Helping, I mean assessing psychiatric communication: An application of incremental self-repair detection

Christine Howes[1], **Julian Hough**[1,2], **Matthew Purver**[1] and **Rose McCabe**[3]
[1]Cognitive Science Research Group, School of Electronic Engineering and Computer Science,
Queen Mary University of London
[2]Dialogue Systems Group, Faculty of Linguistics and Literature, Bielefeld University
[3]Medical School, University of Exeter
c.howes@qmul.ac.uk

Abstract

Self-repair is pervasive in dialogue, and models thereof have long been a focus of research, particularly for disfluency detection in speech recognition and spoken dialogue systems. However, the generality of such models across domains has received little attention. In this paper we investigate the application of an automatic incremental self-repair detection system, STIR, developed on the Switchboard corpus of telephone speech, to a new domain – psychiatric consultations. We find that word-level accuracy is reduced markedly by the differences in annotation schemes and transcription conventions between corpora, which has implications for the generalisability of all repair detection systems. However, overall rates of repair are detected accurately, promising a useful resource for clinical dialogue studies.

1 Introduction

Self-repairs are known to be pervasive in human dialogue and there has been much research into the identification and modelling of repair from both computational and psychological perspectives. In computational linguistics, the focus is on removal of disfluency: for the creation of accurate and useful dialogue systems, disfluencies (including self-repair) need to be identified and removed from the speech input to yield interpretable input for downstream processors (especially when using off-the-shelf parsers). Psycholinguistic research, on the other hand, investigates what the presence and type of repair can tell us about psychological and interactional factors in dialogue. For example, the presence of repair can aid comprehension (Brennan and Schober, 2001) and affect the backchannelling of listeners (Healey et al., 2013). In the psychiatric domain, levels of repair have been found to be associated with verbal hallucinations, and patient adherence to treatment (Leudar et al., 1992; McCabe et al., 2013). Identifying repair in these types of dialogue therefore has the potential to be a diagnostic tool, and offer insights into developing training for psychiatrists, e.g. in detecting that a patient is in difficulty, or shaping their own talk more effectively.

1.1 Self-repair

In the conversation analysis literature (e.g. Schegloff et al. (1977)), repairs are described in terms of the dialogue participant (DP) who initiates the (need for) repair (oneself or another), the DP who completes the repair (self or other), and in which position the repair is completed. For the purposes of this paper, we are interested in cases where a DP repairs their own utterance in the course of producing it – a *position one self-initiated self-repair*, which can repeat part of the utterance (an *articulation* repair, as in (1)), reformulate part of the utterance (a *formulation*, as in (2)), or add something clarificatory to the utterance at a point at which it might have been considered complete (a *transition space* repair (3)).[1]

(1) **Dr:** You probably have seen so many psychiatrists *o- o-* **over** the years

(2) **Dr:** *Did you feel that* **did you despair so much that** you wondered if you could carry on?

(3) **P:** Where I go to do some *printing*. **Lino printing**

Rates of self-repair are known to differ over a startling variety of factors; for example, in different domains and dialogue roles (Colman and

[1] These examples are taken from the psychiatric consultation corpus detailed in Section 2.1, with the reparandum shown in italics and the repair phase shown in bold.

Healey, 2011), modalities (Oviatt, 1995), dialogue moves (Lickley, 2001) gender and age groups (Bortfeld et al., 2001) and clinical populations (Lake et al., 2011). For this reason, there is much discussion in the literature over the underlying cause of self-repair – is it merely an index of difficulty for the speaker, for example when planning or producing an utterance (Bard et al., 2001), or is repair interactively designed for the benefit of the listener(s) (Clark and Fox Tree, 2002; Goodwin, 1979)? While we do not address these questions here, we note that this uncertainty causes repair annotation protocol differences, and makes it unclear whether automatic repair detection trained on any single corpus will generalise to any other.

1.2 Repair in psychiatry

In the psychiatric domain, aspects of doctor-patient communication have been shown to be associated with patient outcomes, in particular patient satisfaction, treatment adherence and health status (Ong et al., 1995). Studies specifically investigating repair show associations between repair and clinical populations known to have language difficulties. For example, Lake et al. (2011) found that participants on the autistic spectrum revised their speech less often than controls, and used fewer filled pauses. For patients with schizophrenia, different rates of repair have been linked to specific types of symptoms, such as verbal hallucinations (Leudar et al., 1992), and whether or not a patient is likely to adhere to their treatment (McCabe et al., 2013) as well as psychiatrist assessments of the therapeutic relationship (McCabe, 2008). These studies rely on accurately hand-annotated repair data, and are not directly comparable to each other as different annotation schemes have been used. Assessing the veracity of these results, and exploring the relationship between repair and outcome – for example, how increased levels of repair are associated with a better therapeutic relationship – requires large datasets to be annotated according to the same schema. This is impractical where expensive and time-consuming hand annotations are required. A domain-general automatic repair identification system would enable us to address some of the specific questions raised by these preliminary results.

1.3 Identifying repair

By hand Self-repairs, which are the repair type of interest in this paper, are often annotated according to a well established structure from (Shriberg, 1994) onwards, and as described in Meteer et al.'s (1995) Switchboard corpus annotation handbook:

$$\underbrace{\text{John and Bill}}_{\text{original utterance}}\ \underbrace{[\text{ like}}_{\text{reparandum}} \ +\ \underbrace{\{\text{uh}\}}_{\text{interregnum}}\ \underbrace{\text{love]}}_{\text{repair}}\ \underbrace{\text{Mary}}_{\text{continuation}} \tag{4}$$

This structure affords three principal subtypes of self-repairs: *repetitions*, *substitutions* and *deletions*. Repetitions have identical reparandum and repair phases; substitutions have a repair phase that differs from its repair phase lexically but is clearly substitutive of it; and deletions have no obvious repair phase that is substitutive of their reparandum, with utterance-initial deletions often termed *restarts*. Despite the clarity the structure affords, there is often low agreement between annotators deciding between substitutions and deletions; in fact, considering gradient boundaries between these categories may be more useful (Hough and Purver, 2013). Presence of a repair alone is agreed upon more often than structure.

While this annotation scheme has been widely used in the computational linguistics community, this is not as common for repair corpus studies interested in the dialogue function of repair, rather than their surface structure. Healey et al. (2005) present a systematic effort to test the reliability of a human annotation scheme for repair, building on Healey and Thirlwell's (2002) annotation protocol for identifying the different CA types of repair in dialogue transcripts. They divide repairs into the CA categories of Position 1 repair (*Articulation*, *Formulation*, *Transition space* as shown in (1)-(3), above), Position 2 repair (*Clarification Request/NTRI*, *Correction*) and Position 3 (*Follow-up* and *reformulate*). Healey et al. (2005) tested the validity and reliability of the protocol through an analysis of two of the authors coding a corpus of repair sequences drawn from the CA repair literature with their original coding removed. The validity of the protocol was shown to be encouraging overall, with 75% of the repairs being assigned the same category as that of the original papers, though detection agreement rates were not reported.

Automatically There has been considerable work on detecting reparandum words from transcripts, with the motivation of filtering them out before parsing. However, while the computational linguistics community focusses on the Switchboard corpus disfluency challenge (Charniak and Johnson, 2001), which has been met with considerable success in terms of reparandum word detection (Honnibal and Johnson, 2014; Rasooli and Tetreault, 2014), these models have rarely been applied outside of this domain. This is because there is a lack of gold-standard disfluency annotation in the format shown in (4) available: in fact, Switchboard provides the only large consistently annotated corpus available for this purpose. Furthermore, the fine-grained utterance unit segmentation as carried out by the Switchboard disfluency scheme (Meteer et al., 1995) is uncommon in other corpus mark-ups. For this reason, cross-domain efforts have been rare and performance dips considerably across domains (Lease et al., 2006; Zwarts et al., 2010b). Furthermore, such models are often not designed with word-by-word incremental processing (as required in an incremental dialogue system) in mind; the only effort to develop a system that could function incrementally in a reliable way (Zwarts et al., 2010a) suffers from latency issues, not detecting repairs until an average of 4.6 words after the repair onset.

While the fine-grained structural detection of repairs is necessarily the focus in computational work, to allow reconstruction of a “cleaned” utterance, high accuracy on detecting the structure may be unnecessary for tasks focussing on inter-subjective *rates* of repair. Use of gold-standard Switchboard-style repair annotations in supervised machine learning approaches has a tendency to cause tight fitting to the Switchboard annotation and transcription conventions. While this data can be used as a basis to train a system, it needs to be suitably adaptable to different corpora.

1.4 Research questions

This study applies an incremental repair detection system (STIR; see Section 2.2, below) trained and initially tested on the Switchboard corpus, to a corpus of face-to-face clinical dialogues between patients with schizophrenia and their psychiatrists. The questions we are directly concerned with are:

- Can self-repair be consistently detected across domains and modalities?
- How reliably can different annotation schemes for repair be compared?
- How useful is automatic analysis of self-repair in the clinical domain?

2 Methods

2.1 Data

Switchboard The Switchboard disfluency tagged corpus (Godfrey et al., 1992; Meteer et al., 1995) which has Penn Tree Bank III mark-up, consists of 650 dyadic telephone conversations collected between 1990 and 1992 between unfamiliar American participants on a range of topics assigned from a pre-determined list, ranging from 1.5 up to 10 minutes in duration, with the average conversation lasting around 6.5 minutes. The disfluencies annotated include filled pauses, discourse markers, and edit terms, all with standardised spelling e.g. consistent ‘uh’ and ‘uh-huh’ orthography. First-position self-repairs are bracketed with the structure in (4) with reparandum, interregnum and repair phases marked. It has gold standard Penn Tree Bank part-of-speech (POS) tags and is segmented in terms of sub-turn utterance units. Restart repairs (utterance-initial deletions) are coded as two separate units and not in fact annotated as repairs.

Psychiatric consultation corpus (PCC) The clinical corpus was constructed using a subset of data from a study investigating clinical encounters in psychosis (McCabe et al., 2013), collected between March 2006 and January 2008. The corpus consists of transcripts from 51 outpatient consultations of patients with schizophrenia and their psychiatrist. These transcripts relate to 51 different patients, and 17 psychiatrists. The consultations varied in length, with the shortest consisting of only 709 words (lasting approximately 5 minutes), and the longest 8526 (lasting nearly an hour). The mean length of consultation was 3500 words.

Each transcript was hand-annotated for repair using the protocol described in Healey et al. (2005). For each turn, words in repairs and their reparanda were highlighted using Dexter Coder (Garretson, 2006). The resulting annotations are available in a standalone XML format. For the purposes of this study, the data extracted consisted of the transcripts and associated position 1 repairs (annotated with reparandum phrase and corresponding repair phase). Filled pauses are not explicitly annotated, but are identifiable as interregna as the unannotated text between the end of the reparanda and its repair. Filled pauses, while consistently transcribed, were found to be inconsistently spelt (*aammm, er, eerrrrmm, uhmmm* etc). A find-and-replace operation was therefore applied to the corpus prior to analysis to give these a standardised spelling, i.e. a consistent ‘er’. Prior to the analysis, we also tagged the corpus for part-of-speech using the Stanford POS tagger (Toutanova et al., 2003). The Stanford tagger is trained on written text, and previous work applying it to spoken dialogue has shown the error rates to be in the order of 10% (Mieskes and Strube, 2006). Here, we are not concerned with the POS labels per se, but in the parallelism between POS label sequences (see below) - given that errors are likely to be fairly consistent (dependent on transcription spelling or spoken dialogue idiosyncracies) we take this as sufficient for our purposes.

2.2 STIR: Strongly incremental repair detection

As a repair detection framework we use the STIR (STrongly Incremental Repair detection) system, designed with incrementality and domain-generality in mind (see Hough and Purver (2014)). STIR does not require much annotated disfluency data to become practically useful, as its backbone is derived from simple language model features. Additionally, due to its pipelined classifier structure, different phases of the repair structure in (5) can be included or excluded, depending on the detection task and the available annotations. The repair structure in (5) maps directly to that shown in (4), with the start and end word of the reparandum marked by rm_{start} and rm_{end}, the optional interregnum marked as ed and the repair phase delimited by rp_{start} and rp_{end}.

$$...[rm_{start}...rm_{end} + \{ed\}rp_{start}...rp_{end}]... \quad (5)$$

STIR’s pipeline structure is intended to support incremental processing while being cognitively plausible: it first detects edit terms ed (where present), and then the repair onset rp_{start}; subsequent stages then identify the extent of the reparandum rm_{start} and the end of the repair rp_{end}. Here, we are interested only in repair points, so use only the first two steps – for full details see Hough and Purver (2014).

2.2.1 Enriched language models

STIR is driven by probabilistic models of language which approximate *fluency* level. This is in contrast to most machine learning approaches to repair tagging which often use string alignment for repeated words and POS tags as their principal features. This allows STIR to be compatible with annotation protocols such as (Healey et al., 2005; Colman and Healey, 2011) more concerned with the rate, dialogue type and presence of disfluency rather than purely for identifying reparanda. STIR can thus be used for different repair detection tasks, adapting to the available annotations, and the motivations for the repair detection.

Following Hough and Purver (2013), STIR uses enriched Kneser-Ney (Kneser and Ney, 1995) smoothed trigram language models, trained on a corpus *with disfluencies removed*. The most basic fluency feature is the negative log of the smoothed trigram probability value s (equation 6), aka the *surprisal*. We also use features that approximate syntactic fluency, the principal measure being the (unigram) Weighted Mean Log probability (WML) of utterances and their local trigrams (equation 7), a feature that factors out the contribution of lexical rarity. WML was originally used successfully in detecting low grammaticality judgements (Clark et al., 2013) and given the word-by-word Markov independence assumption of n-gram models it serves as an approximation of incremental syntactic fluency.

$$\text{s}(\text{w}_{i-2} \dots w_i) = -\log_2 p^{kn}(w_i \mid w_{i-2}, w_{i-1}) \quad (6)$$

$$\text{WML}(\text{w}_{i-2} \dots w_i) = \frac{\log_2 p^{kn}_{TRIGRAM}(\langle w_{i-2} ... w_i \rangle)}{-\sum \log_2 p^{kn}_{UNIGRAM}(\langle w_{i-2} ... w_i \rangle)} \quad (7)$$

These feature values can now be calculated at each word, with versions based on word (s^{lex}, WML^{lex}) and POS tag (s^{POS}, WML^{POS}) sequence. For the WML values, we also calculate the difference between values at current and previous word/POS ($\Delta\,WML$). This gives 6 features overall.

2.2.2 Additional features

STIR’s classifiers combine these language model features with further specific logical (binary) features. The *alignment* features indicate whether the word/POS W_x in position x in a trigram is identical to the final word/POS in the trigram, $W3$. The *edit* feature is true iff there is an edit term (filled pause, edit term or discourse marker) detected in the position before $W3$ – see Table 1.

Word n-gram features (n=3)	s^{lex}, WML^{lex}, ΔWML^{lex}
POS n-gram features (n=3)	s^{POS}, WML^{POS}, ΔWML^{POS}
Alignment features (n=4)	$W2 = W3$, $W1 = W3$, $POS2 = POS3$, $POS1 = POS3$
Edit term feature (n=1)	$edit$ [1,0]

Table 1: Repair onset detection features

2.2.3 Training and testing

For the 6 language model features, we train word and POS 'fluent' language models on the standard Switchboard training data (all files with conversation numbers beginning sw2*, sw3* in the Penn Treebank III release), consisting of ≈100K utterances, ≈600K words, cleaned of disfluencies (i.e. edit terms and reparanda) and with gold-standard POS tags. We then keep this language model the same when calculating the feature values across different test corpora; these consist of raw dialogue transcripts with disfluencies included. When testing on data other than Switchboard, the POS tags are generated using the Stanford POS tagger (see above).

As the test corpora have disfluencies present, partial words may be present, either explicitly transcribed as such, or detected by observing an unknown word that forms an orthographic prefix of its following word (i.e. 's, so'). As corpus studies suggest that a non-utterance-final partial word presence predicts a disfluency almost perfectly, for multi-word as well as single partial-word disfluent cut-offs (Hough and Purver, 2013), we include them into STIR's language models with a probabilistic penalty (see Hough and Purver (2014) for details).

Edit term detection uses the word and POS n-gram features above, plus the likelihood assigned by an edit term language model derived from Switchboard's training data. After edit word detection, for repair detection we obtain values for the features listed above for each of the remaining words in a word-by-word fashion from the standardly used Switchboard heldout data (files PTB III sw4[5-9]*; 6.4K utterances, 49K words).

2.2.4 Classifier pipeline

STIR's first two stages are then implemented as random forest classifiers (Breiman, 2001): the first classifies whether the last word seen is an edit term (ed) or not, and the second classifies whether the word is a repair onset (rp_{start}) or not. If the ed classifier classifies a word as ed, the word is not considered for rp_{start} classification; consequently edit term detection is the first stage in the disfluency detection pipeline. We employ weighted error functions to balance recall and precision in the desired way for the detection task using MetaCost (Domingos, 1999). This allows fine-grained control over the rate of onset prediction, which proved to be very useful for the clinical data.

2.3 Experimental set-up

We choose the cost functions for MetaCost on Switchboard heldout data to yield the best overall F-score of rp_{start} detection, we then test on the test data on the standard Switchboard test files (PTB III sw4154 - sw4483; 6.7K utterances, 48K words) for the precision, recall and F-scores and 'relaxed' repair-per-turn evaluation of repair detection (see below for details). For the PCC data, while we keep the base classifier the same as Switchboard, we optimise the weights to balance precision and recall on a heldout set of doctor-patient interaction of ≈20K words. This step was carried out as the weights used for Switchboard yielded much higher precision than recall in rp_{start} detection on a word-by-word level, though the overall accuracy was roughly the same. We then test on a different set of ≈25K words.

3 Results and discussion

Edit term detection Edit term detection was evaluated on the Switchboard test data, achieving an F-score of 0.938. While this is not directly comparable to previous work, Heeman and Allen (1999) also report very high accuracy on detecting a subset of edit terms, *discourse markers*, achieving an F-score of ≈0.96. Our system detects a bigger and more variable class of phenomena.

Testing edit-term detection on the PCC data was more difficult, as edit terms were not explicitly annotated. For the PCC data, transcribed filled pauses are automatically tagged as edit terms and then edit term detection is performed using a model trained on the Switchboard data – this serves as an approximation only; due to the lack of gold standard this was not evaluated quantitatively, but see below for discussion.

Repair point detection We then tested STIR in terms of its precision, recall and F-score for repair onset detection as in (8).

$$\text{precision} = \frac{rp_{start}\text{ correct}}{rp_{start}\text{ hypothesised}}$$
$$\text{recall} = \frac{rp_{start}\text{ correct}}{rp_{start}\text{ gold}}$$
$$\text{F-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \qquad (8)$$

We evaluate in two ways: a *strict* evaluation at the word level, requiring the exact repair point word rp_{start} to be identified; and a *relaxed* evaluation at the turn level, with a rp_{start} hypothesis taken as correct if in the same turn as a gold-standard repair annotation, but with every additional hypothesised rp_{start} over the correct number treated as a false positive (i.e. incrementing rp_{start} hypothesised but not rp_{start} correct). The results are shown in Tables 2 and 3.

detection	precision	recall	F-score
strict	0.862	0.755	0.805
relaxed	0.904	0.787	0.841

Table 2: Switchboard test data results

detection	precision	recall	F-score
strict	0.527	0.536	0.532
relaxed	0.682	0.679	0.680

Table 3: PCC test data results

Turn-level data As can be seen in Table 2, on the Switchboard data the system identifies both that there is a repair and its exact position in the turn very well (F-score > 0.8). However, for the PCC data (see Table 3), although the system identifies that there are repairs in the turn reasonably well (F-score ≈ 0.7), there is a large drop in performance when looking at the strict position-based metric (F-score ≈ 0.5).

This is likely to be due to differences in both transcription and annotation conventions. In the PCC data, the emphasis for annotators was on identifying the number and type of repairs in the turn. Although there was good agreement between annotators at this level – with levels comparable to our relaxed evaluation performance (Cohen's κ = 0.73, (McCabe et al., 2013)), it is not clear whether the annotators position repair points systematically or agree on positioning. Examination of the transcripts suggests that annotation differences can abound. For example, as shown in (9), editing phrases such as 'I mean' may be annotated as part of the reparandum (9a), left unannotated between reparandum and repair (9b), or annotated as part of the repair itself (9c). While (9b) maps most directly to the Switchboard annotation schema, these differences do not affect the overall number and type of repairs found in a turn, and are therefore only relevant if our task is the strict one of finding the exact position of repairs. While this is usually important for the purposes of speech recognition or dialogue systems, it is not here – our interest in is the association between outcomes and the presence and rate of different types of repairs.

(9) **(a) Dr:** well *I think I mean* **I think** that's why it's really sensible

(b) Dr: well *I think* I mean **I think** that's why it's really sensible

(c) Dr: well *I think* **I mean I think** that's why it's really sensible

Dialogue level data Given the differences in turn-level data, as outlined above, and the different ways in which automatically annotated repair data might be used, we compared the number of identified repairs over each dialogue.

As can be seen from Table 4, there is a very high correlation (> 0.9) between the number of repairs per transcript detected by the automatic incremental classifier and those annotated by hand. At this coarse-grained level, the system provides a useful overview of self-repair, which can allow us to make comparisons between speakers who typically use a lot of repair and those who do not, as well as looking for associations with outcomes on a by-patient level as in (McCabe et al., 2013). However, as can also be seen in Table 4, the automatic repair numbers are lower than those for the hand-coded data, and this is especially the case where patients are concerned. This indicates that the system is systematically *not* picking up certain types of repair that the patients are using.

When comparing the hand annotations on the PCC data with STIR's output, we see differences

	Hand-coded		Automatic		Correlation	
	Mean	(s.d.)	Mean	(s.d.)	r	p
Patient P1 repair	62.51	(44.87)	48.90	(33.29)	0.945	< 0.001
Doctor P1 repair	41.57	(23.25)	41.02	(23.23)	0.906	< 0.001

Table 4: Relationship between hand-coded and automatically generated repair measures

due to several factors of annotation protocol and behaviour and not just due to inherently poor system performance. See examples (10)-(12) where the hand annotation tags (shown in (a) in each case) differ from STIR's annotations (shown in (b)).

(10) **(a) D:** ... and if you tell me that **that**$[RP_{START}]$ that the depressions kicks in ...
(b) D: ... and if you tell me that **that**$[rp_{start}]$ **that**$[rp_{start}]$ the depressions kicks in ...

(11) **(a) D:** and so **I**$[RP_{START}]$ mean otherwise I'm not too concerned about your mental health...
(b) D: and so **I**$[ed]$ **mean**$[ed]$ otherwise I'm not too concerned about your mental health...

(12) **(a) P:** I don't **I'm**$[RP_{START}]$ not like hearing voices...
(b) P: I don't I'm not like hearing voices...

In (10) the second repeat of 'that' is evaluated as a false positive by STIR, reflecting the embedded repairs often found in Switchboard, while the annotator views this as part of one longer repair. A false negative from STIR can be seen in (11) where an annotator deems this a repair, while according to Switchboard, and STIR, this would be an editing phrase 'I mean'. In (12), another false negative is evaluated as STIR misses the transcribed repair onset from 'I'm not'. Utterance-initial deletions, or 'restarts', are not marked in Switchboard but treated as two separate utterance units, so there is no training data for these types of self-repair.

4 Towards domain-general repair detection

Using a more strict word-by-word evaluation, we saw that the differences in annotation schemes and transcription conventions have a marked effect on the system's performance. Switchboard annotation conventions result in a biasing on particular types of repair, namely, mid-utterance repetitions, deletions and substitutions, whereas it is not marked for restarts, which caused it to perform poorly on detecting them in the clinical data. On the clinical side, the fact that editing terms are often marked as the repair onset means a Switchboard-trained detector will not get the exact position of the repair. This has implications for the generalisability of all repair detection systems that rely on strict word-by-word evaluation, such as those used in dialogue systems – the way in which the training data has been annotated and transcribed will affect what types of repair it reliably detects.

Despite the differences in the type of disfluency annotation available, one can build a system that is practically useful for detection purposes using the set-up as shown in Figure 1. As long as there is some heldout data available of the same type as the target corpus, even if not considerable in size, STIR's error functions can be manually adjusted (or automatically experimented with) to yield the best accuracy results before testing. This technique is effective in terms of giving results with good overall correlations as described above.

The element of Figure 1 not present in the version of STIR here is the *"fluent" corpus* which could form additional training data to the fluent language model in STIR. We hypothesize that the appropriate data, even if from written, rather than spoken sources, could boost results on out-of-domain (non-Switchboard) data. (Zwarts and Johnson, 2011) show how large text-based corpora included in a repair hypotheses re-ranker can improve detection on Switchboard, however we would like to explore the effect of additional resources in improving performance on other data, such as the PCC corpus described here. Other data STIR does not currently use is acoustic information, which has been shown to help disfluency detection (Liu et al., 2003). Incorporating speech signal information will form part of future work.

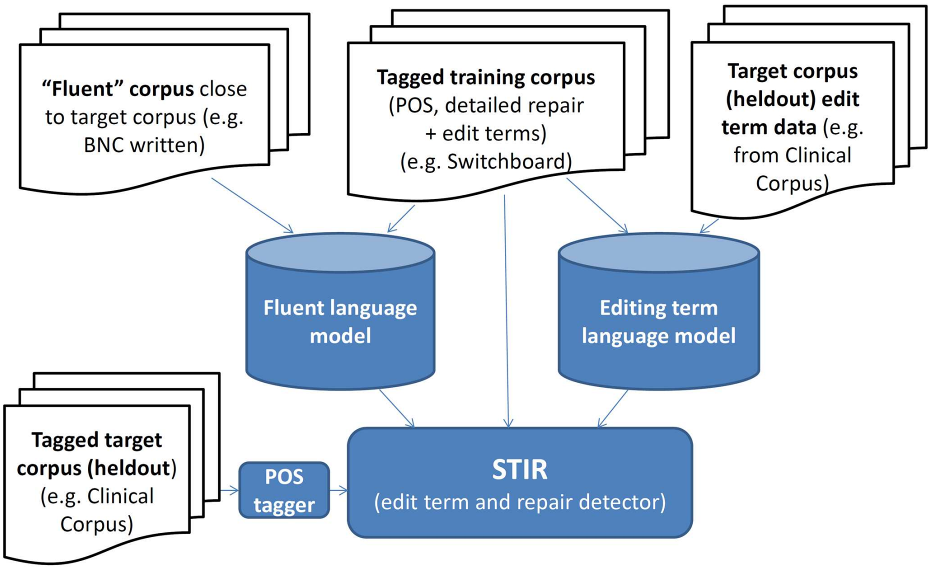
"Fluent" corpus close to target corpus (e.g. BNC written)
Tagged training corpus (POS, detailed repair + edit terms) (e.g. Switchboard)
Target corpus (heldout) edit term data (e.g. from Clinical Corpus)
Fluent language model
Editing term language model
Tagged target corpus (heldout) (e.g. Clinical Corpus)
POS tagger
STIR
(edit term and repair detector)

Figure 1: STIR training and heldout sources for a new target domain

5 Conclusions

In terms of the research questions set out in section 1.4, we can detect self-repair reliably across modalities and domains, but only if we use a relaxed evaluation metric. However, this is sufficient for the purposes of examining overall rates of repair, as used in some clinical studies (McCabe et al., 2013), and automatic self-repair detection using STIR can therefore be usefully applied to these datasets, removing the need for time-consuming and costly hand annotations.

The STIR system is intended to provide a domain-general incremental repair detection system and we are currently experimenting with different language models that allow it to generalise to other data in very different dialogue domains. Issues to consider in future work that have been raised by this preliminary study include (but are by no means limited to) the transcription of filled pauses and overlapping speech, how turns are segmented, and issues arising due to the lack of gold-standard POS tags– joint POS-tagger and repair detection could lead to a more robust final outcome (Heeman and Allen, 1999).

In terms of practical applications, the STIR system is already being used to look at changes in self-repair behaviours before and after training in a psychiatrist communications study, and as it is strictly incremental, it has the capacity to be implemented in artificial mental health worker dialogue agents (Faust and Artstein, 2013).

Acknowledgments

Thanks to the three anonymous SemDial reviewers for their helpful comments.

Howes was supported by the EPSRC-funded PPAT project grant number EP/J501360/1 during this work.

Hough is supported by the DUEL project financially supported by the Agence Nationale de la Research (grant number ANR-13-FRAL-0001) and the Deutsche Forschungsgemainschaft. Much of the work was carried out under an EPSRC DTA scholarship at Queen Mary University of London.

Purver is partly supported by ConCreTe: the project ConCreTe acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET grant number 611733.

The clinical data was collected and transcribed as part of the MRC funded project "Doctor-patient communication in the treatment of schizophrenia: Is it related to treatment outcome?" (G0401323).

References

Ellen G. Bard, Robin J. Lickley, and Matthew P. Aylett. 2001. Is disfluency just difficulty? In *ISCA Tutorial and Research Workshop (ITRW) on Disfluency in Spontaneous Speech*.

Heather Bortfeld, Silvia D Leon, Jonathan E Bloom, Michael F Schober, and Susan E Brennan. 2001. Disfluency rates in conversation: Effects of age, relationship, topic, role, and gender. *Language and speech*, 44(2):123–147.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

S.E. Brennan and M.F. Schober. 2001. How listeners compensate for disfluencies in spontaneous speech. *Journal of Memory and Language*, 44(2):274–296.

E. Charniak and M. Johnson. 2001. Edit detection and parsing for transcribed speech. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, pages 1–9. Association for Computational Linguistics.

Herbert H. Clark and Jean E. Fox Tree. 2002. Using *uh* and *um* in spontaneous speaking. *Cognition*, 84(1):73–111.

Alexander Clark, Gianluca Giorgolo, and Shalom Lappin. 2013. Statistical representation of grammaticality judgements: the limits of n-gram models. In *Proceedings of the Fourth Annual Workshop on Cognitive Modeling and Computational Linguistics (CMCL)*, pages 28–36, Sofia, Bulgaria, August. Association for Computational Linguistics.

M. Colman and P. G. T. Healey. 2011. The distribution of repair in dialogue. In *Proceedings of the 33rd Annual Meeting of the Cognitive Science Society*, pages 1563–1568, Boston, MA.

Pedro Domingos. 1999. Metacost: A general method for making classifiers cost-sensitive. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 155–164. ACM.

Lauren Faust and Ron Artstein. 2013. People hesitate more, talk less to virtual interviewers than to human interviewers. In *Proceedings of the 17th SemDial Workshop on the Semantics and Pragmatics of Dialogue (DialDam)*, pages 35–43, Amsterdam, dec.

Gregory Garretson. 2006. Dexter: Free tools for analyzing texts. In *Actas de V Congreso Internacional AELFE*, pages 659–665.

John J. Godfrey, Edward Holliman, and J. McDaniel. 1992. SWITCHBOARD: Telephone speech corpus for research and development. In *Proceedings of IEEE ICASSP-92*, pages 517–520, San Francisco, CA.

Charles Goodwin. 1979. The interactive construction of a sentence in natural conversation. In G. Psathas, editor, *Everyday Language: Studies in Ethnomethodology*, pages 97–121. Irvington Publishers, New York.

P. G. T. Healey and M. Thirlwell. 2002. Analysing multi-modal communication: Repair-based measures of communicative co-ordination. In *Proceedings of the International CLASS Workshop on Natrual, Intelligent and Effective Interaction in Multimodal Dialogue Systems*, pages 83–92, June 28th-.

P. G. T. Healey, M. Colman, and M. Thirlwell. 2005. Analysing multi-modal communication: Repair-based measures of human communicative co-ordination. In J. van Kuppevelt, L. Dybkjaer, and N. Bernsen, editors, *Natural, Intelligent and Effective Interaction in Multimodal Dialogue Systems*, volume 30 of *Text, Speech and Language Technology*, pages 113–129. Kluwer, Dordrecht.

Patrick G. T. Healey, Mary Lavelle, Christine Howes, Stuart Battersby, and Rose McCabe. 2013. How listeners respond to speaker's troubles. In *Proceedings of the 35th Annual Conference of the Cognitive Science Society*, Berlin, July.

Peter Heeman and James Allen. 1999. Speech repairs, intonational phrases, and discourse markers: modeling speakers' utterances in spoken dialogue. *Computational Linguistics*, 25(4):527–571.

Matthew Honnibal and Mark Johnson. 2014. Joint incremental disfluency detection and dependency parsing. *Transactions of the Association of Computational Linugistics (TACL)*, 2:131–142.

Julian Hough and Matthew Purver. 2013. Modelling expectation in the self-repair processing of annotat-, um, listeners. In *Proceedings of the 17th SemDial Workshop on the Semantics and Pragmatics of Dialogue (DialDam)*, pages 92–101, Amsterdam, December.

Julian Hough and Matthew Purver. 2014. Strongly incremental repair detection. In *Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar. (to Appear).

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 1, pages 181–184. IEEE.

Johanna K Lake, Karin R Humphreys, and Shannon Cardy. 2011. Listener vs. speaker-oriented aspects of speech: Studying the disfluencies of individuals with autism spectrum disorders. *Psychonomic bulletin & review*, 18(1):135–140.

Matthew Lease, Mark Johnson, and Eugene Charniak. 2006. Recognizing disfluencies in conversational speech. *Audio, Speech, and Language Processing, IEEE Transactions on*, 14(5):1566–1573.

Ivan Leudar, Philip Thomas, and Margaret Johnston. 1992. Self-repair in dialogues of schizophrenics: Effects of hallucinations and negative symptoms. *Brain and Language*, 43(3):487 – 511.

Robin J Lickley. 2001. Dialogue moves and disfluency rates. In *ISCA Tutorial and Research Workshop (ITRW) on Disfluency in Spontaneous Speech*.

Yang Liu, Elizabeth Shriberg, and Andreas Stolcke. 2003. Automatic disfluency identification in conversational speech using multiple knowledge sources. In *Preceedings of Eurospeech*, pages 957–960.

R. McCabe, P. G. T. Healey, S. Priebe, M. Lavelle, D. Dodwell, R. Laugharne, A. Snell, and S. Bremner. 2013. Shared understanding in psychiatrist-patient communication: Association with treatment adherence in schizophrenia. *Patient Education and Counselling*.

R. McCabe. 2008. Doctor-patient communication in the treatment of schizophrenia: Is it related to treatment outcome? Technical report, Final report on G0401323 to Medical Research Council.

M. Meteer, A. Taylor, R. MacIntyre, and R. Iyer. 1995. Disfluency annotation stylebook for the switchboard corpus. ms. Technical report, Department of Computer and Information Science, University of Pennsylvania.

Margot Mieskes and Michael Strube. 2006. Part-of-speech tagging of transcribed speech. In *Proceedings of LREC*, pages 935–938.

L.M.L. Ong, J.C.J.M. De Haes, A.M. Hoos, and F.B. Lammes. 1995. Doctor-patient communication: a review of the literature. *Social science & medicine*, 40(7):903–918.

Sharon Oviatt. 1995. Predicting spoken disfluencies during human–computer interaction. *Computer Speech & Language*, 9(1):19–35.

Mohammad Sadegh Rasooli and Joel Tetreault. 2014. Non-monotonic parsing of fluent umm I mean disfluent sentences. *EACL 2014*, pages 48–53.

E.A. Schegloff, Gail Jefferson, and Harvey Sacks. 1977. The preference for self-correction in the organization of repair in conversation. *Language*, 53(2):361–382.

Elizabeth Shriberg. 1994. *Preliminaries to a Theory of Speech Disfluencies*. Ph.D. thesis, University of California, Berkeley.

Kristina Toutanova, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of HLT-NAACL*, pages 252–259.

Simon Zwarts and Mark Johnson. 2011. The impact of language models and loss functions on repair disfluency detection. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 703–711, Stroudsburg, PA, USA. Association for Computational Linguistics.

Simon Zwarts, Mark Johnson, and Robert Dale. 2010a. Detecting speech repairs incrementally using a noisy channel approach. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 1371–1378, Stroudsburg, PA, USA. Association for Computational Linguistics.

Simon Zwarts, Mark Johnson, and Robert Dale. 2010b. Repurposing corpora for speech repair detection: Two experiments. In *Australasian Language Technology Association Workshop 2010*, page 99.