Resolving Referring Expressions in Conversational Dialogs for Natural User Interfaces

Asli Celikyilmaz, Zhaleh Feizollahi, Dilek Hakkani-Tur, Ruhi Sarikaya
Microsoft
asli@ieee.org, zhalehf@microsoft.com
dilek@ieee.org, ruhi.sarikaya@microsoft.com

Abstract

Unlike traditional over-the-phone spoken dialog systems (SDSs), modern dialog systems tend to have visual rendering on the device screen as an additional modality to communicate the system's response to the user. Visual display of the system's response not only changes human behavior when interacting with devices, but also creates new research areas in SDSs. On-screen item identification and resolution in utterances is one critical problem to achieve a natural and accurate human-machine communication. We pose the problem as a classification task to correctly identify intended on-screen item(s) from user utterances. Using syntactic, semantic as well as context features from the display screen, our model can resolve different types of referring expressions with up to 90% accuracy. In the experiments we also show that the proposed model is robust to domain and screen layout changes.

1 Introduction

Todays natural user interfaces (NUI) for applications running on smart devices, e.g, phones (SIRI, Cortana, GoogleNow), consoles (Amazon FireTV, XBOX), tablet, etc., can handle not only simple spoken commands, but also natural conversational utterances. Unlike traditional over-the-phone spoken dialog systems (SDSs), user hears and sees the system's response displayed on the screen as an additional modality. Having visual access to the system's response and results changes human behavior when interacting with the machine, creating new and challenging problems in SDS.

[System]: How can i help you today ?
[User]: *Find non-fiction books by Chomsky.*
[System]: (Fetches the following books from database)

[User]: ***"show details for the oldest production"*** or
"details for the syntax book" or
"open the last one" or
"i want to see the one on linguistics" or
"bring me Jurafsky's text book"

Table 1: A sample multi-turn dialog. A list of second turn utterances referring to the last book (in bold) and a new search query (highlighted) are shown.

Consider a sample dialog in Table 1 between a user and a NUI in the books domain. After the system displays results on the screen, the user may choose one or more of the on-screen items with natural language utterances as shown in Table 1. Note that, there are multiple ways of referring to the same item, (e.g. the last book)[1]. To achieve a natural and accurate human to machine conversation, it is crucial to accurately identify and resolve referring expressions in utterances. As important as interpreting referring expressions (REs) is for modern NUI designs, relatively few studies have investigated withing the SDSs. Those that do focus on the impact of the input from multimodal interfaces such as gesture for understanding (Bolt, 1980; Heck et al., 2013; Johnston et al., 2002), touch for ASR error correction (Huggins-Daines and Rudnicky, 2008), or cues from the screen (Balchandran et al., 2008; Anastasiou et al., 2012). Most of these systems are engineered for a specific

[1] An item could be anything from a list, e.g. restaurants, games, contact list, organized in different lay-outs on the screen.

Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 2094–2104,
October 25-29, 2014, Doha, Qatar.

task, making it harder to generalize for different domains or SDSs. In this paper, we investigate a rather generic contextual model for resolving natural language REs for on-screen item selection to improve conversational understanding.

Our model, which we call FIS (Flexible Item Selection), is able to (1) detect if the user is referring to any item(s) on the screen, and (2) resolve REs to identify which items are referred to and score each item. FIS is a learning based system that uses information from pair of user utterance and candidate items on the screen to model association between them. We cast the task as a classification problem to determine whether there is a relation between the utterance and the item, representing each instance in the training dataset as relational features.

In a typical SDS, the spoken language understanding (SLU) engine maps user utterances into meaning representation by identifying user's intent and token level semantic slots via a semantic parser (Mori et al., 2008). The dialog manager uses the SLU components to decide on the correct system action. For on-screen item selection SLU alone may not be sufficient. To correctly associate the user's utterance with any of the on-screen items one would need to resolve the relational information between the utterance and the items. For instance, consider the dialog in Table 1. SLU engine can provide signals to the dialog model about the selected item, e.g., that "*linguistics*" is a book-genre or content, but may not be enough to indicate which book the user is referring. FIS module provides additional information for the dialog manager by augmenting SLU components.

In §3, we provide details about our data as well as data collection and annotation steps. In §4, we present various syntactic and semantic features to resolve different REs in utterances. In the experiments (§6), we evaluate the individual impact of each feature on the FIS model. We analyze the performance of the FIS model per each type of REs. Finally, we empirically investigate the robustness of the FIS model to domain and display screen changes. When tested on a domain that is unseen to the training data or on a device that has a different NUI design, the performance only slightly degrades proving its robustness to domain and design changes.

2 Related Work

Although the problems of modern NUIs on smart devices are fairly new, RE resolution in natural language has been studied by many in NLP community.

Multimodal systems provide a natural and effective way for users to interact with computers through multiple modalities such as speech, gesture, and gaze. Since the first appearance of the Put-That-There system (Bolt, 1980), a number of multimodal systems have been built, among which there are systems that combine speech, pointing (Neal, 1991), and gaze (Koons et al., 1993), systems that engage users in an intelligent conversation (Gustafson et al., 2000). Earlier studies have shown that multimodal interfaces enable users to interact with computers naturally and effectively (Schober and Clark, 1989; Oviatt et al., 1997). Considered as part of the situated interactive frameworks, many work focus on the problem of predicting how the user has resolved REs that is generated by the system, e.g., (Clark and Wilkes-Gibbs, ; Dale and Viethen, 2009; Gieselmann, 2004; Janarthanam and Lemon, 2010; Golland et al., 2010). In this work, focusing on smart devices, we investigate how the system resolves the REs in user utterances to take the next correct action.

In (Pfleger and J.Alexandersson, 2006) a reference resolution model is presented for a question-answering system on a mobile, multi-modal interface. Their system has several features to parse the posed question and keep history of the dialog to resolve co-reference issues. Their question-answering model uses gesture as features to resolve queries such as "*what's the name of that [pointing gesture] player*?", but they do not resolve locational referrals such as "*the middle one*" or "*the second harry potter movie*". Others such as (Funakoshi et al., 2012) resolve anaphoric ("*it*") or exophoric ("*this one*") types of expressions in user utterances to identify geometric objects. In this paper, we study several types of REs to build a natural and flexible interaction for the user.

(Heck et al., 2013) present an intent prediction model enriched with gesture detector to help disambiguate between different user intents related to the interface. In (Misu et al., 2014) a situated in-car dialog model is presented to answer drivers' spoken queries about their surroundings (no display screen). They integrate multi-modal inputs of

speech, geo-location and gaze. We investigate a variety of REs for visual interfaces, and analyze automatic resolution in a classification task introducing a wide range of syntactic, semantic and contextual features. We look at how REs change with screen layout comparing different devices. To the best of our knowledge, our work is first to analyze REs from these aspects.

3 Data

Crowdsourcing services, such as Amazon Mechanical Turk or CrowdFlower, have been extensively used for a variety of NLP tasks (Callison-Burch and Dredze, 2010). Here we explain how we collected the raw utterances from CrowdFlower platform (crowdflower.com).

For each HITApp (Human Intelligence Task Application), we provide judges with a written explanation about our Media App, a SDS built on a device with a large screen which displays items in a grid style layout, and what this particular system would do, namely search for books, music, tv and movies media result [2] Media App returns results based on the user query using an already implemented speech recognition, SLU and dialog engines. For each HIT, the users are shown a different screenshot showing the Media App's search results after a first-turn query is issued (e.g., "*find non-fiction books by Chomsky*" in Table 1). Users are asked to provide five different second turn text utterances for each screenshot. We launch several hitapps each with a different prompt to cover different REs.

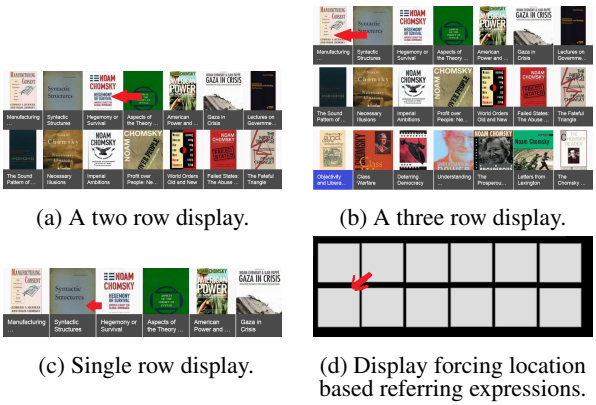
3.1 HITApp Types and Data Collection

A grid of media items is shown to the user with a red arrow pointing to the media result we want them to refer to (see Fig. 1). They can ask to play (an album or an audio book), select, or ask details about the particular media item. Each item in each grid layout becomes a different HIT or screenshot.

3.1.1 Item Layout and Screen Type Variation

The applications we consider have the following row×column layouts: 1×6, 2×6 and 3×6, as shown in Fig. 1 (columns vary depending on the returned item size). By varying the layout, we expect the referent of the last and the bottom layer items to change depending on how many rows, or

[2]Please e-mail the first author to inquire about the datasets.



(a) A two row display.

(b) A three row display.

(c) Single row display.

(d) Display forcing location based referring expressions.

Figure 1: Sketches of different HITApp Screens. The red arrows point to the media we want the annotators to refer.

columns exist in the grid. In addition, phrases like "*middle*", or *"center"* would not appear in the data when there are only one or two rows. Also, we expect that the distribution of types of utterances to vary. For example, in a grid of 1×6, "*the second one*" makes sense, but not so much on a 2×6 grid. We expect similar change based on the number of columns.

We use two kinds of screenshots to collect utterances with variations in REs. The first type of screenshots are aimed to bias the users to refer to items '*directly*' using (full/partial) titles or '*indirectly*' using other descriptors, or meta information such as year the movie is taken, or the author of the book. To collect utterances that indirectly referred to items, we need to show screen shots displaying system results with common titles, eventually forcing the user to use other descriptors for disambiguation. For example, given the first turn query "*find harry potter movies*", the system returns all the *Harry Potter* series, all of which contain the words *Harry Potter* in the title. The user can either refer in their utterance with the series number, the subtitle (e.g. *The prisoners of Azkaban*) or the location of the movie in the grid or by date, e.g., "*the new one*",

Because some media items have long titles, or contain foreign names that are not easy to pronounce, users may chose to refer these items by their location on the display, such as "*top right*", "*first album*", "*the movie on the bottom left*", etc. The second type of screen shots contains a template for each layout with no actual media item (Fig. 1(d)) which simply forces user to use locational references.

3.1.2 Interface Design Variation

In order to test our model's robustness to a different screen display on a new device, we employ an additional collection running another application named *places*, designed for hand-held devices. The places application can assist users in finding local businesses (restaurants, hotels, schools, etc.). and by nature of the device size can display fewer media items and arranges them in a list (one column). The number of items on the screen at any given time depends on the size of the hand-held device screen. The user can scroll down to see the rest of the results. Our collection displays the items in a 3, 4, and 5-rows per 1 column layout as shown in Fig. 2. We use the same variations in prompts as in §3.1. To generate the HitApp screens, we search for nearby places, in the top search engines (Google, Bing) and collect the results to the first turn natural language search queries (e.g.,"*find me sushi restaurants near me*").

Figure 2: A HitApp screen of places app. Items returned by the system regarding the first-turn utterance *"burger places near me?"*

3.2 Data Annotation

We collect text utterances using our media and places application. Using a similar HitApp we labeled each utterance with a domain, intent and segments in utterance with slot tags (see Table 2). The annotation agreement, Kappa measure (Cohen, 1960) is around 85%. Since we are building a relational model between utterances and each item on the screen, we ask the annotators to label each utterance-item as '0' or '1' indicating if the utterance is referring to that item or not. '1' means the item is the intended one. '0' indicates the item is not intended one or the utterance is not referring to any item on the screen, e.g., new search query. We also ask the annotators to label each utterance whether they contain locational (spatial) references.

Domain	Intents (I) & Slots
movie	I: find-movie/director/actor,buy-ticket Slots: name, mpaa-rating (*g-rated*), date,
books	I: find-book, buy-book, Slots: name, genre(*thriller*), author, publisher,
music	I: find-album, find-song, Slots: song-name, genre, album-type,...
tv	I: find-tvseries/play/add-to-queue.. Slots: name, type(*cartoon*), show-time....
places	I: find-place, select-item(*first one*).. Slots: place-type, rating, nearby(*closest*)....

Table 2: A sample of intents and semantic slot tags of utterance segments per domain. Examples for some slots values are presented in parenthesis as *italicized*.

3.3 Types of Observed Referring Expressions

We observe four main categories of REs in the utterances that are collected by varying the prompts and HITApp screens in crowd-sourcing:

Explicit Referential : Explicit mentions of whole or portions of the title of the item on the screen, and no other descriptors, e.g.,"*show me the details of star wars six*" (referring to the item with title "Star wars: Episode VI - Return of the Jedi").

Implicit Referential : The user refers to the item using distinguishing features other than the title, such as the release or publishing date, writers, actors, image content (describing the item image), genre, etc. "*how about the one with Kevin Spacey*".

Explicit Locational : The user refers to the item using the grid design, e.g., "*i want to purchase the e-book on the bottom right corner*".

Implicit Locational : Locational references in relation to other items on the screen, e.g., "*the second of Dan Brown's book*" (showing two of the Dan Brown's book on the same row).

4 Feature Extraction for FIS Model

Here, provide descriptions of each set of features of FIS model used to resolve each expression.

4.1 Similarity Features (SIM)

Similarity features represent the lexical overlap between the utterance and the item's title (that is displayed on the user's screen) and are mainly aimed to resolve *explicit REs*. We represent each utterance u_i and item-title t_k as sequence of words:

$$u_i=\{w_i(1),\ldots,w_i(n_i)\}$$
$$t_k=\{w_k(1),\ldots,w_k(m_k)\}$$

item bigrams	<bos> call five guys and fries <eos>
<bos> five	
five guys	●—●
guys burgers	
burgers and	
and fries	●—●
fries <eos>	●—●

Table 3: Bigram overlap between the item "*five guys burgers and fries*" and utterance"*five guys and fries*".

where $w_i(j)$ and $w_k(j)$ are the *j*th word in the sequence. Since inflectional morphology may make a word appear in an utterance in a different form than what occurs in the official title, we use both the word form as it appears in the utterance and in the item title. For example, *burger* and *burgers*, or *woman* and *women* are considered as four distinct words and all included in the bag-of-words. Using this representation we calculate four different similarity measures:

Jaccard Similarity: A common feature that can represent the ratio of the intersection to the union of unigrams. Consider, for instance, u_i="*call five guys and fries*" and the item t_k="*five guys burgers and fries*" in Fig 2. The Jaccard similarity *S*(*i*,*k*) is:

$$S(i,k)=1-\left(\, c(r_i \cap r_k)/c(r_i \cup r_k)\,\right)$$

where the r_i and r_k are unigrams of u_i and t_k respectively. $c(r_i \cap r_k)$ is the number of common words of u_i and t_k, $c(r_i \cup r_k)$ is the total unigram vocabulary size between them. In this case, the *S*(*i*,*k*)=0.66.

Orthographic Distance: Orthographic distance represent similarity of two text and can be as simple as an edit distance (Levenshtein distance) between their graphemes. The Levenshtein distance (Levenshtein, 1965) counts the insertion, deletion and substitution operations that are required to transform an utterance u_i into item's title t_k.

Word Order: This feature represents how similar are the order of words in two text. Sentences containing the same words but in different orders may result in different meanings. We extend Jaccard similarity by defining bigram word vectors r_i and r_k and look for overlapping bigrams as in Table 3. Among 6 bigrams between them, only 2 are overapping, hence the word-order similarity is *S*(*i*,*k*)=0.33.

Word Vector: This feature is the cosine similarity between the utterance u_i and the item-title t_k that measures the cosine of the angle between them. Here, we use the unigram word counts to represent the word vectors and the word vector similarity is defined as: $S(i,k)=(r_i \cdot r_k)/\|r_i\| \cdot \|r_k\|$.

4.2 Knowledge Graph Features

This binary feature is used to represent overlap between utterance and the meta information about the item and is mainly aimed to resolve *implicit REs*.

First, we obtain the meta information about the on-screen items using Freebase (Bollacker et al., 2008), the knowledge graph that contains knowledge about classes (books, movies, ...) and their attributes (title, publisher, year-released, ...). Knowledge is often represented as the attributes of the instances, along with values for those properties. Once we obtain the attribute values of the item from Freebase, we check if any attribute overlaps with part of the utterance. For instance, given an utterance "*how about the one with Kevin Spacey*", and the item-title "*House of Cards*", the knowledge graph attributes include **year**(*2013*), **cast**(*Kevin Spacey*), **director**(*James Foley*),... We turn the freebase feature 'on' since the actor attribute of that item is contained in the utterance. We also consider partial matches, e.g., last name of the actor attribute.

This feature is also used to resolve implicit REs, with item descriptions, such as "*the messenger boy with bicycle*" referring to the media item *Ride Like Hell*, a movie about a bike messenger. The synopsis feature in Freebase fires the freebase meta feature as the synopsis includes the following passage: "*... in which real messenger boys are used as stunts...* ".

4.3 Semantic Location Labeler (SLL) Feature

This feature set captures spatial cues in utterances and is mainly aimed to resolve *explicit locational* REs. Our goal is to capture the location indicating tokens in utterances and then resolve the referred location on the screen by using an indicator feature. We implement the SLL (Semantic Location Labeler), a sequence labeling model to tag locational cues in utterances using Conditional Random Fields (CRF) (Lafferty et al., 2001).

We sampled a set of locational utterances from each domain to be used as training set. We asked the annotators to label tokens with four different semantic tags that indicate a location.

The semantic tags include row and column indicator tags, referring to the position or pivotal reference. For instance, in "*second from the top*", "*second*" is the `column-position`, and "*top*" is the `row-pivot`, indicating the pivotal reference of the row in a multi-row grid display. Also in "*third from the last*", the "*third*" is the `column-position`, and the "*last*" is the `column-pivot`, the pivotal reference of the column in a multi-column grid display. The fourth tag, `row-position`, is used when the specific row is explicitly referred, such as in "*the Harry Potter movie in the first row*".

To train our CRF-based SLL model we use three types of features: the current word, window words e.g., previous-word, next-word, etc., using five-window around the current word, and syntactic features from the part-of-speech (POS) tagger using the Stanford's parser (Klein and Manning, 2003).

Row Indicator Feature: This feature sets the relationship between the n-gram in an utterance indicated by the `row-position` or `row-pivot` tag and the item's row number on the screen. For instance, given SSL output `row-pivot('top')` and item's location *row*=1, the value of the feature is set to '1'. If no row tag is found by SLL, this feature is set to '0'. We use regular expressions to parse the numerical indicators, e.g., '*top*'='1'.

Column Indicator Feature: Similarly, this feature indicates if a phrase in utterance indicated by the `column-position` or `column-pivot` tag matches the item's column number on the screen. If SLL model tags `column-pivot('on the left')`, then using the item's column number(=1), the value of this feature is set to '1'.

4.4 SLU Features

The SLU (Spoken Language Understanding) features are used to resolve *implicit* and *explicit REs*.

For our dialog system, we build one SLU model per each domain to extract two sets of semantic attributes from utterances: user's intent and semantic slots based on a predefined semantic schema (see examples in Table 2). We use the best intent hypothesis as a categorical feature in our FIS model. Although FIS is not an intent detection model, the intent from SLU is an effective semantic feature in resolving REs. Consider second turn utterance such as "*weather in seattle*", which is a '*find*' intent that is a new search or not related to any item on the screen. We map SLU intents such as *find-book* or *find-place*, to more specific ones, so that the intent feature would have values such as *find*, *filter*, *check-time*, not specific to a domain or device. The intent feature helps us to identify if user's utterance is related to any item on the screen. We also use the best slot hypothesis from the SLU slot model and search if there is full overlap of any recognized slot value with either the item-title or the item meta-information from freebase. In addition, we include the longest slot value n-gram match as an additional feature. We add a binary feature per domain, indicating whether there is a slot value match. Because we are using generic intents as categorical features instead of specific intents, and a slot value match feature instead of domain specific slot types as features, our models are rather domain independent.

5 GBDT Classifier

Among various classifier learning algorithms, we choose the GBDT (gradient boosted decision tree) (Friedman, 2001; Hastie et al., 2009), also known as MART (Multiple Additive Regression Trees). GBDT[3] is an efficient algorithm which learns an ensemble of trees. We find the main advantage of the decision tree classifier as opposed to other non-linear classifiers such as SVM (support vector machines) (Vapnik, 1995) or NN (neural networks) (Bishop, 1995) is the interpretability. Decision trees are "white boxes" in the sense that per-feature gain can be expressed by the magnitude of their weights, while SVM or NN's are generally black boxes, i.e. we cannot read the acquired knowledge in a comprehensible way. Additionally, decision trees can easily accept categorical and continuous valued features. We also present the results of the SVM models.

6 Experiments

We investigate several aspects of the SISI model including its robustness in resolving REs for domain or device variability. We start with the details of the data and model parameters.

We collect around 16K utterances in the media domains (movies, music, tv, and books) and around 10K utterances in places (businesses and

[3]Treenet: http://www.salford-systems.com/products/treenet is the implementation of the GBDT which is used in this paper.

	Movies		Tv		Music		Book		Overall Media		Places	
Feature Description	GBDT	SVM	GBDT	SVM	GBDT	SVM	GBDT	SVM	GBDT	SVM	GBDT	SVM
SLL	79.6	77.1	62.0	62.0	77.1	76.5	63.7	63.0	83.6	82.7	67.9	68.9
SIM	86.6	85.7	78.7	74.1	84.9	84.0	81.6	77.3	88.5	88.3	67.1	66.5
Knowledge Graph (KG)	81.0	82.0	64.8	65.6	86.3	85.4	77.8	77.9	84.4	84.1	76.5	76.5
SLU (Gold)	91.7	91.8	89.1	88.5	87.8	87.5	86.3	84.9	83.7	83.2	77.8	71.1
SLU (Pred.)	75.8	72.6	80.3	79.8	84.3	84.1	82.4	82.4	81.4	80.9	71.4	67.8
SIM+SLL	90.9	90.2	87.2	87.1	85.9	86.2	88.5	87.6	91.9	91.9	78.9	73.4
SIM+SLL+KG	91.7	91.3	89.9	89.1	89.1	87.7	91.4	90.3	93.0	92.7	85.9	82.3
SIM+SLL+KG+SLU(Gold)	**96.2**	95.01	**95.2**	95.09	**90.3**	89.9	**94.6**	94.0	**93.7**	93.2	**86.3**	84.3
SIM+SLL+KG+SLU(Pred.)	90.9	90.8	92.3	92.00	86.9	85.7	93.1	93.0	89.3	88.9	85.7	83.9

Table 5: Performance of the FIS models on test data using different features. Acc:Accuracy,. SIM: similarity features; SLU:Spoken Language Understanding features (intent and slot features); SLL:Semantic Locational Labeler features; Gold: using true intent and slot values, Pred.: using predicted intent and slot values from the SLU models.

Model:	Movies	TV	Music	Books	Places
Intent Acc.	84.5%	87.4%	87.6%	98.1%	89.5%
Slot F-score	92.1F	89.4F	88.5F	86.6F	88.4F

Table 4: The performance of the SLU Engine's intent detection models in accuracy (Acc.) and slot tagging models in F-Score on the test dataset.

locations) domain. We also construct additional negative instances from utterance-item pairs using first turn non-selection queries, which mainly indicate a new search or starting over. In total we compile around 250K utterance-item pairs for media domains and 150K utterance-item pairs for the places domain.[4] We randomly split each collection into 60%-20%-20% parts to construct the train/dev/test datasets. We use the dev set to tune the regularization parameter for the GBDT and SVM using LIBSVM (Chang and Lin, 2011) with linear kernel.

We use the training dataset to build the SLU intent and slot models for each domain. For the intent model, we use the GBDT classifier with n-gram and lexicon features. The lexicon entries are obtained from Freebase and are used as indicator variables, e.g., whether the utterance contains an instance which exists in the lexicon. Similarly, we train a semantic slot tagging model using CRF method. We use n-gram features with up to five-gram window, and lexicon features similar to the intent models. Table 4 shows the accuracy and F-score values of SLU models on the test data. The slot and intent performance is consistent accroess domains. The books domain has only two intents and hence we observe much better intent performance compared to other domains.

[4]In the final version of the paper, we will provide annotated data sets on a web page, which is reserved due to blind review.

6.1 Impact of Individual FIS Features

In our first experiment, we investigate the impact of individual feature sets on FIS model's performance. We train a set of FIS models on the entire media dataset to investigate the per-feature gain on the test dataset for each domain. We also train another set of FIS models with the same feature sets, this time on the places dataset and present the results on the places test set. Table 5 shows the results. We measure the performance starting with individual feature sets, and then incrementally add each feature set. Note that the SLU feature set includes the categorical intent, binary slot-value match and the longest slot value n-gram match with the item's title or meta information. The SLL feature set includes two features indicating the row and column (see §4.3).

As expected, larger gains in accuracy are observed when features that resolve different REs are used. Resolving locational cues in utterances with SLL features considerably impacts the performance when used together with similarity (SIM) features. We see a positive impact on performance as we add the knowledge graph features, which are used to resolve implicit REs. Using only the predicted SLU features in feature generation without golden values degrades the performance. Although the results are not statistically significant, the GBDT outperforms the SVM for almost all models, except for a few models, where the results are similar. However, the models which

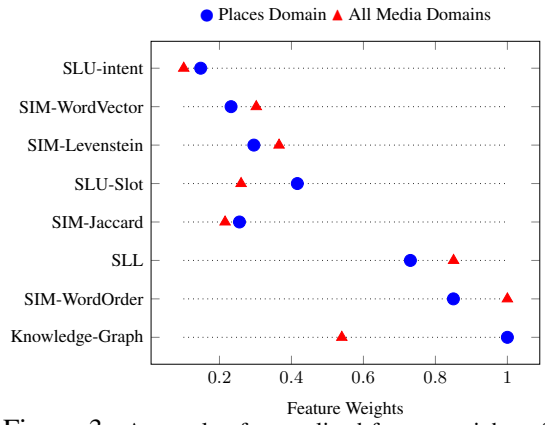


Figure 3: A sample of normalized feature weights of the GBDT FIS models across domains.

combine different features as apposed to individual feature set (the above the line models versusu below the horizantil line models) are statistically significant (based on the student t-test p¡0.01).

Next, we illustrate the significance of individual features across domains as well as devices. Fig. 3 compares the normalized feature weights of media and places domains. Across domains there are similar features with similar weight values such as SLU-intent, some similarity features (SIM-) and even spatial cue features (SLL). It is not surprising to observe that the places domain knowledge-graph meta feature weights are noticeably larger than all media model features. We think that this is due to the way the REs are used when the device changes (places app is on a phone with a smaller screen display). Especially, places application users refer items related to restaurants, libraries, etc., not so much by their names, but more so with implicit REs by using: the location (referring to the address: "*call the one on 31 street*") or cuisine ("*Chinese*"), or the star-rating ("*with the most stars*"), etc.

6.2 Resolution Across REs

We go on to analyze the performance of different RE types. A particularly interesting set of errors we found from the previous experiments are those that involve implicit referrals. Table 6 shows the distribution of different REs in the collected datasets.

Some noticeable instances with false positives for implicit locational REs include ambiguous cases or item referrals with one of its facets that require further resolution including comparison to other items, e.g., "*the nearest one*". Table 7 shows further examples. As might be expected, the locational cues are less common compared to other

	All Media		Places	
Utterance Type	%	Acc.	%	Acc.
All utterances	100%	93.7%	100%	86.3%
Direct/Indirect RE	81%	93.9%	73%	86.9%
Locational RE	19%	92.5%	28%	85.2%
Explicit RE	60%	94.3%	45%	88.4%
Implicit RE	21%	83.4%	28%	72.2%
Explicit Locational RE	15%	75.2%	24%	86.2%
Implicit Locational RE	3%	56.6%	2%	56.7%

Table 6: Distribution of referring expressions (**RE**) in the media (large screen like tv) and places (handheld device like phone) corpus and the FIS accuracies per RE type.

Utterance	Displayed on screen
"*the most rated restaurant*"	★★★'s next to each item
"*first thomas crown affair*"	original release (vs. remake)
"*second one over*"	(incomplete row/col. information)

Table 7: Display screen as user utters.

expressions. We also confirm that the handheld (places domain) users implicitly refer to the items more commonly compared to media app, and use the contextual information about the items such as their location, address, star-rating, etc. The models are considerably better at resolving explicit referrals (both non-spatial and spatial) compared to implicit ones. However, for locational referrals, the difference between the accuracy of implicit and explicit REs is significant (75.2% vs. 56.6% in media and 86.2% vs. 56.7% in places). Although not very common, we observe negative expressions, e.g., "*the one with no reviews*", which are harder for the FIS to resolve. They require quantifying over every other item on the screen, namely the context features, which we leave as a future work.

6.3 New Domains and Device Independence

In the series of experiments below, we empirically investigate the FIS model's robustness to when a new domain or device is introduced.

Robustness to New Domains: So far we trained media domain FIS models on utterances from all domains. To investigate how FIS models would behave when tested on a new domain, we train additional models by leaving out utterances from one domain and test on the left out domain. We used GBDT with all the feature sets. To set up an upper bound, we also train models on each individual domain and test on the same domain.

Table 8 shows the performance of the FIS mod-

	Models tested on:			
Model trained on:	**Movies**	**TV**	**Music**	**Books**
All domains	96.2%	95.2%	90.3%	94.6%
All other domains	94.6%	92.4%	89.7%	%
Only *this domain	96.4%	96.8%	93.4%	%

Table 8: Accuracy of FIS models tested on domains that are: seen at training time (all domains), unseen at training time (all other domains) and trained on individual domains.

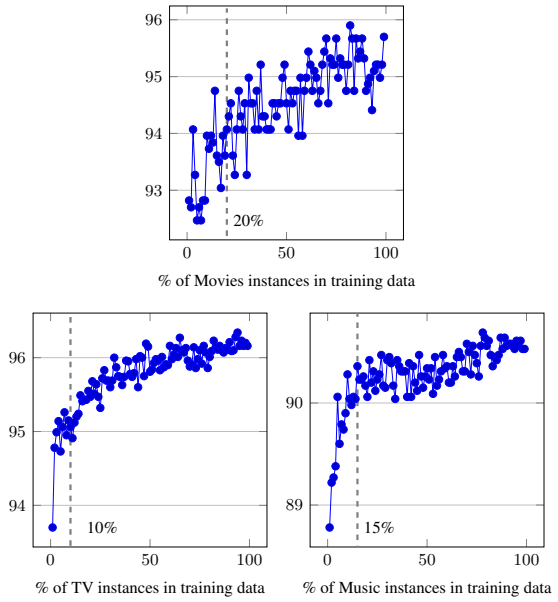


Figure 4: The accuracy (y-axis) versus the percentage (%) of in-domain utterances used in the training dataset. The dashed vertical line indicates an optimum threshold for the amount of in-domain data to be added to the training data.

els in accuracy on each media test domain. The first row shows the results when all domains are used at training time (same as in Table 5). The second row represents models where one domain is unseen at training time. We notice that the accuracy, although degraded for movies and tv domains, is in general not significantly effected by the domain variations. We setup another experiment, where we incrementally add utterances from the domain that we are testing the model on. For instance, we incrementally add random samples from movies training utterances on the dataset that does not contain movies utterances and test on all movies test data. The charts in Fig. 4 show the % improvement in accuracy as in-domain data is incrementally added to the training dataset. The results are interesting in that, using as low as 10-20% in-domain data is sufficient to build a flexible item selection model given enough utterances from other domains with varying REs.

Robustness to a New Device: The difference between the vocabulary and language usage observed in the data collected from the two devices

Media
"*only the new movies*" ; "*second one on the left*"
"*show me the thriller song*"; "*by Lewis Milestone*"
"*the first harry potter book*"
Places
"*directions to Les Schwab tire center*"
"*the closest one*" ;"*show me a map of ...*"
"*get hours of Peking restaurant*"; "*call Mike's burgers*"

Table 9: Sample of utterances collected from media and places applications illustrating the differences in language usage.

Trained on	Tested on Media	Tested on Places
Media	93.7 %	85.9%
Places	85.9%	86.3%
Media+Places	92.7%	85.8%

Table 10: Accuracy of FIS models tested on two separate devices (large screen media, and small screen places) that are unseen at test time.

is mainly due to changes in: (*i*) the screen design (places on phone has one column format wheres the media app has multi-column layout); (*ii*) the domain of the data. Table 9 shows some examples. Here, we add a little bit of complexity, and train one FIS model using the training data collected on one device and test the model on a different one, which is unseen at training time. Table 10 shows the comparisons for media and phone interfaces. The results are interesting. The performance of the places domain on phone does not get affected when the models are trained on the media data and tested on the phone device (86.3% down to 85.9% which is statistically insignificant). But when the data is trained on the places and tested on the media, we see a rather larger degradation on the performance (93.7% down to 85.9%). This is due to the fact that the media display screens are much complicated compared to phone resulting in a larger vocabulary with more variation in REs compared to places domain.

6.4 Conclusion

We presented a framework for identifying and recognizing referring expressions in user utterances of human-machine conversations in natural user interfaces. We use several on-screen cues to interpret whether the user is referring to on-screen items, and if so, which item is being referred to. We investigate the effect of different set of features on the FIS models performance. We also show that our model is domain and device independent which is very beneficial when new do-

mains are added to the application to cover more scenarios or when FIS is implemented on new devices. As a future work, we would like to adapt our model for different languages and include other features from multi modality including gesture or geo-location.

References

Dimitr Anastasiou and Cui Jian and Desislava Zhaekova. 2012. Speech and gesture interaction in an ambient assited living lab. *In. Proc. of the 1st Workshop on Speech and Multimodal Interaction in Assitive Environments at ACL'2012*.

Rajesh Balchandran, and Mark E. Epstein, and Gerasimos Potamianos, and Lsadislav Seredi. 2008. A multi-modal spoken dialog system for interactive tv. *In. Proc. of the 10th International Conference on Multimodal Interfaces*.

Christopher M. Bishop. 1995. *Neural networks for Pattern recognition*.

Kurt Bollacker and Colin Evans and Praveen Paritosh and Ttim Sturge and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. *In. Proc. of the 2008 International Conference on Management of Data (SIGMOD-08)*.

Richard A. Bolt. 1980. Put-that-there: Voice and gesture at the graphics interface. *Computer Graphics*.

Chris Callison-Burch and Mark Dredze. 2010. Creating speech and language data with amazons mechanical turk. *In. Proc. of NAACL*.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27.

Herbert H. Clark and Deanna Wilkes-Gibbs. Referring as colloborative processes.

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

Robert Dale and Jette Viethen. 2009. Referring expression generation through attribute-based heuristics. *In Proc. of the 12th European Workshop on Natural Language Generation (ENLG)*.

Jerome H. Friedman. 2001. Greedy function approximation: A gradient boosting machine. *Annals of Statistics, 2001*.

Kotaro Funakoshi, Mikio Nakano, Takenobu Tokunaga, and Ryu Iida. 2012. A unified probabilistic approach to referring expressions. *In Proc. of the Special Interest Group on Discourse and Dialog (SIGDIAL)*.

Petra Gieselmann. 2004. Reference resolution mechanisms in dialogue management. *In Proc. of the 8th Workshop on the semantics and pragmatics of dialogues (CATALOG)*.

Dave Golland, Percy Liang, and Dan Klein. 2010. A game-theoretic approach to generating spatial descriptions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 410–419, Cambridge, MA, October. Association for Computational Linguistics.

Joakim Gustafson, Linda Bell, Jonas Beskow, Johan Boye, Rolf Carlson, Jens Edlund, Bjorn Granstrom, David House, and Mats Wiren. 2000. Adapt - a multimodal conversational dialogue system in an apartment domain. *In Proc. of the 6th International Conference on Spoken Language Processing (ICSLP)*, pages 134–137.

Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. 2009. *The Elements of Statistical Learning (2nd ed.) Chapter 10. Boosting and Additive Trees, 2009*.

Larry Heck, Dilek Hakkani-Tur, Madhu Chinthakunta, Gokhan Tur, Rukmini Iyer, Partha Parthasarathy, Lisa Stifelman, Elizabeth Shriberg, and Ashley Fidler. 2013. Multi-modal conversational search and browse. *In Proc. of the IEEE Workshop on Speech, Language and Audio in Multimedia*.

Dvaid Huggins-Daines and Alexander I. Rudnicky. 2008. Interactive asr error correction for touchscreen devices. *In Proc. of ACL, Demo session*.

Srinivasan Janarthanam and Oliver Lemon. 2010. Adaptive referring expression generation in spoken dialog systems: Evaluation with real users. *In Proc. of SIGDIAL 2010: the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*.

Mark Johnston, Srinivas Bangalore, Gunaranjan Vasireddy, Amanda Stent, Patrick Ehlen, Marilyn Walker, and Steve Whittaker and Preetam Maloor. 2002. Match: an architecture for multimodal dialog systems. *In. Proc. of the ACL*.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing.

David B. Koons, Carlton J. Sparrell, and Kristinn R. Thorisson. 1993. Integrating simultaneous input from speech, gaze and hand gestures. *In Proc. of the In Maybury, M. (Ed.), Intelligent Multimedia Interfaces*, pages 257–276.

John Lafferty, Andrew McCallum, and Fernando C.N. Pereira. 2001. Conditional random fields: probabilistic models for segmenting and labeling sequence data. *In Proc. ICML*.

Vlademir Levenshtein. 1965. Binary codes capable of correcting deletions, insertions and reversals. *In Proc. of the Doklady Akademii Nauk SSSR*, 163:845–848.

Teruhisa Misu, Antoine Raux, Rakesh Gupta, and Ian Lane. 2014. Situated language understanding at 25 miles per hour. *In. Proc. of the SIGDIAL - Annual Meeting on Discourse and Dialogue*.

Renato De Mori, Frederic Bechet, Dilek Hakkani-Tur, Michael McTear, Giuseppe Riccardi, and Gokhan Tur. 2008. Spoken language understanding: A survey. *IEEE Signal Processing Magazine*, 25:50–58.

Joseph G. Neal. 1991. Intelligent multimedia interface technology. *In Proc. of the Intelligent User Interfaces: In Sullivan, J., and Tyler, S. (Eds.)*, pages 45–68.

Sharon Oviatt, Antonella DeAngeli, and Karen Khun. 1997. Integration and synchronization of input modes during multimodal human-computer interaction. *In Proc. of the Human Factors in Computing Systems: CHI*, pages 415–422.

Nobert Pfleger and Jan Alexandersson. 2006. Towards resolving referring expressions by implicitly activated referents in practical dialog systems. *In. Proc. of the 10th Workshop on the Semantics and Pragmatics of Dialog (SemDial-10)*.

Michael F. Schober and Herbert H. Clark. 1989. Understanding by addressees and overhearers. *In Proc. of the Cognitive Psychology*, pages 211–232.

Vlademrr Vapnik. 1995. *The nature of statistical learning theory*.