pdf

Fall Detection in EHR using Word Embeddings and Deep Learning

2022. 03. 04.

이동건

인하대학교 정보통신공학과

E-mail: time@inha.edu

GitHub: https://github.com/oneonlee

Contents

I. INTRODUCTION

contributions

- Use a state-of-the-art natural language processing neural network to detect fall events from text information present in EHR.
- Detect falls with an F-Measure of 90% in a dataset extract from a tertiary research hospital.

contributions

- 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).

II. RELATED WORK

previous works on predicting comorbidities through clinical notes

previous works

- unsupervised approach to extract term importance weightings

- supervised learning to classify fall events in progress notes

previous works

- syntactic rules to search events based on queries[9],[10]
  - cannot be directly applied to other languages

- Support Vector Machine [11]– [13] is the main algorithm used for fall detection, followed by Random Forest [14].

previous works using word embeddings or DL

- Other studies have already used word embeddings and deep learning to predict adverse events in text information concerning health.

- no previous studies addressing fall event detection from text using word embeddings or deep learning.

III. MATERIALS AND METHODS

A. Data Source and Preparation
B. Fall Annotation Process
C. Language Models
D. Recurrent Neural Network, RNN
E. Evaluation
F. Baseline

A. Data Source and Preparation

- The following steps were performed to prepare the dataset to train the machine learning models:
  - Selection: identifying all inpatient with at least one re-ported 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.
    최적 표준

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.

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 – Algorithms

- 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
  - Also a word vector representation based on the skip-gram model, where each word is represented as a bag of character n-grams.
  - 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
  - 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].

C. Language Models

- Both Word2Vec and FastText are context-free representations of the words.

C. Language Models – 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.
- 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, RNN

- We used a deep learning algorithm for text classification:
  - word embedding representations over a RNN called LSTM (Long Short-Term Memory Network).

- RNNs are modifications of feed-forward neural networks with recurrent connections.

- We used the FLAIR implementation: an open-source framework for state-of-the-art NLP [26].

E. Evaluation

- 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.

- 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.
  - used Scikit-learn implementation of such algorithms [27].

IV. RESULTS

results

- 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.

results

- The best detected fall events was the biomedical model (EHR-Note) computed with the FastText approach and Skip-gram strategy.

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

results

- 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

discussion

- Our experiments focused on **the ability of the proposed models to detect fall incidents** among progress notes extracted from patients with fall reports.
- Biomedical-domain word embeddings (EHR-Notes) prove to be the best model language for fall detection.

limitations

- However, to apply such technique in a real scenario,
  the model should be trained over a natural imbalanced dataset.

VI. CONCLUSION

conclusion

- We were able to detect fall events automatically from clinical notes using deep learning methods and textual features with 90% of F-Measure.
- The RNN 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.

conclusion

- 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.