}), we first removed all instances with length less than $max(d_{train}, d_{test})$, and then perform the cross validation split. In this way we ensure that the number of instances in the folds are not unbalanced. Moreover, each instance was classified as *easy* or *hard* depending on the number of visible objects at time t_b. An instance was considered *easy* if no more than three objects were visible at that point, or *hard* otherwise. For $-d_{test} = 0$, 59.5% of all instances are considered *hard*, but this proportion decreases as $-d_{test}$ increases. At $-d_{test} = -6$, the number of hard instances amounts to 72.7%.

We evaluated both the original P_{obs} model and the P_{Eobs} model on the same data set. We also calculated accuracy values for each feature function, in order to test whether a single function could outperform P_{obs}. We included as baselines two versions of P_{obs} using only the features *InRoom* and *Visual Salience* proposed by Engonopoulos et al. (2013).

The accuracy results on Figure 2 show our observations for $-6 \leq -d_{train} \leq -2$ and $-d_{train} \leq -d_{test} \leq 0$. The graph shows that P_{Eobs} performs similarly as P_{obs} on the *easy* instances, i.e. the eye-tracking features are not contributing in those scenarios. However, P_{Eobs} shows a consistent improvement on the *hard* instances over P_{obs}.

For each permutation of the training and testing parameters (d_{train}, d_{test}), we obtain a set of episodes that fulfil the length criteria for the given parameters. We apply P_{obs} and P_{Eobs} on the obtained set of instances and measure two corresponding accuracy values. We compared the accuracy values of P_{obs} and P_{Eobs} over all 25 different (d_{train}, d_{test}) pairs, using a paired samples t-test. The test indicated that the P_{Eobs} performance ($M = 83.72$, $SD = 3.56$) is significantly better than the P_{obs} performance ($M = 79.33$, $SD = 3.89$), $(t(24) = 9.51, p < .001, Cohen's\ d = 1.17)$. Thus eye-tracking features seem to be particularly helpful for predicting to which entity an RE is resolved in hard scenes.

The results also show a peak in accuracy near the -3 seconds mark. We computed a 2x2 contingency table that contrasts correct and incorrect predictions for P_{obs} and P_{Eobs}, i.e. whether o_i was

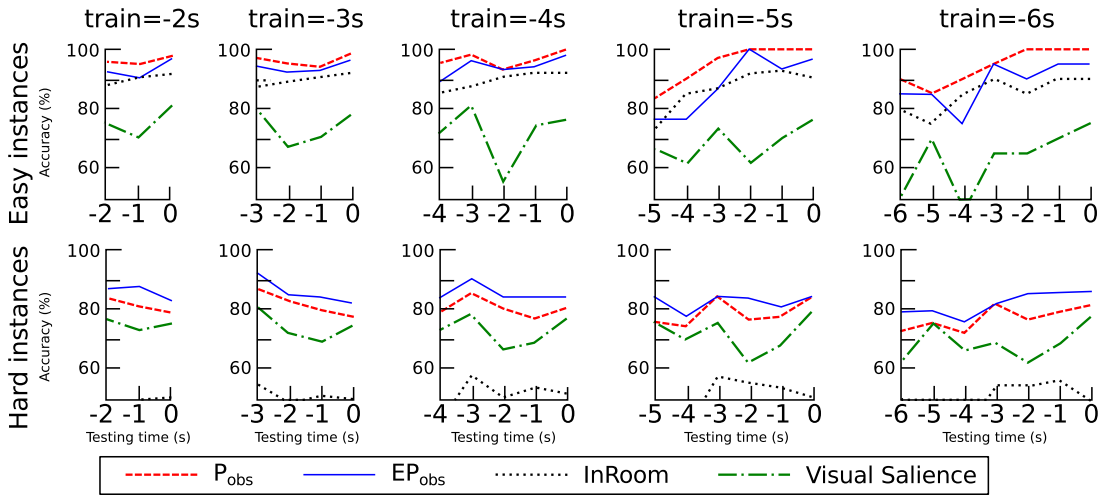

Figure 2: Accuracy as a function of training and testing time.

classified as target object or not. Data for this table was collected from all episode judgements for models trained at times in the $[-6\,sec., -3\,sec.]$ range and tested at -3 seconds. McNemar's test showed that the marginal row and column frequencies are significantly different ($p < 0.05$). This peak is related to the average required time between an utterance and the resulting target manipulation. This result shows that our model is more accurate precisely at points in time when we expect fixations to a target object.

7 Conclusion

In this paper we have shown that listener's gaze is useful by showing that accuracy improves over P_{obs} in the context of predicting the resolution of an RE. In addition, we observed that our model P_{Eobs} proves to be more robust than P_{obs} when the time interval between the prediction ($t_b - d_{test}$) and the button press (t_b) increases, i.e. gaze is especially beneficial in an early stage of an interaction. This approach shows significant accuracy improvement on hard referential scenes where more objects are visible.

We have also established that gaze is particularly useful when combined with some other simple features, as the features that capture listeners visual behaviour are not powerful enough to outperform even the simplest baseline. Gaze only benefits the model when it is added on top of features that capture the visual context, i.e. the current scene.

The most immediate future line of research is the combination of our P_{Eobs} model with the semantic model P_{sem}, in order to test the impact of the extended features in a combined model. If successful, such a model could provide reliable predictions for a significant amount of time before an action takes place. This is of particular importance when it comes to designing a system that automatically generates and online outputs feedback to confirm correct and reject incorrect intentions.

Testing with users in real time is also an area for future research. An implementation of the P_{obs} model is currently in the test phase, and an extension for the P_{Eobs} model would be the immediate next step. The model could be embedded in an NLG system to improve the automatic language generation in such scenarios.

Given that our work refers only to NLG systems, there's no possible analysis of speaker's gaze. However, it may be interesting to ask whether a human IG could benefit from the predictions of P_{Eobs}. We could study whether predictions based on the gaze (mis-)match between both interlocutors are more effective than simply presenting the IF's gaze to the IG and trusting the IG to correctly interpret this data. If such a system proved to be effective, it could point misunderstandings to the IG before either of the participants becomes aware of them.

Acknowledgements

This work was funded by the Cluster of Excellence on "Multimodal Computing and Interaction" of the German Excellence Initiative and the SFB 632 "Information Structure".

References

Susan E. Brennan, Katharina S. Schuhmann, and Karla M. Batres. 2013. Entrainment on the move and in the lab: The walking around corpus. In *Proceedings of the 35th Annual Conference of the Cognitive Science Society*, Berlin, Germany.

Herbert H. Clark and Meredyth A. Krych. 2004. Speaking while monitoring addressees for understanding. *Journal of Memory and Language*, 50(1):62–81, January.

Herbert H. Clark. 1996. *Using Language*. Cambridge University Press, May.

Nikos Engonopoulos, Martín Villalba, Ivan Titov, and Alexander Koller. 2013. Predicting the resolution of referring expressions from user behavior. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Seattle.

Andrew Gargett, Konstantina Garoufi, Alexander Koller, and Kristina Striegnitz. 2010. The give-2 corpus of giving instructions in virtual environments. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Joy E. Hanna and Susan E. Brennan. 2007. Speakers' eye gaze disambiguates referring expressions early during face-to-face conversation. *Journal of Memory and Language*, 57(4):596–615, November.

John M. Henderson and Tim J. Smith. 2007. How are eye fixation durations controlled during scene viewing? further evidence from a scene onset delay paradigm. *Visual Cognition*, 17(6-7):1055–1082.

Casey Kennington and David Schlangen. 2014. Comparing listener gaze with predictions of an incremental reference resolution model. *RefNet workshop on Psychological and Computational Models of Reference Comprehension and Production.*

Alexander Koller, Kristina Striegnitz, Andrew Gargett, Donna Byron, Justine Cassell, Robert Dale, Johanna Moore, and Jon Oberlander. 2010. Report on the Second NLG Challenge on Generating Instructions in Virtual Environments (GIVE-2). In *Proceedings of the 6th International Natural Language Generation Conference (INLG)*.

Alexander Koller, Maria Staudte, Konstantina Garoufi, and Matthew Crocker. 2012. Enhancing referential success by tracking hearer gaze. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, SIGDIAL '12, pages 30–39, Stroudsburg, PA, USA. Association for Computational Linguistics.

Maria Staudte, Alexander Koller, Konstantina Garoufi, and Matthew Crocker. 2012. Using listener gaze to augment speech generation in a virtual 3D environment. In *Proceedings of the 34th Annual Meeting of the Cognitive Science Society (CogSci)*, Sapporo.