Learning Semantic Tags from Big Data for Clinical Text Representation

Yanpeng Li, PhD and Hongfang Liu, PhD
Mayo Clinic, Rochester, MN

Abstract

In clinical text mining, it is one of the biggest challenges to represent medical terminologies and n-gram terms in sparse medical reports using either supervised or unsupervised methods. Addressing this issue, we propose a novel method for word and n-gram representation at semantic level. We first represent each word by its distance with a set of reference features calculated by reference distance estimator (RDE) learned from labeled and unlabeled data, and then generate new features using simple techniques of discretization, random sampling and merging. The new features are a set of binary rules that can be interpreted as semantic tags derived from word and n-grams. We show that the new features significantly outperform classical bag-of-words and n-grams in the task of heart disease risk factor extraction in i2b2 2014 challenge. It is promising to see that semantics tags can be used to replace the original text entirely with even better prediction performance as well as derive new rules beyond lexical level.

Introduction

Text representation is an important fundamental step in various research areas such as information retrieval (IR), information extraction (IE), natural language processing (NLP), machine learning and artificial intelligence. In clinical domain, it is one of the biggest challenges to model the vast amount of medical and biological terminologies as well as multi-word expressions in medical reports. The most straightforward approach known as "bag-of-words" is to represent text as a vector of unique words where the positions of words are ignored. In the past decades, this simple approach has shown surprisingly good results in many tasks such as named entity recognition, relation extraction, text classification and information retrieval, although there are obvious drawbacks such as ignorance of word order, lack of semantic-level knowledge and etc. To our best knowledge, there is still not a general approach that can replace bag-of-words by semantic level features with consistent better performance as well as the ability of new rule discovery, although there are a lot of effort to transform bag-of-words into higher-level representation such as WordNet [1], UMLS[2], latent semantic analysis (LSA) [3] or clustering methods [4]. These techniques can lead to improvement in some special cases when combined with bag-of-words, but usually cannot be the direct substitution of words or n-grams, and it is difficult to interpret the new features generated by LSA or clustering based methods.

This work was based on our previous works on feature coupling generalization (FCG) [5][6] and reference distance estimator (RDE) [7]. FCG is a framework for learning new features from unlabeled data via linear combination of the co-occurrence measure of each feature and a reference feature. It achieved top performance on benchmark data such as BioCreative and TREC[5][6]. RDE can be viewed as a special case of FCG, which was justified by theory to perform well on some special case and experimentally achieved state-of-the-art performance on the task of text classification [7] and gene ontology (GO) annotation in BioCreative IV challenge [8][9]. However, there are two major issues of these methods: 1) the linear combination could lose the non-linear dependency between features. 2) It is still difficult to give a clear interpretation of each new feature generated by FCG or RDE, since each feature is a real value that describes the characteristic of the whole feature set rather than individual words and n-grams.

Rather than linear combination of word co-occurrence, in this work we proposed a novel method for generating semantic tags for word and n-gram representation for clinical text harnessing a large amount of unlabeled medical records. The method represents each word by the distances with a set of reference features [7], discretizes each distance metric into binary rules and then randomly merges the binary rules into novel features (also called semantic tags or rules). For n-gram representation, we used similar technique of random sampling to merge the semantic tags of consecutive words into novel semantic tags. The semantic tags can be used to replace words or n-grams directly, so that there is no information loss from linear combination. Since the novel semantic tags are binary rules derived from a set of reference distance estimators [7], they are easy to be interpreted by human as if each reference feature is an ontology concept. In multiple sentence classification tasks derived from i2b2 2014 challenge, we showed that

the generated semantic tags were able to replace the word and n-grams with better performance and discover novel rules beyond lexical level.

Method

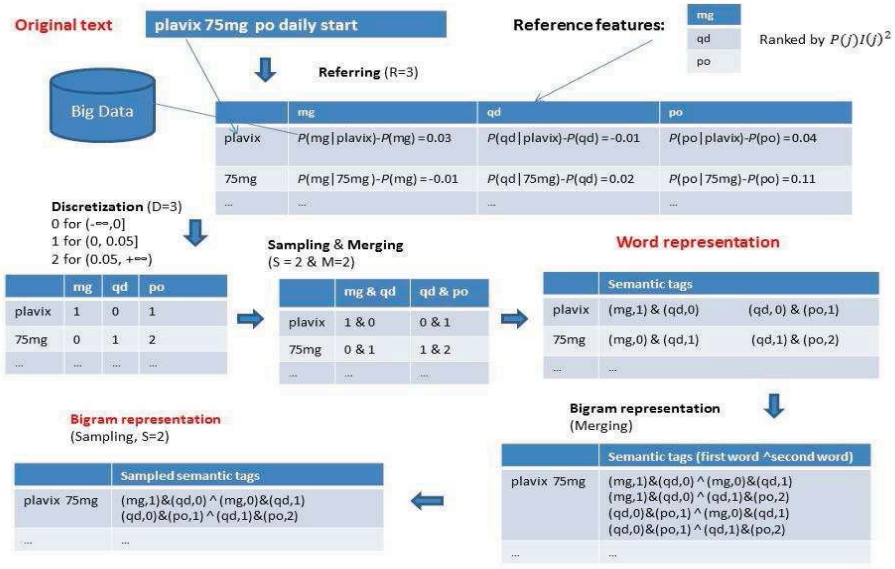

Figure 1. An example of RDSM algorithm for semantic tag generation. The reference features are for the task of identifying sentences related to medications in i2b2 2014. Features in red color are final features for classification.

In our method we used the metric from *feature imbalance coefficient* [7] to measure the individual performance of each word j for predicting class y:

$$I(j) = \frac{P(j,y) - \alpha P(j,\bar{y})}{P(j)} \quad (1)$$

where $\alpha = P(y)/P(\bar{y})$, $P(j,y)$ and $P(j,\bar{y})$ are the joint probabilities of j and class labels. The metric $I(j)$ measures the imbalance degree of feature j over the positive (y) and negative $(\bar{y})$ classes. The core of our approach for text representation is word representation. The n-gram representation is based on the output of word representation. The algorithm includes the steps of referring, discretization, sampling and merging, so we call it RDSM framework. The algorithm for word representation is described as bellow:

1) Referring: for each class y, select R reference features (words) by the metric $P(j)I(j)^2$, and represent each word j as its reference distance $P(r|j) - P(r)$ with each reference feature r.
2) Discretization: divide each real valued reference distance into D binary intervals and use each interval as a binary rule. Therefore, each word can be represented by R*D binary rules.
3) Sampling: Randomly select S subsets from the R reference features, where each subset contains M ($M \leq R$) reference features.
4) Merging: for each subset, merge all the binary rules by the union to generate a new binary feature. Use the set of merged binary features to replace the original words.

The algorithm for n-gram generation is simply to merge the semantic tags of consecutive word by the product and then find a subset from the merged word semantic tags by random sampling. The referring step is to represent each word by its co-occurrence with reference features in big data, which can incorporate rich information beyond the small size of annotated data [5][6][7]. Discretization is to generate interpretable binary rules from real valued distance. Merging step is necessary because the combination of the discretized distances from multiple references

tend to provide more accurate information than each individual reference feature. However, to enumerate all the possible combination of reference features is a NP hard problem, so we used random sampling for subset selection within the computational power of the current computers. In the experiment section we will show that the reference feature size R, discretization size D, sample size S and merging size M are important parameters to achieve good performance. In our experiment, R was set at 200, and the discretization intervals were divided by {0, 0.005}. For word representation we combined the two feature sets (S=40, M=8) and (S=40, M=10). For bigrams, we used the model (S=60, M=4) for both the first and second words. For trigrams, (S=20, M=3) was used for all the three words. Figure 1 shows an example of the RDSM method for word and bigram representation.

Experiment

Design

The data set we used was from i2b2 2014 Heart Disease Task (https://www.i2b2.org/NLP/HeartDisease/), which aims to extract risk factor information for heart disease from clinical notes on diabetic patients, such as high blood pressure and cholesterol levels, obesity, smoking status, etc. The annotation at sentence level was provided. We approached it as multiple binary classification tasks for sentences. We treated the concepts such as medications and smokers as well as XML tags in labeled data such as time="during DCT", type1="metformin", and status="never" as classes, and got 39 classes in training and development data in total. In our approach, if a sentence has overlap with the offset the gold standard annotation, it would be assigned as positive labels. Otherwise, it would be negative. For each sentence there was a 39 dimensional vector where each element referred to a class. There were 48765 and 23522 examples from training and development (for testing) corpora respectively. Therefore, it will be straightforward to apply our RDSM algorithm to this task. The unlabeled data for learning reference features was around 500k medical records from Mayo Clinic. Since the labeled (i2b2 2014 corpus) and unlabeled (Mayo) datasets were from different hospitals, it will be interesting to see if the method can learn better feature representation from similar but not completely identical domains. The classifier we used was SVM-Light (http://svmlight.joachims.org/) with default parameters. We evaluated the performance on both the macro-average of the 39 binary classification tasks as well as the official metrics in i2b2 2014. Note that our goal in this work is not to derive the best system in i2b2 but to compare different feature representations in NLP tasks.

Result

Table 1. Lexical V.S. semantic representation

Features	**Precision / Recall / F1 / AUC (Macro, Sentence-level)**	**Precision / Recall / F1 (Micro, i2b2 2014)**
Words	0.6815 / 0.7105 / 0.6717 / 0.9572	0.8546 / 0.8276 / 0.8409
Semantic tags from words	**0.6846 / 0.7431 / 0.6981 / 0.9722**	**0.8586 / 0.8401 / 0.8493**
Bigrams	0.6056 / 0.5702 / 0.5579 / 0.8814	0.7989 / 0.7368 / 0.7666
Semantic tags from bigrams	0.6168 / 0.6452 / 0.6144 / 0.9593	0.817 / 0.7838 / 0.8001
Trigrams	0.5227 / 0.4164 / 0.4452 / 0.7792	0.7222 / 0.6183 / 0.6662
Semantic tags from trigrams	0.5936 / 0.6018 / 0.5831 / 0.9504	0.755 / 0.7509 / 0.753

In Table 1, we compared the performance of semantic words and lexical words in different measures at micro or macro level. It is promising to see that the performance of semantic tags was much better than lexical representation. We can conclude that at least for the 39 tasks the semantic tags can replace the bag-of-words and n-gram features without the loss of accuracy. For n-gram features, the performance was much worse than word features probably due to data sparseness. The good news was that semantic n-grams performed over 5% or 10% better than lexical n-grams, since the sparseness of semantic n-grams can be adjusted by tuning parameters such as the merging size. Note that for human learning, n-gram should be a better representation t. But for lexical feature based NLP approaches, it is well known that the individual performance of n-grams are usually much worse than word

representation in various tasks, due to the data sparseness issues [6] [7]. The results in Table 1 show the promising trend that makes smaller the gap between n-gram features and word features.

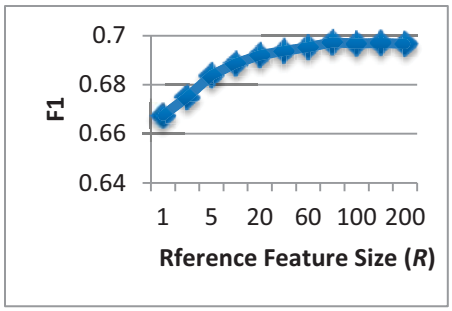

Figure 2a. R v.s. F1. (D=3, S=80,M=8)

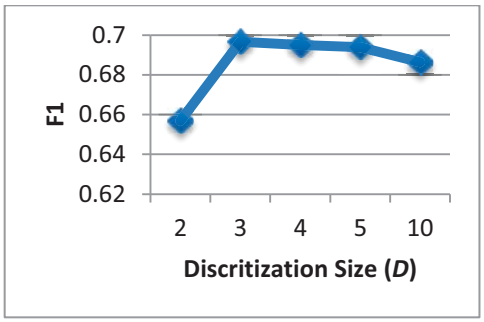

Figure 2b. D v.s. F1. (R=200, S=80,M=8)

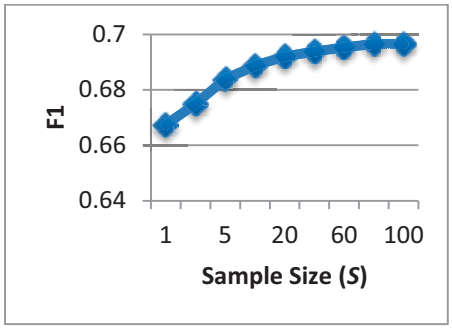

Figure 2c. S v.s. F1. (R=200, D=3, M=8)

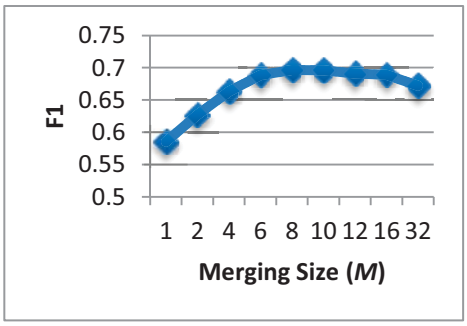

Figure 2d. M v.s. F1. (R=200, D=3, S=80)

Figure 2. Relation between RDSM parameters and prediction performance. Macro F1 of 39 binary classification tasks was used for evaluation. Semantic features for word were applied.

Table 2. Examples of indicative rules discovered by words and semantic tags. The numbers in {} are F1 scores in training and testing data respectively obtained by individual rule. Each semantic tag is the conjunction of several pairs in the format (reference feature, distance). For distance, 0 for (-∞, 0], 1 for (0, 0.005], and 2 for (0.005, +∞) (See Method section).

	Words	**Semantic tags from words**
Medication	mg {0.44, 0.38} po {0.41, 0.36} qd {0.41, 0.36} tablet{0.25,0.18} take {0.25,0.11}	(insulin,1)&(bid,1)&(acetylsalicylic,1)&(s,2)&(captopril,0)&(increase,2)&(zetia,0)&(twice,0)&(blockade,1)&(x1,1) **{0.54,0.49}** (aspirin,2)&(asa,1)&(start,0)&(succinate,1)&(novolog,1)&(sr,0)&(100mg,2)&(125,0) {**0.54, 0.48**} (atenolol,2)&(50,1)&(20mg,0)&(81,1)&(verapamil,1)&(diltiazem,0)&(childrens,1)&(cont,1)&(once,0)&(fosinopril,1) {**0.51,0.47**}
CAD	cad{0.35,0.35} coronary{0.29,0.3} artery {0.23, 0.26} disease{0.21,0.22} lad (0.21, 0.18)	(collaterals,0)&(male,0)&(2144,0)&(dm,0)&(stented,1)&(ramus,0)&(which,1)&(2148,0)&(2080the,0)&(depresion,0) {**0.52, 0.5**} (cad,0)&(cabg,1)&(ptca,1)&(a,0)&(placement,0)&(w,0)&(zone,0)&(had,0)&(intramyocardial,0)&(chest,0) {**0.45, 0.42**} (myocardial,0)&(lima,0)&(lcx,0)&(%,0)&(status,2)&(hyperaldosteronemia,0)&(vt,1)&(left,0) {**0.45, 0.46**}

In Figure 2, we can see the impact of parameter selection for RDSM algorithm. It is promising to see the trend that the F1 score increased with parameter sizes for all parameters, indicating that the new information introduced by these factors contribute to prediction accuracy and the results are big support of the successes of our proposed algorithm since each novel step introduced in this work e.g., discretization, sampling and merging improved the

result significantly. It was surprising to find that using random sampling, the most naïve method, the performance can beat classical bag-of-words. Therefore, we believe that there should be further progress if we try some better sampling or parameter selection methods in the future.

In Table 2, we observed the top ranked features by F1 to investigate if it was possible to discover some interesting rules from the semantic tags. It is surprising to find that the semantic rules learned from big data performed over 10% better than top word features. Due to page limitation, we only listed a few examples from two representative classes "medication" and "CAD". The semantic tags for each word can be interpreted as the coupling degree with a set of reference features that can be viewed as automatically generated ontology concepts. Obviously, one way to make the learned semantic rules more attractive to domain experts is to develop individual reference features they may be interested in. In all, our RDSM method was able to learn powerful rules (e.g., Table 2) beyond lexical level, which were not reported in previous studies.

Conclusion

In this study, we investigate a novel method for word and n-gram representation based on reference distance learning, discretization, sampling and merging. The experimental results showed that the effort in each step contributed to prediction performance as well as new rule discovery. Furthermore, we found only the random sampling approach could beat classical method in benchmark data, so we believe that advance sampling technique as well as reference features selection could improve the performance further. Also, we will work on theoretical study of such approach for word representation as well as feature representation for machine learning in general. Since our framework is novel, there are a lot of open questions for automatic selection of parameters. We are not able to give a fully automatic solution for all the steps in this work. We will focus on these issues in our future works.

Acknowledgment

The authors acknowledge that the study was supported by the following grants: R01GM102282, R01LM11369, R01LM11829, and R01LM11934.

References

1. Miller, G. A. (1995). WordNet: a lexical database for English. Communications of the ACM, 38(11), 39-41.
2. Bodenreider, O. (2004). The unified medical language system (UMLS): integrating biomedical terminology. Nucleic acids research, 32(suppl 1), D267-D270.
3. Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990). Indexing by latent semantic analysis. JASIS, 41(6), 391-407.
4. Turian, J., Ratinov, L., and Bengio, Y. (2010, July). Word representations: a simple and general method for semi-supervised learning. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (pp. 384-394). Association for Computational Linguistics.
5. Li, Y., Lin, H., and Yang, Z. (2009). Incorporating rich background knowledge for gene named entity classification and recognition. BMC bioinformatics, 10(1), 223.
6. Li, Y., Hu, X., Lin, H., and Yang, Z. (2011). A framework for semisupervised feature generation and its applications in biomedical literature mining. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 8(2), 294-307.
7. Li, Y. (2013). Reference Distance Estimator. arXiv preprint arXiv:1308.3818.
8. Mao, Y., Van Auken, K., Li, D., Arighi, C. N., McQuilton, P., Hayman, G. T., ... and Lu, Z. (2014). Overview of the gene ontology task at BioCreative IV. Database, 2014, bau086.
9. Li, Y., and Yu, H. (2014). A robust data-driven approach for gene ontology annotation. Database, 2014, bau113.