16-610: The KDD Lab
Assistive Intelligent Environments

Daniel Wilson
dwilson+@cs

February 5, 2001

Abstract

About 1 in 5 Americans have some kind of disability, and 1 in 10 have a severe disability [12]. With an aging population and a greater likelihood of disability with age, the presence of disabilities can only be expected to grow. The burgeoning field of intelligent environments can augment lost abilities, bringing many people closer to independent, increasingly productive lives. This technology can only succeed if it is usable through a natural interface to a user with disabilities. The most important interface is speech recognition. We propose using reinforcement learning to allow our speech recognition package to apply environmental information to automatically tailor its listening to specific users. An environment will be instrumented with motion detectors. Several language models will be generated, based upon differing tasks. The data from the detectors will be used by our system to decide which language model to use. We predict that over time the system will come to choose the best language model based upon user location, and thereby achieve a higher recognition accuracy than is possible using a general language model. The speech recognition system will adapt itself to user habits and anticipate user needs, helping to make intelligent environments a realistic choice that could help millions of individuals with disabilities live more comfortably and independently.

1 Introduction

Although the most efficient form of interpersonal communication is often speech, it is usually impractical for use in a human-computer interface. Limitations in speech technology hurt the capabilities of intelligent environments, necessitating the use of unnatural and awkward speaking methods [25]. In order to bring the benefits of intelligent environments to those who most need them, the physically disabled, it is vital that we improve this interaction.

Our goal is to use the inherent multimodal capability of intelligent environments to improve speech recognition in a manner that will allow automatic adaptation to user speech patterns. The purpose of an intelligent environment is to use input gathered from sensors to help make decisions or take actions that will benefit the user. One such sensor is a pyroelectric motion detector. We suggest that by using a network of these detectors placed in each room of a house it will be possible for the system to associate certain detectors with corresponding speech patterns, i.e. spoken commands or queries. The next step is to turn this information into an increase in recognition accuracy.

Our speech recognition software uses a language model to improve recognition accuracy. As the language model becomes more specific recognition accuracy increases. We will use a reinforcement learning algorithm, given motion detector readings as the state and a selection of language models as actions. We believe that the system will learn to select the best language model for a situation given motion detector readings. The subsequent increase in recognition accuracy could bring intelligent environments one step closer to helping the ten percent of Americans that suffer from a severe physical disability.

2 Prior Work

Over the last several years much effort has been put into improving the construction of language models over simple trigram models, including higher-order n-grams, skipping [6, 7, 8], modified Kneser-Ney smoothing [8, 10], clustering [8, 9], and combinations of these techniques [5]. Intelligent environments have been applied to several domains, including automatic control of utilities such as heating [11], automatic control of mixed media such as DVD movies based upon user presence [13], facilitation of group meetings [14], and accident detection [22]. However, none of these environments have been used to improve language model selection.

Though intelligent environments remain an unexplored resource for automatic language model selection, there have been past efforts to use multi-modal information to adaptively resolve language model selection. For instance, at the CMU Interactive Systems Laboratories the information gained from gaze tracking has been used to help a Janus III spoken language system decide when to swap language models [15]. Unfortunately, in order to benefit this approach requires a camera in every room and considerable computational power.

The problem has also been approached from a theoretical standpoint. Popovici and Baggia have attempted to instantiate automatic language model selection based upon dialogue predictions [16]. Also, IBM researchers have attempted to adapt trigram language models to different situations based upon the observed topic of conversation [19]. In the spontaneous, unstructured conversation of the home environment, however, these methods face serious challenges.

Many researchers have noted the significance of this problem and attempted to grapple with it. In the past intelligent environments have been criticized as solutions looking for problems [17, 18], but they offer concrete solutions to the problems faced by those with physical disabilities. By applying the information inherent in an intelligent environment to solving the problem of automatic language model selection, we can improve speech recognition accuracy and help solve problems that are clearly definable and clearly significant both to individuals and to America as a whole.

3 Approach

The field of intelligent environments goes by many names, including ubiquitous computing and pervasive computing. The field was introduced by Weiser in 1990. The goal is to augment human physical and cognitive abilities, hopefully in a manner that is invisible to the user. On command the environment could perform physical tasks for the occupant, opening doors, manipulating lights, operating a telephone, reading a book, etc., Also, it could perform cognitive tasks such as reminding the user to take medecine or answering queries, i.e., "Is the front door locked?".

A preliminary survey was conducted during the fall of 2000. Twenty five surveys were collected through personal interviews with individuals suffering from paraplegia (paralyzed from the waist down), quadroplegia (paralyzed from the neck down), and disabilities related to advanced age. The survey revealed a disturbing trend. Speech recognition systems connected to home automation devices existed in the homes of three surveyed quadroplegic individuals. Speech recognition on home computers only existed in the homes of eight surveyed individuals. Speech recognition software was in common use in none of these homes. The survey results reflected a pessimistic view as to the real world utility of speech recognition systems. When prompted about this discrepancy participants responded with varying takes on the same idea: continuous speech recognition is maddeningly inaccurate, and discrete recognition (speaking one word at a time and letting the computer repeat) is too unnatural.

By definition the 'invisible' computer of an intelligent environment requires a maximally natural human-computer interface. It must accept spoken commands and queries from a human user with high accuracy. These commands should also be spoken in a natural manner, as though the person were speaking to a human assistant. Unfortunately, this is a very difficult task given the large amount of commands and queries a user may use at any time. This is a problem that can be approached by drawing upon the multimodal resources available in an intelligent environment.

The groundwork was laid for this project during the fall of 2000. A home environment has been outfitted with an X10 home automation system. X10 is a communications language that allows compatible devices to communicate via existing 110V electrical wiring. These devices may also be accessed by wireless RF signals. Such devices include light switches, motion detectors, thermostats, wall sockets, contact switches, cameras, etc., Many of these devices have 2-way communication, i.e. they can both accept commands and report status. These devices are well tested, widely available, and inexpensive. Currently the environment uses only automated light switches and lamp switches. These devices can be accessed through a serial connection, using a transmitter module controlled by Java code. The X10 system has been interfaced through Java code to a Sphinx speech recognition system [3]. Sphinx is a speaker independent continuous speech recognition system that allows the X10 system to be controlled verbally through a wireless microphone.

Statistical language models (LM) improve the performance of speech recognition systems by providing estimates of a priori probabilities of word sequences. An LM is simply a statistical description of which words are said, and which words are said together. Our Sphinx language model is built automatically using LMTool [20], from a sentence corpus that contains examples of what might be said. For instance, the CMU Communicator airline reservation system language model was trained using a corpus of dialogue between travel agents and customers [4].

Our Sphinx system interfaces with a dialogue management module also developed by the CMU Speech group [4]. This dialogue manager classifies each utterance into different categories based upon a pre-written grammar. The grammar is a collection of possible word combinations and their classification. For instance, the following might be a grammar for a "turn on the lights" command/classification:

please TURN on *the LIGHT *please
TURN: turn, flip, switch
LIGHT: light, lights, lamp

The asterix denotes that the word is optional. A capital word denotes a word that has synonyms which are later enumerated. This small grammar could understand, "Turn on light", and "Please switch on the lights," and everything in between. In this manner utterances are categorized and the commands or queries may be answered by the system. These grammars must be hand-written, but the notation is flexible and powerful, and our domain limited enough to make this feasible. Our grammar will contain a limited selection of three main classifications: lights, appliance, and query. These correspond to commands to turn on and off lights, commands to turn on and off appliances such as a coffee maker or a television, and simple queries to the system pertaining to the weather, time, date, etc., respectively.

We will require a sentence corpus in order to generate a language model. Unfortunately, no database exists with suitable information drawn from an instrumented home environment. This lack of information for the training of a language model is common and is referred to as the 'sparse data problem' [21]. Therefore we will use a less desirable method that has nevertheless been tried and tested. We will generate sentences automatically from the structure of our grammar [23]. Optional words will be added or omitted randomly. Synonyms will be chosen randomly. Obviously this will not perfectly represent dialogue in a real environment, however, the goal of this project is not to create perfect language models but to provide a method of choosing the best among many. Future work will include gathering data from the instrumented home environment to be used in upgrades of the language model sentence corpora.

This method will be used to generate seven language models. Each will focus upon one or a combination of different classifiers. This focus will be generated by creating a corpus with a certain percentage more sentences pertaining to that classifier as compared to non-focus classifiers. Figure 1 illustrates the proportion of corpora devoted to each classifier more clearly.

LM	Light	Appliance	Query
1	50%	25%	25%
2	25%	50%	25%
3	25%	25%	50%
4	40%	40%	20%
5	20%	40%	40%
6	40%	20%	40%
7	33%	33%	33%

Figure 1

In order to supply the modified Sphinx system with real world information it will be necessary to augment our existing intelligent environment. This will be done through the addition of six motion detectors, located in the kitchen, living room, study, bathroom, bedroom, and hallway. The wireless motion detectors require both heat and motion to trigger and will therefore be less susceptible to false detections. Once triggered the detectors pause for sixty seconds before taking new readings.

Reinforcement learning explores how to map from situations to actions while maximizing a scalar reward. The learner is initially ignorant of the correct action and must discover through exploration which actions yield the highest reward. The 'situations' that we speak of are described to the system by the status of many motion detectors. Any combination of all or none of the motion detectors may be triggered at any time. There are seven possible actions to take, corresponding to one of the seven language models which may be active at any time.

There are two subproblems often faced in reinforcement learning. The first is known as the temporal difference problem (Sutton, 1984). If a learner negotiates a set of situations and experiences a certain outcome then how does the learner assign credit or blame for each action? In our case the learner will use immediate reinforcement, meaning that the payoff to the learner is provided by the most recent input-output pair only [24]. In other words, an immediate punishment will occur if the learner has selected a language model and the next utterance is misunderstood, and an immediate reward if the utterance is understood and falls into the majority classification of the current language model. The second problem is called the generalization problem. When the problem space is too large to fully explore, a learner must be able to guess about new situations based on past experience. Our problem space is limited by a finite set of actions and states. The learner's search behavior will not be random, instead it will be more likely to choose a more general language model when in doubt. This will allow the learner to gradually choose more specific language models as it gains more information through experience.

When the X10 system and the learner are fully constructed we will begin gathering results. The system will exist in an authentic home environment containing two male college students and a pet cat. The students will use the light manipulation, appliance manipulation, and querying capabilities of the environment on a regular basis through wireless and stationary microphones. The results of these activities will be logged and the successes and failures of the system will be charted as a function of time. As time passes and the learner has more information we hope to see a marked increase in the correct selection of language model, and therefore an increase in speech recognition accuracy.

4 Work Plan

This project must be complete by the start of May 2001, leaving approximately three working months to complete the project and gather results. During February we will continue instrumentation of the home environment with X10 motion detectors. This has already begun and is expected to take one week. We will also add functionality for these detectors to the current Java interface. During March we will create a grammar and use the grammar to create several sentence corpora. These data will be used to create the seven language models. Work on the reinforcement learner will begin in March. In early April the reinforcement learner will be complete and approximately one month will be spent logging results. At the end of April and beginning of May the results will be collected and included in a paper.

5 Conclusion

With such a significant number of people with disabilities in our society we can not afford to deny our citizens the ability to live independently and productively. Intelligent environments are a natural answer to the question of how people with disabilities can live normal lives. Enabling intelligent environments to make the leap from laboratory to home depends upon an accurate speech recognition interface. Using the multimodal capabilities of the environment to improve this interface is a natural and inexpensive solution.

6 References

[1] Ronald Rosenfeld. A maximum entropy approach to adaptive statistical language modeling. Computer Speech and Language, 10:187-228, 1996.

[2] Y. Gotoh and S. Renals. Variable word rate N-grams. IEEE, 1591-1594, 04/2000.

[3] Ward, W. and Issar, S. Recent improvements in the CMU spoken language understanding system. In Proceedings of the ARPA Human Language Technology Workshop, March 1994, 213-216.

[4] Rudnicky, A.I., Thayer, E., Constantinides, P., Tchou, C., Shern, R., Lenzo, K., Xu W., Oh, A. Creating natural dialogs in the Carnegie Mellon Communicator system. Proceedings of Eurospeech, 1999, 4, 1531-1534.

[5] Goodman, J. Putting it all together: language model combination. IEEE, 1647-1650, 04/2000.

[6] R. Rosenfeld. Adaptive statistical language modeling: a maximum entropy approach. PhD thesis, Carnegie Mellon University, April 1994.

[7] X. Huang, F. Alleva, H.-W. Hon, M.-Y. Hwang, K.-F. Lee, and R. Rosenfeld. The SPHINX-II speech recognition system: An overview. Computer, Speech, and Language, 2:137-148, 1993.

[8] H. Ney, U.Essen, and R. Kneser. On structuring probabilistic dependences in stochastic language modeling. Computer, Speech, and Language, 8:1-38, 1994.

[9] P. F. Brown, V. J. DellaPietra, P. V. deSouza, J. C. Lai, and R. L. Mercer. Class-based n-gram models of natural language. Computational Linguistics, 18(4):467-479, December 1992.

[10] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. Computer Speech and Language, 13:359-394, October 1999.

[11] Mozer, M. C. The neural network house: An environment that adapts to its inhabitants. In M. Coen (Ed.), Proceedings of the American Association for Artificial Intelligence Spring Symposium on Intelligent Environments (pp. 110-114). Menlo, Park, CA: AAAI Press. 1998.

[12] McNeil, J. Disabilities affect one-fifth of all Americans: Proportion could increase in coming decades. U.S. Census Brief. 1997.

[13] Brumitt, B., Meyers, B., Krumm, J., Kern, A., and Shafer, S. EasyLiving: Technologies for Intelligent Environments. Handheld and Ubiquitous Computing, September 2000.

[14] Cohen, M. The future of human-computer interaction or how I learned to stop worrying and love my intelligent room. IEEE Intelligent Systems. March/April 1999.

[15] Stiefelhagen, R. and Yang, J. Gaze tracking for multimodal human-

computer interaction. IEEE. 2617-2620. 1997.

[16] Popovici, C. and Baggia, P. Specialized Language Models Using Dialogue Predictions. IEEE. 815-818. 1997.

[17] Huang, A., Ling, B., Ponekanti, S., and Fox, A. Pervasive Computing: What Is It Good For? Communications of the ACM. 84-91. 5/1999.

[18] Araya, A. Questioning Ubiquitous Computing. ACM Transactions on Computer-Human Interaction. 230-237. 5/1995.

[19] Rao, P., Monkowski, M., and Roukos, S. Language Model Adaptation Via Minimum Discrimination Information. IEEE. 161-164. 5/1995.

[20] Clarkson, P. and Rosenfeld, R. Statistical Language Modeling Using the CMU-Cambridge Toolkit. From Proceedings ESCA Eurospeech. 1997.

[21] Song, F. and Croft, B. A General Language Model for Information Retreival. ACM. 1999.

[22] Abowd, G., Atkeson, C., Bobick, A., Essa, I., MacIntyre, B., Mynatt, E., and Starner, T. Living Laboratories: The Future Computing Environments Group at the Georgia Institute of Technology. Proceedings of the 2000 Conference on Human Factors in Computing Systems (CHI 2000), The Hague, Netherlands, April 1-6, 2000.

[23] Kellner, A. Initial Language Models for Spoken Dialogue Systems. IEEE. 6/1998

[24] Sutton, R. Reinforcement Learning. Kluwer Academic Publishers, Boston/Dordrecht/London, 1992.

[25] Abowd, G. and Mynatt, E. Charting Past, Present, and Future Research in Ubiquitous Computing. ACM Transactions on Computer-Human Interaction, Vol. 7, No. 1, Pages 29-58, March 2000.