Challenges in Implementing an AI-Based Autocoding Tool to Enhance Coding Efficiency

Introduction:

At present medical coding is a manual process that is time-consuming and threatened by a shortage of skilled coders. Rule-based tools can assist in coding simple cases but often lack adequate coding quality. Artificial intelligence introduces a more complex way of assisting and even replacing the tedious manual coding process. In this article, a leading university hospital in Belgium describes how they collaborated with a third party to assist in the development of an autocoding tool and the challenges that were faced during implementation.

Methods:

The autocoding software is based on Machine Learning and is able to predict ICD-10-CM and PCS codes for daycare stays. The University hospital's collaboration involved the extraction of historical discharge letters and MBDS data, pseudonymization of sensitive free text, development of interfaces and finally the deployment and testing of the tool in their own production environment. The whole coding team was involved in testing the autocoding tool on production data. Their role shifted from coding to a more supervisory role. The software utilized a sophisticated algorithm to identify multiple cases as suitable for autocoding, providing coders with an indication of the expected coding quality. Afterwards, an analysis was done based on the adjustments the medical coders performed on each case.

Results:

The university hospital was a pioneer in successfully implementing this AI-based autocoding software in their production environment. The medical coders reviewed 740 cases from various specialties using the tool. Due to data drift, the performance was lower than expected, but still reached a F1-score of 67.6% for diagnoses and 65.5% for procedure codes. 120 (16.6%) cases were accepted without any changes. 215 (29%) only needed one code adjustment. 339 (45%) required two code changes. More complex cases were handled accurately. Data drift was handled by retraining the machine learning model on these reviewed cases.

Conclusion:

The use of AI tools to code medical encounters is a promising technique that has the capability to code even complex cases. Machine Learning requires a lot of data to perform well, but the sensitive nature of this data makes it hard to collect. A relatively small training set, intercoder variability, and data drift are important hurdles to tackle. By combining the pseudonymized data of four hospitals, a larger training set was created which provided a significant performance increase. A real-time updating case confidence algorithm, based on coders actions, tackled the negative influence of intercoder variances in the golden standard. Data drift imposed a significant impact on coding performance. However, by retraining the model on the reviewed cases, its performance increased.

This study shows the performance of an AI-based Autocoding tool in a real-world healthcare setting. The implementation marks a paradigm shift in medical coding, offering a promising solution to the challenge of coder shortages. The study highlights the capabilities of the tool, but also emphasizes the hurdles, specific for the healthcare sector.