Proceedings of DiSS'03, Disfluency in Spontaneous Speech Workshop. 5–8 September 2003, Göteborg University, Sweden.
Robert Eklund (ed.), Gothenburg Papers in Theoretical Linguistics 89, ISSN 0349–1021, pp. 49–52.

Disfluency and speech recognition profile factors

Matthew P. Aylett

Department of Linguistics, University of Edinburgh and Rhetorical Systems Ltd.

Abstract

This paper reports on work bringing together disfluency coding carried out by Lickley [7] and recognition work carried out as part of the ERF project (Bard, Thompson & Isard, [2]) at Edinburgh University. A set of factors are investigated which characterise the behaviour of the ASR during recognition based on an analysis of the resulting word lattice. These factors can be grouped as: Entropy Factors – the entropy of the acoustic and language model likelihoods, within the word lattice, over a 10 ms frame, and, Arc Factors – the number of non-unique and unique arcs in the word lattice in any given 10ms time frame, together with the variance of start and end times of these arcs, and the number of arcs starting or ending in the frame.

The values of all factors were used to train a simple CART model. The CART model was used to predict: recognition failure, interruption point location (the point where a disfluency begins), and whether the location was in a repair or a reparandum.

The entropy of the language model values contributed most to the models prediction of recognition failure, and whether a frame was in a repair or reparandum. In contrast, the number of unique word hypotheses contributed most to the successful prediction of a frame being close to an interruption point.

1. Introduction

Disfluency is common in normal speech but automatic speech recognisers (ASRs) suffer disproportionate and sometimes disastrous deficits when confronted with normal, abandoned or amended utterances. We examined the behaviour of an automatic recogniser built using HTK [10], when applied to data which forms part of the HCRC Map Corpus [1], and compared this to the extensive and detailed disfluency coding which is available for this corpus [7].

We address the following questions:

1. Can we use the behaviour of an ASR to predict disfluency coding?
2. If we can, how might we use this knowledge to improve the performance of the ASR?

The results show we can predict disfluency phenomena using an ASR to a certain extent. However, the accuracy of this prediction is low making it difficult to integrate this knowledge into a conventional ASR to improve performance.

1. In this paper we will:
2. Give a brief example of the disfluency coding and the disfluency factors we tried to predict.
3. Give a description of the factors we took from the ASR, go into some depth concerning the rationale behind selecting these factors and describe the techniques we used to examine them.
4. Present a detailed example of a disfluent sentence and discuss how the ASR factors relate to disfluency.
5. Present the results from a predictive CART model based on the ASR factors.
6. Discuss possible strategies for integrating disfluency knowledge into ASRs.

2. Disfluency Coding

The disfluency coding used in this study is described in depth in [1]. We used in the study, disfluencies categorised as:

repetitions:
"right to my... my right"

substitutions:
"I don't suppose you've got the ballons... the baboons?"

insertions:
"parallel with the ravine... the word ravine"

deletions:
"oh no what... the line stops at the flagship"

The three dots in the above examples mark the interruption point (IP) (which may or may not be followed by a pause). Before the IP there is material which, if removed, would produce a fluent utterance. This material is termed the reparandum. Following the IP their may be material which has replaced the reparandum, termed the repair. In addition we marked filled pauses, editing expressions (i.e. "I mean...") and pauses following interruption points. It is possible (and not uncommon) for disfluencies to be nested, and to be multiple. See [1] for more detail. In this paper we will concern ourselves only with the four simple disfluency types described above.

3. Speech Recognition Profile Factors

One possible output of an ASR is a word lattice. This is a set of transition probabilities for a various hypothesised sequences of words. The transition probabilities are divided into acoustic likelihoods (the probability of the sounds present in the word to be present in the input), and the language model likelihoods (in our case based on a bigram model – the probability of a word following a previous word). The most probable path though this lattice is regarded as the best hypothesis and usually is the final output of the system. See Figure 1 for a simple example of a word lattice.

Lattices, even for simple sentences, are potentially huge. In general, pruning is used during recognition to remove very unlikely arcs. However, even with such pruning it is not uncommon for a word lattice to have tens of thousands of arcs.

In previous work (e.g. [4, 6, 9]), these lattices have been examined to produce confidence measures. A confidence measure is a value which indicates how likely any word in the ASR's output was in the input. One hypothesised means of estimating a confidence value is to calculate the entropy of the likelihoods in the lattice. A high entropy (all the likelihoods are around the same value) would reflect noise and uncertainty, a low entropy (some likelihoods much higher than others) would suggest certainty of a particular arc, or arcs.

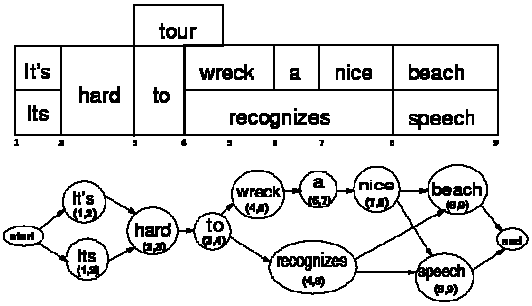

Figure 1: A simple word lattice without transition probabilities (Adapted from a figure in [10]).

In order to calculate these entropy scores we sliced the lattice into 10ms sections. The acoustic, and language model, likelihood, of each arc present during this slice, is taken and normalised by the length of the arc. In section 4 we will examine figures which give some idea how these likelihoods behave across disfluencies. The entropy across these likelihoods in the slice, is then calculated both for the acoustic likelihoods, and the language model likelihoods.

In addition to the entropy of the likelihoods, previous work [6] has examined the way arcs are distributed within a lattice. The more arcs that persist at any point the more uncertain the lattice could be regarded as being, especially if all the arcs represent many different words. From this we produce two values: number of arcs and number of unique word arcs.

The lattice can also give information on boundaries that may or may not exist in the input. If all the arcs present start and end in the frame then the lattice gives a high probability of a word boundary being present. If however the arcs are all ending and beginning at different times in other frames then the lattice could be regarded as being uncertain concerning what boundaries are where. From this we generate three more values: the number of arcs ending in the frame, and the variance of the start points and the end points of all arcs present in the frame.

To summarise, the ASR factors which we used to describe the profile of the recognition that had occurred was as follows:

1. Entropy of Language Model likelihoods normalised by arc length.
2. Entropy of Acoustic Model likelihoods normalised by arc length.
3. Number of arcs.
4. Number of unique arcs.
5. Number of arcs ending in the frame.
6. Variance of arc start times.
7. Variance of arc end times.

These factors are connected with recognition certainty within a lattice. Disfluency has been shown to reduce recognition rates, and thus we might expect, to be related to these factors.

4. Qualitative Example

In order to visualise the way acoustic and language model entropy within the lattice might relate to disfluency we constructed a 3D plot. Two examples of these plots are shown in Figure 2 and Figure 3. Figure 2 was constructed using the language model likelihoods, and Figure 3 using the acoustic likelihoods, for the disfluent section of the utterance:

a level with the... the word giraffes.

Figure 2 shows the log likelihoods of the language model, normalised by arc length, for the most probable hundred arcs, sorted by likelihood, for the section “the... the word”. The result is an escarpment of values. The higher the entropy, the flatter *(not higher)* this escarpment will tend to become. The plot shows that the entropy is higher across the reparandum (the first ‘the’), drops during the pause (indicating the language model is more confident it has found the sequence ‘silence the’), rises at the beginning of repair and drops during the body of the repair (indicating the language model is more confidence in recognising the sequence ‘the word’).

Figure 3 shows the same plot for the normalised acoustic likelihoods. The entropy starts to rise through the first ‘the’ (i.e. the escarpment gets flatter). Then drops briefly, suddenly and strongly, for the acoustic burst of confidence surrounding the pause, then returns to the previous value and finally increases gradually within the sequence ‘the word’.

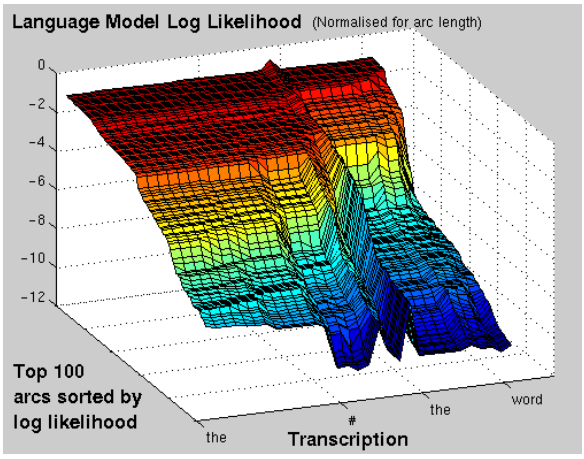

Figure 2: Normalised language model entropy.

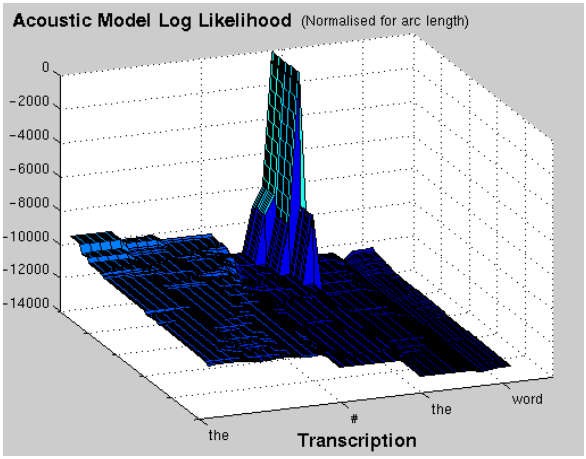

Figure 3: Normalised acoustic model entropy.

The recogniser produced the following hypothesis for this utterance:

no no level with where the words first

Despite the fact arcs matching the transcription were present, within the top 100 arcs, for both language and acoustic models, apart from the word 'giraffe'.

We believe these plots do help understand what is occurring within the entropy of the likelihoods. However more information regarding arcs is required to produce a clear picture of the recognisers behaviour. For example the majority of the arcs present during the sequence "word" in the top 100 arcs sorted by normalised acoustic likelihood were either "word" or "words" suggesting the apparently high entropy at this point reflected the uncertainty between choosing 'word' and 'words' not that the recogniser was uncertain about the acoustic information in general.

The acoustic and language model entropy does indeed correlate negatively with recognition success *(Acoustic Entropy v Recognition Success n=1257, r=-0.352, p < 0.001, Language Model Entropy v Recognition Success n=1257, r=-0.430, p < 0.001).* And more strongly than the overall likelihood of the best hypothesis *(likelihood of best hypothesis v. Recognition Success n=1257, r=0.117, p < 0.001)* which although significant is a very poor predictor indeed.

To investigate their relationship with disfluency we constructed a Decision Tree CART model.

5. Decision Tree CART Model

Using the speech tools wagon program [3] a cart model was built and used to categorise output from an ASR. The cart model tried to learn:

1. Did the speech recognition profile factors suggest this frame would be recognised correctly.
2. Did they indicate that the frame was close to an Interruption point.
3. Could they indicate what part of a disfluency a 10ms frame was within.

Cart models were trained on 75% of the corpus data and then applied to the remaining 25%. In addition to the factors described above, we also used the entropy values from the previous frame as factors. This was so the CART model could potentially make use of the delta of the entropy scores as well as the absolute values.

The results from the CART model show the contribution to increasing predictive power for each parameter, the number of frames correctly classified and the overall success rate. For example, in Table 1a below, Normalised Language Model Entropy together with Number of Arcs led to a 76.68% percent correct classification, adding Number of unique arcs to the model increased classification to 78.33%. If a parameter is not shown its contribution to the classification was negligible. For predicting recognition failure Table 1b shows that the entropy of the language model contributes most to the model with the other factors increasing success by 6.5%. Table 1b shows the classification results. The CART model correctly categorises 80% of the frames according to recognition failure and success.

Table 1a: Predicting Recognition Failure: Contribution of Factors.

Factor	%Accuracy With Factor
Normalised Language Model Entropy:	0.7425
Number of Arcs	0.7668
Number of Unique Arcs:	0.7833
Variance of Arc End Times:	0.7964
Variance of Arc Start Times:	0.8034
Number of Arcs Ending in Frame:	0.8064
Normalised Acoustic Model Entropy:	0.8085
Normalised Language Model Entropy of Previous Frame:	0.8088

Table 1b: Predicting Recognition Failure: Classification Results.

	Model Predicts			
	Recognition Failure	Recognition Success	total	Correct
ASR Result				
Recognition Failure	7483	2801	10284	72.764%
Recognition Success	2723	15891	18614	85.371%

Total 28898 Correct 23374 - Percentage Correct 80.884%

Table 2a and Table 2b show the results for predicting that a frame is within 60ms of an interruption point (IP). The number of frames next to IPs were balanced in number by a random selection that were not close to IPs. Otherwise the CART model gives a misleading 'percentage correct' by **never** positing an IP because they occur rarely. The number of unique arcs contributes most to the model with the other factors increasing success by 6.5% (Table 2a), the frames were categorised correctly 69% of the time (Table 2b).

Table 2a: Frame within 60ms of IP: Contribution of Factors.

Factor	%Accuracy With Factor
Number of Unique Arcs:	0.6297
Variance of Arc Start Times:	0.6561
Number of Arcs	0.6714
Normalised Language Model Entropy:	0.6736
Variance of Arc End Times:	0.6889
Normalised Acoustic Model Entropy:	0.6911
Number of Arcs Ending in Frame:	0.6945

Table 2b: Frame within 60ms of IP: Classification Results

	Model Predicts			
	No IP	IP	total	Correct
Disfluency Coding				
No IP	869	418	1287	67.521%
IP	402	995	1397	71.224%

Total 2684 Correct 1864 - Percentage Correct 69.449%

Table 3a and 3b show the results for a CART model which categorised frames as being either in a reparandum, a repair or not in a disfluency. As for predicting recognition failure the language model makes the biggest contribution to the result with other factors improving the predictive power by 2.7%. These results are skewed by the large number of non disfluent frames. If the results were calculated on chance we would expect: 74% correct categorisation of non-disfluent, 18% correct classification of reparandum and 11% correct classification of repairs.

Table 3a: Predicting Location in Disfluency: Contribution of Factors.

Factor	%Accuracy With Factor
Normalised Language Model Entropy:	0.7439
Number of Arcs	0.7499
Variance of Arc Start Times:	0.7578
Variance of Arc End Times:	0.7667
Number of Unique Arcs:	0.7710
Normalised Language Model Entropy of Previous Frame:	0.7711
Number of Arcs Ending in Frame:	0.7712

Table 3b: Predicting Location in Disfluency: Classification Results.

	Model	**Predicts**			
	Fluent	Reparandum	Repair	total	Correct
Disfluency Coding					
Fluent	33938	1354	633	35925	94.464%
Reparandum	5303	2390	173	7866	30.384%
Repair	3309	353	1168	18614	24.182%

Total 48621 Correct 37469 - Percentage Correct 77.119%

6. Discussion

Overall the results are promising but not ideal. A number of factors make a CART model far from the ideal classifier.

1. The CART model regards each frame as independent when we know that, especially within disfluencies, what was classified previously or subsequently is related. e.g. an IP follows a Reparandum and a Repair follows an IP.
2. Prosodic structure other than IPs may well confound these results. For example the IP classifier might be categorising Intonational Phrase Boundaries as IPs. This would still give a result better than chance but if it cannot tell the difference between an IP and Phrase break the result is not useful.
3. To improve the recognition rate of the base recogniser we would like to produce a probabilistic hypothesis of the location of disfluencies. The CART models do not do this.

A major problem faced in this work was that state of the art recognisers perform poorly on the Map Task Data. This is partly because the data required to train a recogniser for the glaswegian accent is nor readily available but also because of the type of material, it is very natural dialogue, full of disfluencies and other features that typify normal spontaneous speech. Thus the very features which make it interesting material for research into disfluency make it hard for an ASR to deal with. Our baseline recogniser managed 50% word accuracy. This is poor but not dissimilar to the best results on Switchboard (a similarly difficult Corpus) of around 60%. We did find that if we could spot disfluencies and remove reparandum the recogniser improved by 5% to 55% word error rate. This is a significant possible improvement.

Other work carried out in the ERF project tried to amalgamate disfluency spotting with the recogniser by post processing the lattice to allow the recogniser to remove reparandum. This met with some limited success [2].

It is interesting that the language model entropy predicted location in disfluency, while the number of unique arcs predicted the location of the IP. This could be because the loss of context has a ripple effect left and right through the lattice on each side of the IP, while the increase in the number of the unique arcs at the IP relate more closely to it acting as a boundary. However, the relationship between recognition failure, the contents of a lattice produced by an ASR, and disfluency, is still far from clear. It is the case that human beings appear to edit out reparandum during recognition [8]. It is true that if ASRs could do this they would perform better. It is the case that disfluencies are prevalent in normal speech and do appear to have structure. However, for it to significantly improve recognition scores we need to be able to detect disfluencies much more accurately than has so far been achieved.

7. Acknowledgements

This work was supported by EPSRC Project Grant GR/L50280/01.

8. References

[1] Anderson, Anne H., Miles Bader, Eller Gurman Bard, Elizabeth Boyle, Gwyneth Doherty-Sneddon, Simon Garrod, Steve Isard, Iaqueline Kowtko, Jan McAllister, Jim Miller, Cathy Sotillo, Henry Thompson & Regina Weinert. 1991. The HCRC Map Task Corpus. *Language and Speech*, vol. 34(4), pp. 351–366.

[2] Bard, Ellen G., Henry S. Thompson & Steve Isard, 2000, ERF: Exploiting Recognition Failures in Automatic Recognition of Disfluent Speech, EPSRC, *SALT GR/L50280 Final Report.*

[3] Black, Alan W., Paul Taylor & Richard Caley. 1998. The Festival Speech Synthesis System: system documentation. CSTR, University of Edinburgh. http:/ / www.cstr.ed.ac.uk/ projects/ festival/ festival_toc.html

[4] Chase, Lin. 1997. Word and Acoustic Confidence Annotation for Large Vocabulary Speech Recognition. *Proc. Eurospeech '97*, Rhodes, vol. 2, pp. 815–818.

[5] Helzerman, Randall A. & Mary P. Harper. 1996. MUSE CSP: An Extension to the Constraint Satisfaction Problem, *JAIR*, vol. 5, pp. 239–288.

[6] Kemp, Thomas & Thomas Schaaf. 1997. Estimating Confidence Using Word Lattices. *Proc. Eurospeech '97*, Rhodes, vol. 2, pp. 827–830.

[7] Lickley, Robin J., 1998. HCRC Disfluency Coding Manual. *HCRC Technical Report*. HCRC/TR-100.

[8] Lickley, Robin J., David McKelvie & Ellen G. Bard. 1999. Comparing Human and Automatic Speech Recognition Using Word Gating. *Proceedings of the ICPhS Satellite meeting on Disfluency in Spontaneous Speech*, UC Berkeley, pp. 23–26.

[9] Willet, Daniel, Andreas Worm, Christoph Neukirchen & Gerhard Rigoll. 1998. Confidence Measures for HMM-Based Speech Recognition. *Proc. ICSLP '98*, Sydney, Australia.

[10] Young, Steve, Joop Jansen, Julian Odell, Dave Ollason, & Phil Woodland. 1996. The HTK Book. Entropic. Version 2.00.