6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

STUDIES IN ACOUSTIC TRAINING AND LANGUAGE MODELING USING SIMULATED SPEECH DATA

Don McAllaster *Larry Gillick*

Dragon Systems, Inc.
320 Nevada Street
Newton, MA 02460

10.21437/Eurospeech.1999-358

ABSTRACT

We continue our study of the use of fabricated data in the investigation of speech recognition algorithms. After reviewing the basic data generation algorithm and some earlier results involving the recognition of fabricated conversational speech data, we go on to describe some new and intriguing experiments concerning on the one hand, training acoustic models from simulated speech data, and on the other, recognizing fabricated data with different amounts of context in the language model. Among other things, we conclude that standard training algorithms are remarkably good at recovering the underlying structure in acoustic models when they are given 30 hours of data generated from those models. We also propose an alternative to perplexity for measuring the quality of a language model — the word error rate on fabricated data — that takes into account the inherent acoustic confusability of words.

1. INTRODUCTION

In this paper we continue our examination of the use of fabricated data as a means of investigating speech recognition algorithms and ideas [1, 2]. A large part of the point of the earlier work was to introduce and popularize the method of data simulation as a means of studying speech recognition algorithms, in particular, of uncovering their strengths and weaknesses. Our underlying philosophy is that experiments in which one has control of the underlying data generation mechanism provide a crucial complement to the usual sorts of experiments done on actual speech data. We believe that controls of this sort will, at least in the long run, be a source of considerable enlightenment in a field where it is often unclear why algorithmic changes turn out to be either beneficial or detrimental.

The present work presents two new applications of the method of simulated data to the evaluation of speech recognition algorithms. In the first, we study the behavior of our standard acoustic training algorithms when they are set to work on fabricated data rather than real data. Are our standard training algorithms capable of capturing all of the structure that we have put into the fabricated acoustic data? How much training data is required to capture all of the structure? What is the degradation in performance when we decrease the quantity of data? Will we still be able to recognize real data successfully if we train only on fabricated data? We address these and other related questions in the third section of this paper after we describe our data generation methods and review our earlier results in the second section.

The second application of simulation that we explore here lies in the area of language modeling. At present, our ability to recognize conversational telephone speech, while considerably better than it was just a few years ago [3], is still limited: even the best systems still obtain 30-40% word error rates. We anticipate that there will continue to be steady progress in bringing down these error rates over the next few years, and therefore, we raise the question: what would the effect of various sorts of language models be if we had much better acoustic models, acoustic models that achieved a 10% word error rate, for example?

In essence, by using fabricated data, we can ensure that data will match the models (perfectly if we like) and that all of the errors will derive from subtle acoustic confusions. (Of course there can still be acoustic errors even when the data matches the model perfectly, because of the overlap in the acoustic probability distributions.) We submit that this is an intriguing regime in which to study language modeling, for these errors are the ones that it will be very difficult to fix by acoustic means alone: they are the language model's job. In Section 4 we begin to address these issues by examining the relative merits of various length ngram models. We also propose an alternative to perplexity — word error rate on fabricated data — as a means of comparing language models or the difficulty of texts. This alternative measure takes explicit account of the acoustic confusability in the language in a way that the classic perplexity measure cannot.

2. THE DATA — (REAL AND SIMULATED) AND SOME EARLIER RESULTS

Our key methodological tool is the ability to simulate speech data that corresponds to a particular text from a set of acoustic models, and so we begin by briefly reviewing how that is done (for more details see [1, 2]); all of the experiments presented in this paper are based on the Switchboard corpus. We begin by converting the string of words to a string of phonemes by looking up the pronunciations of the words in a lexicon. If there is more than one pronunciation, we choose among the set of possible pronunciations using a uniform distribution (of course, a more informative distribution would be preferable). We insert initial and final silence, and then randomly insert internal silences between words. Next we convert the sequence of phonemes to a sequence of triphones. Each triphone model itself consists of several nodes, and so, in the end, we have a sequence of silence and speech nodes, each of which has corresponding output and duration distributions specified by decision trees [3]. We now are able to produce a string of frames as follows. For each node, generate a duration D (in frames) from the duration model for the node, pick a Gaussian (according to the mixture probabilities), and then generate D frames from the Gaussian with given means and variances using standard methods.

In our earlier papers we implemented the data fabrication using two contrasting strategies: the simpler one involves using pronunciations drawn from our standard recognizer lexicon as described above, while the more elaborate one relies on a human-labeled phonetic transcription of the conversation, as provided by ICSI (the International Computer Science Institute at the University of California at Berkeley) [4]. We transliterated the ICSI phonemes into one or two of Dragon's phonemes, and derived phoneme strings that were then converted to random frame sequences as described above.

The former procedure for generating data is likely to be less realistic than the latter method, since all words are constrained to be "pronounced" according to the dictionary, while real conversational speech has much greater variability. For example, the word *that* is uttered 328 times in 72 minutes of Switchboard conversation, and is pronounced in 117 different ways [4]. By contrast, our recognition dictionary has just two pronunciations for *that* .

In those experiments we trained two acoustic models AM1 and AM2 from disjoint sets of acoustic training data, each containing around 30 hours of speech. The LM, a backoff bigram model, was trained from approximately 3 million words of transcribed Switchboard conversations, and has 28k distinct words, with 32k pronunciations. Our test set here and throughout this paper is the "test-ws96dev-i" devtest, as used in the 1996 and 1997 summer workshops at Johns Hopkins [5]; it is composed of 23 minutes of speech and 4703 words from 12 speakers. In Table 1, the data is simulated with AM1. Note that the error rate when recognizing using AM2 is much higher than when using AM1. Using AM1 to recognize data fabricated with AM1 is akin to recognizing one's own training data, so we tend, in the experiments that follow, to use different models to fabricate and recognize. Most important, the error rate when the data was fabricated from the ICSI phoneme strings was nearly as high as for actual speech.

Test Set	AM1 WER (%)	AM2 WER (%)
Real Data	48.2	48.8
Data simulated from dictionary	4.3	10.8
Data simulated from phonetic transcription	41.3	43.9

Table 1: WERs when recognizing data simulated with AM1, along with either a dictionary, or phonetic transcripts.

This and other evidence led us to the conclusion [2] that the varied pronunciations of spontaneous speech, along with our failure properly to model them, is to blame for most of the errors made by this recognition system.

3. BUILDING MODELS FROM SIMULATED DATA

In this section we describe three experiments that seek to shed light on the performance of our acoustic training algorithms. In the first experiment we generate 32 hours of data from AM1 and train models from that data, beginning from a flat start (in which we use no *a priori* knowledge about the models). We report performance after successive training iterations on both real and fabricated data. The second experiment examines the question of how the recognition performance on models trained from fabricated data depends on the quantity of training data. A natural question to ask is whether the flat start procedure (described below) is a source of error, and whether we can train more accurate models from the fabricated data if we cheat and use some prior knowledge concerning the alignments. Thus, in the third experiment, we compare the performance of models trained (a) using the exact alignment information (known to us because we manufactured the data) or (b) using Viterbi alignments generated from AM1.

3.1. Training from a Flat Start with Simulated Data

In this experiment, we begin with a model trained on 32 hours of real data (AM1), and proceed to generate 32 hours of training data by using word transcriptions of the 32 hours of Switchboard conversations from which the AM1 model was trained, together with our standard recognition dictionary. It would be interesting to perform this experiment with hand–labeled phonetic transcriptions, as provided for the ICSI test set, but only a few hours of telephone conversation have been so carefully transcribed. We perform a flat start procedure, as we typically do when bringing up a recognizer in a new language, using three hours selected from the 32 hours of fabricated data. This procedure consists of training tied mixture monophone models — each node of each phoneme has an output distribution that is a mixture of a fixed set of 256 Gaussian components — by repeated passes of the Baum-Welch algorithm. This resulting model is used to produce a Viterbi alignment of the 32 hours of training data. From these alignments, we train a decision-tree clustered mixture model with 9000 output distributions, each with up to 16 Gaussian components. We may then iterate, using the models to re-align the training data, and build another set of models.

We use the models resulting from each iteration to perform recognition using both real test data, as well as data fabricated from the other 30 hour acoustic model AM2, using a trigram LM. We compare the word error rate of the model AM1 which generated the training data, to the WERs for models trained from data simulated from AM1 in Table 2.

	Real test data WER (%)	Fabricated test data WER (%)
AM1	42.2	9.0
first iteration	48.2	12.6
second iteration	46.6	10.4
third iteration	45.8	10.6

Table 2: WERs for real and fabricated test data, for the original model AM1, and three iterations of models trained from 32 hours of data fabricated by AM1.

As we have seen in other experiments, the error rates (in Table 2) on fabricated data are systematically lower than on real data. On both real and fabricated data, we see the error rate decreasing significantly after the first iteration. The error rates after the third iteration are reasonably close to, but significantly worse than, the error rate of AM1 (the model that produced the data from which we trained these new models). Presumably, we cannot exceed the performance of AM1 even if we were to generate vast amounts of training data. However, it would be interesting to know whether we could asymptotically achieve AM1's performance if we trained on more data, and how much data would be needed to get to that point. A related question, which we address next, is how much performance we would lose if we trained on less data.

3.2. Varying the Quantity of Simulated Training Data

Here, we explore the effect of reducing the amount of fabricated training data on the WER for both fabricated and real test data. Using the same flat start model in each case, we build three

rounds of full recognition models using 32, 16, 8, 4, and 2 hours of randomly selected fabricated data, and compare recognition performance of the final model iteration to that of the original fabricating model.

	Real test data WER (%)	Fabricated test data WER (%)
AM1	42.2	9.0
32 hour training	45.8	10.6
16 hour training	46.3	10.3
8 hour training	50.3	12.0
4 hour training	51.6	13.2
2 hour training	54.2	16.4

Table 3: WERs for real and fabricated test data, for the original model AM1, and the third iteration of models trained from 32, 16, 8, 4, and 2 hours of data fabricated by AM1. The fabricated test data is generated by AM2.

From Table 3, it is clear that we lose performance when we drop below 16 hours of fabricated training data, but it is less certain whether performance has saturated around 16-32 hours. Could it be that some information present in AM1 has been irretrievably lost? Could there be alignment information that we are somehow not recovering? In the next experiment we use some auxiliary information that would not be available to us if we were building models on real data, in order to investigate these questions.

3.3. Cheating on the Time Alignments

We can evaluate how much is lost in the flat start process by adding some auxiliary information about the alignments. We explore here two kinds of auxiliary alignment information, one based on perfect knowledge of where each node of each triphone starts and ends in the fabricated data, and one based on good but only approximate knowledge obtained by making use of what AM1 can figure out about the alignments.

Instead of performing a flat start on 3 of the 32 hours of fabricated data, we can generate alignments for the 32 hours of data by taking advantage of our knowledge of the data fabrication process itself: namely, how many frames were actually generated in each state. In this scenario, every frame is guaranteed to be labeled with the correct triphone state. Alternatively, we can also use the model that generated the training data (AM1) to do the alignment. We would expect the resulting alignments to be quite good, better than alignments done on real data (just as for recognition), but to contain some small amount of mislabeling. The differences in performance among models built from (1) perfect alignments, (2) good alignments, and (3) flat start based alignments should shed some light on the importance of labeling every frame correctly. As before, we iterate the model build three times, to get an idea of the asymptotic performance of these models, and present the results, for test data fabricated by AM2, and for real data in Table 4. Note that for the second and third iterations, the so-called "perfect alignment" models induce their own alignments, upon which models are built; these induced alignments are no longer "perfect".

"Good" and "perfect" alignments yield word error rates on both real and fabricated data that are not significantly different. On fabricated data, the performance of models initiated from a flat start is not significantly different from the other two models, but is significantly worse when recognizing real test data, at least after three iterations. All of these models are significantly worse than the original AM1, recognizing both real and simulated data.

While our standard model-building algorithms do quite a good job of capturing structure placed in training data through the fabrication process, some information is lost when building from alignments (even perfect ones), and still more in a flat start. It may be that more iterations, or using more data would restore some of the loss.

	Real test data WER (%)	Fabricated test data WER (%)
AM1	42.2	9.0
perfect alignment first iteration	43.2	10.1
second iteration	43.9	9.8
third iteration	44.0	10.0
good alignment first iteration	43.8	9.8
second iteration	43.8	10.2
third iteration	43.4	10.1
flat start alignment first iteration	48.2	12.6
second iteration	46.6	10.4
third iteration	45.8	10.6

Table 4: WERs for real and fabricated test data, for the original model AM1, and three iterations of models trained from 32 hours of data fabricated by AM1. Perfect alignments are created as a part of the fabrication process, good alignments are made in the ordinary way by AM1 and the recognizer, and flat start alignments have no auxiliary model information.

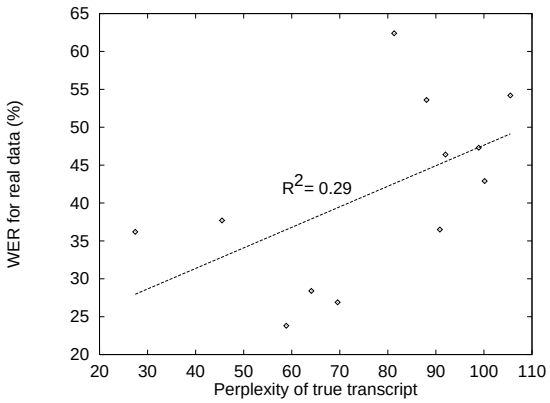

Figure 1: Perplexity vs WER of real data.

4. LANGUAGE MODELING AND FABRICATED DATA

The use of fabricated test data can be seen as a crystal ball, allowing us to see far into the future, when acoustic models for conversational telephone speech match the data far better than they do today. One can argue that an interesting regime in which to do language modeling research is that futuristic one in which the overall error rate is much lower than it is at present, one in which the only errors are, in fact, based on very subtle (or even nonexistent) acoustic distinctions. These are errors where there is no alternative but to turn to the language model for help.

We propose, in fact, an alternative measure of the quality of a language model (or the difficulty of text) that is simply the word error rate using that language model to recognize acoustic data fabricated from a given text. The same acoustic model is used for both fabrication and recognition, to reduce the problem of acoustic mismatch to the absolute minimum. In this way, we take account of some of the inherent acoustic confusability of the text in question (which perplexity ignores), but we do so

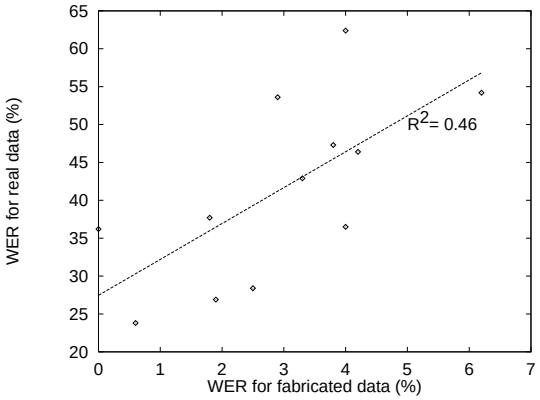

Figure 2: WER of fabricated vs real data.

without the need to record a speaker reading all of the corresponding material, and without the need to exclude senseless or stupid errors due to inadequate or quirky acoustic modeling. In Figure 1 we plot the real WER against the traditional perplexity measure of the true transcript, and Figure 2 shows the fabricated WER against the real WER, where each point represents an individual speaker or fabricated data file; in both plots, we provide an R^2 value for the least-squares linear fit.

The error rate on real speech is, of course, highly speaker dependent, and adds a great deal of noise to the real error rate, but the higher R^2 in the fit of the Figure 2 is at least consistent with the notion that the WER on fabricated data is a better predictor of the WER of real data than the perplexity of the true transcript. We continue to investigate this measure of LM quality.

In our final experiment, we compare performance on fabricated data using standard trigram, bigram, and unigram language models, and, in addition, using no language model at all. We fabricate data using two different models: either the same model that we use for recognition (AM1), or a model trained on 30 different hours of data (AM2). The results in Table 5 illustrate that when using AM2 to fabricate data, there is a factor of two drop in the error rate when we introduce the unigram LM instead of using no LM, and that there is a similar factor of 2 when we add the bigram language model. However, the trigram LM drops the WER by only 15%. Perhaps this reduced reduction reflects the small size of the training corpus for the language model (only 3 million words). We present this experiment as a preliminary example of the sort of experiment that could be done, and hope to pursue this further in the future. Clearly, it would be quite interesting to study the WER on fabricated data as a function of the quantity of LM training data.

	AM1 test data WER (%)	AM2 test data WER (%)
AM1 trigram	3.1	9.0
AM1 digram	3.8	10.5
AM1 unigram	8.7	22.0
AM1 no LM	28.8	44.0

Table 5: WERs for AM1– and AM2– fabricated test data, recognized with AM1, using various LMs.

5. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper we have again sought to illustrate the value of performing experiments on simulated data as a means of probing the strengths and weaknesses of a speech recognition system. Our experiments on training from fabricated data reveal that our standard training algorithms are really quite effective. Indeed, one way of looking at our results is to say that we have shown that if one trains a set of acoustic models, generates 30 hours of fabricated data (keeping the transcriptions, of course), and then discards both the original models and the original data, one can regenerate models from the fabricated data that are almost as good as the originals. We have also proposed a line of investigation involving the study of language modeling by recognizing fabricated data. In this way, one simulates the much lower error rate regime of the future and focuses attention on the very subtle errors that language modeling must be ultimately responsible for fixing.

Modern speech recognizers are immensely complicated software systems, and it is often very difficult to come to understand their sources of error. We remain optimistic that by sending in "known signals" — fabricated data — and examining what happens, we may in the long run be led to a deeper understanding of our algorithms and inspired to invent better ones.

6. REFERENCES

[1] Don McAllaster, Larry Gillick, Mike Newman, Francesco Scattone. Studies with fabricated Switchboard data: exploring sources of model-data mismatch. Proceedings of the Broadcast News Transcription and Understanding Workshop, 1998, pp. 306-310.

[2] Don McAllaster, Larry Gillick, Mike Newman, Francesco Scattone. Fabricating conversational speech data with acoustic models: a program to examine model-data mismatch. ICSLP '98, to be published.

[3] B. Peskin *et al.*, Improvements in recognition of conversational telephone speech. ICASSP '99, submitted for publication.

[4] Greenberg, S. The Switchboard Transcription Project. Johns Hopkins Workshop on Innovative Techniques for Large Vocabulary Continuous Speech Recognition Technical Report, Baltimore.

[5] 1996 CLSP/JHU Workshop on Innovative Techniques for Large Vocabulary Continuous Speech Recognition, July 15 - August 23, 1996, Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD 21218.