Toward Never Doing Known Use Problem Analysis Again

Carolyn Hartzell, MS, & Piyusha Gade, PhD

Background:

Overall Goal: Create a fully* automated Known Use Problem Analysis Tool

*to the extent possible

Current Scope: Create an **automated model** to significantly **reduce time** spent doing adverse event review for relevance

Problem Statement:

Analyzing post-market data for usability issues & trends, i.e., Known Use Problem Analysis *can* be invaluable for avoiding use problems *before* a product goes to market.

However, KUPA are also **time-intensive** (i.e., expensive) and **tedious**. This tends to mean:

a) **They get put off** and only completed after the device design is established, or

b) The **scope** of data analysis **gets pared down** to make it more feasible

Prior Work:

Last year Evolution presented an AI-based tool that analyzed MAUDE data and determined with good results whether a given event was use-related.

The model leveraged a publicly available tokenizer trained on PubMed articles and fine-tuned it using MAUDE data for autoinjectors (and similar). Last year's PubMedBert results:

Accuracy: **0.73** Precision: **0.71** Recall: **0.73**

Procedure:

- Leverage existing labeled data to determine generalizability of previous model

  Datasets include:

  ~50k client-internal adverse event data points (surgical device)

  ~6k MAUDE events on related surgical devices

  ~500 MAUDE events for injection devices

  → *Result*: Poor performance metrics

- Identify & evaluate new model(s)
- Calculate performance metrics of model using realistic use cases, optimizing for Recall
- Estimate time spent on data review for each of the above
- Compare with a Manual Review model (see Key Terms)

Model Methods:

1. Data Preprocessing

Standard text pre-processing steps/tokenization

2. Text Feature Extraction

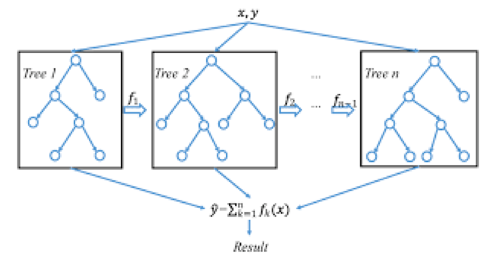
Used TF-IDF (Term Frequency-Inverse Document Frequency) to convert to numerical features. Weights words by frequency but down-weights too-frequent words

3. Our Classification Model: XGBoost (eXtreme Gradient Boosting)

- *Ensemble Learning*: Builds an ensemble of decision trees, combining multiple weak learners to create a strong predictive model.
- *Sequential Tree Building*: Trees are added to the model one at a time, with each new tree trying to correct the errors made by the previous trees.
- *Gradient Boosting*: It uses gradient descent to minimize the loss function, which measures the difference between predicted and actual values.

4. Recall-Focused Tunable Threshold

Set classification threshold to maximize recall while maintaining 'acceptable' precision.



5. Cross validation

Results:

Use Case 1. Client has past, labeled Known Use Problem Analysis (KUPA) data. Client needs to do a KUPA for a similar device.

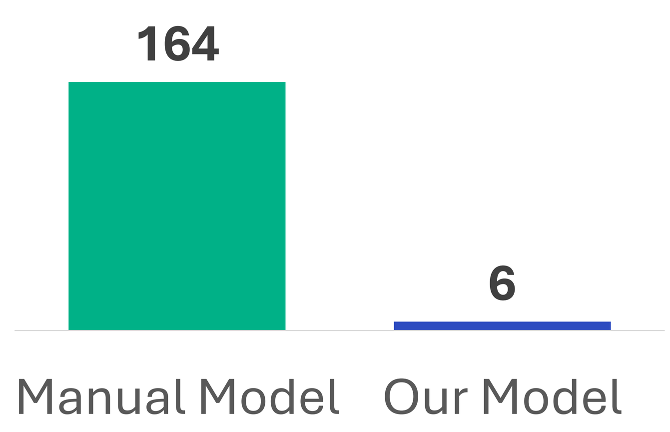
Procedure to evaluate model: Train on 80% of internal and MAUDE datasets from surgical device data. Imagine that the remaining 20% is the 'new' unlabeled data and use as Test set.

Key Takeaway: Using the XGBoost model on similar device data achieved **96% reduction in review time** and **found 84%** of use-related problems.

Performance Metrics

	Manual Model	Our Model
Recall	**1**	**84%**
Precision	1	28%
Accuracy	1	97%

Review Time (Hrs)



Confusion Matrix

	−	+
−	11,841	329
+	24	126

Only 20 missed Use Events

Events to review

Note: Performance was similar for similar use case of having labeled internal data and wanting to avoid manually labeling MAUDE data.

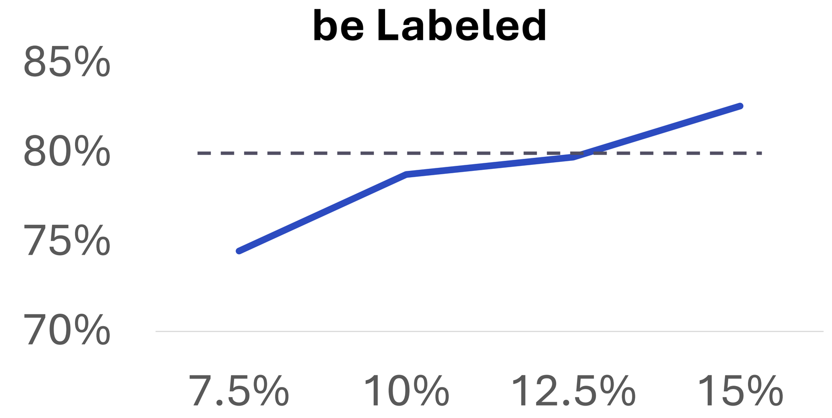
Use Case 2. Client has labeled data and needs to do a KUPA for a different device.

Procedure to evaluate model: Train on internal and MAUDE datasets from surgical device data. Imagine MAUDE injection device dataset is the 'new' non-annotated data and use as Test set.

Result: Poor performance metrics. Recall <0.10

Follow-up Procedure: Determine what percentage of MAUDE injection device data would need to be added to Training set to achieve at least 80% Recall.

Recall versus % New Data to be Labeled

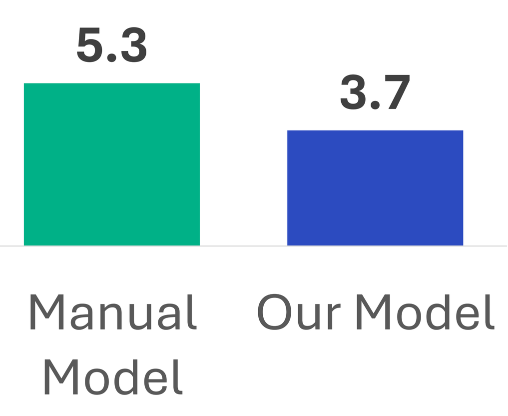


Key Takeaways: **(1)** While the model did not work well on the new device set, if we were to **annotate only ~12.5% of the new data**, it would perform adequately. **(2)** Time savings are more significant with larger datasets.

Performance Metrics

	Manual Model	Our Model
Recall	**1**	**80%**
Precision	1	29%
Accuracy	1	56%

Review Time (Hrs)



Confusion Matrix

	−	+
−	154	159
+	17	66

Performance metrics below assumed 12.5% of MAUDE data was manually labeled

Key Terms:

Confusion Matrix:

True Label \ Predicted Label	−	+
−	Correct Negatives	Incorrect Positives
+	Incorrect Negatives	Correct Positives

Recall: $\frac{Correct\ Positives}{True\ Positives} = \frac{CP}{CP + IN}$

High Recall means few missed events

Precision: $\frac{True\ Positives}{Labeled\ Positive} = \frac{CP}{CP + IP}$

High Precision means few false positive events

Accuracy: $\frac{Labeled\ Correctly}{Total} = \frac{CN + CP}{Total}$

Review Time:

Time spent manually reviewing events for use-related relevance. Assumes:

- A reviewer reads all events labeled by model as Positive (i.e., Use-related)
- A reviewer (re)labels events as use-related or not at a rate of **75 events/hour** (based on internal analysis of our past KUPA).

Manual Model:

This model assumes one HFE reviewed every event to determine use-related relevance. It also assumes 100% accurate labeling, though this does not account for Human Error in high volume datasets.

Discussion:

Current Model Takeaways

- Our XGBoost Model can **significantly cut review time** on large datasets and **maintain excellent performance** metrics.

  →Reduced time can make development teams more likely to take on this effort **EARLY** in development

- The current model **needs only a small amount of new labeled data** to expand to new device types and maintain the same performance

  →Spending more time to label event data for new device can significantly improve performance

Future Work

- Expand training datasets to include more data from more diverse devices, so model can be more broadly applicable without further annotation from future users of the tool.

SCAN HERE FOR PDF VERSION or visit us at out website: EvolutionEngineering.us

You can also reach us at pdeshmukh@evolutionengineering.us