Fall Detection in EHR using Word Embeddings and Deep Learning

Henrique D. P. dos Santos*, Amanda P. Silva‡, Maria Carolina O. Maciel‡,
Haline Maria V. Burin‡, Janete S. Urbanetto‡ and Renata Vieira*
Pontifical Catholic University of Rio Grande do Sul,
*School of Technology and ‡School of Health Sciences
Email: {henrique.santos.003, amanda.pestana, maria.maciel.001, haline.burin}@acad.pucrs.br,
{jurbanetto, renata.vieira}@pucrs.br

***Abstract*—Electronic health records (EHR) are an important source of information to detect adverse events in patients. In-hospital fall incidents represent the largest category of adverse event reports. The detection of such incidents leads to better understanding of the event and improves the quality of patient health care. In this work, we evaluate several language models with state-of-the art recurrent neural networks (RNN) to detect fall incidents in progress notes. Our experiments show that the deep-learning approach outperforms previous works in the task of detecting fall events. Vector representation of words in the biomedical domain was able to detect falls with an F-Measure of 90%. Additionally, we made available an annotated dataset with 1,078 de-identified progress notes for replication purposes.**

***Index Terms*—Fall Detection, Electronic Health Records, Biomedical Language Processing, Word Embeddings, Deep Learning**

I. Introduction

The adoption of Electronic Health Records in hospital environments brings many benefits for patient safety and health care quality [1]. This amount of data could be used as the source of many clinical decision support systems. Recent studies show the advantage of using EHR data to perform comorbidity index [2], potential prescription errors [3], and several risk models [4].

However, hospitals clinical data is more widely used for diagnoses and treatment predictions than for the detection of adverse events [5]. Falls are critical adverse events that occur in the hospital environment. Within hospitals and nursing homes, falls constitute the largest category of adverse event reports. Approximately 30% of inpatient falls result in injury, with 4% to 6% resulting in serious injury [6]. The starting point for falls prevention programmes should always be a critical review of such evidence [7]. An automated system to detect fall could help in the adverse events smart screening.

This work presents a new approach in terms of fall detection in clinical notes. We use a state-of-the-art natural language processing neural network to detect fall events from text information present in EHR. Our models are able to detect falls with an F-Measure of 90% in a dataset extract from a tertiary research hospital.

This work was partially supported by CAPES (Coordenao de Aperfeioamento de Pessoal de Nvel Superior) Foundation (Brazil), UFRGS (Federal University of Rio Grande do Sul), and Google Latin America Research Awards.

The main contributions of our work are the following:

- New approach in fall event detection from text in Electronic Health Records;
- Annotated dataset with 1,078 progress notes, with the presence of fall events and their structured description for replication purposes;
- Results of an evaluation of several language models (two general- and one biomedical-domain model generated with two shallow word embeddings algorithms).

The rest of this paper is organized as follows: Section II presents previous works on predicting comorbidities through clinical notes. Section III describes the dataset used and the experiment setup, followed by the results in Section IV. In addition, we perform a qualitative analysis of the results in Section V. Finally, in Section VI we summarize our contributions and present further research directions.

II. Related Work

Automated approaches for fall detection in EHR have been studied since patient records became digital. One of the first works using machine learning methods to detect falls [8] used an unsupervised approach to extract term importance weightings and supervised learning to classify fall events in progress notes.

In the last few years, more studies have been focusing on the detection of fall events in clinical texts. Two of them developed syntactic rules to search events based on queries [9], [10]. This cannot be directly applied to other languages. Other studies using classical machine learning techniques adopted annotated datasets to train their models. Support Vector Machine [11]–[13] is the main algorithm used for fall detection, followed by Random Forest [14]. All studies above used text information: clinical notes, progress notes, incident reports, image orders, and radiology reports.

Other studies have already used word embeddings and deep learning to predict adverse events in text information concerning health. Some studies for adverse drug events (ADE) used recurrent architecture to detect events in EHR [15] and attention neural networks to highlight important words related to these events [16]. Deep learning was also applied to identify harm events in patient care [17] and biomedical word sense disambiguation [18].

2471-7819/19/$31.00 ©2019 IEEE
DOI 10.1109/BIBE.2019.00054

To the best of our knowledge, there are no previous studies addressing fall event detection from text using word embeddings or deep learning. In the next section, we cover the dataset, the neural network and the language models (word embeddings) used in the experiments.

III. Materials and Methods

We designed the experiments to evaluate several automated approaches to detect fall events in progress notes. In this Section, we cover the dataset we used, data preparation, and the methods we evaluated.

A. Data Source and Preparation

A retrospective cohort study was developed in a large public tertiary hospital in Porto Alegre, in southern Brazil. The population consisted of 1,694 patients who had a fall in the years 2012 to 2017; this generated 1,971 voluntary incident reports and 2,698 progress notes in the patients' charts. The sample calculation considered an estimated percentage of 10% of falls, a sampling error of 2.5%, and a statistical significance of 5%, which indicated a minimum sample of 433 reports. Therefore, 367 patients, 441 incident reports, and 1,078 progress notes with records of possible fall incidents were included in this study. Progress notes were collected based on the date of the incident reports and the patients numbers. Each incident report had on average 2.4 progress notes referring to the same patient and the same date.

The following steps were performed to prepare the dataset to train the machine learning models:

- Selection: identifying all inpatient with at least one reported fall incident and their progress notes;
- De-identification: de-identifying this data to ensure patient anonymity;
- Annotation: creating a "gold standard" with the charts reviewed by nursing students.

Ethical approval to use the hospital dataset in this research was granted by the Research Ethics Committee of the Hospital Group under the number 71571717.7.0000.5530.

B. Fall Annotation Process

The data collection of the incident reports and data annotation of progress notes in the WebAnno system [19] lasted four months, being carried out through the careful reading by three different nurse students, with double checking. In cases of incongruities or doubts, notes were taken in a spreadsheet and later discussed in meetings of the research group.

Each word or phrase was annotated with several definitions related to the fall, according to the WHO Technical Report about Patient Safety [20]. Some of the annotated concepts are: procedure after fall; medical assessment; damage level (none, low, medium, high, death); damage type (physical, psychological, social). The annotated dataset resulted in 723 (68%) progress notes without fall incidents and 355 (32%) notes with fall-related incidents annotated by nursing students. In our experiments, we designed the task as a classification problem and used the fall or non-fall related note.

The study included 367 patients with a total of 441 fall incident reports and 1,078 progress notes on the day of the fall, with a median of two (2) progress notes per report (a minimum of 1 and a maximum of 12). Of these, although all suffered a fall, 342 (32.97%) developments did not contain the progress notes of the incident. Table I shows the distribution of in-hospital fall incidents among the patients.

TABLE I
Fall per Patient in Annotated Dataset

# of Patients	% of Total	# of Falls
316	87.0%	1fall
36	10.1%	2 falls
11	3.0%	3 falls
1	0.3%	4 falls
2	0.5%	5 falls
1	0.3%	6 falls

C. Language Models

Word vector representations (word embeddings) bring new perspective for Natural Language Processing. This approach outperforms traditional rule-based or machine learning methods [21]. To evaluate word embeddings, we used four model architectures using three data sources that were combined to build 12 language models for our experiments. This approach focuses on evaluating biomedical-domain and general-domain language models in the task of fall detection in health records.

The strategies of the algorithms to compute language models and model architectures are listed below:

- Word2Vec: Word vectors are a way of mapping words in a numerical space, called Word2Vec [22]. A latent syntactic/semantic vector for each word is induced from a large unlabeled corpus.
- FastText: It is also a word vector representation based on the skip-gram model, where each word is represented as a bag of character n-grams. It allows to compute word representations for words that did not appear in the training data [23].
- CBOW: Continuous Bag-of-Words Model uses a continuous distributed representation of the context; the order of words in the history does not influence the projection [22].
- SKIP: Continuous Skip-gram Model is similar to CBOW, but instead of predicting the current word based on the context, it tries to maximize the classification of a word based on another word in the same sentence [22].

Both Word2Vec and FastText are context-free representations of the words. The following list presents the data source used to build the language models:

- Wikipedia: A simple language model build with Portuguese articles from Wikipedia-PTs dump of May 2019. This corpus has a total of 250 million tokens. The model was trained with 300 dimensions per word and a minimum word count of 10.

- NILC: These are pre-computed language models that feature vectors generated from a large corpus of Brazilian Portuguese and European Portuguese, from varied sources and genres. Seventeen different corpuses were used, totaling 1.3 billion tokens [24].
- EHR-Notes: We used 24 million sentences with 603 million tokens from the hospital progress notes extracted from electronic health records. The generated model has 300 dimensions per word and contains words with a minimum of 100 occurrences. This model resulted in 79 thousand biomedical word vectors used as a semantic model in the neural network below.

D. Recurrent Neural Network

Deep learning algorithms are extensively used in biomedical language processing tasks [25]. Neural network algorithms are often associated with word vector representation. In our experiments, we used a deep learning algorithm for text classification: word embedding representations over a recurrent neural network (RNN) called LSTM (Long Short-Term Memory Network). RNNs are modifications of feed-forward neural networks with recurrent connections. In our experiments, we used the FLAIR implementation: an open-source framework for state-of-the-art NLP [26].

E. Evaluation

For each classification algorithm, we ran a cross validation with five stratified folds. The folds were made by preserving the proportion of samples for each class: fall and non-fall notes. For every iteration, four folds were used in the training stage, and one fold was used for model evaluation. The mean for all validations was used as the algorithm score. We chose the F-Measure as the main metric to evaluate the quality of the models. F-Measure corresponds to the harmonic mean between precision and recall.

F. Baseline

Classical machine learning models are used as the baseline models. We selected the main algorithms used for fall event detection in text mining: SVM and Random Forest with TF-IDF word weighting. The machine learning methods have no ability to process word vectors as a feature of the instances. We used Scikit-learn implementation of such algorithms [27].

IV. Results

We used all 12 language models with the LSTM neural network to train over progress notes with fall events. Table II shows the overall results of our experiments.

Overall, in our experiments, all deep learning models outperform classical machine learning methods. Word embeddings themselves add great value to automated word understanding and disambiguation. However, the feature extraction capabilities of LSTM layers are able to select the finest sequence of words that predict the fall outcome. In some cases, the word "fall" does not represent a fall incident, e.g. "blood pressure fall", "patient did not fall." Classical machine learning unigram features are not able to distinguish these cases.

TABLE II
F-Measure of each Language Model

	WV-CBOW	**FT-SKIP**	**WV-SKIP**	**FT-SKIP**
Wikipedia	0.88 ± 0.14	0.87 ± 0.11	0.77 ± 0.05	0.81 ± 0.09
NILC	0.77 ± 0.06	0.89 ± 0.13	0.79 ± 0.06	0.77 ± 0.06
EHR-Notes	0.88 ± 0.14	$\mathbf{0.90 \pm 0.13}$	0.82 ± 0.08	0.85 ± 0.10
R. Forest	0.73 ± 0.03			
SVM	0.60 ± 0.05			

WV: Word2Vec, FT: FastText, CBOW: Continuous Bag-Of-Words, SKIP: Continuous Skip-Gram

The language model that best detected fall events in our experiments was the biomedical model (EHR-Note) computed with the FastText approach and Skip-gram strategy. FastTexts ability to represent words that did not appear in the training data improved the model precision and recall harmonic mean (F-Measure).

Besides the result, RNN requires some overhead: word embeddings need a vast amount of text to train the word vector representation, and the training time of RNN is exponential, higher than the machine learning methods.

The best classical machine learning algorithm was Random Forest (RF), an ensemble of decision trees with an F-Measure of 0.73 using unigram features. Random Forest is a good alternative for fall detection when there is less amount of text to train the language model.

V. Discussion and Limitations

Results of this study point to the validity and feasibility of the classification method to detect fall events in clinical notes. We were able to detect fall incidents with minimal error using natural language processing (NLP) features, without the need for specialized software to process the texts in this dataset.

Biomedical-domain word embeddings (EHR-Notes) prove to be the best model language for fall detection. Despite this result, NILC general-domain could also be a proper alternative in datasets with lower clinical note density (not enough text to train word vectors).

Our experiments focused on the ability of the proposed models to detect fall incidents among progress notes extracted from patients with fall reports. However, to apply such technique in a real scenario, the model should be trained over a natural imbalanced dataset. Our dataset detected 32% of falls in progress notes, when generally falls among hospital inpatients range from 2.3 to 7 falls per 1,000 patientdays [6]. Further work should annotate a dataset that is more similar to the natural distribution.

VI. Conclusion

We were able to detect fall events automatically from clinical notes using deep learning methods and textual features with 90% of F-Measure. This approach could be replicated at other hospitals with the same type of labeled dataset. The Recurrent Neural Network with Word Embedding outperforms the other methods, but Random Forest with Unigrams could

also be a suitable alternative in datasets with less labeled clinical notes.

All the content of the work (algorithm, sample dataset, language models, and experiments) is available at the project's GitHub Page[1] in order to be easily replicated. The sample dataset has no patient data; it contains the dataset, with the information regarding fall events and their structured description (damage level, location, fall type, etc).

Several research groups have been developing language representations to perform many natural language processing tasks. Further work should evaluate other model languages like BERT [28], FLAIR [29], and GPT-2 [30]. Different from Word2Vec and FastText, these strategies implement context-aware language models with backward and forward capabilities improving sentence understanding.

ACKNOWLEDGEMENT

We thank the prestige cooperation of Ana Helena D. P. S. Ulbrich and Graziella G. Baiocco providing the incident report dataset from the hospital.

REFERENCES

[1] M. B. Buntin, M. F. Burke, M. C. Hoaglin, and D. Blumenthal, "The benefits of health information technology: a review of the recent literature shows predominantly positive results," *Health affairs*, vol. 30, no. 3, pp. 464–471, 2011.

[2] H. D. P. d. Santos, A. H. D. P. S. Ulbrich, V. Woloszyn, and R. Vieira, "An initial investigation of the charlson comorbidity index regression based on clinical notes," in *2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS)*, June 2018, pp. 6–11.

[3] ——, "Ddc-outlier: Preventing medication errors using unsupervised learning," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 2, pp. 874–881, March 2019.

[4] B. Goldstein, A. Navar, M. Pencina, and J. Ioannidis, "Opportunities and challenges in developing risk prediction models with electronic health records data: A systematic review," *Journal of the American Medical Informatics Association*, vol. 24, no. 1, pp. 198–208, 2017.

[5] P. B. Jensen, L. J. Jensen, and S. Brunak, "Mining electronic health records: towards better research applications and clinical care," *Nature reviews. Genetics*, vol. 13, no. 6, p. 395, 2012.

[6] E. B. Hitcho, M. J. Krauss, S. Birge, W. Claiborne Dunagan, I. Fischer, S. Johnson, P. A. Nast, E. Costantinou, and V. J. Fraser, "Characteristics and circumstances of falls in a hospital setting: a prospective analysis," *Journal of general internal medicine*, vol. 19, no. 7, pp. 732–739, 2004.

[7] D. Oliver, "Preventing falls and fall injuries in hospital: a major risk management challenge," *Clinical Risk*, vol. 13, no. 5, pp. 173–178, 2007.

[8] M. Tremblay, D. Berndt, S. Luther, P. Foulis, and D. French, "Identifying fall-related injuries: Text mining the electronic medical record," *Information Technology and Management*, vol. 10, no. 4, pp. 253–265, 2009.

[9] S.-I. Toyabe, "Detecting inpatient falls by using natural language processing of electronic medical records," *BMC Health Services Research*, vol. 12, no. 1, 2012.

[10] B. Shiner, J. Neily, P. Mills, and B. Watts, "Identification of inpatient falls using automated review of text-based medical records," *Journal of Patient Safety*, 2016.

[11] J. McCart, D. Berndt, J. Jarman, D. Finch, and S. Luther, "Finding falls in ambulatory care clinical documents using statistical text mining," *Journal of the American Medical Informatics Association*, vol. 20, no. 5, pp. 906–914, 2013.

[12] S. Luther, J. McCart, D. Berndt, B. Hahm, D. Finch, J. Jarman, P. Foulis, W. Lapcevic, R. Campbell, R. Shorr, K. Valencia, and G. Powell-Cope, "Improving identification of fall-related injuries in ambulatory care using statistical text mining," *American Journal of Public Health*, vol. 105, no. 6, pp. 1168–1173, 2015.

[1]https://github.com/nlp-pucrs/fall-detection

[13] J. Bates, S. Fodeh, C. Brandt, and J. Womack, "Classification of radiology reports for falls in an hiv study cohort," *Journal of the American Medical Informatics Association*, vol. 23, no. e1, pp. e113–e117, 2016.

[14] M. Topaz, L. Murga, K. Gaddis, M. McDonald, O. Bar-Bachar, Y. Goldberg, and K. Bowles, "Mining fall-related information in clinical notes: Comparison of rule-based and novel word embedding-based machine learning approaches," *Journal of Biomedical Informatics*, vol. 90, 2019.

[15] A. N. Jagannatha and H. Yu, "Bidirectional rnn for medical event detection in electronic health records," in *Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting*, vol. 2016. NIH Public Access, 2016, p. 473.

[16] T. Huynh, Y. He, A. Willis, and S. Rueger, "Adverse drug reaction classification with deep neural networks," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 877–887.

[17] A. Cohan, A. Fong, R. M. Ratwani, and N. Goharian, "Identifying harm events in clinical care through medical narratives," in *Proceedings of the 8th ACM international conference on bioinformatics, computational biology, and health informatics*. ACM, 2017, pp. 52–59.

[18] A. Sabbir, A. Jimeno-Yepes, and R. Kavuluru, "Knowledge-based biomedical word sense disambiguation with neural concept embeddings," in *2017 IEEE 17th International Conference on Bioinformatics and Bioengineering (BIBE)*, Oct 2017, pp. 163–170.

[19] S. M. Yimam, I. Gurevych, R. E. de Castilho, and C. Biemann, "Webanno: A flexible, web-based and visually supported system for distributed annotations," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2013, pp. 1–6.

[20] W. H. Organization *et al.*, "The conceptual framework for the international classification for patient safety," *World Health Organization*, vol. 2009, pp. 1–149, 2009.

[21] Y. Li and T. Yang, "Word embedding for understanding natural language: A survey," in *Guide to Big Data Applications*. Springer, 2018, pp. 83–104.

[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[23] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[24] N. Hartmann, E. Fonseca, C. Shulby, M. Treviso, J. Silva, and S. Aluísio, "Portuguese word embeddings: Evaluating on word analogies and natural language tasks," in *Proceedings of the 11th Brazilian Symposium in Information and Human Language Technology*, 2017, pp. 122–131.

[25] Z. Jiang, L. Li, D. Huang, and L. Jin, "Training word embeddings for deep learning in biomedical text mining tasks," in *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*. IEEE, 2015, pp. 625–628.

[26] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf, "Flair: An easy-to-use framework for state-of-the-art nlp," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, 2019, pp. 54–59.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[29] A. Akbik, T. Bergmann, and R. Vollgraf, "Pooled contextualized embeddings for named entity recognition," in *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2019, p. 724728.

[30] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, 2019.