Automatic Editing in a Back-End Speech-to-Text System

Maximilian Bisani Paul Vozila Olivier Divay Jeff Adams
Nuance Communications
One Wayside Road
Burlington, MA 01803, U.S.A.
{maximilian.bisani,paul.vozila,olivier.divay,jeff.adams}@nuance.com

Abstract

Written documents created through dictation differ significantly from a true verbatim transcript of the recorded speech. This poses an obstacle in automatic dictation systems as speech recognition output needs to undergo a fair amount of editing in order to turn it into a document that complies with the customary standards. We present an approach that attempts to perform this edit from recognized words to final document automatically by learning the appropriate transformations from example documents. This addresses a number of problems in an integrated way, which have so far been studied independently, in particular automatic punctuation, text segmentation, error correction and disfluency repair. We study two different learning methods, one based on rule induction and one based on a probabilistic sequence model. Quantitative evaluation shows that the probabilistic method performs more accurately.

1 Introduction

Large vocabulary speech recognition today achieves a level of accuracy that makes it useful in the production of written documents. Especially in the medical and legal domains large volumes of text are traditionally produced by means of dictation. Here document creation is typically a "back-end" process. The author dictates all necessary information into a telephone handset or a portable recording device and is not concerned with the actual production of the document any further. A transcriptionist will then listen to the recorded dictation and produce a well-formed document using a word processor. The goal of introducing speech recognition in this process is to create a draft document automatically, so that the transcriptionist only has to verify the accuracy of the document and to fix occasional recognition errors. We observe that users try to spend as little time as possible dictating. They usually focus only on the content and rely on the transcriptionist to compose a readable, syntactically correct, stylistically acceptable and formally compliant document. For this reason there is a considerable discrepancy between the final document and what the speaker has said literally. In particular in medical reports we see differences of the following kinds:

- Punctuation marks are typically not verbalized.
- No instructions on the formatting of the report are dictated. Section headings are not identified as such.
- Frequently section headings are only implied. ("vitals are" → "PHYSICAL EXAMINATION: VITAL SIGNS:")
- Enumerated lists. Typically speakers use phrases like "number one . . . next number . . . ", which need to be turned into "1. . . . 2. . . . "
- The dictation usually begins with a preamble (e.g. "This is doctor *Xyz* ...") which does not appear in the report. Similarly there are typical phrases at the end of the dictation which should not be transcribed (e.g. "End of dictation. Thank you.")

Proceedings of ACL-08: HLT, pages 114–120,
Columbus, Ohio, USA, June 2008.

- There are specific standards regarding the use of medical terminology. Transcriptionists frequently expand dictated abbreviations (e.g. "CVA" → "cerebrovascular accident") or otherwise use equivalent terms (e.g. "nonicteric sclerae" → "no scleral icterus").

- The dictation typically has a more narrative style (e.g. "She has no allergies.", "I examined him"). In contrast, the report is normally more impersonal and structured (e.g. "ALLERGIES: None.", "he was examined").

- For the sake of brevity, speakers frequently omit function words. ("patient" → "the patient", "denies fever pain" → "he denies any fever or pain")

- As the dictation is spontaneous, disfluencies are quite frequent, in particular false starts, corrections and repetitions. (e.g. "22-year-old female, sorry, male 22-year-old male" → "22-year-old male")

- Instruction to the transcriptionist and so-called normal reports, pre-defined text templates invoked by a short phrase like "This is a normal chest x-ray."

- In addition to the above, speech recognition output has the usual share of recognition errors some of which may occur systematically.

These phenomena pose a problem that goes beyond the speech recognition task which has traditionally focused on correctly identifying speech utterances. Even with a perfectly accurate verbatim transcript of the user's utterances, the transcriptionist would need to perform a significant amount of editing to obtain a document conforming to the customary standards. We need to look for what the user wants rather than what he says.

Natural language processing research has addressed a number of these issues as individual problems: automatic punctuation (Liu et al., 2005), text segmentation (Beeferman et al., 1999; Matusov et al., 2003) disfluency repair (Heeman et al., 1996) and error correction (Ringger and Allen, 1996; Strzalkowski and Brandow, 1997; Peters and Drexel, 2004). The method we present in the following attempts to address all this by a unified transformation model. The goal is simply stated as transforming the recognition output into a text document. We will first describe the general framework of learning transformations from example documents. In the following two sections we will discuss a rule-induction-based and a probabilistic transformation method respectively. Finally we present experimental results in the context of medical transcription and conclude with an assessment of both methods.

2 Text transformation

In dictation and transcription management systems corresponding pairs of recognition output and edited and corrected documents are readily available. The idea of transformation modeling, outlined in figure 1, is to learn to emulate the transcriptionist. To this end we first process archived dictations with the speech recognizer to create approximate verbatim transcriptions. For each document this yields the spoken or *source* word sequence $S = s_1 \dots s_M$, which is supposed to be a word-by-word transcription of the user's utterances, but which may actually contain recognition errors. The corresponding final reports are cleaned (removal of page headers etc.), tagged (identification of section headings and enumerated lists) and tokenized, yielding the text or *target* token sequence $T = t_1 ... t_N$ for each document. Generally, the token sequence corresponds to the spoken form. (E.g. "25mg" is tokenized as "twenty five milligrams".) Tokens can be ordinary words or special symbols representing line breaks, section headings, etc. Specifically, we represent each section heading by a single indivisible token, even if the section name consists of multiple words. Enumerations are represented by special tokens, too. Different techniques can be applied to learn and execute the actual transformation from S to T. Two options are discussed in the following.

With the transformation model at hand, a draft for a new document is created in three steps. First the speech recognizer processes the audio recording and produces the source word sequence S. Next, the transformation step converts S into the target sequence T. Finally the transformation output T is formatted into a text document. Formatting is the

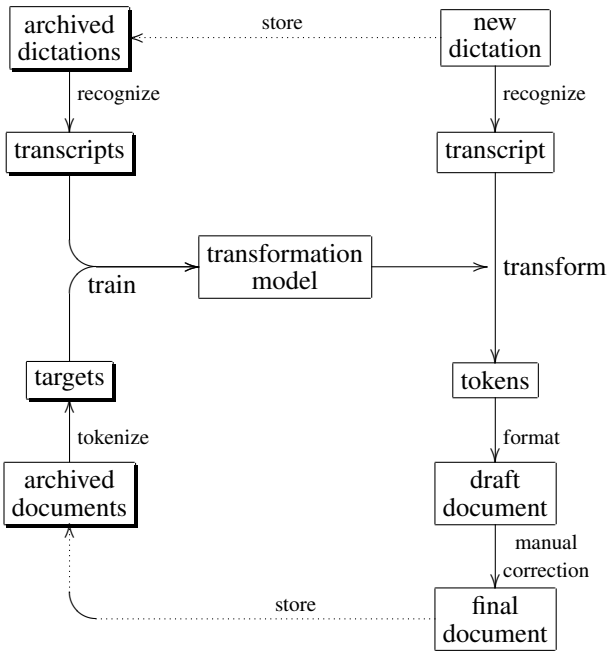

Figure 1: Illustration of how text transformation is integrated into a speech-to-text system.

inverse of tokenization and includes conversion of number words to digits, rendition of paragraphs and section headings, etc.

Before we turn to concrete transformation techniques, we can make two general statements about this problem. Firstly, in the absence of observations to the contrary, it is reasonable to leave words unchanged. So, a priori the mapping should be the identity. Secondly, the transformation is mostly monotonous. Out-of-order sections do occur but are the exception rather than the rule.

3 Transformation based learning

Following Strzalkowski and Brandow (1997) and Peters and Drexel (2004) we have implemented a *transformation-based learning* (TBL) algorithm (Brill, 1995). This method iteratively improves the match (as measured by token error rate) of a collection of corresponding source and target token sequences by positing and applying a sequence of *substitution rules*. In each iteration the source and target tokens are aligned using a minimum edit distance criterion. We refer to maximal contiguous subsequences of non-matching tokens as *error regions*. These consist of paired sequences of source and target tokens, where either sequence may be empty. Each error region serves as a candidate substitution rule. Additionally we consider refinements of these rules with varying amounts of contiguous context tokens on either side. Deviating from Peters and Drexel (2004), in the special case of an empty target sequence, i.e. a deletion rule, we consider deleting all (non-empty) contiguous subsequences of the source sequence as well. For each candidate rule we accumulate two counts: the number of exactly matching error regions and the number of false alarms, i.e. when its left-hand-side matches a sequence of already correct tokens. Rules are ranked by the difference in these counts scaled by the number of errors corrected by a single rule application, which is the length of the corresponding error region. This is an approximation to the total number of errors corrected by a rule, ignoring rule interactions and non-local changes in the minimum edit distance alignment. A subset of the top-ranked non-overlapping rules satisfying frequency and minimum impact constraints are selected and the source sequences are updated by applying the selected rules. Again deviating from Peters and Drexel (2004), we consider two rules as overlapping if the left-hand-side of one is a contiguous subsequence of the other. This procedure is iterated until no additional rules can be selected. The initial rule set is populated by a small sequence of hand-crafted rules (e.g. "impression colon" → "IMPRESSION:"). A user-independent baseline rule set is generated by applying the algorithm to data from a collection of users. We construct speaker-dependent models by initializing the algorithm with the speaker-independent rule set and applying it to data from the given user.

4 Probabilistic model

The canonical approach to text transformation following statistical decision theory is to maximize the text document posterior probability given the spoken document.

$$T^* = \operatorname*{argmax}_T p(T|S) \quad (1)$$

Obviously, the global model $p(T|S)$ must be constructed from smaller scale observations on the cor-

respondence between source and target words. We use a 1-to-n alignment scheme. This means each source word is assigned to a sequence of zero, one or more target words. We denote the target words assigned to source word s_i as τ_i. Each replacement τ_i is a possibly empty sequence of target words. A source word together with its replacement sequence will be called a *segment*. We constrain the set of possible transformations by selecting a relatively small set of allowable replacements $A(s)$ to each source word. This means we require $\tau_i \in A(s_i)$. We use the usual m-gram approximation to model the joint probability of a transformation:

$$p(S,T) = \prod_{i=1}^{M} p(s_i, \tau_i | s_{i-m+1}, \tau_{i-m+1}, \ldots s_{i-1}, \tau_{i-1}) \quad (2)$$

The work of Ringger and Allen (1996) is similar in spirit to this method, but uses a factored source-channel model. Note that the decision rule (1) is over whole documents. Therefore we processes complete documents at a time without prior segmentation into sentences.

To estimate this model we first align all training documents. That is, for each document, the target word sequence is segmented into M segments $T = \tau_1 \frown \ldots \frown \tau_M$. The criterion for this alignment is to maximize the likelihood of a segment unigram model. The alignment is performed by an expectation maximization algorithm. Subsequent to the alignment step, m-gram probabilities are estimated by standard language modeling techniques. We create speaker-specific models by linearly interpolating an m-gram model based on data from the user with a speaker-independent background m-gram model trained on data pooled from a collection of users.

To select the allowable replacements for each source word we count how often each particular target sequence is aligned to it in the training data. A source target pair is selected if it occurs twice or more times. Source words that were not observed in training are immutable, i.e. the word itself is its only allowable replacement $A(s) = \{(s)\}$. As an example suppose "patient" was deleted 10 times, left unchanged 105 times, replaced by "the patient" 113 times and once replaced by "she". The word patient would then have three allowables: $A(\text{patient}) = \{(), (\text{patient}), (\text{the}, \text{patient})\}$.)

The decision rule (1) minimizes the document error rate. A more appropriate loss function is the number of source words that are replaced incorrectly. Therefore we use the following *minimum word risk* (MWR) decision strategy, which minimizes source word loss.

$$T^* = (\operatorname*{argmax}_{\tau_1 \in A(s_i)} p(\tau_1|S)) \frown \ldots \frown (\operatorname*{argmax}_{\tau_M \in A(s_M)} p(\tau_M|S)) \quad (3)$$

This means for each source sequence position we choose the replacement that has the highest posterior probability $p(\tau_i|S)$ given the entire source sequence. To compute the posterior probabilities, first a graph is created representing alternatives "around" the most probable transform using beam search. Then the forward-backward algorithm is applied to compute edge posterior probabilities. Finally edge posterior probabilities for each source position are accumulated.

5 Experimental evaluation

The methods presented were evaluated on a set of real-life medical reports dictated by 51 doctors. For each doctor we use 30 reports as a test set. Transformation models are trained on a disjoint set of reports that predated the evaluation reports. The typical document length is between one hundred and one thousand words. All dictations were recorded via telephone. The speech recognizer works with acoustic models that are specifically adapted for each user, not using the test data, of course. It is hard to quote the verbatim word error rate of the recognizer, because this would require a careful and time-consuming manual transcription of the test set. The recognition output is auto-punctuated by a method similar in spirit to the one proposed by Liu et al. (2005) before being passed to the transformation model. This was done because we considered the auto-punctuation output as the status quo ante which transformation modeling was to be compared to. Neither of both transformation methods actually relies on having auto-punctuated input. The auto-punctuation step only inserts periods and commas and the document is not explicitly segmented into sentences. (The transformation step always applies to entire documents and the interpretation of a period as a sentence boundary is left to the human

Table 1: Experimental evaluation of different text transformation techniques with different amounts of user-specific data. Precision, recall, deletion, insertion and error rate values are given in percent and represent the average of 51 users, where the results for each user are the ratios of sums over 30 reports.

	user	sections		punctuation		all tokens		
method	docs	precision	recall	precision	recall	deletions	insertions	errors
none (only auto-punct)		0.00	0.00	66.68	71.21	11.32	27.48	45.32
TBL	SI	69.18	44.43	73.90	67.22	11.41	17.73	34.99
3-gram	SI	65.19	44.41	73.79	62.26	18.15	12.27	36.09
TBL	25	75.38	53.39	75.59	69.11	10.97	15.97	32.62
3-gram	25	80.90	59.37	78.88	69.81	11.50	12.09	28.87
TBL	50	76.67	56.18	76.11	69.81	10.81	15.53	31.92
3-gram	50	81.10	62.69	79.39	70.94	11.31	11.46	27.76
TBL	100	77.92	58.03	76.41	70.52	10.67	15.19	31.29
3-gram	100	81.69	64.36	79.35	71.38	11.48	10.82	27.12
3-gram without MWR	100	81.39	64.23	79.01	71.52	11.55	10.92	27.29

reader of the document.) For each doctor a background transformation model was constructed using 100 reports from each of the *other* users. This is referred to as the speaker-independent (SI) model. In the case of the probabilistic model, all models were 3-gram models. User-specific models were created by augmenting the SI model with 25, 50 or 100 reports. One report from the test set is shown as an example in the appendix.

5.1 Evaluation metric

The output of the text transformation is aligned with the corresponding tokenized report using a minimum edit cost criterion. Alignments between section headings and non-section headings are not permitted. Likewise no alignment of punctuation and non-punctuation tokens is allowed. Using the alignment we compute precision and recall for sections headings and punctuation marks as well as the overall token error rate. It should be noted that the so derived error rate is not comparable to word error rates usually reported in speech recognition research. All missing or erroneous section headings, punctuation marks and line breaks are counted as errors. As pointed out in the introduction the reference texts do not represent a literal transcript of the dictation. Furthermore the data were not cleaned manually. There are, for example, instances of letter heads or page numbers that were not correctly removed when the text was extracted from the word processor's file format. The example report shown in the appendix features some of the typical differences between the produced draft and the final report that may or may not be judged as errors. (For example, the date of the report was not given in the dictation, the section names "laboratory data" and "laboratory evaluation" are presumably equivalent and whether "stable" is preceded by a hyphen or a period in the last section might not be important.) Nevertheless, the numbers reported do permit a quantitative comparison between different methods.

5.2 Results

Results are stated in table 1. In the baseline setup no transformation is applied to the auto-punctuated recognition output. Since many parts of the source data do not need to be altered, this constitutes the reference point for assessing the benefit of transformation modeling. For obvious reasons precision and recall of section headings are zero. A high rate of insertion errors is observed which can largely be attributed to preambles. Both transformation methods reduce the discrepancy between the draft document and the final corrected document significantly. With 100 training documents per user the mean token error rate is reduced by up to 40% relative by the probabilistic model. When user specific data is used, the probabilistic approach performs consistently better than TBL on all accounts. In particular it always has much lower insertion rates reflecting its supe-

rior ability to remove utterances that are not typically part of the report. On the other hand the probabilistic model suffers from a slightly higher deletion rate due to being overzealous in this regard. In speaker independent mode, however, the deletion rate is excessively high and leads to inferior overall performance. Interestingly the precision of the automatic punctuation is increased by the transformation step, without compromising on recall, at least when enough user specific training data is available. The minimum word risk criterion (3) yields slightly better results than the simpler document risk criterion (1).

6 Conclusions

Automatic text transformation brings speech recognition output much closer to the end result desired by the user of a back-end dictation system. It automatically punctuates, sections and rephrases the document and thereby greatly enhances transcriptionist productivity. The holistic approach followed here is simpler and more comprehensive than a cascade of more specialized methods. Whether or not the holistic approach is also more accurate is not an easy question to answer. Clearly the outcome would depend on the specifics of the specialized methods one would compare to, as well as the complexity of the integrated transformation model one applies. The simple models studied in this work admittedly have little provisions for targeting specific transformation problems. For example the typical length of a section is not taken into account. However, this is not a limitation of the general approach. We have observed that a simple probabilistic sequence model performs consistently better than the transformation-based learning approach. Even though neither of both methods is novel, we deem this an important finding since none of the previous publications we know of in this domain allow this conclusion. While the present experiments have used a separate auto-punctuation step, future work will aim to eliminate it by integrating the punctuation features into the transformation step. In the future we plan to integrate additional knowledge sources into our statistical method in order to more specifically address each of the various phenomena encountered in spontaneous dictation.

References

Beeferman, Doug, Adam Berger, and John Lafferty. 1999. Statistical models for text segmentation. *Machine Learning*, 34(1-3):177 – 210.

Brill, Eric. 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543 – 565.

Heeman, Peter A., Kyung-ho Loken-Kim, and James F. Allen. 1996. Combining the detection and correction of speech repairs. In *Proc. Int. Conf. Spoken Language Processing (ICSLP)*, pages 362 – 365. Philadelphia, PA, USA.

Liu, Yang, Andreas Stolcke, Elizabeth Shriberg, and Mary Harper. 2005. Using conditional random fields for sentence boundary detection in speech. In *Proc. Annual Meeting of the ACL*, pages 451 – 458. Ann Arbor, MI, USA.

Matusov, Evgeny, Jochen Peters, Carsten Meyer, and Hermann Ney. 2003. Topic segmentation using markov models on section level. In *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 471 – 476. IEEE, St. Thomas, U.S. Virgin Islands.

Peters, Jochen and Christina Drexel. 2004. Transformation-based error correction for speech-to-text systems. In *Proc. Int. Conf. Spoken Language Processing (ICSLP)*, pages 1449 – 1452. Jeju Island, Korea.

Ringger, Eric K. and James F. Allen. 1996. A fertility channel model for post-correction of continuous speech recognition. In *Proc. Int. Conf. Spoken Language Processing (ICSLP)*, pages 897 – 900. Philadelphia, PA, USA.

Strzalkowski, Tomek and Ronald Brandow. 1997. A natural language correction model for continuous speech recognition. In *Proc. 5th Workshop on Very Large Corpora (WVVLC-5):*, pages 168 – 177. Beijing-Hong Kong.

Appendix A. Example of a medical report

Recognition output. Vertical space was added to facilitate visual comparison.

doctors *name* dictating a progress note on *first name last name* patient without complaints has been ambulating without problems no chest pain chest pressure still has some shortness of breath but overall has improved significantly

vital signs are stable she is afebrile lungs show decreased breath sounds at the bases with bilateral rales and rhonchi heart is regular rate and rhythm two over six crescendo decrescendo murmur at the right sternal border abdomen soft nontender nondistended extremities show one plus pedal edema bilaterally neurological exam is nonfocal

white count of five point seven H. and H. eleven point six and thirty five point five platelet count of one fifty five sodium one thirty seven potassium three point nine chloride one hundred carbon dioxide thirty nine calcium eight point seven glucose ninety one BUN and creatinine thirty seven and one point one

impression number one COPD exacerbation continue breathing treatments number two asthma exacerbation continue oral prednisone number three bronchitis continue Levaquin number four hypertension stable number five uncontrolled diabetes mellitus improved number six gastroesophageal reflux disease stable number seven congestive heart failure stable

new paragraph patient is in stable condition and will be discharged to *name* nursing home and will be monitored closely on an outpatient basis progress note

Automatically generated draft (speech recognition output after transformation and formatting)

Progress note

SUBJECTIVE: The patient is without complaints. Has been ambulating without problems. No chest pain, chest pressure, still has some shortness of breath, but overall has improved significantly.

PHYSICAL EXAMINATION:

VITAL SIGNS: Stable. She is afebrile.

LUNGS: Show decreased breath sounds at the bases with bilateral rales and rhonchi.

HEART: Regular rate and rhythm 2/6 crescendo decrescendo murmur at the right sternal border.

ABDOMEN: Soft, nontender, nondistended.

EXTREMITIES: Show 1+ pedal edema bilaterally.

NEUROLOGICAL: Nonfocal.

LABORATORY DATA: White count of 5.7, hemoglobin and hematocrit 11.6 and 35.5, platelet count of 155, sodium 137, potassium 3.9, chloride 100, CO2 39, calcium 8.7, glucose 91, BUN and creatinine 37 and 1.1.

IMPRESSION:

1. Chronic obstructive pulmonary disease exacerbation. Continue breathing treatments.
2. Asthma exacerbation. Continue oral prednisone.
3. Bronchitis. Continue Levaquin.
4. Hypertension. Stable.
5. Uncontrolled diabetes mellitus. Improved.
6. Gastroesophageal reflux disease, stable.
7. Congestive heart failure. Stable.

PLAN: The patient is in stable condition and will be discharged to *name* nursing home and will be monitored closely on an outpatient basis.

Final report produced by a human transcriptionist without reference to the automatic draft.

Progress Note

DATE: July 26, 2005.

HISTORY OF PRESENT ILLNESS: The patient has no complaints. She is ambulating without problems. No chest pain or chest pressure. She still has some shortness of breath, but overall has improved significantly.

PHYSICAL EXAMINATION:

VITAL SIGNS: Stable. She's afebrile.

LUNGS: Decreased breath sounds at the bases with bilateral rales and rhonchi.

HEART: Regular rate and rhythm. 2/6 crescendo, decrescendo murmur at the right sternal border.

ABDOMEN: Soft, nontender and nondistended.

EXTREMITIES: 1+ pedal edema bilaterally.

NEUROLOGICAL EXAMINATION: Nonfocal.

LABORATORY EVALUATION: White count 5.7, H&H 11.6 and 35.5, platelet count of 155, sodium 137, potassium 3.9, chloride 100, co2 39, calcium 8.7, glucose 91, BUN and creatinine 37 and 1.1.

IMPRESSION:

1. Chronic obstructive pulmonary disease exacerbation. Continue breathing treatments.
2. Asthma exacerbation. Continue oral prednisone.
3. Bronchitis. Continue Levaquin.
4. Hypertension-stable.
5. Uncontrolled diabetes mellitus-improved.
6. Gastroesophageal reflux disease-stable.
7. Congestive heart failure-stable.

The patient is in stable condition and will be discharged to *name* Nursing Home, and will be monitored on an outpatient basis.