Data-Driven Language Understanding for Spoken Language Dialogue*

Nick Webb
Cristian Ursu
Yorick Wilks
Natural Language Processing Group
Department of Computer Science
University of Sheffield, UK
{n.webb,c.ursu,y.wilks}@sheffield.ac.uk

Hilda Hardy
Min Wu
Tomek Strzalkowski
ILS Institute
University at Albany
SUNY, Albany, NY, USA
{hhardy,minwu,tomek}@albany.edu

Abstract

We present a natural-language customer service application for a telephone banking call center, developed as part of the AMITIES dialogue project (Automated Multilingual Interaction with Information and Services). Our dialogue system, based on empirical data gathered from real call-center conversations, features data-driven techniques that allow for spoken language understanding despite speech recognition errors, as well as mixed system/customer initiative and spontaneous conversation. These techniques include robust named-entity extraction and vector-based task identification and dialogue act classification. Preliminary evaluation results indicate efficient dialogues and high user satisfaction, with performance comparable to or better than that of current conversational information systems.

Introduction

Spoken language dialogue systems (SLDS) have been in evidence for some time. Early systems were usually confined to highly structured domains, where a restricted, regularized language set could be expected, such as train timetabling scenarios, for example SUNDIAL (Peckham 1993) and ARISE (Lamel *et al.* 1999). Later dialogue systems, like those in the US Defense Advanced Research Projects Agency (DARPA) Communicator program (Walker *et al.* 2001; **?**), allowed more complete travel planning but still relied heavily on the regular nature of information interchange in these scenarios. Although by now competent at performing this role, with reasonable task completion rates (an average of 56% across Communicator), the systems are can be somewhat inflexible and abrupt in approach, often as a consequence of poor speech recognition.

Spoken language understanding in the context of these systems means the ability to recognise key concepts or entities in user input which relate to the domain. For example, in train time-tabling, we need the departure station, destination station, and dates of travel. This can be extremely simple, such as spotting keywords, or expecting specific input in response to directed system questions. More complex systems use deeper models of understanding, which try to construct deep linguistic representations of user input. Additionally, there is the effort involved in constructing or adapting understanding components to new tasks, domains or languages.

This paper outlines work performed as part of the AMITIES project to discover to what extent we can leverage recorded human-human dialogue data to build a data-driven SLDS. One effect of this work is the use of human-human data to characterize elements of an information-seeking dialogue which existing human-computer interfaces fail to represent. There are arguments which claim specifically that human-human data is not useful as a basis for building human-computer systems, because of the level of dysfluencies, ellipsis and anaphora (Dahlbäck & Jönsson 1992). Others point out that the usability of natural language systems, especially spoken language dialogue systems in the telecommunications setting, could profit from techniques that allow users to engage in more natural dialogues (Karis & Dobroth 1991).

We recognize that in recent years, the performance of key components in SLDS has increased dramatically. In particular, the level of word error rate (WER) for automatic speech recognition (ASR) engines has fallen to a level where perhaps we can forgo strategies of understanding which minimize the effect of poor speech recognition, and can experiment with established language processing technologies from other related fields of research, such as information retrieval (IR) and information extraction (IE).

The AMITIES project seeks to develop novel technologies for building empirically induced dialogue processors to support multilingual human-computer interaction, and to integrate these technologies into systems for accessing information and services[1].

Corpora

A large corpus of recorded, transcribed telephone conversations between real agents and customers gives us a unique opportunity to analyze and incorporate features of human-human dialogues into our automated system. The data we have collected spans two different application areas: soft-

*This paper is based on work supported in part by the European Commission under the 5th Framework IST/HLT Program, and by the U.S. Defense Advanced Research Projects Agency.

[1] http://www.dcs.shef.ac.uk/nlp/amities

ware support and customer banking. The financial domain corpus currently stands at 1,000 dialogues, comprising some 30,000 utterances and a vocabulary size of around 8,000 words. This corpus forms the basis of our initial multilingual triaging application, implemented for English, French and German (Hardy, Strzalkowski, & Wu 2003); as well as our prototype automatic financial services system, presented in this paper, which completes a variety of tasks in English (Hardy *et al.* 2004). The much larger software support corpus (10,000 calls in English and French) is still being collected and processed and will be used to develop the next AMITIES prototype. All conversations were recorded in call centers in Europe, and are anonymized following strict guidelines whereby all names, account numbers and personal information are removed from the audio data and replaced by generic substitutes in the word transcripts. For annotation, we have used a modified DAMSL tag set (Core & Allen 1997) to capture the functional layer of the dialogues, and a frame-based semantic scheme to record the semantic layer (Hardy *et al.* 2003). By adopting the general DAMSL markup scheme, we hope in a later stage to be able to acquire domain-independent models of dialogue, although we recognize that we might sacrifice some highly specific functional information. For example, travel domains have dialogue acts such as give-departure-airport, which indicate specific information inside the utterance. Models of dialogue derived from these tags would then be unsuitable if moved to the financial banking domain.

Related Work

Approaches to designing dialogue systems range from 1) finite state models, which are suitable for only the simplest of tasks, with a limited number of possible states and transitions; to 2) form-filling models, in which the user can answer system prompts with additional or out-of-order information to fill slots in a frame; to 3) more complex plan-based models, such as the classic TRAINS project, which require the system to reason about an explicit model of the domain, so that the system can conduct collaborative problem-solving between itself and the user (Allen *et al.* 1995), (Allen *et al.* 2001). There has been a proliferation of dialogue systems in the last decade, although the similarity of application domains is no accident. Early systems, such as SUNDIAL (Peckham 1993) and PEGASUS (Zue *et al.* 1994), dealt with flight and travel reservations. Such domains are highly structured, and the language used is easy to predict and model. SUNDIAL, for example, had a vocabulary of no more than 2000 words. Later systems tried new domains, such as the JUPITER conversational interface for weather information (Zue *et al.* 2000), or added new modalities to the spoken-language travel application, like a touch-screen for both the MASK kiosk, which provides train information and reservations (Lamel *et al.* 2002) and the MATCH system for restaurant and subway information (Johnston *et al.* 2002).

The US DARPA Communicator program has been instrumental in bringing about practical implementations of spoken dialogue systems. The goal was to support rapid development of speech-enabled dialogue systems, in the domain of travel planning. To achieve this, all sites involved in the program were encouraged to use a single dialogue platform, the Galaxy Communicator Software Infrastructure (GCSI) (Seneff *et al.* 1998). Systems developed under this program include CMU's script-based dialogue manager, in which the travel itinerary is a hierarchical composition of frames (Xu & Rudnicky 2000). The AT&T mixed-initiative system uses a sequential decision process model, based on concepts of dialogue state and dialogue actions (Levin *et al.* 2000). MIT's Mercury flight reservation system uses a dialogue control strategy based on a set of ordered rules as a mechanism to manage complex interactions (Seneff & Polifroni 2000). CU's dialogue manager is event-driven, using a set of hierarchical forms with prompts associated with fields in the forms. Decisions are based not on scripts but on current context (Ward & Pellom 1999).

Data-Driven Understanding

When talking about spoken language understanding, we have to differentiate between deep and shallow methods of understanding content. Shallow understanding can be as simple as spotting keywords, or having lists of, for example, every location recognised by the system. Several systems are able to decode directly from the acoustic signal into semantic concepts precisely because the speech recogniser already has access to this information (Young 2002; Wang, Mahajan, & Huang 2000). Deeper analysis can leverage a number of linguistic methods, including part-of-speech tagging, syntactic parsing and verb dependency relationships.

The trade-off between these two approaches is usually one of speed versus the ability to deal with ambiguity of one form or another. It is possible that deeper approaches can offer more help to the user in the case of misunderstandings or error correction; however, they can often be slower and it is not clear to what extent complex ambiguities occur in the kind of dialogue systems being deployed at present.

Here we present the spoken language understanding components of the AMITIES system. We have adopted existing language processing software to recognise key concepts in our target domain. This approach uses a series of shallow approaches, which in combination appear to achieve good results. We attempt understanding of the utterances based on the semantic content, in terms of key concepts defined by our domain, and functional structure, in terms of dialogue act detection.

Semantics

The goal of the natural language understanding component (NLU) is to take the word string output of the ASR module, and identify key semantic concepts relating to the target domain. Increasingly, recognition engines can perform direct acoustic wave to semantic concept recognition with some success, although this tends to be limited to highly specific domain-based applications (Young 2002). An alternative approach would be simple keyword spotting, although again this is restricted to straightforward slot-filler dialogues, where utterance length and context are such that

ambiguities, both across utterances and within the utterance (such as multiple keywords), are likely to be a rare phenomenon. Some speech providers, Nuance among them, offer an ASR-NLU integrated component with the possibility of directly filling slots with information from the recognized text. This has the advantage of an easy implementation, but at the cost of being heavily dependent on the ASR engine. Reliance on features implemented by a specific off-the-shelf product would have serious consequences for the system's flexibility. Within AMITIES, our goal is to create a multilingual system with the possibility of working with different recognizers. This led us to the decision of creating a multilingual, stand-alone NLU engine.

IE for dialogue

The recognition of key concepts in the utterance can be seen as a specialized kind of information extraction (IE) application, and we have adapted existing IE technology to this task. IE has made significant advances in recent years, in no small part thanks to the successful Message Understanding Conferences, also known as MUC (SAIC 1998). Although composed of many tasks, central to IE is the idea of named entity (NE) recognition. The NEs used in MUC tasks were specific to each application, but there are core entities used across domains, including person, location, organization, date, time, money and percent. The domain of AMITIES, that of financial banking, is such that we need to be able to recognize some additional NEs in order to perform the required tasks. Specifically, we are interested in account numbers, debit card numbers, person names, dates, amounts of money, addresses and telephone numbers.

For the AMITIES NLU component we have used a modified version of the ANNIE engine (A Nearly-New IE system; (Cunningham *et al.* 2002) (Maynard 2003). ANNIE is distributed as the default built-in IE component of the GATE framework (Cunningham *et al.* 2002). GATE is a pure Java-based architecture developed over the past eight years at the University of Sheffield Natural Language Processing Group. ANNIE has been used for many language processing applications, in a number of languages both European and non-European. This versatility makes it an attractive proposition for use in a multilingual speech processing project. ANNIE includes customizable components necessary to complete the IE task: tokenizer, gazetteer, sentence splitter, part of speech tagger and a named entity recognizer based on a powerful engine named JAPE (Java Annotation Pattern Engine) (Cunningham, Maynard, & Tablan 2000). If gazetteer information for each target language is available, the nature of ANNIE and the patterns needed to recognize entities mean that the process is to some extent language-independent, if appropriate processing resources are available. For example, some patterns used for entity recognition rely on part-of-speech tags, produced in the English version using a modified version of the Brill POS tagger (Brill 1992).

Given an utterance from the user, the NLU unit produces both a list of words for detecting dialogue acts, an important research goal inside this project, and a frame with the possible named entities specified by our application. By using a named entity recognizer based on a fast pattern-matching engine, we could use patterns to spot likely occurrences of entities despite the presence of recognition errors. This is a significant advantage over keyword spotting systems. In order to extract the entities used in AMITIES, we have updated the gazetteer, which works by explicit look-up tables of potential candidates, and modified the rules of the transducer engine, which attempts to match new instances of named entities based on local grammatical context. There are some significant differences between the kind of prose text more typically associated with information extraction, and the kind of text we are expecting to encounter. Current models of IE rely heavily on certain orthographic information, such as capitalized words indicating the presence of a name, company or location, as well as punctuation. We have access to neither of these in the output of the ASR engine, and so it was necessary to retune our processors to data which reflected that. In addition, we created new processing resources in the modified ANNIE, such as those required to spot number units and translate them into textual representations of numerical values; for example, to take twenty thousand one hundred and fourteen pounds, and produce 20,114. The ability to do this is of course vital for the performance of the system.

If none of the main entities can be identified from the token string, we create a list of possible fallback entities, in the hope that partial matching would help narrow the search space. For instance, when trying to recognise an account number, represented by a seven-digit numeric sequence, if we fail to recognise all the numbers, we still send an incomplete sequence to the database server for partial matching. The main strategy used by the NLU module in AMITIES was to present as much information as possible to the dialogue manager, where an interpretation could be made considering the context. For instance, the utterance twenty three four nineteen five oh (23 4 19 5 0) could be interpreted both as an account number (2341950) and as a date (23/4/1950). The NLU offers both formats to the dialogue manager, which can then choose based on context (e.g., account number has been supplied, but birth date is missing).

Functional Properties

In conjunction with recognising the semantic content, we try to ascertain the functional property of the utterance, in terms of the associated dialogue act. Dialogue act (DA) recognition is an important component of most spoken language systems. Searle (1969) introduced speech acts (extending the work of Austin (1962)) [2] as a fundamental concept of linguistic pragmatics, analyzing, for example, what it means to ask a question or make a statement. Although major dialogue theories treat DAs as a central notion (see, for example, Grosz & Sidner (1986) and Allen *et al.* (1996)), the conceptual granularity of the DA labels used varies considerably among alternative analyses, depending on the application or domain.

Two key approaches use machine learning over annotated corpora to recognize dialogue acts. First, there are n-

[2] the terms speech acts, dialogue acts and dialogue moves are often used interchangeably

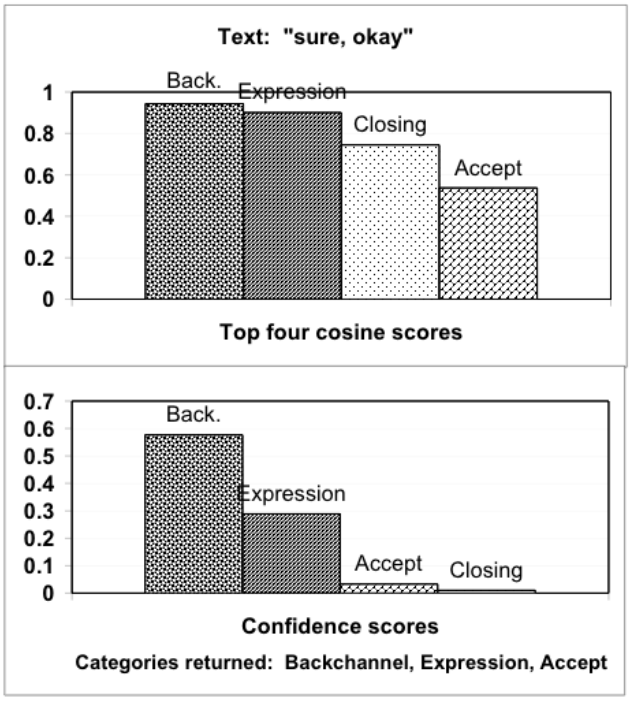

Figure 1: DA classification example one

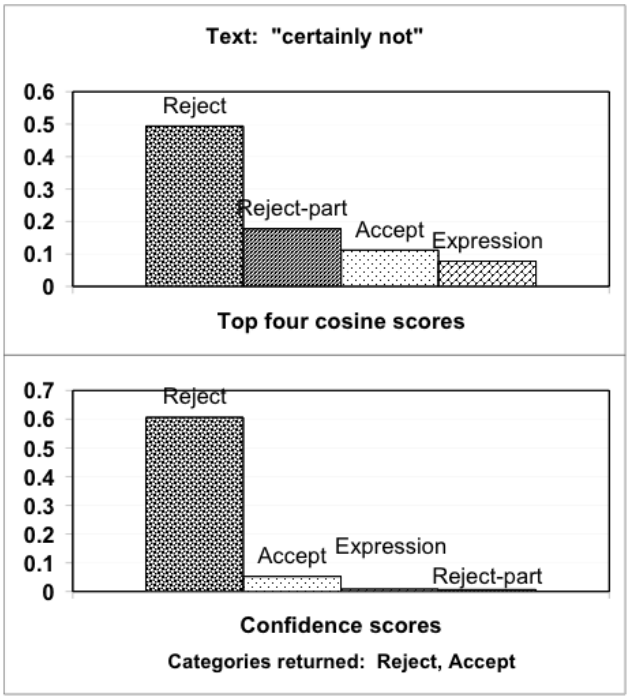

Figure 2: DA classification example two

gram or HMM language modelling approaches, exploiting techniques from speech recognition. Reithinger & Klesen (1997), for example, apply such an approach to the VERBMOBIL corpus, which provides rather limited training data, and report a 74.7% tagging accuracy. Stolcke *et al.* (2000) apply a more complicated n-gram method to the SWITCHBOARD corpus (Jurafsky *et al.* 1998), using HMM models of both individual utterances and DA sequences. They report 71% accuracy, using 198,000 utterances for training and 4,000 utterances for testing. Recent work by Webb, Hepple, & Wilks (2005) shows that a simple classification technique using word n-grams as significant DA cues can achieve cross-validated results at around 70% accuracy.

A second approach, based on transformation-based learning (Samuel, Carberry, & Vijay-Shanker 1998), achieves a tagging accuracy of 75.1% on VERBMOBIL, using features such as utterance length, speaker turn and the DA tags of adjacent utterances.

These performance differences, with a higher tagging accuracy score for the VERBMOBIL corpus despite significantly less training data, can be seen to reflect the differential difficulty of tagging for the two corpora. The SWITCHBOARD corpus comprises conversations of an unstructured, non-directed character, which hence exhibit much greater semantic variability than VERBMOBIL, and have therefore been thought to present a more difficult problem for accurate DA modelling.

The purpose of our DA Classifier Frame Agent is to identify a caller's utterance as one or more domain-independent dialogue acts. These include Accept, Reject, Non-understanding, Opening, Closing, Backchannel, and Expression. We have trained the DA classifier on our corpus of transcribed, labeled human-human calls, and we have used vector-based classification techniques. Importantly, based solely on intra-utterance features (such as the words) an utterance may have multiple correct classifications. We filter out the usual stops, including speech dysfluencies, proper names, number words, and words with digits; but we need to include words such as *yeah*, *uh-huh*, *hi*, *ok*, *thanks*, *pardon* and *sorry*.

Some examples of DA classification results are shown in Figures 1 & 2. For *sure, ok*, the classifier returns the categories Backchannel, Expression and Accept. If the dialogue manager is looking for either Accept or Reject, it can ignore Backchannel and Expression in order to detect the correct classification. In the case of *certainly not*, the first word has a strong tendency toward Accept, though both together constitute a Reject act. Our classifier performs well if the utterance is short and falls into one of the selected categories (86% accuracy on the British data); and it has the advantages of automatic training, domain independence, and the ability to capture a great variety of expressions. However, it can be inaccurate when applied to longer utterances, and it is not yet equipped to handle domain-specific assertions, questions, or queries about a transaction.

System Evaluation

Where possible, we have presented evaluation of individual understanding components, even if these are only indicative. The ANNIE system for information extraction has under-

gone some significant evaluation on written prose. The NE recognition of ANNIE over a news corpus (looking for the standard MUC NE classes) returned figures of 89% precision and 91.8% recall, for a combined f-measure score of 90.4% (Maynard, Bontcheva, & Cunningham 2003). We fully expect the AMITIES ANNIE system to perform at a lower accuracy rate, due in part to errors in speech recognition output, and no orthographic information to leverage; however an indicative evaluation shows promising results. Ten users (5 at our UK site and 5 in the US) interacted with the system 9 times each, creating a total of 90 interactions. Each user had to perform 9 scenarios, the performance of which required the system to recognise on average around 9 concepts or semantic entities each. For example, if the systems asks for *name* and *postcode*, and the user replies *John Smith, S10 1SL*, this represents three concepts – *firstname, lastname* and *postcode*. Over the 90 calls, the NLU component had an average recognition accuracy of 80%, which is acceptable, if only indicative at this stage. The concept recognition performance varied depending on the user, from a best of 89% to a low of 70% over all concepts.

It is possible to gain extra information from our full system evaluation. Ten native speakers of English, 6 female and 4 male, were asked to participate in a preliminary in-lab system evaluation (half in the UK and half in the US). The AMITIES system developers were not among these volunteers. Each made 9 phone calls to the system from behind a closed door, according to scenarios designed to test various customer identities as well as single or multiple tasks. After each call, participants filled out a questionnaire to register their degree of satisfaction with aspects of the interaction.

Overall call success was 70%, with 98% successful completions for the VerifyId and 96% for the CheckBalance subtasks. Failures were not system crashes but simulated transfers to a human agent. There were 5 user terminations. Task completion rates for each sub-task can be seen in Figure 3.

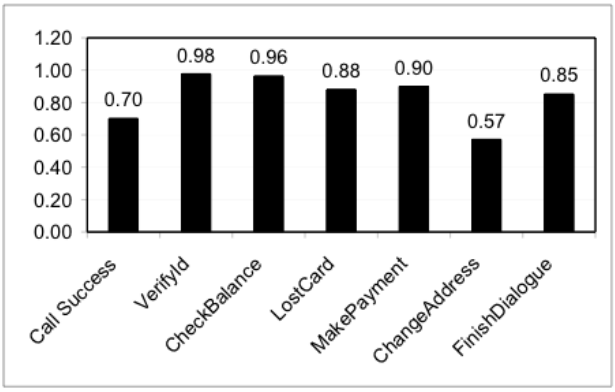

Figure 3: Task Completion

Average word error rates were 17% for calls that were successfully completed, and 22% for failed calls. Word error rate by user ranged from 11% to 26%. Call duration was found to reflect the complexity of each scenario, where complexity is defined as the number of concepts needed to complete each task. Figures 4 & 5 illustrate the relationship between length of call and task complexity. It should be noted that customer verification, a task performed in every dialogue, requires a minimum of 3 personal details to be verified against a database record, but may require more in the case of recognition errors.

The overall average number of turns per dialogue was 18.28. Users spoke an average of 6.89 words per turn and the system spoke an average of 11.42 words. User satisfaction for each call was assessed by way of a questionnaire containing five statements. These covered the clarity of the instructions, ease of doing the task, how well the system understands the caller, how well the system works, and the caller's enjoyment of the system. Participants rated each on a five-point Likert scale. Summed results showed an average score of 20.45 over all users (range 5-25; higher = stronger agreement).

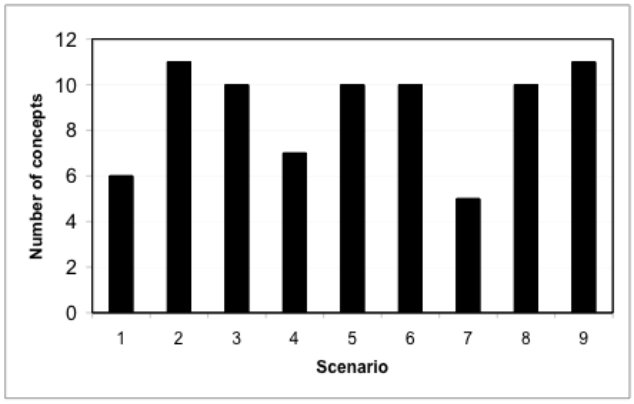

Figure 4: Number of Concepts by scenario

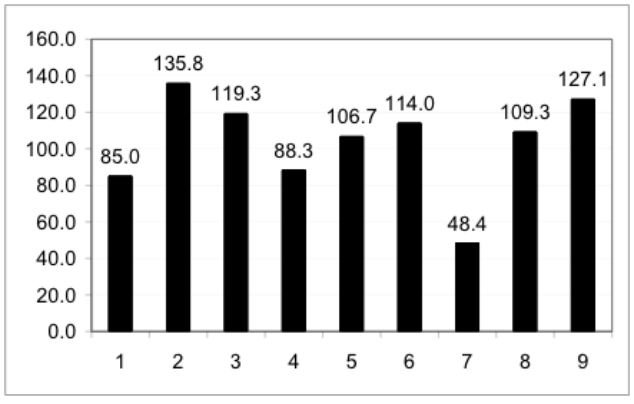

Figure 5: Call Duration by scenario

Although user satisfaction was high, we were more interested in identifying the major problems for the callers. Users were often frustrated by recognition failures and/or unsuccessful attempts to capture values such as a new street

address, county, or phone number. Sometimes the system failed to determine that the user was finished. Because the system is designed to be non-restrictive in accepting users' input, misidentifications were common. We plan to revise our strategy so that we are a little more cautious in our re-prompts. Occasionally, the system misidentified the user's desired task and had difficulty backing off gracefully and starting the correct task. We are working on improving our recovery strategies for these cases.

Discussion

The preliminary evaluation reported here indicates promise for an automated dialogue system such as ours, which incorporates robust, data-driven techniques for information extraction and dialogue act classification. Task duration and number of turns per dialogue both appear to indicate greater efficiency and corresponding user satisfaction than many other similar systems. In the DARPA Communicator evaluation, for example, between 60 and 79 calls were made to each of 8 participating sites (Walker *et al.* 2001) (Walker *et al.* 2002). A sample scenario for a domestic round-trip flight contained 8 concepts (airline, departure city, state, date, etc.). The average duration for such a call was over 300 seconds; whereas our overall average was 104 seconds for a similar average number of concepts (around 8.8). In part this can be attributed to the ability of our system to prompt for, and receive, multiple variables in a single utterance. Our robust IE techniques have proved invaluable to the efficiency and spontaneity of our data-driven dialogue system. In a single utterance the user is free to supply several values for attributes, prompted or unprompted, allowing tasks to be completed with fewer dialogue turns.

This flexibility is made possible by the significant improvement in ASR accuracy rates. In 2001, the Communicator ASR word error rates were about 40% for airline itineraries not completed and 25% for those completed; and task completion rates were 56%. By comparison, our WER of 17% for successful calls is a huge improvement, and is reflected by high user confidence in calling the system. Our average number of user words per turn, 6.89, is also higher than that reported for Communicator systems. This number seems to reflect lengthier responses to open prompts, responses to system requests for multiple attributes, and greater user initiative.

Exactly how much of this system improvement is attributable to improved ASR performance is an interesting question. In one evaluation metric devised for SLDS, the PARADISE framework (Walker 2000), dialogue management strategies can be optimized using machine learning, to provide higher task completion (and so better user satisfaction). The resultant dialogue strategy minimizes the impact of the worst performing component, the speech recognizer. Short, system-initiative utterances, which heavily constrain user input, are favoured, as might be expected. However, if the ASR does not place such a heavy restriction on the operation of the system, it appears that we can make the interaction more natural, users more comfortable, and the system achieves the goal of increased conversational competency.

We are currently working on transfering the system to a new domain: from telephone banking to computer helpdesk support. As part of this effort we are again collecting and analyzing data from real human-human calls. For advanced speech recognition, we hope to train our ASR on new acoustic data. We also plan to expand our dialogue act classification so that the system can recognize more types of acts, and to improve our classification reliability.

References

Allen, J.; Schubert, L.; Ferguson, G.; Heeman, P.; Hwang, C.; Kato, T.; Light, M.; Martin, N.; Miller, B.; Posesio, M.; and Traum, D. 1995. The TRAINS project: a case study in building a conversational planning agent. *Journal of Experimental and Theoretical Artificial Intelligence* 7:7–48.

Allen, J. F.; Miller, B. W.; Ringger, E. K.; and Sikorski, T. 1996. A robust system for natural spoken dialogue. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*.

Allen, J.; Byron, D.; Dzikovska, M.; Ferguson, G.; Galescu, L.; and Stent, A. 2001. Towards conversational human-computer interaction. *AI Magazine*.

Austin, J. L. 1962. *How to Do Things with Words*. Oxford: Oxford University Press.

Brill, E. 1992. A simple rule-based part of speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing, ACL*.

Core, M. G., and Allen, J. 1997. Coding dialogs with the DAMSL annotation scheme. In *AAAI Fall Symposium on Communicative Action in Humans and Machines*.

Cunningham, H.; Maynard, D.; Bontcheva, K.; and Tablan, V. 2002. Gate: a framework and graphical development environment for robust nlp tools and applications. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 02), Philadelphia, Pennsylvania*.

Cunningham, H.; Maynard, D.; and Tablan, V. 2000. JAPE: a Java Annotation Patterns Engine (second ed.). Technical Report CS-00-10, Department of Computer Science, University of Sheffield.

Dahlbäck, N., and Jönsson, A. 1992. An empirically based computationally tractable dialogue model. In *Proceedings of the 14th Annual Conference of the Cognitive Science Society (COGSCI-92)*.

Grosz, B., and Sidner, C. 1986. Attention, Intentions, and the Structure of Discourse. *Computational Linguistics* 19(3).

Hardy, H.; Baker, K.; Bonneau-Maynard, H.; Devillers, L.; Rosset, S.; and Strzalkowski, T. 2003. Semantic and dialogic annotation for automated multilingual customer service. In *Eurospeech, Geneva, Switzerland*.

Hardy, H.; Biermann, A.; Inouye, R. B.; Mckenzie, A.; Strzalkowski, T.; Ursu, C.; Webb, N.; and Wu, M. 2004. Data driven strategies for an automated dialogue system. In

Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), Barcelona.

Hardy, H.; Strzalkowski, T.; and Wu, M. 2003. Dialogue management for an automated multilingual call center. research directions in dialogue processing. In *Proceedings of the HLT-NAACL 2003 Workshop, Edmonton, Alberta, Canada*.

Johnston, M.; Bangalore, S.; Vasireddy, G.; Stent, A.; Ehlen, P.; Walker, M.; Whittaker, S.; and Maloor, P. 2002. Match: an architecture for multimodal dialogue systems. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 02), Philadelphia, Pennsylvania*.

Jurafsky, D.; Bates, R.; Coccaro, N.; Martin, R.; Meteer, M.; Ries, K.; Shriberg, E.; Stolcke, A.; Taylor, P.; and Ess-Dykema, C. V. 1998. Switchboad discourse language modeling project final report. Research Note 30, Center for Language and Speech Processing, Johns Hopkins University, Baltimore.

Karis, D., and Dobroth, K. M. 1991. Automating Services with speech recognition over the public switched telephone network: Human factors considerations. *IEEE Journal of Selected Areas in Communications* 9(4):574–585.

Lamel, L.; Rosset, S.; Gauvain, J.; and Bennacef, S. 1999. The LIMSI ARISE system for train travel information. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*.

Lamel, L.; Bennacef, S.; Gauvain, J.; Dartigues, H.; and Temem, J. 2002. User evaluation of the MASK kiosk. *Speech Communication* 38(1):131–139.

Levin, E.; Narayanan, S.; Pieraccini, R.; Biatov, K.; Bocchieri, E.; Fabbrizio, G. D.; Eckert, W.; Lee, S.; Pokrovsky, A.; Rahim, M.; Ruscitti, P.; and Walker, M. 2000. The AT&T-DARPA Communicator mixed-initiative spoken dialog system. In *ICSLP*.

Maynard, D.; Bontcheva, K.; and Cunningham, H. 2003. Towards a semantic extraction of named entities. In *Recent Advances in Natural Language Processing, Bulgaria*.

Maynard, D. 2003. Expert Update. Technical report, Department of Computer Science, University of Sheffield.

Peckham, J. 1993. A new generation of spoken dialogue systems: results and lessons from the SUNDIAL project. In *Proceedings of the 3rd European Conference on Speech Communication and Technology*, 33 – 40.

Reithinger, N., and Klesen, M. 1997. Dialogue act classification using language models. In *Proceedings of EuroSpeech-97*.

SAIC. 1998. Proceedings of the seventh message understanding conference (muc-7). Technical report.

Samuel, K.; Carberry, S.; and Vijay-Shanker, K. 1998. Dialogue act tagging with transformation-based learning. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*.

Searle, J. R. 1969. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press.

Seneff, S., and Polifroni, J. 2000. Dialogue management in the Mercury flight reservation system. In *Satellite Dialogue Workshop, ANLP-NAACL, Seattle, Washington*.

Seneff, S.; Hurley, E.; Lau, R.; Pao, C.; Schmid, P.; and Zue, V. 1998. Galaxy-II: a reference architecture for conversational system development. In *ICSLP*.

Stolcke, A.; Ries, K.; Coccaro, N.; Shriberg, E.; Bates, R.; Jurafsky, D.; Taylor, P.; Martin, R.; Ess-Dykema, C. V.; and Meteer, M. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. In *Computational Linguistics 26(3), 339–373*.

Walker, M.; Aberdeen, J.; Boland, J.; Bratt, E.; Garofolo, J.; Hirschman, L.; Le, A.; Lee, S.; Narayanan, S.; Papineni, K.; Pellom, B.; Polifroni, J.; Potamianos, A.; Prabhu, P.; Rudnicky, A.; Sanders, G.; Seneff, S.; Stallard, D.; S.; and Whittaker. 2001. DARPA Communicator dialog travel planning systems: the June 2000 data collection. In *Eurospeech*.

Walker, M.; Rudnicky, A.; Aberdeen, J.; Bratt, E.; Garofolo, J.; Hastie, H.; Le, A.; Pellom, B.; Potamianos, A.; Passonneau, R.; Prasad, R.; Roukos, S.; Sanders, G.; Seneff, S.; and Stallard, D. 2002. DARPA Communicator evaluation: progress from 2000 to 2001. In *ICSLP*.

Walker, M. 2000. An application of reinforcement learning to dialogue strategy selection in a spoken dialogue system for email. *Journal of Artificial Intelligence Research* 12:387–416.

Wang, Y.-Y.; Mahajan, M.; and Huang, X. 2000. A unified context-free grammar and n-gram model for spoken language processing. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing ICASSP*.

Ward, W., and Pellom, B. 1999. The CU Communicator system. *IEEE ASRU* 341–344.

Webb, N.; Hepple, M.; and Wilks, Y. 2005. Empirical determination of thresholds for optimal dialogue act classification. In *Proceedings of the Ninth Workshop on the Semantics and Pragmatics of Dialogue*.

Xu, W., and Rudnicky, A. 2000. Task-based dialog management using an agenda. In *ANLP/NAACL Workshop on Conversational Systems*.

Young, S. 2002. Talking to machines (statistically speaking). In *International Conference on Spoken Language Processing, Denver, Colorado*.

Zue, V.; Glass, J.; Goodine, D.; Leung, H.; Phillips, M.; Polifroni, J.; and Seneff, S. 1994. PEGASUS: a spoken dialogue interface for online air travel planning. *Speech Communication* 15:331–340.

Zue, V.; Seneff, S.; Glass, J.; Polifroni, J.; Pao, C.; Hazen, T.; and Hetherington, L. 2000. JUPITER: a telephone-based conversational interface for weather information. *IEEE Transactions on Speech and Audio Processing* 8(1).