Towards Multimodal Coreference Resolution for Exploratory Data Visualization Dialogue: Context-Based Annotation and Gesture Identification *

Abhinav Kumar and **Barbara Di Eugenio** and **Jillian Aurisano** and **Andrew Johnson**
Abeer Alsaiari and **Nigel Flowers**
Computer Science, University of Illinois at Chicago
Chicago, IL, USA
{akumar34/jauris2/bdieugen/aej}@uic.edu

Alberto Gonzalez and **Jason Leigh**
Information & Computer Sciences, University of Hawai'i at Manoa
Honolulu, HI, USA
{agon/leighj}@hawaii.edu

Abstract

The goals of our work are twofold: gain insight into how humans interact with complex data and visualizations thereof in order to make discoveries; and use our findings to develop a dialogue system for exploring data visualizations. Crucial to both goals is understanding and modeling of multimodal referential expressions, in particular those that include deictic gestures. In this paper, we discuss how context information affects the interpretation of requests and their attendant referring expressions in our data. To this end, we have annotated our multimodal dialogue corpus for context and both utterance and gesture information; we have analyzed whether a gesture co-occurs with a specific request or with the context surrounding the request; we have started addressing multimodal co-reference resolution by using Kinect to detect deictic gestures; and we have started identifying themes found in the annotated context, especially in what follows the request.

1 Introduction

The goals of our work are twofold. The first is to gain insight into how humans interact with complex data in order to make discoveries. It is well known that visualization is very effective for exploring large datasets and gaining insight into the underlying phenomena. However, users (particularly visualization novices) struggle with translating higher-level natural language queries to appropriate visualizations that could assist in answering their questions. Our first step has been to collect and analyze naturalistic dialogues in which novices explore such datasets. Based on the insights from the data collection, our second goal is that of developing a conversational interface that will automatically generate the appropriate visualizations by participating in a natural interaction with users. We already have a pipeline in place that creates visualizations in response to a limited type of spoken requests.

In this paper, we focus on the role that context and gestures play in the interpretation of both requests and referring expressions. We are certainly not the first ones to suggest that the context of a request and multimodality, specifically deictic gestures, are essential to providing a more natural interactive system. Already (Sinclair, 1992) showed that having knowledge of utterances prior to the current one helps the human better interpret the utterance, which can lead to improved disambiguation. Similarly, multimodal systems have been shown to be advantageous over unimodal systems (Jaimes and Sebe, 2007). One reason is that receiving multiple input signals rather than just speech can reduce the chances of misunderstandings as well as resolve ambiguities. Also, humans are able to interact more naturally by using gestures along with speech, making the experience more effective and natural.

This paper builds on our previous work (Aurisano et al., 2015; Aurisano et al., 2016; Kumar

*Supported by NSF awards IIS-1445751 and IIS-1445796

et al., 2016) and focuses on the following new contributions: annotation of context and gestures on our multimodal corpus; gesture detection using Kinect; and classification of contextual themes.

2 Related Work

There are several areas of research that are relevant to our work, the first one being the vast literature on multimodality – we will just focus on multimodal referring expressions in this paper. As is well known (Sinclair, 1992; Kehler, 2000; Goldin-Meadow, 2005; Landragin, 2006; Navarretta, 2011), in natural dialogue, the antecedents of linguistic referring expressions are often introduced via gestures; for example in our environment, the user can point to a street intersection on a map yet never have mentioned it earlier. Crucially from a computational point of view, including hand gestures information improves the performance of the reference resolution module (Eisenstein and Davis, 2006; Baldwin et al., 2009). Other sources of multimodal information are important as well, including eye gaze (Prasov and Chai, 2008; Iida et al., 2011; Liu et al., 2013), or haptic (force exchange) information (Foster et al., 2008; Chen et al., 2015), but we will not address those in this paper.

Several additional challenges concerning resolving referring expressions arise when humans interact with graphical representations. First, the user will likely expect that any visible object can be discussed (Byron, 2003). Second, the same expression can be used to refer to an entity in the domain or in the visualization (Qu and Chai, 2008). For example, in our domain, users can refer to a type of crime in the world (*Look how much theft around UIC*), or to the visual elements, e.g. dots, that represent theft (*Can you color theft red?*). As far as we know, only (LuperFoy, 1992) tried to account for different perspectives on a referent, by linking them to a so-called discourse peg; interestingly, she applied her approach to an interface for manipulating visualizations (Hollan et al., 1988).

If the graphical representation is presented on a large display, as in our case, yet additional challenges arise as concerns how humans interact with it, including window management problems (Robertson et al., 2005). Closer to our interests, not much work exists on interpreting deictic gestures directed to large displays, especially as concerns recognizing the target at a semantic level (Kim et al., 2017).

Finally, as regards interactive systems that generate data visualizations more in general, the vast majority of those are not focused on natural, conversational interaction: (Gao et al., 2015) does not provide two-way communication; the number of supported query types are limited in both (Cox et al., 2001) and (Reithinger et al., 2005), while (Sun et al., 2013) uses simple NLP methods that limit the extent of natural language understanding possible. EVIZA (Setlur et al., 2016), perhaps the closest project to our own, does provide a dialogue interface for users to explore visualizations; however, EVIZA focuses on supporting a user interacting with one existing visualization, and doesn't cover creating a new visualization, modifying the existing one, or interacting with more than one visualization at a time.

3 Foundational Work

As we describe in previously published work (Aurisano et al., 2015; Aurisano et al., 2016; Kumar et al., 2016) and briefly summarize here, our work rests on a new multimodal corpus that we collected, transcribed and started annotating, and on an NLP pipeline that can currently interpret a subset of the requests we observed in our data.

3.1 Corpus and Initial Annotations

The corpus was built by collecting spoken conversations from 15 subjects. Each subject interacted with a remote Data Analysis Expert (DAE) in a Wizard-of-Oz setup, to explore data visualizations on Chicago crime data to understand when and where to deploy police officers. In each session users went through multiple cycles of visualization construction, interaction and interpretation; these sessions lasted between 45 and 90 minutes. Users were invited to interact with the DAE as naturally as possible, and to think aloud about their reasoning. They viewed visualizations and limited communications from the DAE on a large, tiled-display wall. The DAE viewed the subject through two high-resolution, direct video feeds, and also had a mirrored copy of the tiled-display wall on two 4K displays. The DAE generated responses to questions using Tableau,[1] and used SAGE2 (Marrinan et al., 2014), a collaborative large-display middleware, to drive the display wall. The DAE could also communicate via a

[1] http://www.tableau.com

Words	Utterances	Directly Actionable Utts.
38,105	3,179	490

Table 1: Corpus size

chat window, but tried to behave like a system with limited dialogue capabilities would. Apart from greetings, and status messages (*sorry, it's taking long*) the DAE would occasionally ask for clarifications, e.g. *Did you ask for thefts or batteries*.[2] However, the DAE never responded with a message, if the query could be directly visualized; neither did the DAE engage in multi-turn elicitations of the user requirements.

The dialogues were transcribed in their entirety: some basic distributional statistics are presented in Table 1, which includes *directly actionable utterances*, the focus of our initial annotation effort. Three coders identified the directly actionable utterances, namely, those utterances[3] which directly affect what the DAE is doing; the rest are non-actionable think-aloud utterances (during which the user was expressing out-loud what he or she was thinking at the time). This was achieved by leaving an utterance unlabelled or labeling it with one of six directly actionable request types: 1. create new visualization (*Can I see number of crimes by day of the week?*); 2. modify existing visualization (*Umm, yeah, I want to take a look closer to the metro right here, umm, a little bit eastward of Greektown*); 3. window management operations (on windows on the screen) (*If you want you can close these graphs as I won't be needing it anymore*); 4. fact-based requests that don't need a visualization to be answered (*During what time is the crime rate maximum, during the day or the night?*) 5. clarification questions (*Okay, so is this statistics from all 5 years? Or is this for a particular year?*); 6. expressing preferences (*The first graph is a better way to visualize rather than these four separately*). After annotation, it was found that only 15% of the dialogue consisted of actionable requests while the remaining 85% were non-actionable think-aloud. We obtained an excellent intercoder agreement $\kappa = 0.84$ (Cohen, 1960) on labeling an utterance or leaving it unlabeled; and $\kappa = 0.74$ on the six types of actionable requests.

[2] *Batteries* in this context means *an offensive touching or use of force on a person without the person's consent* (Merriam-Webster).

[3] What counts as an utterance was defined at transcription.

3.2 The Articulate2 dialogue architecture

The current system's (Articulate2) process flow can be seen within the rectangular box in Figure 1. It begins by translating the request to logical form using the Google Speech API and NLP parsing. Three NLP structures are obtained: ClearNLP (Choi and McCallum, 2013) is used to obtain PropBank (Palmer et al., 2005) semantic role labels (SRLs), which are then mapped to Verbnet (Kipper et al., 2008) and Wordnet using SemLink (Palmer, 2009). The Stanford Parser is used to obtain the remaining two structures, i.e. the syntactic parse tree and dependency tree. On the basis of these three structures, a standard logical form is obtained. Then, a classifier determines the type of the request among the six request types we just described. At this point in time, Articulate2 can process the first three types of requests we discussed earlier: it will transform the logical form to SQL for *create new visualization* and *modify existing visualization* requests, or skip this step for window management operations (since data retrieval is not needed in this case). Finally, the system generates an appropriate visualization specification which is then executed by the Visualization Executor on the data returned by the execution of the SQL query. The system also stores each generated visualization specification to its dialogue history. At the moment, Articulate2 is limited by its inability to resolve referring expressions (e.g., it closes the most recently created visualization without checking if the user was referring to a different window on the screen). In this paper, we discuss what our data tells us on multimodal referring expressions, and discuss the first steps we have taken to model those computationally.

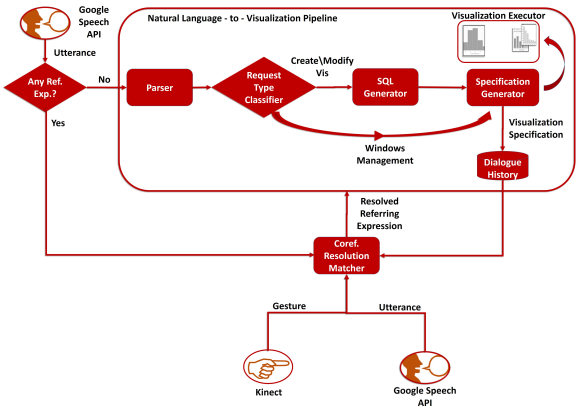


Figure 1: Articulate2 dialogue processing architecture

4 Corpus Analysis: Multimodal references in context

Preliminary analysis of the dialogue data showed that users referred to visualizations through speech, gestures, or both. In addition, sometimes clues about identifying the object referred to by the referring expression (in the form of speech or gesture) were found as part of the think-aloud nearby rather than temporally aligned with the actionable request. This is why we decided to extend our analysis to the context surrounding an actionable request (contextual utterance annotation) as well as any gestures that occurred during that context (contextual gesture annotation).

The context is comprised of three parts: setup, request, and conclusion. For the purpose of this work, we start from one single utterance annotated as an actionable request, and look at its preceding and following context. The setup includes utterances that come prior to the request while the conclusion includes utterances after the request. Since often the utterances just prior or after the request are part of a larger contiguous thought process that can be captured, all utterances up to and including the mention of a data attribute are included.

One example is shown in Figure 2 along with the corresponding annotation in ANVIL (Kipp, 2001) in Figure 3. The setup component includes just one utterance because *"June"*, *"July"*, and *"August"* are part of our data attribute set. The request component is always just the request utterance itself. Finally, in this example the conclusion part is also a single utterance, however not because it mentions data attributes, but rather because it is followed by another request, which signals the start of a new context. Also note that U_R in Figure 2 mentions a deictic referring expression *"that map"*; in the conclusion utterance, clues are provided about the referent by means of language (*"It's like that one right there or maybe it's that one"*) and gesture (the user points to multiple visualizations). We believe that the interplay between different components of the context, the referring expressions and the deictic gestures is crucial to properly resolving a referential expression, and to interpret a request.

4.1 Context and Gesture Annotation

We performed two separate annotations on the corpus, one to determine which utterances belong to the set-up and conclusion for a certain utterance

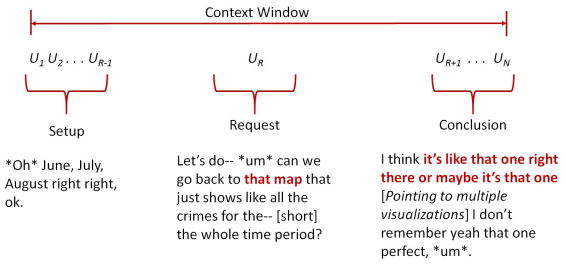


Figure 2: Context is comprised of setup, request, and conclusion utterances.

U_R, the second for gestures and their context.

Utterances. We use the label *Timestep* to apportion utterances to the three components of a context. By default, we start with an utterance U_R previously marked as an actionable request, which will be assigned the default *Timestep* value of *Current*. As we noted when discussing what type of requests Articulate2 currently processes, also here we focus only on the first three types of actionable requests (1. *create new visualization*; 2. *modify existing visualization*; 3. *window management operations*). This is a total of 449 requests out of the 490 in Table 1.

The utterances preceding the *Current* utterance, i.e. U_R, are coded as *Previous*; and those that follow the request and are pertaining to it as conclusion, are coded as *Next*. For an actionable utterance U_R then, the context includes all preceding utterances marked as *Previous* (the context set-up) and all following utterances marked as *Next* (the context conclusion). We obtained a very good $\kappa = 0.783$ on *Timestep* annotations for utterances.

Figure 4 shows the distribution of the coded *Timestep* values; and then two derived distributions, *Type* and *Context*. In all three plots, the "anchor" so to speak, is the *Current* utterance, i.e. U_R, the request of interest; hence, to the 449 *Current* utterances in Figure 4(a) correspond the 449 *Requests* in Figures 4(b) and 4(c).

Figure 4(a) shows the distribution of utterances preceding and following U_R, whereas Figure 4(b) shows how those utterances are apportioned within the context, i.e. as set-up or conclusions. By comparing Figures 4(a) and 4(b) we can conclude that set-up includes about 1.8 utterances on average, and conclusions about 2 utterances on average. Finally, Figure 4(c) simply confirms that no utterance either in the set-up or the conclusion is a directly actionable utterance. Whereas this follows by construction, the data confirms that no hu-

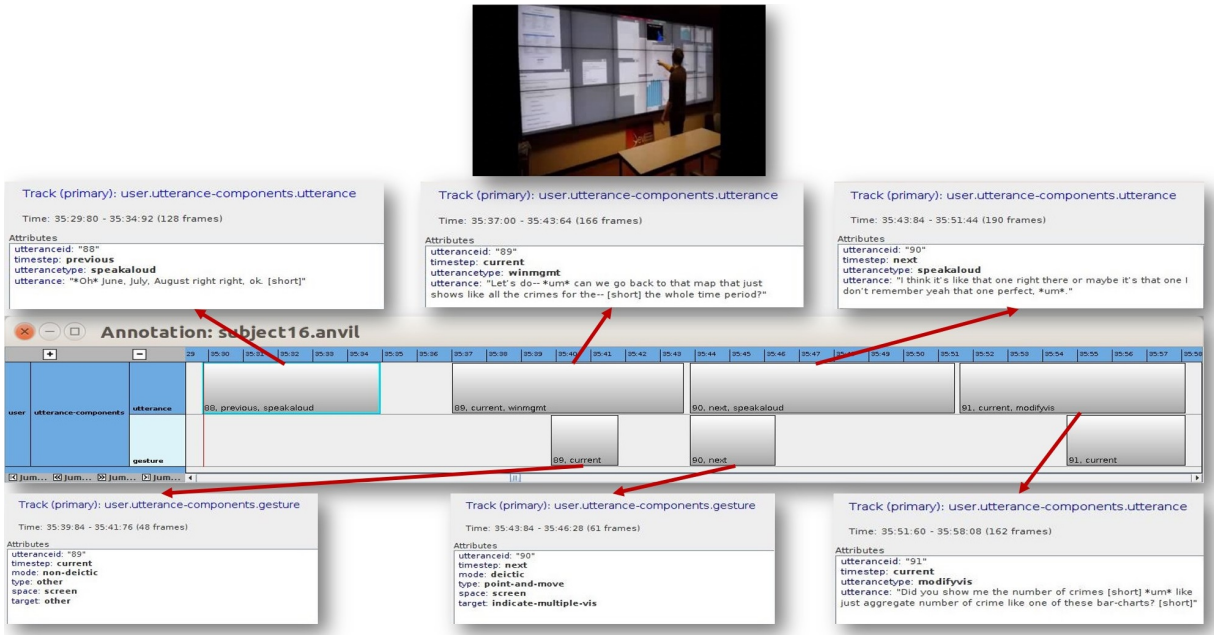


Figure 3: Annotation of a context in ANVIL (Kipp, 2001).

man errors occurred during annotation.

Gestures. The annotation for gestures includes various components, as shown in Figure 5. First, we mark the gesture with *Timestep*, as described above: the value for *Timestep* will be *Previous/Current/Next* depending on where the gesture occurs within the context of utterance U_R. Second, *Mode* is used to encode whether the gesture is *Deictic* (that is, whether the gesture is pointing to objects on the screen). If not, then it is *Non-Deictic* and the *Space* is assigned to *Peripheral* or *Screen*. The *Screen* value for *Space* pertains to gestures that the user makes in front of him or herself while interacting with the screen, while *Peripheral Space* is used if the gesture is made without screen interaction (Wagner et al., 2014). Note that for *Deictic* gestures, the *Space* will always be assigned to *Screen*, since pointing to objects on the screen is clearly interactive. Finally, if the gesture is *Deictic*, then its *Type* and *Target* are also assigned – the values for these two labels will be discussed shortly.

Table 2 provides intercoder agreement for various labels associated with gestures. Whereas κ values for *Timestep, Mode* and *Space* are substantial, the values for *Type* and *Target* are lower. This is not surprising: it is difficult to determine if the user is moving the hand while pointing, or keeping it stationary; and even more so, to distinguish between the four values the *Target* label can have, including deciding whether the user is pointing to a visualization, or to objects within a visualization.

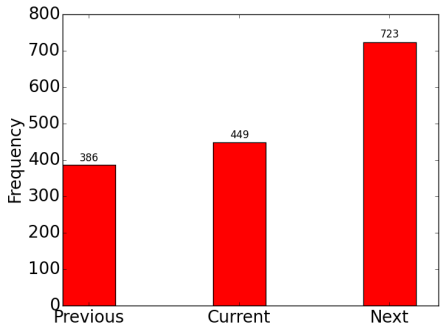
Code	κ_g
Timestep	0.718
Mode	0.748
Space	0.764
Type	0.659
Target	0.639

Table 2: Intercoder agreement for gestures

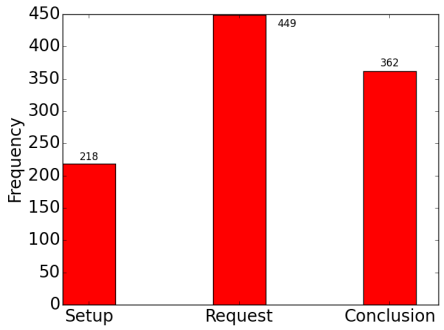
Figure 6 provides distributions for all the labels that are included in the gesture annotation. Figures 6(a) and 6(b) provide information about where gestures fall with respect to request U_R. These two graphs show that only about 50% of gestures are aligned with the actual request (*Current* in Figure 6(a) and *Request* in Figure 6(b)); about 17% of gestures co-occur with an utterance preceding U_R (*Previous/Setup*), and the remaining 33% co-occur with an utterance following U_R (*Next/Conclusion*).

Figure 6(c) shows that about 70% of gestures are deictic; and Figure 6(d) shows that subjects used gestures to interact with the screen far more than peripherally, since apart from the 380 deictic gestures, also 38 non-deictic gestures interact with the screen. Finally, Figures 6(e) and 6(f) focus on deictic gestures. [4] Figures 6(e) shows that

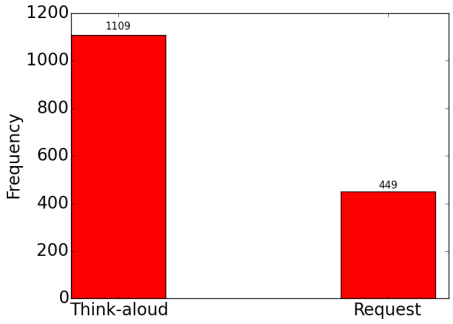
[4]The attentive reader will note that totals in Figures 6(e)



(a) Timestep Frequency



(b) Context Components Frequency



(c) Type Frequency

Figure 4: Utterance contextual labels

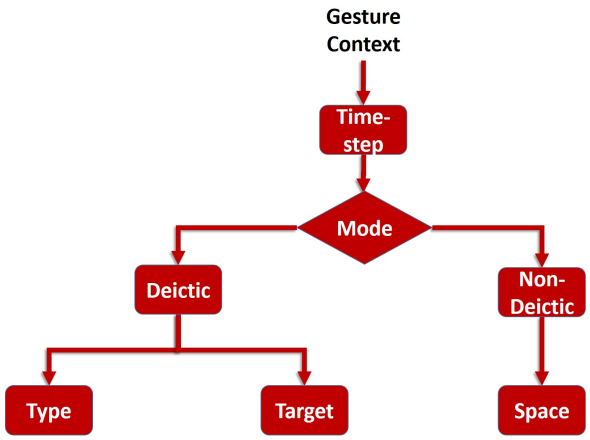


Figure 5: Coding scheme for gestures

most deictic gestures are exclusively pointing, or pointing while also moving the hand. Finally, Figure 6(f) provides the distributions of the targets for the deictic gestures. Three targets occur with similar frequencies: it is not surprising that users point to either individual visualizations, or individual objects within a visualization; it is less expected than they point to more than one individual object within a visualization so frequently. On the other hand, pointing to more than one visualization at the same time is not as common.

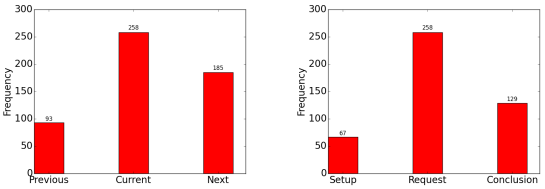
4.2 Lessons from Context Annotation

The most important lesson is that U_R does not occur in a vacuum: as demonstrated by Figure 4(b), about half of the time, an actionable request U_R is preceded by contextual information directly relevant to U_R itself; and even more frequently, about 80% of the times U_R is followed by pertinent information. The second important lesson is that about half of the gestures relevant to the interpretation of referring expressions contained within U_R are not aligned with U_R either. This is a crucial insight for coreference resolution.

5 Towards Multimodal Coreference Resolution

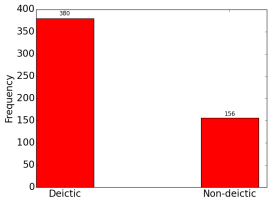
Our coreference resolution approach begins with the spoken contextual utterances from the user. If no referring expressions are detected, then the process flow described in Section 3.2 will be followed. Otherwise, if a gesture has been detected

and 6(f) are slightly lower (378 and 373 respectively), than 380, the number of deictic gestures in Figure 6(c). In both cases a very small number of gestures has been assigned an *Other* Type or Target. For the sake of brevity, we will not discuss the *Other* categories.

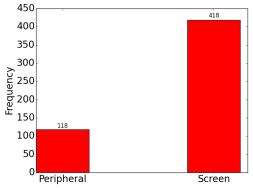


(a) Timestep Frequency

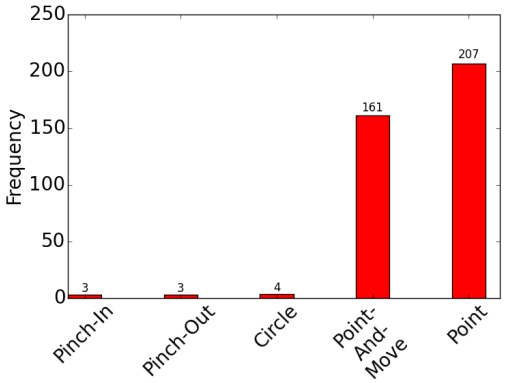
(b) Context Frequency



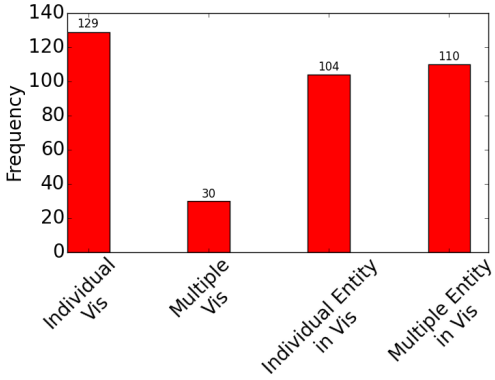
(c) Mode Frequency



(d) Space Frequency



(e) Deictic Gesture Type



(f) Deictic Gesture Target

Figure 6: Gesture features distributions

by the gesture detection process we will discuss in the next section, information about any objects pointed to by the user will be provided to the *Matcher*. The *Matcher* will then be invoked and attempt to find a best match between properties of each of the relevant entities. A difficulty we still need to address is to select the properties of visualizations and objects we will keep track of. A first inventory of good properties to extract from a visualization include, statistics in the data (e.g., neighborhoods with lowest and highest crime rates), trends in the data (e.g., top 5 and bottom 5 crime and location types), the title, plot type, and any more prominent objects within the visualization, such as hot-spots, street names, bus stops, and so on.

As noted earlier, when users are faced with a graphical representation, any object in the representation can become a referent. However an additional difficulty is that we do not have a declarative representation for all these potential referents. For example, in a map representation of crime occurrences, each crime is represented by a dot; however, the dot is procedurally generated by the graphics software to render one data point in the data; that individual dot does not exist as an individuated object in some declarative representation of the visualization. The reason for a lack of representation is that the language we used for generating visualizations (Vega (Trifacta, 2014)) abstractly performs behind-the-scenes operations on the data when producing graphs and does not directly provide access to individual objects.

5.1 Deictic Gesture Recognition

Whereas the *Matcher* still needs to be developed, we have made considerable progress on recognizing deictic gestures, to which we turn now.

Several approaches are proposed to estimate the pointing direction using Computer Vision techniques. One common method is to model the pointing direction as the line of sight that connects the joints of head and hand (Kehl and Van Gool, 2004). Using regular cameras to detect body joints is still a challenging task in Computer Vision since they lack information about the depth of the users body and surrounding environment.

Since its release in 2011, the Microsoft Kinect camera provided the capability of depth detection at a low cost. It combines depth and infrared cameras with a regular RGB camera for depth stream

acquisition and skeletal tracking. The Kinect camera has the ability to track 24 distinct joints of the human body in which the 3D coordinates of body joints can be obtained. Using the 3D information from the Kinect camera, we constructed a virtual touch screen originally defined by (Cheng and Takatsuka, 2006) and adapted later by (Jing and Ye-peng, 2013) to enable an efficient pointing gesture interaction with the large display. The user interacts with the large display through the constructed virtual touch screen to point to a specific visualization on the display.

5.1.1 Virtual Touch Screen Construction

First, we set up the interaction space by defining the physical space that will model the Kinect position and orientation in relation to the large display position. Each acquired joint position by the Kinect is rotated and translated so the center of the display represents the origin of the world coordinate. We receive data from the Kinect camera as a stream of 3D positions of body joints per frame. Although we can track all body joints, we focused only on the head and the fingertip of the right hand as dominant hand.

We created a virtual touch screen using head-fingertip positions to estimate the pointing target. As shown in Figure 7, the virtual screen is assumed to be at the position of the fingertip from the large display. Since the large display and the Kinect are in the same plane, the z coordinate of the large display is zero. Each point (x, y) on the large display is mapped to a point $(x^{'}, y^{'})$ on the virtual screen through a line from the large display to the head joint position (x_h, y_h, z_h). Therefore:

$$\frac{x_h - x}{x_h - x^{'}} = \frac{y_h - y}{y_h - y^{'}} = \frac{z_h - z}{z_h - z^{'}} \quad (1)$$

Hence, we can estimate any point (x, y) on the large display by calculating x and y from Equation 1.

$$x = \frac{z_h * (x^{'} - x_h)}{z_h - z^{'}} + x_h \quad (2)$$

$$y = \frac{z_h * (y^{'} - y_h)}{z_h - z^{'}} + y_h \quad (3)$$

The user interacts with the large display as if it was brought forward in front of him/her and we can map any point on the virtual screen to its corresponding point on the large display using the above equations. The position and dimensions of the virtual screen are calculated based on the positions of the head and fingertip, and subsequently, it is adaptive to the positions of the user head and fingertip. Using pointing data, it is possible to infer which visualization the user is pointing to – in particular, we are now also able to identify the window or windows that point *(x,y)* belongs to.

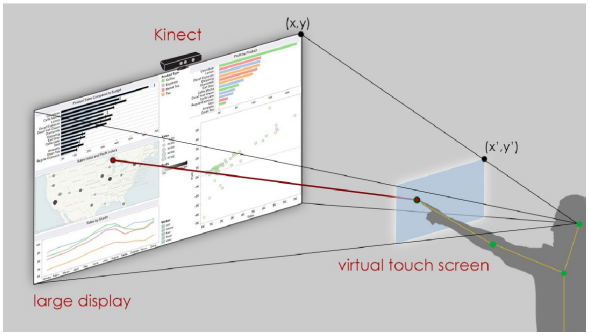


Figure 7: User interaction with large display through constructed virtual touch screen at user's fingertip.

6 Towards interpreting requests in context

As we noted earlier, requests don't occur in isolation: they are preceded by a set-up in 50% of the cases, and followed by a conclusion in 80% of the cases. The conversational interface clearly needs to take this information into account: the set-up in order to further refine the request, and the conclusion, in order to further the task itself. As a first step towards these goals, we focused on analyzing the conclusion component of a context, and specifically, on uncovering any relevant themes that may occur. In the conclusion part of the context, via additional annotation, it was found that the user would either: discuss resulting graphs produced from the current request (e.g., *"ok so it shows that the theft, battery, deceptive-practice and criminal-damage have the highest rate of *uh* crime."*), (2) refine the current request (e.g., *"thank you, i shall take a look at these, by the hour."*), (3) provide some insights (e.g., *"so then maybe if may–, if it gets cold, crime goes down at least the cops can go where its warm, maybe take their vacations in the winter."*, (4) or some unrelated utterances (e.g., *"ok, thank you. ok, thank you."*).

Figure 8 shows the distribution of these themes: 66% of conclusions discuss what the user gleaned from the request; of these, about 60% discuss the results directly, whereas an additional 6% dis-

cuss more general insights into the phenomenon at hand. About 20% represent a further refinement of the request, which sets the stage for the next request.

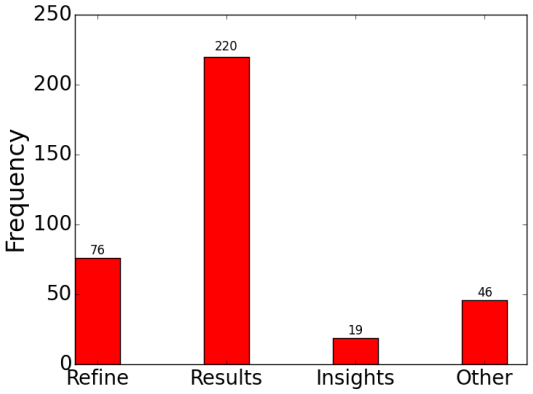


Figure 8: Conclusion utterances label frequency.

Given these annotations, we trained a supervised classification model to predict the overall theme of a set of conclusion utterances. The model used three different categories of feature types: syntactic, semantic, and miscellaneous. The syntactic feature types include unigrams, bigrams, and trigrams for words, part-of-speech, and tagged part-of-speech. The semantic category is based on the Word2Vec word embedding representation. Specifically, the utterances within a conclusion were added together by their corresponding Word2Vec vectors and then normalized. Finally, the remaining feature types include total number of words across a given conclusion, the total number of Chicago crime data attributes mentioned across a given conclusion, and the total number of utterances in the conclusion that ended with a question mark (because such utterances were observed to occur in the conclusions). The feature vector dimensions was *17,904* (feature selection was applied to reduce the dimensionality). Accuracy results when using different classifiers are shown in Table 3. Apart from Multinomial Naive Bayes, the other three classifiers all perform similarly. We will further investigate sources of confusion in classfication to improve their performance.

Classifier	**Accuracy**
Support Vector Machine	**74%**
Decision Tree	74%
Random Forest	73%
Multinomial Naive Bayes	64%

Table 3: Thematic conclusion classification accuracy.

7 Conclusions and Future Work

In this paper we presented our work on investigating the role context plays in interpreting requests and referential expressions in task-oriented dialogues about exploring complex data via visualizations. This work takes place in the context of our Articulate2 project. Our goals are both to gain insight into how people use visualizations to make discoveries about a domain, and to use our findings in developing an intelligent conversational interface to a visualization system. In previous work, we had collected a new corpus of dialogues, started annotating and analyzing it, and set up the NLP pipeline for the Articulate2 system.

Specifically as concerns context, in this paper we have presented how we annotated the context surrounding each of our directly actionable requests, and how we annotated for gestures also in context. We found that indeed an actionable request is preceded by a set-up 50% of the times, and followed by a conclusion 80% of the times. As concerns gestures, we found that (not surprisingly) the majority of them are interactional with respect to the screen and in fact deictic; however, we also found that half of the gestures relevant to the interpretation of referring expressions contained within the request are not aligned with the request, but with the setup, or more often, with the conclusions.

As concerns the computational modeling of our findings, so far, we have focused on recognizing deictic gestures via Kinect, and on learning classifiers for the themes contained in the conclusion component of a context.

Much work remains to be done. Apart from taking advantage of the context to refine and disambiguate requests, our most pressing work regards resolving referring expressions. As we noted, we still need to understand what specific properties of visualizations and objects within visualizations are the most useful for resolving referring expressions in our domain. From our findings on gestures and where they occur in the context, it is clear that our algorithm must be incremental. We also need to analyze the referring expressions that users use in our data, to assess how prevalent the phenomenon of a single referent playing a dual role (in the domain, or as a graphical element) is.

References

Jillian Aurisano, Abhinav Kumar, Alberto Gonzales, Khairi Reda, Jason Leigh, Barbara Di Eugenio, and Andrew Johnson. 2015. 'show me data': observational study of a conversational interface in visual data exploration. In *Information Visualization Conference, IEEE VisWeek*, Chicago, IL.

Jillian Aurisano, Abhinav Kumar, Alberto Gonzales, Khairi Reda, Jason Leigh, Barbara Di Eugenio, and Andrew Johnson. 2016. Articulate2: Toward a conversational interface for visual data exploration. In *Information Visualization Conference, IEEE VisWeek*, Baltimore, MD.

Tyler Baldwin, Joyce Y. Chai, and Katrin Kirchhoff. 2009. Communicative gestures in coreference identification in multiparty meetings. In *Proceedings of the 2009 International Conference on Multimodal Interfaces*, pages 211–218. ACM.

Donna K. Byron. 2003. Understanding referring expressions in situated language: Some challenges for real-world agents. In *Proceedings of the First International Workshop on Language Understanding and Agents for the Real World.*, pages 80–87.

Lin Chen, Maria Javaid, Barbara Di Eugenio, and Miloš Žefran. 2015. The roles and recognition of haptic-ostensive actions in collaborative multimodal human-human dialogues. *Computer Speech & Language*, 32:201–231, Nov.

Kelvin Cheng and Masahiro Takatsuka. 2006. Estimating virtual touchscreen for fingertip interaction with large displays. In *Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments*, pages 397–400. ACM.

Jinho D. Choi and Andrew McCallum. 2013. Transition-based dependency parsing with selectional branching. In *ACL*, pages 1052–1062.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46.

Kenneth Cox, Rebecca E Grinter, Stacie L Hibino, Lalita Jategaonkar Jagadeesan, and David Mantilla. 2001. A multi-modal natural language interface to an information visualization environment. *International Journal of Speech Technology*, 4(3):297–314.

Jacob Eisenstein and Randall Davis. 2006. Gesture Improves Coreference Resolution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 37–40.

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.

Tong Gao, Mira Dontcheva, Eytan Adar, Zhicheng Liu, and Karrie G Karahalios. 2015. Datatone: Managing ambiguity in natural language interfaces for data visualization. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, pages 489–500. ACM.

S. Goldin-Meadow. 2005. *Hearing gesture: How our hands help us think*. Harvard University Press.

James Hollan, Elaine Rich, William Hill, David Wroblewski, Wayne Wilner, Kent Wittenburg, Jonathan Grudin, and Members Human Interface Laboratory. 1988. An introduction to hits: Human interface tool suite. Technical report, MCC.

Ryu Iida, Masaaki Yasuhara, and Takenobu Tokunaga. 2011. Multi-modal reference resolution in situated dialogue by integrating linguistic and extra-linguistic clues. In *IJCNLP*, pages 84–92.

Alejandro Jaimes and Nicu Sebe. 2007. Multimodal human–computer interaction: A survey. *Computer vision and image understanding*, 108(1):116–134.

Pan Jing and Guan Ye-peng. 2013. Human-computer interaction using pointing gesture based on an adaptive virtual touch screen. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 6(4):81–92.

Roland Kehl and Luc Van Gool. 2004. Real-time pointing gesture recognition for an immersive environment. In *Automatic face and gesture recognition, 2004. proceedings. sixth ieee international conference on*, pages 577–582. IEEE.

Andrew Kehler. 2000. Cognitive Status and Form of Reference in Multimodal Human-Computer Interaction. In *AAAI 00, The 15th Annual Conference of the American Association for Artificial Intelligence*, pages 685–689.

Hansol Kim, Kun Ha Suh, and Eui Chul Lee. 2017. Multi-modal user interface combining eye tracking and hand gesture recognition. *Journal on Multimodal User Interfaces*, pages 1–10, March. Published on line.

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

Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. 2008. A large-scale classification of english verbs. *Language Resources and Evaluation*, 42(1):21–40.

Abhinav Kumar, Jillian Aurisano, Barbara Di Eugenio, Andrew E. Johnson, Alberto Gonzalez, and Jason Leigh. 2016. Towards a dialogue system that supports rich visualizations of data. In *SIGDIAL Conference*, pages 304–309, Los Angeles, CA.

F. Landragin. 2006. Visual perception, language and gesture: A model for their understanding in multimodal dialogue systems. *Signal Processing*, 86(12):3578–3595.

Changsong Liu, Rui Fang, and Joyce Y. Chai. 2013. Shared gaze in situated referential grounding: An empirical study. In *Eye Gaze in Intelligent User Interfaces*, pages 23–39. Springer.

Susann LuperFoy. 1992. The representation of multimodal user interface dialogues using discourse pegs. In *ACL*, pages 22–31.

Thomas Marrinan, Jillian Aurisano, Arthur Nishimoto, Krishna Bharadwaj, Victor Mateevitsi, Luc Renambot, Lance Long, Andrew Johnson, and Jason Leigh. 2014. Sage2: A new approach for data intensive collaboration using scalable resolution shared displays. In *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)*, pages 177–186. IEEE.

Costanza Navarretta. 2011. Anaphora and gestures in multimodal communication. In *Proceedings of the 8th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 2011), Faro, Portugal, Edicoes Colibri*, pages 171–181.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–105, March.

Martha Palmer. 2009. Semlink: Linking PropBank, Verbnet and FrameNet. In *Proceedings of the Generative Lexicon Conference*, pages 9–15.

Zahar Prasov and Joyce Y. Chai. 2008. What's in a gaze?: the role of eye-gaze in reference resolution in multimodal conversational interfaces. In *Proceedings of the 13th international conference on Intelligent user interfaces*, IUI '08, pages 20–29, New York, NY, USA. ACM.

S. Qu and J. Chai. 2008. Beyond attention: The role of deictic gesture in intention recognition in multimodal conversational interfaces. In *ACM 12th International Conference on Intelligent User interfaces (IUI)*.

Norbert Reithinger, Dirk Fedeler, Ashwani Kumar, Christoph Lauer, Elsa Pecourt, and Laurent Romary. 2005. Miamma multimodal dialogue system using haptics. In *Advances in Natural Multimodal Dialogue Systems*, pages 307–332. Springer.

George Robertson, Mary Czerwinski, Patrick Baudisch, Brian Meyers, Daniel Robbins, Greg Smith, and Desney Tan. 2005. The large-display user experience. *IEEE Computer Graphics and Applications*, 25(4):44–51.

Vidya Setlur, Sarah E. Battersby, Melanie Tory, Rich Gossweiler, and Angel X. Chang. 2016. Eviza: A natural language interface for visual analysis. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pages 365–377. ACM.

Melinda Sinclair. 1992. The effects of context on utterance interpretation: Some questions and some answers. *Stellenbosch Papers in Linguistics*, 25.

Yiwen Sun, Jason Leigh, Andrew Johnson, and Barbara Di Eugenio. 2013. Articulate: Creating meaningful visualizations. *Innovative Approaches of Data Visualization and Visual Analytics*, page 218.

Trifacta. 2014. Vega: A Visualization Grammar. https://vega.github.io/vega/.

Petra Wagner, Zofia Malisz, and Stefan Kopp. 2014. Gesture and speech in interaction: An overview. *Speech Communication*, 57:209–232.