Multi-task learning for interpretable cause-of-death classification using key phrase prediction

Serena Jeblee
Dept of Computer Science
University of Toronto
Toronto, Ontario, Canada
sjeblee@cs.toronto.edu

Mireille Gomes
St. Michael's Hospital
Toronto, Ontario, Canada
mireille.m.gomes@gmail.com

Graeme Hirst
Dept of Computer Science
University of Toronto
Toronto, Ontario, Canada
gh@cs.toronto.edu

Abstract

We introduce a multi-task learning model for cause-of-death classification of verbal autopsy narratives that jointly learns to output interpretable key phrases. Adding these key phrases outperforms the baseline model and topic modeling features.

1 Introduction

Verbal autopsies (VAs) are written records of the events leading up to a person's death, typically in situations where there was no physical autopsy and the cause of death (CoD) was not determined by a physician. As per World Health Organization recommendations, most VAs contain structured information from answers to a questionnaire, and may also contain a free-text narrative that captures additional information, such as the time and sequence of the subject's symptoms and details of their medical history (Nichols et al., 2018). VAs are used in countries such as India to gain a better idea of the most prevalent causes of death, which are not accurately captured by only the small number of well-documented deaths that occur in health facilities.

Typically, VAs are collected by non-medical surveyors who record the information on a form that is later reviewed by physicians who assign the record an International Classification of Diseases (ICD-10) code (World Health Organization, 2008). Automating some of this coding process would reduce the time and costs of VA surveys.

Previous work has shown that machine learning methods can be useful for medical text classification. However, many models do not provide interpretable explanations for their output, which are crucial in health care.

Multi-task learning methods use a shared architecture to learn several classification tasks, which has been shown to improve performance especially when the tasks are closely related. In this work we aim to use a multi-task learning model to classify VA narratives according to CoD and simultaneously provide automatically determined key phrases in order to increase the interpretability of the model.

2 Related work

Several automated methods for coding VAs are currently in use, including InterVA (Byass et al., 2012), InSilicoVA (McCormick et al., 2016), and the Tariff Method (Serina et al., 2015). However, these methods are largely based on the structured data (which varies depending on the particular VA survey form used) and on physician-curated conditional probabilities of symptoms and diseases, which are time-consuming to collect. The performance of these methods is typically less than .60 precision for 15 to 30 CoD categories (Desai et al., 2014).

Miasnikof et al. (2015) used a naïve Bayes classifier with structured data and achieved comparable or better results than the expert-driven models. Danso et al. (2013) used linguistic features to classify VA narratives of neonatal deaths into 16 CoD categories with a support vector machine (SVM), achieving .406 recall.

TextRank (Mihalcea and Tarau, 2004) is a popular method that uses document graphs to extract key phrases. However, unsupervised models such as TextRank can extract text only from the document itself, in which the physician-generated key phrases that we use in this work might or might not be explicitly present. Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is a topic modeling framework that is often used for text classification. We will compare our key phrase clusters to LDA topics learned from the same narrative data.

Proceedings of the BioNLP 2018 workshop, pages 12–17
Melbourne, Australia, July 19, 2018. ©2018 Association for Computational Linguistics

3 Data

Our dataset consists of 12,045 records of adult deaths from the Million Death Study (MDS) (Westly, 2013; Aleksandrowicz et al., 2014; Gomes et al., 2017), which is a program to collect and code VAs from India. In the MDS coding process, two physicians separately assign an ICD-10 code to each record and disagreements are resolved by a third physician if necessary. Because there are hundreds of possible ICD-10 codes and our dataset is fairly small, the codes are grouped into 18 CoD categories, which are broader groupings of the WHO 2012 VA categories (World Health Organization, 2012).

The narratives, written by non-medical surveyors, range from a couple of sentences to a few paragraphs and describe the person's medical history and symptoms before death. In addition to the free-text narratives, the VA records from the MDS also contain key phrases assigned by the coding physicians. By highlighting important symptoms and events, these phrases are meant to explain the code assigned. They may be taken directly from the narrative or written in by the physician.

We represent the narrative text and key phrases with 100-dimensional word embeddings trained with the word2vec CBOW algorithm[1], which learns vector space representations for words based on their context (Mikolov et al., 2013). The key phrase representation for clustering is the average of the embeddings of each word in the phrase. The narrative representation is a matrix containing the embeddings for each word in order, padded with zero vectors to a maximum length of 200 words.

Because the dataset is rather small for training word2vec, we include Indian English text from the International Corpus of English[2] and 1.7M posts from MedHelp[3], an online medical advice forum that contains informal health-related language.

The text of both the narratives and the key phrases is lowercased, punctuation is removed, and spelling is corrected using PyEnchant's English dictionary (Kelly, 2015) and a 5-gram language model for scoring candidate replacements, using KenLM (Heafield et al., 2013). After preprocessing we remove duplicate key phrases.

4 Model

The model used for both key phrase cluster prediction and CoD classification is a neural network that contains a gated recurrent unit layer (GRU) (Cho et al., 2014) with 0.1 dropout followed by a convolutional layer (CNN) with filters of size $1 \times d$ through $5 \times d$ where d is the word embedding size (100), followed by a max-pooling layer. The output of the pooling layer is then used as input to a dense softmax layer that outputs the classification. The models are implemented in Python using Keras (Chollet, 2015), with the Theano backend (Theano Development Team, 2016).

For CoD classification, the prediction layer outputs the probabilities over the 18 CoD categories, and we choose the one with the highest probability. For key phrase prediction, it outputs the probabilities over the key phrase clusters, and we take each cluster as a label if it has a probability of 0.1 or higher (since there can be any number of key phrases per record). A higher cutoff will result in slightly higher precision but lower recall. The loss functions are categorical cross-entropy for CoD classification and mean squared error for key phrase cluster prediction.

The multi-task learning model consists of a shared GRU/CNN model that generates a vector representation that is then used by two parallel prediction layers, one for the CoD category and one for the key phrase clusters. The key phrase loss function has a weight of 0.1 to emphasize the CoD coding task. All three of these models use only the narrative word embedding matrix as input.

5 Key phrase clustering

5.1 Unsupervised clustering

The key phrases from the training data are grouped into 100 clusters using the k-means algorithm with Euclidean distance from scikit-learn (Pedregosa et al., 2011).

We need a sufficient number of clusters to capture specific symptoms and event, but not so many that we cannot predict them accurately. In our case, we want to favor precision over recall because we would rather generate fewer, more-correct key phrases than more phrases that are less accurate. We chose 100 clusters based on early experiments to maximize precision and the number of clusters.

[1] We used a context window of 5, min count of 1 (due to the small dataset), and no negative sampling.

[2] http://ice-corpora.net/ice/avail.htm

[3] http://www.medhelp.org

Label	Key phrases in cluster
cough	cough, cough with sputum, cough with phlegm, had sputum cough, ...
rigours	fear, sudden chest pain one day and died in short while, h/o headache, epileptic, ...
h/o chest pain	sudden chest pain, occasional chest pain, sudden pain in middle of chest, ...
breathing difficulty	difficulty in eating, difficulty in urination, ...

Table 1: Examples of key phrase clusters with generated labels ('h/o' means 'history of')

Model	**CoD classification**			**Key phrase cluster prediction**		
	Precision	**Recall**	**F_1**	**Precision**	**Recall**	**F_1**
Majority class	.027	.163	.046	.292	.070	.105
Key phrase only	-	-	-	**.498**	**.283**	**.317**
CoD only	.755	.746	.743	-	-	-
Multi-task	**.760**	**.753**	**.750**	.481	.276	.310

Table 2: Weighted average scores from 10-fold cross-validation using the GRU/CNN model

CoD classification			
Features	**Precision**	**Recall**	**F_1**
Majority class	.027	.163	.046
Embeddings	.757	.752	.747
Emb + LDA	.726	.703	.699
Emb + key phrases	**.779**	**.778**	**.774**

Table 3: Results using a CNN model with a parallel feed-forward network (inputs are word embeddings and key phrases or LDA topics respectively)

5.2 Cluster prediction

For new, uncoded records, we will have only the narrative and therefore will need to predict the key phrase clusters. For evaluation, because the clustering is unsupervised and we have no gold standard mapping of key phrases in the test data to clusters, we assign each test key phrase to a cluster using a k-nearest neighbor classifier ($k = 5$). We treat these clusters as the "true" labels for the key phrase prediction task.

5.3 Cluster interpretation

In order for these clusters to be useful to physicians, we need a text label for each. We could simply take the most frequent key phrase in each cluster as the label, but many key phrases are variations of the the same idea, or have extra details in them, so the most frequent phrase might not be the most representative. Therefore, to get a text label that is representative of the cluster, we choose the key phrase that is closest to the center of the cluster in vector space.

However, there are some key phrases which are much longer than average. Since the vector representation of each phrase is the average of the word embeddings, a phrase with many words is more likely to be closer to the center. Also, we want to favor shorter labels that are general enough to describe the members of the cluster. Therefore we introduce a length penalty: the score used for selecting the label phrase is the distance of the phrase embedding from the center of the cluster multiplied by the number of words in the phrase. This gives us cluster labels that are usually one or two words.

Table 1 shows some of the generated cluster labels and the associated key phrases.[4]

6 Results

Table 2 shows the results from 10-fold cross-validation for key phrase cluster prediction and CoD classification, using the multi-task learning model, as well as separate models. The majority-class baseline is the scores obtained by assigning every record to the most frequent class in the training set ('road traffic incidents').

Some key phrase clusters are much larger and more frequent than others, which can render them unhelpful if too many different key phrases are grouped together. For the key phrase majority

[4] All examples are from the first iteration of 10-fold cross-validation, since different clusters are generated for each training set.

Record CoD category	Physician-assigned key phrases	Nearest-neighbor clusters	Predicted clusters
Ischemic heart disease	stroke patient, fever, dizziness for days, severe abdominal pain, diggings's, sudden pain abd.	oliguria, fever, sometime, abdominal pain, oliguria, diahorrea	pain abdomen, fever
Chronic respiratory infections	cough, wheezing, breathlessness edema	cough, h/o cough, breathlessness, h/o edema	h/o cough, breathlessness
Liver and alcohol	heavy alcohol intake, less food, not having food at regular interval, excess consumption of alcohol	pesticide, pesticide, oliguria, pesticide	died in5 mts., oliguria, progressive

Table 4: Examples of predicted key phrase clusters and CoD categories from the test set. Nearest neighbor clusters are the clusters from the training set that are closest to the embeddings of the physician key phrases.

baseline, we assign the most frequent key phrase cluster from the training set to each record in the test set. Even though there are 100 possible clusters and multiple clusters per record, we get .292 precision from the most frequent cluster alone.

We also use the predicted key phrase clusters as features for CoD classification. We use the clusters predicted by the 'key phrase only' model as input to a CNN CoD classifier. The input to the CNN layer is the matrix of word embeddings from the narratives, as in the previous model, and key phrase clusters are represented as a binary array that is the input to a feed-forward layer of 100 nodes. The outputs of the CNN module and the feed-forward module are concatenated and used as input for the final softmax classification layer, which outputs the CoD category.

Table 3 shows the results of this model, compared to the same model architecture using 100 LDA topics as the second feature set. The model using predicted key phrase features performs much better than that using the LDA topics. It also outperforms both the CNN model using only the narrative embeddings (without the feed-forward layer), and the majority class baseline.

7 Discussion

Table 4 shows some examples of the key phrase clusters predicted by the multi-task model. As we can see from the first two examples, many of the predicted phrases capture the same type of information as the physician-generated key phrases, although not as thoroughly.

However, as seen in Table 1, the clustering doesn't always capture the type of similarity we're interested in, such as the 'breathing difficulty' cluster, which captures phrases containing 'difficulty', although these often represent different symptoms. In Table 4 we see that the cluster representing alcohol intake has been labeled as 'pesticide' (along with several other clusters), and the predicted clusters for the third record do not contain any useful information related to the CoD (alcohol consumption).

Despite the key phrase prediction accuracy being fairly low, adding these predicted clusters as features for CoD classification improves both the precision and recall of the model.

We suspect that topic modeling does not help in this case because the distribution of words is very similar between documents, since they all deal with symptoms leading up to death. In addition, the key phrases are generated by physicians, and can capture information that is not explicitly present in the narrative.

8 Conclusion

We have demonstrated that learning key phrases along with CoD categories can improve CoD classification accuracy for verbal autopsies. The text representation of the key phrase clusters also adds interpretability to the model. In future work, we will aim to improve the cluster prediction accuracy, and we will investigate unsupervised methods of extracting important information from VA narratives.

Acknowledgments

We thank Prabhat Jha of the Centre for Global Health Research for providing the dataset. Our work is supported by funding from the Natural Sciences and Engineering Research Council of Canada and from a Google Faculty Award.

References

Lukasz Aleksandrowicz, Varun Malhotra, Rajesh Dikshit, Rajesh Kumar Prakash C Gupta, Jay Sheth, Suresh Kumar Rathi, Wilson Suraweera, Pierre Miasnikof, Raju Jotkar, Dhirendra Sinha, Shally Awasthi, Prakash Bhatia, and Prabhat Jha. 2014. Performance criteria for verbal autopsy-based systems to estimate national causes of death: Development and application to the Indian Million Death Study. *BMC Medicine*, 12:21.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Peter Byass, Daniel Chandramohan, Samuel Clark, Lucia D'Ambruoso, Edward Fottrell, Wendy Graham, Abraham Herbst, Abraham Hodgson, Sennen Hounton, Kathleen Kahn, Anand Krishnan, Jordana Leitao, Frank Odhiambo, Osman Sankoh, and Stephen Tollman. 2012. Strengthening standardised interpretation of verbal autopsy data: The new InterVA-4 tool. *Global Health Action*, 5:19281.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

François Chollet. 2015. Keras. https://github.com/fchollet/keras.

Samuel Danso, Eric Atwell, and Owen Johnson. 2013. Linguistic and statistically derived features for cause of death prediction from verbal autopsy text. In *Language Processing and Knowledge in the Web*, pages 47–60. Springer Berlin Heidelberg.

Nikita Desai, Lukasz Aleksandrowicz, Pierre Miasnikof, Ying Lu, Jordana Leitao, Peter Byass, Stephen Tollman, Paul Mee, Dewan Alam, Suresh Kumar Rathi, Abhishek Singh, Rajesh Kumar, Faujdar Ram, and Prabhat Jha. 2014. Performance of four computer-coded verbal autopsy methods for cause of death assignment compared with physician coding on 24,000 deaths in low- and middle-income countries. *BMC Medicine*, 12:20.

Mireille Gomes, Rehana Begum, Prabha Sati, Rajesh Dikshit, Prakash C Gupta, Rajesh Kumar, Jay Sheth, Asad Habib, and Prabhat Jha. 2017. Nationwide mortality studies to quantify causes of death: Relevant lessons from India's Million Death Study. *Health Affairs*, 36(11):1887–1895.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 690–696, Sofia, Bulgaria.

Ryan Kelly. 2015. Pyenchant. http://pythonhosted.org/pyenchant/.

Tyler H McCormick, Zehang Richard Li, Clara Calvert, Amelia C Crampin, Kathleen Kahn, and Samuel Clark. 2016. Probabilistic cause-of-death assignment using verbal autopsies. *Journal of the American Statistical Association*, 111(15):1036–1049.

Pierre Miasnikof, Vasily Giannakeas, Mireille Gomes, Lukasz Aleksandrowicz, Alexander Y Shestopaloff, Dewan Alam, Stephen Tollman, Akram Samarikhalaj, and Prabhat Jha. 2015. Naïve Bayes classifiers for verbal autopsies: Comparison to physician-based classification for 21,000 child and adult deaths. *BMC Medicine*, 13(1):286–294.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Erin K. Nichols, Peter Byass, Daniel Chandramohan, Samuel J. Clark, Abraham D. Flaxman, Robert Jakob, Jordana Leitao, Nicolas Maire, Chalapati Rao, Ian Riley, and Philip W. Setel. 2018. The WHO 2016 verbal autopsy instrument: An international standard suitable for automated analysis by InterVA, InSilicoVA, and Tariff 2.0. *PLOS Medicine*, 15(1):e1002486.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Peter Serina, Ian Riley, Andrea Stewart, Spencer L James, Abraham D Flaxman, Rafael Lozano, et al. 2015. Improving performance of the Tariff method for assigning causes of death to verbal autopsies. *BMC Medicine*, 13(1):291.

Theano Development Team. 2016. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688. http://arxiv.org/abs/1605.02688.

Erica Westly. 2013. Global health: One million deaths. *Nature*, 504(7478):22–23.

World Health Organization. 2008. *International statistical classifications of diseases and related health problems. 10th rev*, volume 1. World Health Organization, Geneva, Switzerland.

World Health Organization. 2012. *The 2012 WHO Verbal Autopsy Instrument*. World Health Organization, Geneva, Switzerland.