Interspeech 2018
2-6 September 2018, Hyderabad

An automated assistant for medical scribes

Greg P. Finley, Erik Edwards, Amanda Robinson, Najmeh Sadoughi, James Fone, Mark Miller, David Suendermann-Oeft, Michael Brenndoerfer[1], Nico Axtmann[2]

EMR.AI Inc., San Francisco, CA, USA
[1]University of California, Berkeley, USA
[2]DHBW, Karlsruhe, Germany

greg.finley@emr.ai

Abstract

Healthcare professionals spend a significant amount of their time on administrative tasks rather than direct patient care. One effective way to alleviate some of this burden is to employ a medical scribe, who charts patient-physician encounters in real time. We present a complete implementation of an automated medical scribe assistant, which listens to the encounter and produces a draft of report text and the information used to create it, which can dramatically streamline the scribe's task. This system is, to our knowledge, the first automated scribe ever presented, and relies on multiple speech and language technologies, including speaker diarization, medical speech recognition, knowledge extraction, and natural language generation.

Index Terms: speech recognition, speaker diarization, medical speech, natural language processing

1. Introduction

A recent study concluded that primary care physicians spend almost two hours on tasks related to electronic medical record (EMR) systems for every one hour of direct patient care [1]. One strategy for reducing this overhead is to employ a *medical scribe*: a clinical paraprofessional who interacts with EMR systems in real time during patient-physician encounters. Scribes have been shown to save physicians substantial time, improve their work-life balance, and enhance productivity [2]. Scribes do have some disadvantages, however: a high rate of turnover, as they tend to be students; extended training time required; and cost [3]. To help offset these disadvantages, we have developed an automated scribe assistant that employs a full stack of state-of-the-art speech and natural language processing components to transform a recorded conversation into an initial draft of a fully formatted report, which can be marked up and corrected by human scribes. This proposed workflow is identical to one frequently employed for transcribing dictated medical notes, in which human transcribers mark up a draft produced by speech recognition and formatting systems [4]. Our system has the potential to substantially increase the throughput of human scribes, making them a more affordable and attractive proposition for healthcare providers. To the best of our knowledge, this is the first automated scribe implementation ever presented to the scientific community.

The processing pipeline comprises four major stages: speaker diarization, automatic speech recognition (ASR) [4], knowledge extraction (KE), and natural language generation (NLG). The goal of the latter two is exemplified in Figure 1, in which a segment of a conversation is mined for appropriate information used to produce a formatted section of report.

2. Speaker diarization

Speaker diarization is the "who spoke when" problem, also called speaker indexing [5]. The input is audio features sampled at 100 Hz frame rate, and the output is frame-labels indicating speaker identity for each frame. Four labels are possible: speaker 1 (e.g. the doctor), speaker 2 (e.g. the patient), overlap (both speakers), and silence (within-speaker pauses and between-speaker gaps). Note that the great majority of doctor-patient encounters involve exactly two speakers. Although our method is easily generalizable to more speakers, we currently report on the two-speaker problem.

The diarization literature broadly distinguishes "bottom-up" vs. "top-down" approaches. The former [6] operate by merging neighboring frames according to similarity (clustering); we found initial results unsatisfactory. The latter operate with a prior model such as HMM-GMM (Hidden Markov, Gaussian mixture model) to represent the likely audio features and timing (transition) characteristics of dialogs. We have introduced our own top-down approach that utilizes a modified expectation maximization algorithm at decoding time to learn the current speaker and background silence characteristics in real time.

Diarization requires an expanded set of audio features compared to ASR. In ASR, only phoneme identity is of final interest, so audio features are generally insensitive to speaker characteristics. Also note that, as diarization performs a *de facto* speech activity detection (SAD), features successful for SAD [7] are helpful to diarization as well. Accordingly, we use an expanded set of gammatone-based audio features for the total SAD + diarization + ASR problem.

3. Speech recognition

ASR operates on the audio segments produced by the diarization stage, where each segment contains one conversational turn (1 speaker + possibly a few frames of overlap). Currently, the diarization and ASR stages are strictly separated and the ASR decoding operates by the same neural network (NN) methodology that we recently reported for general medical ASR [4]. In brief, the acoustic model (AM) consists of a NN trained to predict context-sensitive phones from the audio features; and the language model (LM) is a 3- or 4-gram statistical LM prepared with methods of interpolation and pruning that we developed to address the massive medical-vocabulary challenge. Decoding operates in real time by use of weighted finite-state transducer methodology [8]. Our current challenge is in adapting the AM and LM to medical conversations, which have somewhat different statistics compared to medical dictations.

Conversation	Report
Dr: "okay great and in terms of your past medical history do you have any other medical conditions you have"	FAMILY MEDICAL HISTORY The patient's aunt had lung cancer.
Pt: "no i have not had any medical conditions but my auntie actually she had lung cancer so that's why i kind of…"	

Figure 1: *An excerpt from a typical input and output for the natural language processing stages of the scribe. Note that the ASR output has no punctuation or case. The doctor ('Dr.') and patient ('Pt.') identifiers illustrate the contribution of speech diarization.*

4. Knowledge extraction

The KE stage locates any relevant unstructured information in the conversation and converts it into a structured representation. Extracting information from spontaneous conversational speech is a notoriously difficult problem. There has been some recent work in this area, although it is unclear whether any known methods are suitable to clinical conversation specifically. We apply a novel strategy to simplify the KE problem by classifying turns in the conversation based upon the information they likely contain using hierarchical recurrent NNs. These classes overlap largely with sections in the final report—chief complaint, medical history, etc.

We then apply a variety of information extraction strategies on these sections of text, including: rule-based processing to identify predictable elements (medication dosages, dates and durations, certain ontology concepts); knowledge-based strategies, such as calculating a phrase's semantic overlap with dictionary definitions, for concepts that can vary widely in expression (e.g., symptom descriptions); and fully supervised machine learning approaches for difficult or highly specialized tasks—e.g., identifying highly variable events such as symptoms generally worsening. This module also relies on extractive summary techniques where necessary, in which entire sentences may be kept if they refer to information that is relevant but is difficult to represent in structured form—for example, a description of how a patient sustained a workplace injury.

Keeping structured data as an intermediate output has a number of advantages. Most relevantly to our system, it allows a human scribe to see what information the system has detected and amend it directly if necessary. Structured data can also be kept to assist in transcribing later visits by the same patient or for use by other systems that read structured data (e.g., billing systems, decision support). Wherever possible, data is encoded in structures compatible with common medical informatics standards to facilitate interoperability.

5. Natural language generation

The NLG module produces and formats the final report. Medical reports often follow a loosely standardized format: sections appear in a generally predictable order and have well-defined scope. Our strategy is a data-driven templatic approach in conjunction with a finite-state "grammar" of report structure.

Sentence templates, annotated for the structured data types necessary to complete them, constitute a sentence bank, which we fill by clustering sentences from a large corpus of medical reports according to semantic and syntactic similarity. Results are manually curated to ensure that strange or imprecise sentences cannot be generated and to ensure parsimony in the information type system. (See Kondadadi *et al.* [9] for a similar method.)

Using the same corpus, we induce a document-level grammar using a probabilistic finite-state graph, where each arc is a sentence and a single path through the graph represents one full report. Decoding jointly optimizes the maximal use of structured data and the likelihood of the path. The grammar helps to improve upon one common criticism of templatic NLG approaches, which is the lack of variation in sentences [10], in a way that does not require any "inflation" of the template bank with synonyms or paraphrases. As note structure can vary considerably between specialty and hospital, we build separate NLG models to handle each type of output.

Finally, all notes pass through a processor that handles reference and anaphora (e.g., replacing references to the patient with gendered pronouns), truecasing, formatting, etc.

6. Development status

Our system is currently in an early prototype stage, with all modules functioning, but limited in scope. We will demonstrate a full end-to-end system for patient–physician conversations captured in a general practitioner's office.

7. References

[1] B. Arndt, J. Beasley, M. Watkinson, J. Temte, W.-J. Tuan, C. Sinsky, and V. Gilchrist, "Tethered to the EHR: primary care physician workload assessment using EHR event log data and time-motion observations," *Ann Fam Med*, vol. 15, no. 5, pp. 419–426, 2017.

[2] S. Earls, J. Savageau, S. Begley, B. Saver, K. Sullivan, and A. Chuman, "Can scribes boost FPs' efficiency and job satisfaction?" *J Fam Pract*, vol. 66, no. 4, pp. 206–214, 2017.

[3] K. Walker, W. Dunlop, D. Liew, M. Staples, M. Johnson, M. Ben-Meir, H. Rodda, I. Turner, and D. Phillips, "An economic evaluation of the costs of training a medical scribe to work in emergency medicine," *Emerg Med J*, vol. 33, no. 12, pp. 865–869, 2016.

[4] E. Edwards, W. Salloum, G. Finley, J. Fone, G. Cardiff, M. Miller, and D. Suendermann-Oeft, "Medical speech recognition: reaching parity with humans," in *Proc SPECOM*, vol. LNCS 10458. Springer, 2017, pp. 512–524.

[5] M. Moattar and M. Homayounpour, "A review on speaker diarization systems and approaches," *Speech Commun*, vol. 54, no. 10, pp. 1065–1103, 2012.

[6] H. Gish, M.-H. Siu, and J. Rohlicek, "Segregation of speakers for speech recognition and speaker identification," in *Proc ICASSP*, vol. 2. IEEE, 1991, pp. 873–876.

[7] S. Sadjadi and J. Hansen, "Unsupervised speech activity detection using voicing measures and perceptual spectral flux," *IEEE Signal Process Lett*, vol. 20, no. 3, pp. 197–200, 2013.

[8] C. Allauzen, M. Riley, J. Schalkwyk, and M. Mohri, "OpenFst: a general and efficient weighted finite-state transducer library," in *Proc CIAA*, vol. LNCS 4783. Springer, 2007, pp. 11–23.

[9] R. Kondadadi, B. Howald, and F. Schilder, "A statistical NLG framework for aggregated planning and realization," in *Proc ACL*. ACL, 2013, pp. 1406–1415.

[10] K. van Deemter, M. Theune, and E. Krahmer, "Real versus template-based natural language generation: a false opposition?" *Comput Linguist*, vol. 31, no. 1, pp. 15–24, 2005.