SAS® GLOBAL FORUM 2021

Paper 1058-2021

Developing a Deep Learning MedDRA encoder (MedDRA-DeepCoder) for Patient Narratives

Qais Y. Hatim[1], PhD; Sundaresh Sankaran[2], MS; Thomas W Sabo[2], MS; Emily McRae[2], PhD, Lauren Laufe[2], Robert Blanchard[2]
1 U.S. Food and Drug Administration, 2 SAS Institute Inc.

ABSTRACT

Patient narratives reported in clinical study reports (CSRs) provide clinical evidence of adverse events that occurred to a patient and help scientific reviewers during pharmacovigilance activities. The manual review of these narratives is a daunting task for safety reviewers as it is time consuming and resource intensive. How can we improve the efficiency of identifying safety signals from patient narratives? Can deep learning technology help to overcome the review challenges in an automated way?

This paper suggests an implementation to accurately categorize one adverse event term, "Serotonin Syndrome", as an example of what SAS® deep learning technology is capable of. We first generate sentence level embeddings from terms contained in patient narratives. Following this, we generate term embeddings within a SAS deep learning framework. Subsequently, we obtain a category decision on whether or not the narrative text relates to Serotonin syndrome as an output. Finally, we compare this method to other deep learning and machine learning methods, including the SAS supervised Boolean rule builder algorithm, which provide a layer of interpretability. We expect that use of a Deep Learning methodology in SAS shall improve the accuracy of the medical coding (example MedDRA coding) process for adverse events. It will also help in identifying drug-event pairs, drug interactions, and clinical evidence from narratives benefiting safety reviewers during the safety review process.

INTRODUCTION

The FDA Adverse Event Reporting System (FAERS) is a database that contains detailed free-text narratives regarding Adverse Events (AEs) occurring to a patient/subject. However, in order to identify these AEs and manually apply Medical Dictionary for Regulatory Activities (MedDRA) coded preferred terms on a significant volume of patient narratives, personnel experienced in medical codes are required. MedDRA Preferred Terms (PTs) are distinct descriptors (single medical concepts) for a symptom, sign, disease diagnosis, therapeutic indication, investigation, surgical or medical procedure, and medical social or family history characteristic[1]. As such, the automation of MedDRA PT coding from FAERS narrative texts to recognize Adverse Events, such as serotonin syndrome (SS), will improve Post Market safety reviews of FDA-regulated drugs.

We hypothesize that Deep Learning could promote the automatic process to accurately extract standard MedDRA preferred terms from FAERS narratives and classify narratives as associated to different conditions. In this paper, we aim to develop an automatic MedDRA encoder – named "MedDRA-DeepCoder" - for accurate encoding of free-text documents to standard MedDRA terms. To test this approach, we will develop models to classify narratives as whether they should be flagged with serotonin syndrome.

We will construct deep learning models in SAS based on the free-text description of FAERS adverse events. We will then fine-tune the process by training models with different network architectures. Finally, comparative analysis will be carried out to investigate the performance of the deep learning approaches, alongside of a machine learning approach that provides a layer of interpretability.

METHODS

We leverage SAS Viya® as the solution to execute and develop all the work described in this paper. We use this environment to define term embeddings and topic weights, train different deep learning models against these embeddings, and validate the models by scoring a validation set of FAERS data. We run these using SAS Cloud Analytic Services (CAS) actions. CAS enables us to invoke SAS in a cloud-based, run-time environment for data management and analytics that Viya provides2.

DATA PREPARATION

We collected the training dataset from FAERS by querying the FDA internal Empirica Signal tool and Molecular Health EFFECT systems. We used the Empirica Signal to generate two lists of mutually exclusive initial molecular targets for product active moieties associated with serotonin syndrome. With these lists, we queried the FDA EFFECT system, which returns drug safety narrative reports. The drugs associated with these results demonstrate a pharmacokinetic and/or pharmacodynamic relationship with CYP450 enzymes and historically reported “true” serotonin syndrome events. These were selected independent of age, sex, and ethnicity.

To better understand the impact of concomitant serotonin-active drug pairs on serotonin syndrome diagnoses, we created 6 individual cohort datasets. We first augmented each dataset with 2 new binary flag variables (List1 and List2), indicating which set of drug targets were involved in the adverse event. By defining the universe under consideration as those narratives involving drugs that fall either in List 1, List 2, or both, we made sure that the analysis of serotonin syndrome occurs on an appropriate dataset consisting of narratives that had some likelihood to lead to serotonin syndrome in the first place. Were we to have not followed this principle and considered the entire AE database, the heterogeneity of such a diverse database, and a reduced proportion of serotonin syndrome (our outcome of interest) would have significantly reduced our chances of identifying accurate indicators.

Following the creation of these descriptive flag variables, we appended all 6 cohorts into a single dataset, with 26,274 records, of which 28.5% were flagged as ‘SS’, designating the presence of serotonin syndrome from the narrative. To balance the proportion of events containing serotonin syndrome with those not, we applied oversampling. This gave us a balanced dataset of 14,976 observations, where there were 7488 samples of SS = 1 and SS = 0 each. Finally, we designated 60% of the data to a model training set, and 40% to a validation set. Figure 1 shows the distribution of the training and validation data.

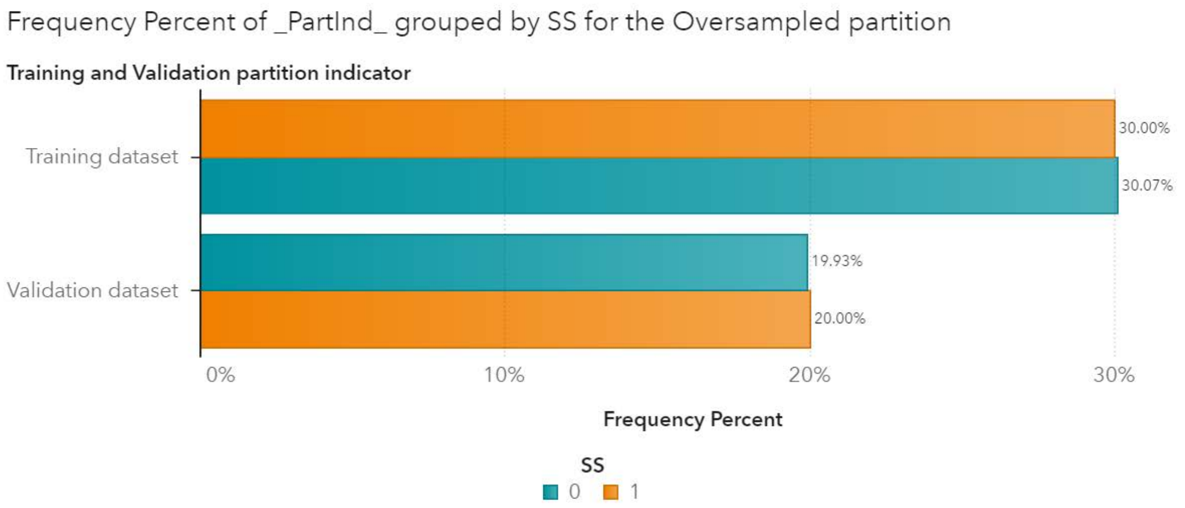


Figure 1: Graph indicating size by frequency percent of training and validation FAERS narrative data

DEEP LEARNING AND MODELING

After oversampling and partitioning the data as indicated above, we prepared for the exercise in modeling and deep learning. We selected several different approaches, to compare different algorithms with a sliding window approach alongside a document-level approach to deep learning. Simultaneously, we wanted to compare these results with a machine learning approach to determine if the payoff in the efficacy of deep learning was worth the computational overhead and investment in the development of the deep learning models. The five different methods we applied are outlined in Table 1.

Approach	Type	Description
TmCooccur	Deep Learning	Sliding window approach based on *tmCooccur* algorithm. Custom developed using training FAERS data. This uses a sentence context rather than sliding windows. A Recurrent Neural Network was trained using these term embeddings.
GloVe	Deep Learning	A Deep Learning approach using pre-trained embeddings generated using the GloVe algorithm. Standardized third-party embedding file. A Recurrent Neural Network was created using these embeddings.
Deep Neural Network on Topic Weights	Deep Learning	Document level approach based on weights generated from topic model and run through a convolutional neural network.
Rules-Based approach	Machine Learning	Document level approach which generated sets of Boolean rules combining terms and phrases which when present, designate an outcome.
TmCooccur Averaging	Deep Learning	Leveraged tables generated from the *tmCooccur* embeddings which contained co-occurring term pairs and averaged their 200 dimensions at a document level. Ran this through a deep

		learning model, similar to the Deep Neural Network on Topic Weights.

Table 2: Deep Learning and Machine Learning modeling approaches to serotonin syndrome classification

We will begin by describing term embedding approaches to deep learning, before detailing methods and intermediate steps involved in generating the term embeddings for SAS *tmCooccur* and GloVe. Term embeddings are a mapping of each term into a multi-dimensional space, which can place words and phrases that appear in a similar context near to each other[3]. This can be assessed at a document level, at a sentence level, or by using various sliding windows of 3 or more terms.

TmCooccur and GloVe Deep Learning Methods

Our first deep learning method involved the SAS CAS *tmCooccur* action to enable term embeddings on a document corpus similar to what is available in Word2Vec[4] and GloVe[5]. Word2Vec and GloVe train a model using sliding windows of words, such as 3-5 words immediately preceding and following terminology. First, we invoke a document parsing algorithm, the SAS CAS *tmMine* action, on the training dataset of narratives to compute an offset table. Next, we calculate term co-occurrence through the SAS CAS *tmCooccur* action as a pair-wise combination with every other term at the sentence level (rather than a 3-5 word sliding window), which generates an association column to designate how strongly the terms are connected. In the third step in the process we generate the term embeddings by applying the SAS CAS *tmSvd* action to the term-by-term matrix of associations calculated in the previous step. This essentially compresses the matrix down into a lower dimensional space of numerical interval values, organized by term rows and term columns, which we can feed into a deep learning model.

The second deep learning approach we applied involved GloVe term embeddings. This approach is very similar to the *tmCooccur* approach. The difference is that we leverage a third-party embeddings file – GloVe – instead of generating customized embeddings. As such, these embeddings are generated independent of the FAERS narratives. We downloaded pre-trained word embeddings for inclusion in network training[6]. We applied both a GloVe 100-dimension file and a GloVe 300-dimension file for this purpose.

For both the customized set of term embeddings from *tmCooccur* and the standardized set of embeddings from GloVe, the next step is to train a deep learning neural network model and subsequently validate the embeddings using the serotonin syndrome tagged data from FAERS, partitioned as indicated in the previous section. We used a Recurrent Neural Network for this exercise.

Neural networks attempt to mimic key aspects of a brain, in particular its ability to learn from experience[7]. Viewing a Neural Network as a hierarchy of many hidden layers, RNNs are structured in such a way that they perform the same task for every element in a sequence of data. A layer within an RNN network, when processing an element in a sequence, retains some information (context) about the preceding element and may use the same within processing as well. Sequential data could refer to both data occurring in text (a sequence of words) or temporal (time series) data. These qualities render RNNs specifically attractive and suitable for problems involving unstructured textual data[8].

We applied Gated Recurrent Unit (GRU) RNN model layers of depth. GRU models are perceived as being, on an average, smaller in size (due to lesser training parameters involved) compared to LSTMs (Long short-term memory, another model based on RNN). We tuned the models by applying hyperparameter tuning. We trained the deep learning models using the reserved training dataset and assess for misclassification rate on the validation dataset. Table 2 illustrates the high-level architecture of the models we applied to the tmCooccur embeddings and the GloVe 300-dimension term embeddings.

	tmCooccur model	GloVe 300-dimention model
Model Type	Recurrent Neural Network	Recurrent Neural Network
Number of Layers	4	7
Number of Input Layers	1	1
Number of Output Layers	1	1
Number of Convolutional Layers	0	0
Number of Pooling Layers	0	0
Number of Fully Connected Layers	0	0
Number of Recurrent Layers	2	5
Number of Weight Parameters	101632	250496
Number of Bias Parameters	386	962
Total Number of Model Parameters	102018	251458
Approximate Memory Cost for Training (MB)	2469	5542

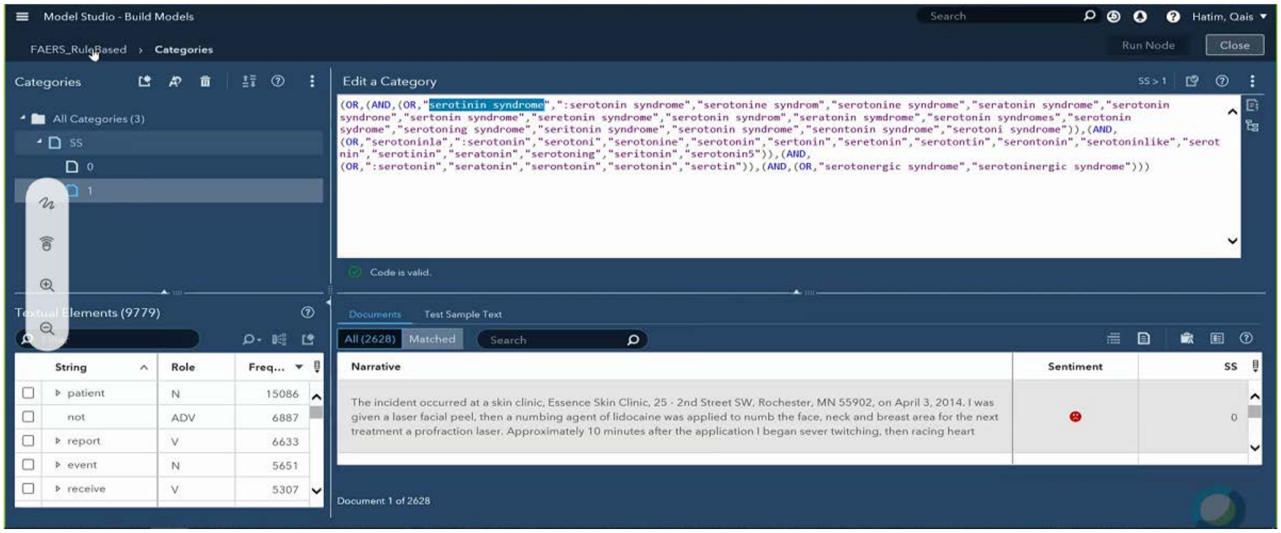
Table 2: High level model architecture for tmCooccur and GloVe 300-dimension approach

Deep Learning with Topic Weights

Prior research suggests that document level approaches can outperform the combination of term embeddings and RNN approach, but not extensively[3]. Therefore, for our third method and as a point of comparison to the sliding window/sentence level approaches of the *tmCooccur* and GloVe methods, we chose to generate topics using SAS Visual Text Analytics® and provided the topic weights for each training narrative from FAERS as input to a convolutional neural network deep learning model.

Machine Learning Boolean Rules Approach

For our fourth method, there is much we can do to reverse engineer the categorization model through SAS rule-building capabilities, so we examine this application in concert with term embeddings, RNNs and categorization. Here, we are specifically discussing using machine learning methods to automatedly generate sets of Boolean rules which map to outcome variables. To generate a set of rules against a labeled dataset, we utilize the SAS CAS *brTrain* action*.* Against these results, we leveraged stop lists in five iterations to remove rules as reviewed by an FDA expert to designate the generated combinations of terms and phrases in the narratives, particularly symptoms, that likely designate serotonin syndrome. The goal was to define a model which could evaluate narratives that had not been classified as serotonin syndrome but looked very much like serotonin syndrome. This is similar to our deep learning approach, only that it also provides a layer of interpretability.



```
(OR,(AND,(OR,"serotinin syndrome",":serotonin syndrome","serotonine syndrom","serotonine syndrome","seratonin syndrome","serotonin
syndrone","sertonin syndrome","seretonin syndrome","serotonin syndrom","seratonin symdrome","serotonin syndromes","serotonin
sydrome","serotoning syndrome","seritonin syndrome","serotonin syndrome","serontonin syndrome","serotoni syndrome")),(AND,
(OR,"serotoninla",":serotonin","serotoni","serotonine","serotonin","sertonin","seretonin","serotontin","serontonin","serotoninlike","serot
nin","serotinin","seratonin","serotoning","seritonin","serotonin5")),(AND,
(OR,":serotonin","seratonin","serontonin","serotonin","serotin")),(AND,(OR,"serotonergic syndrome","serotoninergic syndrome")))
```

	String	Role	Freq...
□	▸ patient	N	15086
□	not	ADV	6887
□	▸ report	V	6633
□	▸ event	N	5651
□	▸ receive	V	5307

Narrative	Sentiment	SS
The incident occurred at a skin clinic, Essence Skin Clinic, 25 - 2nd Street SW, Rochester, MN 55902, on April 3, 2014. I was given a laser facial peel, then a numbing agent of lidocaine was applied to numb the face, neck and breast area for the next treatment a profraction laser. Approximately 10 minutes after the application I began sever twitching, then racing heart	●	0

Display 1: Boolean rules generated from SAS Visual Text Analytics ML process on first pass to designate instances of serotonin syndrome in input data

TmCooccur Averaging Approach

The fifth and last method we applied was an averaging approach that leveraged tables from the *tmCooccur* method. We joined one table from the SAS CAS *tmCooccur* action where each row represented a set of co-occurring terms (~5M observations) with the original FAERS dataset to label each document projection ID (14976 observations). As each document was represented numerous times in the 5M observations, one for each set of co-occurring terms, we could take an average of the 200 dimensions identified per case ID. This method is consistent with practices followed when creating embeddings for documents through a singular vector decomposition. In both cases, there is an aggregate measure carried out on the individual embeddings to get the overall document representation. This gave us our original number of observations (14976) – one for each FAERS case. Again, each observation had 200 numerical dimensions. We subsequently ran a deep learning model against this dataset. Table 3 depicts the high-level architecture for this approach.

	tmCooccur averaging model
Model Type	Convolutional Neural Network
Number of Layers	14
Number of Input Layers	1
Number of Output Layers	1
Number of Convolutional Layers	0
Number of Pooling Layers	0
Number of Fully Connected Layers	11
Number of Concat Layers	1
Number of Weight Parameters	18900404
Number of Bias Parameters	11704
Total Number of Model Parameters	18912108
Approximate Memory Cost for Training (MB)	984

Table 3: Hi-level model architecture for tmCooccur averaging approach

RESULTS

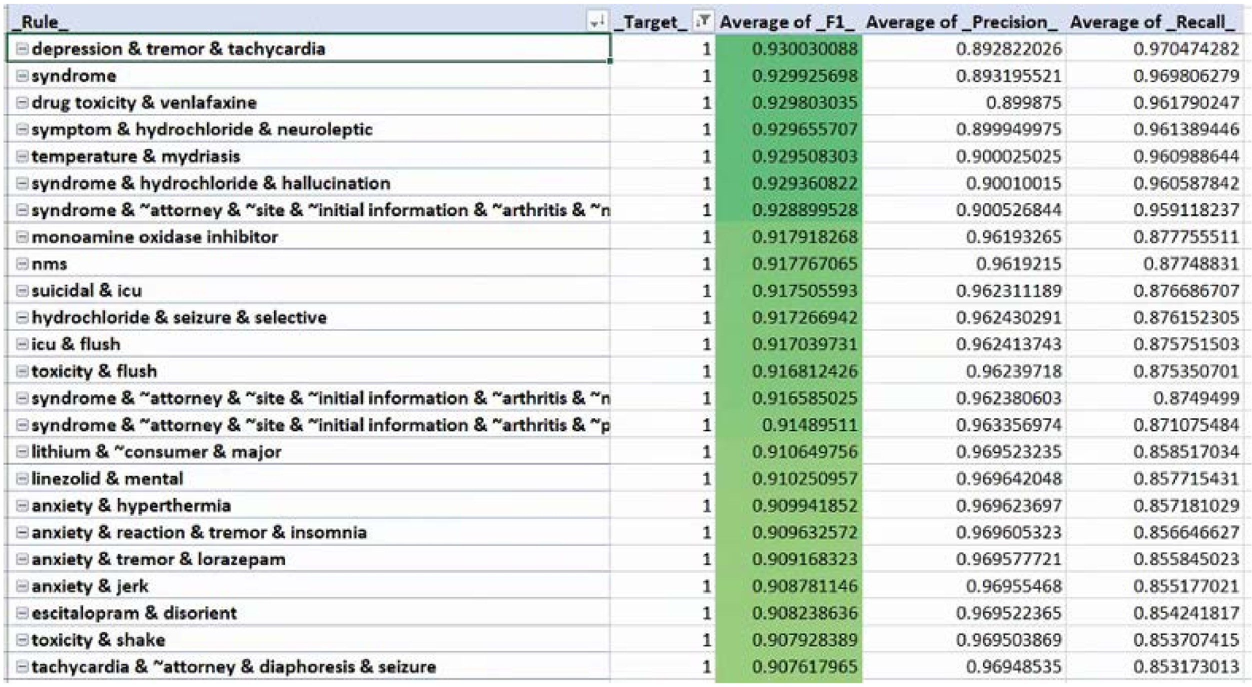
We generated and validated a model for each approach as designated above, using the same training and validation dataset. Table 4 presents the misclassification rates achieved for each approach.

Approach	Type	Misclassification Rate
TmCooccur	Deep Learning	**2.8461 %**
GloVe on 300 Dimensions	Deep Learning	**19.37%**
Deep Neural Network on Topic Weights	Deep Learning	**1.9%**
Rules-Based approach	Hybrid (Machine Learning Suggested Linguistics)	**5.6%**
TmCooccur Averaging Approach	Deep Learning	**1.115385 %**

Table 4: Modeling results by type and misclassification rate

Our misclassification rate for the *tmCooccur* approach is 2.8461%. This low misclassification rate is attributed to a systematic round of iterations which adjusted both model definition (adding new layers, adding specific GRU layers), model parameters (changing mini-batch sizes and learning rates) and utilizing the correct embeddings files (through the *tmCooccur-tmSvd* method). Our misclassification rate for the GloVe approach was 19.37% with the 300-dimension glove standard embedding file. We applied a learning rate of .001 and .0005 and found better results with the .001 learning rate. Our misclassification rate for the deep learning approach with topic weights was 1.9%.

Regarding the machine learning Boolean rules approach, our first iteration of a rules-based model after removing giveaway terms such as serotonin syndrome from consideration is 5.6%. An example set of these rules can be referenced in Table 5. In this table, the character '&' indicates a combination of terms, while '~' indicates that the term should not be present. We also employed additional rulesets after removing additional terms on recommendation from FDA subject matter experts, which identified additional patterns, and resulted in a higher misclassification rate.



Rule	_Target_	Average of _F1_	Average of _Precision_	Average of _Recall_
depression & tremor & tachycardia	1	0.930030088	0.892822026	0.970474282
syndrome	1	0.929925698	0.893195521	0.969806279
drug toxicity & venlafaxine	1	0.929803035	0.899875	0.961790247
symptom & hydrochloride & neuroleptic	1	0.929655707	0.899949975	0.961389446
temperature & mydriasis	1	0.929508303	0.900025025	0.960988644
syndrome & hydrochloride & hallucination	1	0.929360822	0.90010015	0.960587842
syndrome & ~attorney & ~site & ~initial information & ~arthritis & ~n	1	0.928899528	0.900526844	0.959118237
monoamine oxidase inhibitor	1	0.917918268	0.96193265	0.877755511
nms	1	0.917767065	0.9619215	0.87748831
suicidal & icu	1	0.917505593	0.962311189	0.876686707
hydrochloride & seizure & selective	1	0.917266942	0.962430291	0.876152305
icu & flush	1	0.917039731	0.962413743	0.875751503
toxicity & flush	1	0.916812426	0.96239718	0.875350701
syndrome & ~attorney & ~site & ~initial information & ~arthritis & ~n	1	0.916585025	0.962380603	0.8749499
syndrome & ~attorney & ~site & ~initial information & ~arthritis & ~p	1	0.91489511	0.963356974	0.871075484
lithium & ~consumer & major	1	0.910649756	0.969523235	0.858517034
linezolid & mental	1	0.910250957	0.969642048	0.857715431
anxiety & hyperthermia	1	0.909941852	0.969623697	0.857181029
anxiety & reaction & tremor & insomnia	1	0.909632572	0.969605323	0.856646627
anxiety & tremor & lorazepam	1	0.909168323	0.969577721	0.855845023
anxiety & jerk	1	0.908781146	0.96955468	0.855177021
escitalopram & disorient	1	0.908238636	0.969522365	0.854241817
toxicity & shake	1	0.907928389	0.969503869	0.853707415
tachycardia & ~attorney & diaphoresis & seizure	1	0.907617965	0.96948535	0.853173013

Table 5: Boolean rules generated from machine learning rules-based method after adding direct references to serotonin syndrome to a stop list

Our misclassification rate for the deep learning approach with topic weights was 1.115385% using an averaging approach. The dataset we put into the deep learning algorithm was very similar to what we used for topic weights (also created through an SVD method but projecting onto the entire document). However, our results here were based on the averaging of the specific term pairs per document.

CONCLUSION

We explored how deep learning models built upon term embeddings can be leveraged to assist classification tasks, namely, determining if a narrative is likely associated with serotonin syndrome. We explored these approaches alongside document-level deep learning approaches and a rules-based machine learning approach.

The high misclassification rate of the GloVe approach shows us that the general embeddings do not outperform term embeddings trained on FAERS narratives. This implies that more information and embedding data can enhance this process. A medically oriented dataset of standardized embeddings could improve results. Our document level approaches yielded the best misclassification rates. This suggests that there could either be a loss of information, or additional noise present in the term embeddings approaches that are not present in an approach that leverages topic weights, drawing this information from the document level. Our best misclassification rate was achieved from the fifth approach, which leveraged document level averaging. This gives us alternative methods in deep learning for refinement versus a typical machine learning single vector decomposition approach, where the embeddings are represented in another way as a term by document matrix.

Regarding the machine learning approach, these rules provide a layer of interpretability around the decision-making process of when to label an adverse event as serotonin syndrome. Example generated rules highlight key symptom terminology such as “anxiety” and “hyperthermia” or “depression” present with “tremors” and “tachycardia”. When these terms are present in narratives associated to the drug lists given earlier, they are highly indicative of serotonin syndrome.

There are several other verification tasks we can leverage in future work. We can explore how the term embeddings from FAERS data could be extended to other public health use cases, such as assessing the vaccine adverse event reporting system (VAERS) which contains information on unverified reports of adverse events following immunization with US-licensed vaccines.

Term embedding which drew from a wider variety of sources could be applied to a wider variety of tasks. Furthermore, the classification of documents is only one way to leverage deep learning. These embeddings could leverage to perform entity extraction tasks as well. One area we could contribute to is essentially creating a set of standardized embeddings for medical diseases and, in particular, adverse event assessment. This would require an extensive set of labeled data. As such, an area of continued interest is semi-automating the process of generating training data for the purposes of deep learning. Further work could explore capabilities of label spreading for expanding known instances of a target. We could also use SAS capabilities for autogenerating concept definitions given a set of known, related terms, such as symptoms or drugs, to identify and characterize the context in which these entities appear.

Once we create an embeddings file, as done in the above tasks, visualizing it in a way non-technical users can understand may be very important to more tightly embed this into the process. Developing easily accessible interfaces can be used to interpret the FAERS data and the connections embedded within it. Furthermore, such exercises will enable us to select parameters that should be more heavily weighted in deep learning efforts, which would enable model refinement. We would be interested in this and other exercises that can assist with fusing subject matter experts with deep learning in ways other than simply pushing more data at a model.

In this effort, we did not leverage the MedDRA dictionary directly for adverse events for a couple of reasons

1. We are dealing with Postmarket data, which was not standardized. Therefore, utilizing MedDRA would not be efficient.
2. We wanted to explore public sources of deep learning, or custom generated term embeddings in this exercise.

Leveraging MedDRA directly is an area that can be explored in future work. Leveraging the structure and defined terms from MedDRA would be powerful in Premarket data as it is more tightly tied with their submissions. Therefore, we could develop custom weighting algorithms for MedDRA terms in Premarket data or explore how the term embeddings generated from Premarket data map over to the MedDRA dictionaries to better classify adverse events. As such, we would use MedDRA to weight terms in addition to our proposed deep learning and rules-based ensemble approach discussed in this paper.

In future work, we will consider the application of other deep learning approaches, namely, BERT[9]. BERT combines the benefits of a pre-trained model on a standardized and very-wide corpus of data, such as we attempted with GloVe, with the tunability available from a smaller document corpus, such as what we have available for serotonin syndrome from FAERS. Furthermore, BERT has developed to account for polysemy, that is, assessing multiple meanings and multiple embeddings for word depending on the context. While there are generalized embeddings available for BERT, there are also BERT models developed, specifically for biomedical applications. BioBERT[10] is one such model that has been pre-trained on a large-scale biomedical corpus. Leveraging this as a starting point, tuned with data from the FAERS corpus on serotonin syndrome or some other adverse event classification could yield great results. A follow-on study could compare leveraging BERT and a BioBERT, then subsequently applying a rule-based machine learning model for a layer of interpretation. This transparency would enhance the utility of BERT in classification-based approaches to deep learning.

In conclusion, we envision an accurate, interpretable model based on machine learning and deep learning, with results surfaced via dashboards for analysis to interpret and interact. The application of these methods can shorten response cycles and improve assessments in determining adverse drug events, ultimately improving quality of life.

REFERENCES

1. MedDRA: Medical Dictionary for Regulatory Activities. Accessed December 8th, 2020. https://www.meddra.org/how-to-use/basics/hierarchy
2. SAS® Viya® 3.5 Administration: SAS® Cloud Analytic Services. SAS Help Center. 2019. Accessed December 3rd, 2020. https://go.documentation.sas.com/?cdcId=calcdc&cdcVersion=3.5&docsetId=calserverscas&docsetTarget=titlepage.htm&locale=en
3. Cox, J., Albright, R. 2019. “The Wondrous New tmCooccur SAS® Cloud Analytic Services (CAS) Action and Some of Its Many Uses.” *Proceedings of the SAS Global Forum 2019 Conference. Cary NC: SAS Institute Inc.*
4. Mikolov Tomas, Corrado Greg, Chen Kai, Dean Jeffrey. “Efficient Estimation of Word Representations in Vector Space."CoRR, abs:1301.3781.
5. Pennington Jeffrey, Socher Richard, Manning Christopher D. "GloVe: Global Vectors for Word Representation." EMNLP. Vol. 14.2014.
6. GloVe: Global Vectors for Word Representations. Stanford University. August 2014. Accessed December 3rd, 2020. https://nlp.stanford.edu/projects/glove/
7. Hatim Qais, Rosario Lilliam, Almario EN, Worthy Kendra, Sabo Tom, McRae Emily, et al. 2018. “Modeling and Text Analysis to Empower FAERS Adverse Event Assessment.”, *PhUSE Connect 2018*.
8. Le, L., Xie, Y. 2019. “Deep Learning with SAS® and Python: A Comparative Study” *Proceedings of the SAS Global Forum 2019 Conference. Cary NC: SAS Institute Inc.*
9. Devlin, Jacob, Chang Ming-Wei, Lee Kenton, Toutanova Kristina. 2018. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.” CoRR, abs:1810.04805.
10. Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, et al. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics, Volume 36, Issue 4, 15 February 2020.

ACKNOWLEDGMENTS

This work would not have been possible without a host of professional support from FDA and SAS. Experts in artificial intelligence and text analytics are hard to come by, and we were fortunate to have several contribute to this project. Thanks to Doug Cairns for reviewing our deep learning models and helping to architect and tune them. Thanks to Sam Leeman-Munk, Xiangxiang Meng, Jim Cox, David Styles, Ari Zitin, Terry Woodfield, and Busi Akanbi for reviewing our approaches, helping to interpret our results, and providing recommendations for future work. We would also like to thank Doug Bell for his expertise as a technical architect in navigating and helping to provision an environment to support this deep learning work.

CONTACT INFORMATION

Dr. Qais Y. Hatim, Computer and Data Scientist
U.S. Food and Drug Administration (FDA)

O:(301)796-7932
C:(646)596-5035
Qais.hatim@fda.hhs.gov

Tom Sabo, Principal Solutions Architect
SAS Federal LLC
+1` (703) 310-5717
tom.sabo@sas.com
https://www.linkedin.com/in/tomsabo/

Disclaimer: This is not a formal dissemination of information by FDA and does not represent Agency position of policy.