

Computer-assisted Coding for Inpatients—a Case Study

by Cheryl Servais MPH, RHIA

Abstract

Computer-assisted coding (CAC) has often been marketed for its ability to increase productivity. We undertook two studies to determine if productivity for inpatient coding was increased using CAC software. A number of significant problem areas were identified and it was determined that the CAC software did not increase coder productivity.

Introduction

We have been testing computer-assisted coding (CAC) software since 2002 in order to increase the productivity of our remote coding staff. This paper describes several of our initiatives and the results.

Background

“The term computer-assisted coding is currently used to denote technology that automatically assigns codes from clinical documentation for a human...to review, analyze, and use,” explains Rita Scichilone, MHSA, RHIA, CCS, CCS-P, director of coding products and services at AHIMA.¹ We became familiar with this technology and sought a way to harness its potential. We did recognize that the development of such software was not mature, as results of other projects have been reported in the literature. One example of the issues surrounding computer assisted coding (CAC) was provided by Beth Friedman: “More often, questions were raised about CAC quality and reliability. Results indicated that 29 percent of respondents are concerned that CAC systems would result in high productivity but poor quality, and 38 percent expressed concern that these systems were not yet proven.”²

There are a variety of methodologies employed by developers of CAC software to “read” text and assign codes. The software can use structured input (SI) or natural language processing (NLP). Even within the NLP range of products, there are a variety of approaches with varying levels of sophistication. The methodology used has a tremendous impact on data transmission and the output reviewed by the coders

Study Method

Our general study method consisted of selecting transcribed files for 100 patients. The files consisted of the History and Physical report, any consultation report or operative report, and discharge summary.

Not all reports were available for all patients. Both studies focused on inpatients because we felt that if a company had developed software sophisticated enough to provide accurate code assignment for inpatient records, they could easily adapt the software to handle outpatient records.

We reviewed the software product on several criteria:

1. Accuracy of code assignment
2. Ease of use (including transmission of transcribed data, initial coder instruction, and functionality of output information)
3. Ability of the software and review process to enhance coder productivity

Pilot 1

Our first pilot involved a vendor whose product assigned codes for inpatients and outpatients. The software used SI coding. In order for the software to “read” the clinical terms to be codes, the diagnostic or procedural phrase had to start on the same line and at the same column in each different transcribed document. It was not possible for us to meet this requirement with transcription coming from multiple clients with varying report formats. The only alternative was to place delimiters around the text to be read for coding. This proved tedious and time-consuming. It negated any productivity increases by the coders. In addition, the requirement for the strict identification of phrases meant that any diagnostic or procedural information embedded in the body of the report could not be “read” by the software.

Other problems became apparent during the study:

The software was unable to assign Current Procedural Terminology (CPT) codes, so this eliminated its use for outpatients.

The software was fairly accurate in ICD-9-CM code assignment (approximately 70 percent accuracy rate), but the list of assigned codes included extensive notes to physicians regarding possible documentation improvement and extensive notes to coders about possible additional or different codes. While interesting the first time they appeared, the notes became overwhelmingly repetitive. Since each note was interspersed with its associated code in the list of codes assigned by the software, it was not possible for a coder to quickly review the codes for accuracy or completeness. The code listing spanned several screens or, if printed out, several pages. There was no way to turn off this feature other than totally eliminating all edits. Because of these operational issues—the pilot was discontinued.

Pilot 2

Our second pilot involved a product that only assigned codes for inpatients. It truly “read” the text and assigned ICD-9-CM codes for both diagnoses and procedures. It did not sequence codes, but left that task to the coder. Human intervention was required to determine the principal diagnosis. Once a coder made that selection, the software determined the diagnosis-related group (DRG). With the selection of additional codes for secondary diagnoses and procedures, the software updated the DRG assignment.

The computer assigned codes appeared in a window. If coders clicked on the code, they were taken to the report and text that served as the basis for the code assignment. If there were multiple text references, the coder would be led to each one for review.

An encoder and grouper were embedded in the CAC product, so coders could “recode” any diagnostic or procedural term. All activity occurred on one screen through various windows. The CAC software analyzed documentation for 100 patients as called for in our study method.

Results of Pilot 2

1. Coding Accuracy

Codes assigned were accurate, but not always appropriate. The coders did not accept 75 percent of the diagnosis (dx) codes and 90 percent of the procedure (px) codes. In 58 percent of the cases, coders added diagnosis codes. In 45 percent of the cases, procedure codes were added. The codes required to determine the correct DRG were present in 48 percent of the cases.

Specific Problem Areas

- The codes assigned by software were listed in numeric order, making review and resequencing tedious.
- The software had difficulty distinguishing between status codes and follow-up codes (V-codes).
- For obstetric codes or other code sections in which the 5th digit needed to be determined as a distinct thought process, the software selected code with an “x” for the 5th digit or reported back a range of codes.
- The software had difficulty distinguishing between poisonings and adverse drug events, although even human coders have trouble with this. There was no E code (External Cause of Injury Code) selection by the software.
- The software had a problem selecting E codes for accidents. No “Place of Occurrence” E Codes were assigned.
- The determination of “history of” conditions versus “active” conditions also proved problematic for the software.
- The software had difficulty in determining context of “cervical”—neck versus cervix. It coded Cervical Spine Exam to Examination of cervix.
- There were some misses on code assignment—chronic lower leg edema was coded to unequal leg length.
- Chest pain with radiation was coded as chest pain with radiotherapy.
- Some procedures were not coded at all (such as biventricular pacemaker).
- Some grouping issues were noted (certain principal diagnosis codes would not group to the correct DRG).
- The system did not allow for disposition code effect on DRG. It assumed all discharges were to be assigned to the status of “home.”
- The system defaulted birth defects to “congenital” codes.
- The software missed the code for the specific organisms for many infections.
- The software had difficulty dealing with some acronyms and abbreviations or distinguishing an abbreviation from letters imbedded in a word—for example, CAD (academic).
- The software had great difficulty dealing with procedures with multiple components (which parts to bundle and which to code separately).

2. Ease of Use

- The software was very intuitive. It required minimal training (less than one/half day) and the coders liked using it.
- The software had no problems importing the transcribed documents. No data manipulation required.

3. Enhanced Coder Productivity

Productivity was not enhanced due to the coder needing to research each of the numerous codes returned by the program (average of 20 codes per case). Coder A reported coding two and a half charts

per hour. Coder B reported coding four charts per hour. Our productivity range for inpatient is three to five charts coded per hour. Clearly, this software did not enhance productivity.

Of note is that the key documents were not present for all patients. The History and Physical report was handwritten or came from another source (physician office, clinic) The Discharge Summary was handwritten or not required (in cases of OB, NB, short-stay patients). A key issue for CAC will be the presence of complete documentation in a form that the software can read.

Discussion

The CAC products we tested did reflect improvements in the technology from software that required structured text or parsed data in order to recognize the phrases to be coded to software that could select key phrases within text to be coded. The software tested second also reflected improved ease in use and sophistication of code selection.

Because of the vast number of codes presented to the coder, the software did not improve productivity as expected. Future versions of CAC software for inpatient documentation need to reduce the number of codes presented to the coder. The software can do this by removing symptoms when related to specific diagnoses and by eliminating the unspecified and specified forms of the same code. It needs to be “smart” enough to assign fifth digits, V-codes and E codes.

Correct coding is more than just following the rules in the codebook. Billing guidelines also determine correct selection and sequencing of codes. At this point, CAC is not a panacea for poor coding. It cannot reduce undercoding. In order to do this, the software must recognize all terms and assign the correct code. Currently, the software we tested often missed subsequent documentation or failed to link separated diagnostic or procedural statements that could result in a higher level of ICD-9-CM or CPT code.

The software also cannot reduce overcoding. In order to do this, the software must incorporate rules related to unbundling of codes that are included with the main diagnostic or procedural code. It must also eliminate codes that are not being treated (such as history or rule outs).

Conclusion

Computer-assisted coding (CAC) from inpatient documentation needs additional refinements before it will enhance coder productivity and coding accuracy.

It truth, computer-assisted inpatient coding is the most sophisticated task that anyone could ask a CAC software module to perform. As opposed to studies that report on the successful performance of CAC for outpatient coding, these studies did not have parallel success for inpatient coding.

Cheryl Servais, MPH, RHIA, is Vice President of Compliance & Privacy Officer for Precyse Solutions, LLC in Wayne, PA.

Notes

1. Friedman, Beth. “Coding Technology Today.” *Journal of AHIMA*. 77, no. 4 (2006): 67.
2. Friedman, Beth. “Coding Technology Today.” *Journal of AHIMA*. 77, no. 4 (2006): 66.

References

Blumenfeld, Barry, Joseph Ketcherside, and Robert Rashke. “Using Automated Rules to Improve Documentation and Quality.” Presented at AHIMA 2000 National Convention, Chicago, IL.

D’Amato, Cheryl. “Auto-Coding.” Presented at AHIMA 2003 National Convention, Minneapolis, MN.

American Health Information Management Association. “Delving into Computer-assisted Coding.” Practice Brief Part 9. 30-40. Available at <http://library.ahima.org>.

Friedman, Beth. “A New Day Dawning for Remote Coding.” *Advance for Health Information Professionals*. November 21, 2005: 18–20.

Garvin, Jennifer Hornung, Valerie Watzlaf, and Sohrab Moeini. “Fighting Fraud Automatically.” *Journal of AHIMA* 77, no. 4 (2006): 32–36.

Diagnosis Acceptance Findings

	Total # Diagnosis (Dx) Codes by Software	# Dx Codes Accepted by Reviewer	# Dx Codes Added by Reviewer	# Cases Where Dx Codes Added	Complications/ Co-morbid Conditions (CCs) Added by Reviewer
Number	1969	470	113	58	25
Percentage	100	24 (470/1969)	5 (113/1969)	58 (58/99)	22 (25/113)

Procedure Acceptance Findings

	Total # Procedure (Px) Codes by Software	# Px Codes Accepted by Reviewer	# Px Codes Added by Reviewer	# Cases Where Px Codes Added
Number	791	80	68	45
Percentage	100	10 (80/791)	8 (68/791)	45 (45/99)

Diagnosis Related Group (DRG) Acceptance Findings

	Software Found Code Selected for Principal Dx?	Software Found One or More CCs?	Software Found Code Selected for Principal Procedure?	Codes Needed to Identify DRG Found by Software?
Number	75	83	58	48
Percentage	76	84	59	48

DRG Acceptance by Length of Stay (LOS)

	1-4 days	5-15 days
Number with Codes for Correct DRG	31/68	15/31
% with Codes for Correct DRG	48.5%	48.4%

DRG Acceptance by Medical/Surgical Division

	Medicine	Surgery
Number with Codes for Correct DRG	32/52	16/47
Percentage with Codes for Correct DRG	62	34

DRG Acceptance by Medical Specialty

	Number with Codes for Correct DRG	Percentage with Codes for Correct DRG
Cardiovascular	15/28	54
Dermatology	0/1	0
Endocrinology	2/3	66
Gastrointestinal	7/14	50
Genitourinary	2/4	50
Gynecology	4/5	80
Hepatology	1/1	100
Infectious Disease	0/2	0
Neurology	3/5	60
Obstetrics	3/7	42
Orthopedics	5/17	43
Pulmonary Medicine	5/9	55
Trauma	0/2	0