ASR for Dialogue Systems in Noisy Environments

Term paper, GSLT course Speech Technology
Jessica Villing
Göteborg University

Abstract

This paper is a review of some of the techniques that can be used to improve the performance of spoken dialogue systems in noisy environments. We will look at methods to improve the quality of the input signal by filtering out additive noise in different ways. We will also look at methods to improve a dialogue system's capability of choosing the correct recognition hypothesis and how to identify recognition errors by prosodic as well as NLU (Natural Language Understanding) features.

1 Introduction

The traditional input and output devices keyboard, mouse and screen have in recent years entered into competition with devices like speakers, microphones and touch screens as ubiquitous computing has become more common. This trend will continue in the future. Instead of typing, the user - who might be driving a car or walking down the street carrying her hand-held device - will probably speak to the computer, possibly in combination with pointings at a touch screen. This means that the user will not always sit in a quiet room, but instead talk to the computer in an environment with other people speaking or shouting, cars driving by etc. This puts heavy demands on the speech recognizer being robust. Not only does it have to be able to recognize the user's way of pronouncing the words (even if she happens to be for example hoarse, under stress or speaking a language other than her native language), the system also has to be able to disregard everything that is considered to be "noise", i.e. everything that is not relevant words spoken by the user.

2 Background

Basically, ASR (Automatic Speech Recognition) research is about building systems that maps the input, i.e. an acoustic signal, into a string of words. For the system to be able to interpret the acoustic wave the wave is sampled, quantized and converted to some sort of parametric representation, e.g. an LPC (Linear Predictive Coding) cepstrum which provides a vector of features for each time-slice of the input wave. The feature vectors are used to estimate the *phonetic likelihoods*, i.e. the system compares the input with the stored acoustic and language models to find a model that matches the input and thereby recognize the spoken word. The technique for this matching often uses HMM's (Hidden Markov Model). To decode, or search for, the sequence of model states that optimize the sequence of input observations the Viterbi algorithm is often used. Early ASR systems could only recognize a few words, for example any of the 10 digits, spoken by a single speaker. The words had to be spoken one at a time, with pauses in between. Nowadays the technique has advanced in a way that it is possible for ASR systems to have a vocabulary of thousands of words, and the speaker can use natural speech instead of short commands.

An ASR system can be speaker dependent or speaker independent. A speaker dependent system, e.g. a dictation system, has to be trained (since it might have a vocabulary of hundreds of thousands of words) by the user so that the system learns how to recognize that unique user. The more training the system gets the better it becomes at recognizing that user. The advantage of these systems is the high recognition rate the system eventually gets, the disadvantage is that every user who wants to use the system has to spend time training the system. Speaker independent systems on the other hand can be used by any user without any training time at all, which is suitable for public systems, e.g. telephone services like timetable information. The challenge for these systems is to be able to recognize all kinds of voices; male, female, young, old, different dialects etc.

(Jurafsky and Martin, 2000)

3 Methods for handling noisy speech signals in a dialogue system

Current speech recognition systems often work quite well in quiet settings, e.g. used by a single user in an office, but for the systems to be able to manage noisy settings there is still some work left to be done. The market for speech applications that can handle conversational speech is fast growing. Robust speech recognition in all practical acoustic environments are therefore crucial, since environmental noise has become one of the primary causes that limit speech system performance in real world environments. Current systems are usually customized to operate under specific environments. If the device is hand-held and the surrounding noise therefore changes, there is often a rapid decrease in system performance. For speech recognition to be robust the accuracy needs to be good even when the quality of the input speech is degraded, or when the characteristics of the training and testing

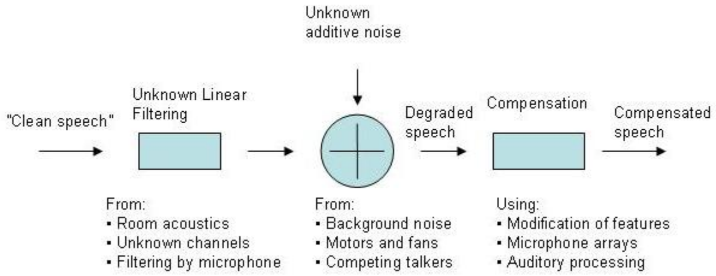

Figure 1: *Schematic representation of some of the sources of variability that can degrade speech recognition accuracy, along with compensation procedures that improve environmental robustness.*

environments differ in any sense. (Deng and Huang, 2004), (Krishnamurthy and Hansen, 2006)

Figure 1 shows how the clean speech signal is degraded by various sources of variability. Acoustic degradations can be caused by the effects of linear filtering, non-linearities in transduction or transmission, impulsive interfering sources or additive noise. The accuracy can also be reduced by changes in articulation, e.g. increased intensity, produced by the presence of high-intensity noise sources (the so-called Lombard effect).

To adapt an ASR system to different environments and speakers Stern (1996) brings up three major approaches; *optimal parameter estimation* (where one either use a formal statistical model to describe the difference between speech used to train and speech used to test the system or use knowledge of background noise to estimate how the clean speech signal changes in noisy environments), *empirical feature comparison* (where features derived from high-quality speech is compared with features of simultaneously recorded speech under degraded conditions) and *cepstral high-pass filtering* (where a high-pass filter passes high frequencies but reduces frequencies lower than the cut-off frequency).

Next follows a review of techniques to improve the ASR in dialogue systems. First we will explore techniques to improve the quality of the input signal by filtering out additive noise from the speech signal, then look at dialogue management techniques to choose among recognition hypotheses and finally look at techniques for error identification.

3.1 Filtering the speech signal

ASR front-ends usually do not have the ability to compensate the effects of noise on feature extraction. This means that there is a risk that they

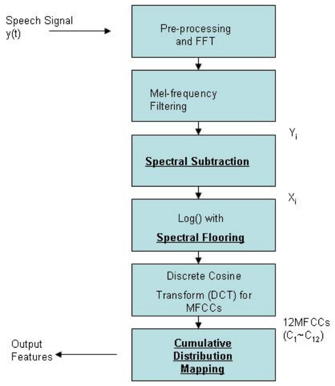

Figure 2: *Front-end processing which incorporates spectral subtraction, spectral flooring and cumulative distribution mapping for noise compensation.*

extract more information about the noise than the speech, if the speech signal is noisy.

Choi (2004) uses spectral subtraction, spectral flooring and cumulative distribution mapping (CDM) to compensate the effects of additive noise during the front-end processing. The spectral subtraction is used to subtract the estimated noise from the noisy signal spectrum by assuming that the first 10 frames of each utterance are noise only. These 10 frames are used to compute the average noise spectrum. Spectral flooring masks out the potential effect of noise by limiting the lower-bound of a Mel-filterbank output to an appropriate value. Looking at the output sequences for a clean model set (speech data with high SNR[1]) and noisy data (speech data with low SNR) reveals a large mismatch. By spectral flooring one can maintain the dynamic range of a feature component in the model set to a desired level and thus help to reduce the potential mismatch between a noisy utterance and the acoustic models. The CDM method is based on the use of histogram equalisation in image processing described by (Russ, 1995) and the idea is to map the distribution of a time sequence of noisy speech fea-

[1] Signal-to-noise ratio (SNR) is an engineering term for the power ratio between a signal and the background noise, i.e. the ratio of useful information to false or irrelevant data. A low SNR implies a low level of useful information and a high level of noise. The noise robust front-end needs to have knowledge about the noise in order to extract only relevant information about the speech, and not the noise.

tures into a target distribution with a pre-defined PDF[2]. A diagram of the processing flow is shown in Figure 2. An evaluation was performed on the Aurora II database (Pearce and Hirsch, 2000), which is a database developed to be used for testing a defined HMM recognition back-end or a complete automatic speech recognition system under noisy conditions. The database contains connected digit tasks spoken by American English speakers. 8 different real-world noises (including subway, babble, car and exhibition noise) have been added to the speech with controlled filtering of the speech and noise. The evaluation demonstrated that the CDM provided the greatest improvement in recognition accuracy and even better results were obtained using CDM with spectral flooring.

In demanding environments hands-free devices are sometimes necessary, e.g. in a car the driver needs to keep her hands on the steering wheel and the eyes on the road. Hands free cellphones are therefore frequent in vehicles, but the poor sound quality and acoustic feedback (echo) of the far-end speech signal produced by the loudspeaker are disadvantages that come in addition. In environments with low SNR microphone arrays (multiple microphones placed at different locations) may be effective for filtering out noise and for echo cancelling. By directing the microphones towards the user they can increase sensitivity to the speaker, and reduce sensitivity of competing sound sources. The individual microphone signals can be filtered and combined to enhance the speech signal of the user. Microphone arrays enable for hands-free devices to use speech as an input modality. (Stern, 1996). Dahl and Claesson (1999) proposes a method of using microphone arrays for echo cancelling. The method is divided into two phases; the gathering phase and the continuous filtering and adaption phase. The gathering phase takes place while the car is parked, by sending representative sequences from each hands free loudspeaker and target position in order to get a fair SNR during data collection. This data is stored and will be used for training. In phase 2 the multichannel calibration signals that contain information on the acoustical environment, the variations in the electrical equipment and the spatial and frequency responses will be used to form the input and reference for the echo canceller. The evaluation of the method revealed that the placement of the microphones seems to be very important, more important than the quality of the microphones. The number of microphones was also essential, the target distortion decreased when the number of microphones was increased. The method yielded good suppression, 19 dB, of the hands free loudspeaker with two microphones and 256 filter taps, and good suppression of the ambient noise in the car.

Many dialogue systems placed in noisy environments use a push-to-talk (PTT) button to reduce the amount of noise from the speech signal. The

[2] Probability Density Function is a function that is non-negative everywhere and presents a probability distribution in terms of integrals.

user pushes a button to switch from transmit mode to voice reception mode. It is only in the voice reception mode that the system listens, which reduces the amount of noise in the speech signal (Wikipedia, 2006). A PTT button simplifies barge-in since the system does not have to listen while talking. Instead the user interrupts the system by pushing the button which decreases the risk of a noisy speech signal due to the system listening at itself.

3.2 Choosing correct recognition hypothesis

In a dialogue system a parser makes semantic representations of the user's input. A dialogue manager then chooses the right dialogue move to make depending on the interpretation of the users utterance. However, as the recognition is seldom perfect the parser might give several outputs with different semantical representations, resulting in an N-best list where N is the number of hypotheses. The hypotheses are ranked, the topmost hypothesis is the one that the ASR system considers to be the most probable. It is, however, not always so that the hypothesis chosen by the ASR system is the best. To get a "second opinion" one might use linguistic knowledge such as semantics, pragmatics, grammar etc. to get a better ranking. The baseline way to choose between the hypotheses is to choose the one that gets the highest score. The dialogue manager then decides how to react depending on the recognition score of the chosen hypothesis; if the recognition score is high enough the system will consider the utterance as correctly recognized and accept the hypothesis, otherwise it might assume that the utterance is misrecognised and reject it, or even ignore it as being noise or speech that is not directed to the system. If accepted, the system can also choose whether the utterance needs to be clarified or confirmed. (Jurafsky and Martin, 2000), (Meza-Ruiz and Lemon, 2005).

However, even if an interpretation has got the highest ranking it is not always so that that hypothesis is the correct one, e.g. noise can make the acoustic signal change considerably so that a hypothesis with a low ranking is more correct than the highest ranked hypothesis. Another way of choosing between the hypotheses is to look at the dialogue context to be able to decide which hypothesis is the most likely to be correct. Gabsdil and Lemon (2004) investigated the use of machine learners trained on a combination of acoustic confidence (N-best recognition lists) and pragmatic plausibility features (dialogue context). The baseline is the WITAS dialogue system, a multi-modal command and control system where the user interacts with the system by combining GUI (Graphical User Interface) actions and spoken commands. It uses the ISU (Information State Update) approach where information relevant to dialogue context is collected in a central data structure. In WITAS the speech recognition is context-sensitive, meaning that the system keeps track of the "most active node" (i.e. conversational contributions that are still in some sense open (under discussion) and hence new utterances are likely to attach) in the dialogue, and loads the grammar corresponding to that node. E.g. if the most active node is a system *yes/no-question*, the language model is defined by a grammar covering phrases such as "yes", "that's right", "okay", "negative", "maybe" and so on. To improve the baseline system, N-best recognition hypotheses for each user utterance were considered. Each hypothesis was classified by the memory-based learner TiMBLE and the rule induction learner RIPPER as either correctly or incorrectly recognized. A simple selection procedure was then made to choose the "best" hypothesis, and decide whether to *accept, clarify, reject* or *ignore* the utterance. The best results, using TiMBLE, show a 25% weighted f-score improvement over the baseline system.

Jonson (2006) examined how well an automatic classifier could manage to find the correct hypothesis compared to a human "classifier". TiMBLE was used to classify ASR hypotheses and re-rank the N-best list depending on these classifications. The classifier was used in the dialogue system GoDiS (Larsson, 2002) to improve the grounding[3] behaviour on the perception level. The classification was done using five classes (optimistic, positive, pessimistic, negative, and ignoring). A comparison classification was made with human classifiers; 16 subjects were asked to re-rank N-best lists given no context at all, immediate context (i.e. the previous system utterance), close context (i.e. the two dialogue moves before the recognition output made by the system or the user) and dialogue context (larger portions of dialogue giving the dialogue history). The classifier was tested on the same N-best lists as the human subjects. The classifier reached 58% sentence accuracy which was slightly better than the humans who got 51% and considerably better than the baseline (which always chose the topmost hypothesis on the N-best list) that only had 10% sentence accuracy.

3.3 Error identification

Litman et al. (2006) have identified certain prosodic features that predict recognition errors better than acoustic confidence scores. They found that speaker turns containing recognition errors are higher in pitch, louder, longer, follow longer pauses and are slower than the turns that are correctly recognized by the system. This points at the fact that there is a strong association between misrecognition and hyperarticulation. It is common that users of a dialogue system start hyperarticulate if the system mishears a word, not knowing that this behaviour will make it even harder for the system to recognize since hyperarticulated words differs considerably from the trained words. Litman et al. showed that the standard use of ASR confidence scores predicted misrecognitions at word level with an error rate of 19%, adding new ASR features (including acoustic confidence score, rec-

[3] Grounding refers to the interaction between dialogue participants that tells whether they perceive, understand and accept each others dialogue moves or not.

ognized string, grammar and features derived from these data) reduced the error rate to 15%, and adding prosodic features further reduced the error rate to 9%.

Improving the NLU performance further increases the ability to recognize early signs of misunderstanding. Walker et al. (2000) tried to identify which of 15 possible tasks the user was attempting, and to detect any items of information that are relevant to that task in the user utterance using RIPPER. The study showed that adding NLU features to measure which task is the most likely improved the accuracy with 21 percentage units compared to the baseline (85% compared to 63%). The NLU features was:

- confidence measure for all of the possible tasks that the user could be trying to do
- salience-coverage (measures the proportion of the utterance which is covered by the salient grammar fragments
- inconsistency
- context-shift (shift of context away from the current task focus)
- top-task (the task with the highest confidence score)
- nexttop-task (second highest confidence score)
- top-confidence (value of the highest confidence score)
- diff-confidence (difference in values between the top and next-to-top confidence score)

4 Discussion and conclusion

ASR techniques has now improved so much that it is reasonable to believe that dialogue systems can be used in many everyday environments. But even though the quality of ASR has improved a good deal the recognition rate is still not sufficient to be reliable, especially when it comes to difficult conditions like noisy environments. Although the accuracy will continue to improve, there will never be 100% accuracy for every user utterance; not even humans achieve this. What humans have, though, is the capability of fast error recognition and error recovery. The challenge now is to make dialogue systems that can handle misrecognition in a way that feels natural and as less distracting as possible for the user. The dialogue system must be able to identify early clues in the dialogue that indicate that there might be a misrecognition and handle the misunderstanding in a proper way.

A study carried out by Skantze (2005) shows that when subjects face speech recognition problems, a common strategy is to ask task-related questions that confirm their hypothesis about the situation instead of signalling non-understanding. A recent, however not yet finished, study carried out by Fernandez et al. (2007) seems to be confirming these results. More studies need to be done to find out how dialogue partners make implicit corrections and confirmations, and whether the behaviour is different in different environments, in order to let the dialogue run smoothly without to many tedious clarifying questions.

5 Future Work

There is a need to make ecologically valid user tests to discover signs of misrecognition, since laboratory tests do not always reflect the reality. For example in-vehicle tests are often carried out in a simulator. The test persons then might be more concerned about carrying out fictive tasks (e.g. different math tasks) than to make sure that they are not causing any accidents, since it does not really matter if you crash the car in a simulator. To make the tests ecologically valid they should be carried out in real traffic, preferably with everyday tasks that are natural for the test person to carry out while driving. Misrecognition might occur for several reasons, from the basic level of not hearing the spoken words at all, to hearing the words but not understand them (out-of-vocabulary errors) or understanding the word but not the meaning (out-of-grammar errors). Techniques for handling noise, and for adapting the dialogue when one dialogue partner is occupied with something else but the dialogue (driving a car) will be investigated in the ongoing Vinnova project DICO in cooperation with Volvo, KTH and TeliaSonera.

References

Choi, E. H. C. (2004). Noice robust front-end for asr using spectral subtraction, spectral flooring and cumulative distribution mapping. In *Proc. 10th Australian Int. Conf. on Speech Science and Technology*, pages 451–456.

Dahl, M. and Claesson, I. (1999). Acoustic noise and echo cancelling with microphone array. *IEEE Transactions on Vehicular Technology*, 48(5):1518–1526.

Deng, L. and Huang, X. (2004). Challenges in adopting speech recognition. *Commun. ACM*, 47(1):69–75.

Fernandez, R., Corradini, A., Schlangen, D., and Stede, M. (2007). Towards reducing and managing uncertainty in spoken dialogue systems. In *Proceedings of the Seventh International Workshop on Computational Semantics (IWCS-7).*

Gabsdil, M. and Lemon, O. (2004). Combining acoustic and pragmatic features to predict recognition performance in spoken dialogue systems. In *42nd Annual Meeting of the Association for Computational Linguistics*.

Jonson, R. (2006). Dialogue context-based re-ranking of asr hypotheses. In *To appear Proceedings IEEE 2006 Workshop on Spoken Languge Technology*.

Jurafsky, D. and Martin, J. H. (2000). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall PTR, Upper Saddle River, NJ, USA.

Krishnamurthy, N. and Hansen, J. H. L. (2006). Noise update modeling for speech enhancement: When do we do enough? In *Proc of ICSLP, Interspeech 2006*, pages 1431–1434.

Larsson, S. (2002). *Issue-Based Dialogu Management*. PhD thesis, Göteborg university.

Litman, D., Hirschberg, J., and Swerts, M. (2006). Characterizing and predicting corrections in spoken dialogue systems. *Comput. Linguist.*, 32(3):417–438.

Meza-Ruiz, I. and Lemon, O. (2005). Using dialogue context to improve parsing performance in dialogue systems. In *International Workshop on Computational Semantics (IWCS)*.

Pearce, D. and Hirsch, H.-G. (2000). The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions. In *ISCA ITRW Workshop on Automatic Speech Recognition*.

Russ, J. C. (1995). *The Image Processing Handbook*. CRC Press.

Skantze, G. (2005). Exploring human error recovery strategies: Implications for spoken dialogue systems. *Speech Communication*, 45(3):325–341.

Stern, R. M. (1996). Robust speech recognition. Website. `http://cslu.cse.ogi.edu/HLTsurvey/ch1node6.html#SECTION14(2006-12-21)`.

Walker, M., Wright, J., and Langkilde, I. (2000). Using natural language processing and discourse features to identify understanding errors in a spoken dialogue system. In *Proc. 17th International Conf. on Machine Learning*, pages 1111–1118. Morgan Kaufmann, San Francisco, CA.

Wikipedia (2006). Push-to-talk. Website. `http://en.wikipedia.org/wiki/Push-to-talk(2006-12-15)`.