Improving Sentence Completion in Dialogues with Multi-Modal Features

Anruo Wang, Barbara Di Eugenio, Lin Chen
Department of Computer Science
University of Illinois at Chicago
851 S Morgan ST, Chicago, IL 60607, USA
awang28, bdieugen, lchen43@uic.edu

Abstract

With the aim of investigating how humans understand each other through language and gestures, this paper focuses on how people understand incomplete sentences. We trained a system based on interrupted but resumed sentences, in order to find plausible completions for incomplete sentences. Our promising results are based on multi-modal features.

1 Introduction

Our project, called RoboHelper, focuses on developing an interface for elderly people to effectively communicate with robotic assistants that can help them perform Activities of Daily Living (ADLs) (Krapp, 2002), so that they can safely remain living in their home (Di Eugenio et al., 2010; Chen et al., 2011). We are developing a multi-modal interface since people communicate with each other using a variety of verbal and non-verbal signals, including haptics, i.e., force exchange (as when one person hands a bowl to another person, and lets go only when s/he senses that the other is holding it). We collected a medium size multi-modal human-human dialogue corpus, then processed and analyzed it. We observed that a fair number of sentences are incomplete, namely, the speaker does not finish the utterance. Because of that, we developed a core component of our multi-modal interface, a sentence completion system, trained on the set of interrupted but eventually completed sentences from our corpus. In this paper, we will present the component of the system that predicts reasonable completion structures for an incomplete sentence.

Sentence completion has been addressed within information retrieval, to satisfy user's information needs (Grabski and Scheffer, 2004). Completing sentences in human-human dialogue is more difficult than in written text. First, utterances may be informal, ungrammatical or dis-fluent; second, people interrupt each other during conversations (DeVault et al., 2010; Yang et al., 2011). Additionally, the interaction is complex, as people spontaneously use hand gestures, body language and gaze besides spoken language. As noticed by (Bolden, 2003), during face-to-face interaction, the completion problem is not only an exclusively verbal phenomenon but "an action embedded within a complex web of different meaning-making fields". Accordingly, among our features, we will include pointing gestures, and haptic-ostensive (H-O) actions, e.g., referring to an object by manipulating it in the real world (Landragin et al., 2002; Foster et al., 2008).

The paper is organized as follows. In Section 2 we describe our data collection and multi-modal annotation. In Section 3 we discuss how we generate our training data, and in Section 4 the model we train for sentence completion, and the results we obtain.

2 Dataset

In contrast with other sentence completion systems that focus on text input, the dataset we use in this paper is a subset of the ELDERLY-AT-HOME corpus, a multi-modal corpus in the domain of elderly care, which includes collaborative human-human dialogues, pointing gestures and haptic-ostensive (H-O) actions. Our experiments were conducted in a fully functional apartment and included a helper

Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL), pages 290–294,
Seoul, South Korea, 5-6 July 2012.

(HEL) and an elderly person (ELD). HEL helps ELD to complete several realistic tasks, such as putting on shoes, finding a pot, cooking pasta and setting the table for dinner. We used 7 web cameras to videotape the whole experiment, one microphone each to record the audio and one data glove each to collect haptics data. We ran 20 realistic experiments in total, and then imported the videos and audios (in avi format), haptics data (in csv format) and transcribed utterances (in xml format) into Anvil (Kipp, 2001) to build the multi-modal corpus.

Among other annotations (for example Dialogue Acts) we have annotated these dialogues for ***Pointing gestures and H-O actions***. Due to the setting of our experiments, the targets of pointing gestures and H-O actions are real life objects, thus we designed a reference index system to annotate them. We give pre-defined indices to targets which cannot be moved, such as cabinets, draws, and fridge. We also assign runtime indices to targets which can be moved, like pots, glasses, and plates. For example, "Glass1" refers to the first glass that appears in one experiment. In our annotation, a "Pointing" gesture is defined as a hand gesture without any physical contact between human and objects. Hand gestures with physical contact to objects are annotated as H-O actions. H-O actions are further subdivided into 7 subtypes, including "Holding", "Touching","Open" and "Close". In order to verify the reliability of our annotations, we double coded 15% of the pointing gestures and H-O actions. Kappa values of 0.751 for pointing gestures, and of 0.703 for H-O actions, are considered acceptable, especially considering the complexity of these real life tasks (Chen and Di Eugenio, 2012).

In this paper, we focus on specific sub-dialogues in the corpus, which we call interruptions. An interruption can occur at any point in human-human dialogues: it happens when presumably the interrupter (ITR) thinks s/he has already understood what the speaker (SPK) means before listening to the entire sentence. By observing the data from our corpus, we conclude that there are generally three cases of interruptions. First, the speaker (SPK) stops speaking and does not complete the sentence – these are the incomplete sentences whose completion a robot would need to infer. In the second type of interruption, after being interrupted SPK continues with (a) few words, and then stops without finishing the whole sentence: hence, there is a short time overlap between two sentences (7 cases). The third case occurs when the SPK ignores the ITR and finishes the entire sentence. In this case, the SPK and the ITR speak simultaneously (198 cases). The number of interruptions ranges from 1 to 37 in each experiment. An excerpt from an interruption with a subsequent completion (an example of case 3) is shown below. The interruption occurs at the start of the overlap between the two speakers, marked by $<$ and $>$. This example also includes annotations for pointing gestures and for H-O actions.

Elder: I need some glasses from $<$ that cabinet $>$.
[Point (Elder, Cabinet1)]
Helper: $<$ From this $>$ cabinet?
[Point (Helper, Cabinet2)]
Helper: Is this the glass you $<$ 're looking for? $>$
[Touching (Helper, Glass1)]
Elder: $<$ No, that one.$>$
[Point (Elder, Cabinet1, Glass2)]

As concerns annotation for interruptions, it proceeds from identifying *interrupted sentences* to finding $<$*interrupted sentences, candidate structure*$>$ pairs which will be used for generating grammatical completion for an incomplete sentence. Each interrupted sentence is marked with two categories: incomplete form, from the start of the sentence to where it is interrupted, such as "I need some glasses"; complete form, from the start of a sentence to where the speaker stops, "I need some glasses from that cabinet."

Table 2 shows distribution statistics for our ELDERLY-AT-HOME corpus. It contains a total of 4839 sentences, which in turn contain 7219 clauses. 320 sentences are incomplete in the sense of case 1 (after interruption SPK never completes his/her sentence); whereas 205 sentences are completed after interruption (cases 2 and 3).

Sentences	4,839
Clauses	7,219
Pointing Gestures	362
H-O Actions	629
Incomplete sentences	320
Interrupted sentences	205

Table 1: Corpus Distributions

3 Candidate Pairs Generation

The question is now, how to generate plausible training instances to predict completions for incomplete sentences. We use the 205 sentences that have been interrupted **but** for which we have completions; however, we cannot only use those pairs for training, since we would run the risk of overfitting, and not being able to infer appropriate completions for other sentences. To generate additional <*Interrupted sentences, candidate structure*> pairs, we need to match an interrupted sentence **IntS** with its potential completions – basically, to check whether IntS can match the prefix of other sentences in the corpus. We do so by comparing the POS sequence and parse tree of IntS with the POS sequence and parse tree of the prefix of another sentence. Both IntS and other sentences in the corpus are parsed via the Stanford Parser (Klein and Manning, 2003).

Before discussing the details though, we need to deal with one potential problem: the POS sequence for the incomplete portion of IntS may not be correctly assigned. For example, when the sentence 'The/DT, top/JJ, cabinet/NN.' is interrupted as 'The/DT, top/NN', the POS tag of NN is assigned to 'top'; this is incorrect, and engenders noise for finding correct completions.

We first pre-process a dialogue by splitting turns into sentences, tokenizing sentences into tokens, and POS tagging tokens. Although for the interrupted sentences, we could obtain a correct POS tag sequence by parsing the incomplete and resumed portions together, this would not work for a truly incomplete sentence (whose completion is our goal). Thus, to treat both interrupted sentences and incomplete sentences in the same way, we train a POS tag Correction Model to correct fallaciously assigned POS tags. The POS tag Correction Model's feature set includes the POS tag of the token, the word, and the previous tokens' POS tags in a window size of 3. The model outputs the corrected POS tags.

The POS tag Correction model described above was implemented using the Weka package (Hall et al., 2009). Specifically, we experimented with J48 (a decision tree implementation), Naive Bayes (NB), and LibSVM (a Support Vector Machine implementation). All the results reported below are calculated using 10 fold cross-validation.

	J48	**NB**	**LibSVM**
Accuracy	0.829	0.680	0.532

Table 2: POS tag Correction Model Performance

The results in Table 2 are not surprising, since detecting the POS tag of a known word is a simple task. Additionally, it is not surprising that J48 is more accurate than NB, since NB is known to often behave as a baseline method. What is surprising though is the poor performance of SVMs, which are generally among the top performers for a broad variety of tasks. We are investigating why this may be the case. At any rate, by applying the J48 model, we obtain more accurate POS tag assignments for interrupted sentences (and in our future application, for the incomplete sentence we need to complete).

Once we have corrected the POS assignments for each interrupted sentence IntS, we retrieve potential grammatical structures for IntS, by comparing IntS with the prefixes of all complete sentences in the corpus via POS tags and parse trees. Note that due to the complexity of building a parse tree correction model in our corpus, we only build a model to correct the POS tags, but ignore the possible incorrect parse trees of the incomplete portion of an interrupted sentence. The matching starts from the last word in IntS back to the first word, with weights assigned to each position in decreasing order. Due to the size of our corpus, it is not possible to find exactly matched POS tag sequences for every incomplete sentence; thus, we also consider the parsed tree structures and mismatched POS tags between IntS's and complete sentences by reducing weights according to the size of the matched phrases and distances of mismatched POS tags. After this, a matching score is calculated for each incomplete and candidate structure pair.

Due to the large number of candidate structures, only the top 150 candidate structures for each IntS are selected and manually annotated with three classifications: "R", when the candidate structure provides a grammatically "reasonable" structure, which can be used as a template for completion; "U", which means the candidate structure gives an "ungrammatical" structure, thus this candidate structure cannot be used as template for completion;

"T", the candidate structure is exactly the same as what the speaker was originally saying, as judged based on the video and audio records. An example of an incomplete sentence with candidate structures in each of the three categories is shown below.

It/PRP, feels/VBZ | It/PRP, feels/VBZ, good/JJR
[R] It/PRP, 's/VBZ, fine/JJ, like/IN, this/DT]
[U] We/PRP, did/VBD, n't/RB
[T] It/PRP, is/VBZ, better/JJR

10543 interrupted sentences and candidate pairs are generated. 5268 of those 10543 pairs (49.97%) were annotated as "Reasonable", 4727 pairs (44.85%) were annotated as "Unreasonable", and 545 pairs (5.17%) were annotated as "Same with original sentence".

Incomplete Sentence and Structure pairs	10,543
Reasonable structures (R)	5,268
Unreasonable structures (U)	4,729
Exactly same structures (T)	545

Table 3: Distribution of completion classifications

4 Results and Discussion

On the basis of the annotation, we trained a "Reasonable Structure Selection (RSS)" model via supervised learning methods. For each pair <IntS, Candidate>, the feature set includes word and POS tag of the tokens of IntS and its candidate structure sentence. Co-occurring pointing gestures and H-O actions for both IntS and Candidate are also included in the model. Co-occurrence is defined as temporal overlap between the gesture (pointing or H-O action) and the duration of the utterance. For each training instance, we include the following features:
IntS: <words, POS tags>, <Pointing (Person / Object / Location)>, <H-O action (Person / Object / Location / Type)>;
Candidate: <words/POS tags)>, <Pointing (Person / Object / Location)>, <H-O action (Person / Object / Location / Type)>;
<Matching Score>;
<Classification: R, U, or T>.
We trained the RSS model also using the Weka package. The same methods mentioned earlier (J48, NB and SVM) are used, with 10-fold cross-validations. Results are shown in Table 4. We

		J48	**NB**	**LibSVM**
Precision	R, U, T	0.822	0.724	0.567
	R, U	0.843	0.761	0.600
Recall	R, U, T	0.820	0.725	0.512
	R, U	0.842	0.762	0.563
F-Measure	R, U, T	0.818	0.711	0.390
	R, U	0.841	0.761	0.440

Table 4: Reasonable Structure Selection models

ran two different sets of experiments using two versions of training instances: Classification with three classes, R, U and T, and classification with two classes, R and U. When training with only two classes, the T instances are marked as R. We experimented with collapsing R and T candidates since T candidates may lead to overfitting, and some R candidates might even provide better structures for an incomplete sentence than what exactly one speaker had originally said. Not surprisingly, results improve for two-way classification. Based on the J48 model, we observed that the POS tag features play a significant part in classification, whereas the word features are redundant. Further, pointing gestures and H-O actions do appear in some subtrees of the larger decision tree, but not on every branch. We speculate that this is due to the fact that pointing gestures or H-O actions do not accompany every utterance.

5 Conclusions and Future Work

In this paper, we introduced our multi-modal sentence completion schema which includes pointing gestures and H-O actions in the corpus ELDERLY-AT-HOME. Our data shows that it is possible to predict what people will say, even if the utterance is not complete. Our promising results include multimodal features, which as we have shown elsewhere (Chen and Di Eugenio, 2012) improve traditional co-reference resolution models. In the near future, we will implement the last module of our sentence completion system, the one that fills the chosen candidate structure with actual words.

References

G.B. Bolden. 2003. Multiple modalities in collaborative turn sequences. *Gesture*, 3(2):187–212.

L. Chen and B. Di Eugenio. 2012. Co-reference via pointing and haptics in multi-modal dialogues. In *The 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics. short paper, to appear.

L. Chen, A. Wang, and B. Di Eugenio. 2011. Improving pronominal and deictic co-reference resolution with multi-modal features. In *Proceedings of the SIGDIAL 2011 Conference*, pages 307–311. Association for Computational Linguistics.

David DeVault, Kenji Sagae, and David Traum. 2010. Incremental interpretation and prediction of utterance meaning for interactive dialogue. *Dialogue and Discourse*, 2(1):143170.

B. Di Eugenio, M. Zefran, J. Ben-Arie, M. Foreman, L. Chen, S. Franzini, S. Jagadeesan, M. Javaid, and K. Ma. 2010. Towards effective communication with robotic assistants for the elderly: Integrating speech, vision and haptics. In *2010 AAAI Fall Symposium Series*.

M.E. Foster, E.G. Bard, M. Guhe, R.L. Hill, J. Oberlander, and A. Knoll. 2008. The roles of haptic-ostensive referring expressions in cooperative, task-based human-robot dialogue. In *Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction*, pages 295–302. ACM.

K. Grabski and T. Scheffer. 2004. Sentence completion. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 433–439. ACM.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA data mining software: An update. *SIGKDD Explorations*, 11(1). http://www.cs.waikato.ac.nz/ml/weka/.

M. Kipp. 2001. Anvil-a generic annotation tool for multimodal dialogue. In *Seventh European Conference on Speech Communication and Technology*.

D. Klein and C.D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430. Association for Computational Linguistics.

K.M. Krapp. 2002. *The Gale Encyclopedia of Nursing & Allied Health: DH*, volume 2. Gale Cengage.

F. Landragin, N. Bellalem, L. Romary, et al. 2002. Referring to objects with spoken and haptic modalities.

F. Yang, P.A. Heeman, and A.L. Kun. 2011. An investigation of interruptions and resumptions in multi-tasking dialogues. *Computational Linguistics*, 37(1):75–104.