Identifying Tweets with Adverse Drug Reactions

1 Introduction

This report clarifies how to extract features from tweets and use supervised machine learning methods to determine whether a tweet contains an ADR[1], and gain knowledge from it. In this report, we will use Weka(Hall et al., 2009) as the machine learning package.

2 Dataset

The dataset is from Twitter, and is an altered form of a dataset from the DIEGO Lab(Sarker and Gonzalez, 2015). Three parts are included. We will use training data to train the model, use development data to evaluate and predict whether the test data contain ADRs.

3 Related work

Some researches apply NLP techniques and machine learning methods to detect ADR. As described by Sarker and Gonzalez (2015), his research used NLP approches to generate rich semantic and linguistic features from the online texts and use supervised machine learning methods to classify the instances.

4 Machine learning methods

There are bunch of classification methods, such as Naive Bayes, K-nearest, SVM, and etc. when trying Naive Bayes with unigram features, the accuracy is 82.1561%, which seems good, but the F-Measure of positive ADR is only 0.364. The reason is that the class distribution is highly skewed, with 89.4%'s negative ADR. So, even we always predict the negative, the accuracy could be higher(89.4%), but obviously it is not a good model. As a result, except accuracy, we will also focus on the F-Measure in the following analysis.

According to the previous researches, SVM with linear kernal gives a good performance, so we choose it as our classification method and by using Weka, we use LibSVM(Chang and Lin, 2011), which has a better performance than the original SMO package.

[1] `https://en.wikipedia.org/wiki/Adverse_drug_reaction`

5 Basic features

The basic approach used Mutural Information to identify 92 best unigrams from tweets. The Mutural Information is a method to calculate the dependency between features.

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \left(\frac{p(x,y)}{p(x)\, p(y)} \right)$$

As formula is given, the feature is more valuable if it is more related to the class label. We use tweet ID and the 92 unigrams as basic features.

6 New features

By observing bad cases of the unigram-feature model, We find some new features which could help increase the performance.

6.1 Punctuation

Punctuation conveys the user's emotion and attitude to some extent, and the tweets contain ADRs may be likely to use more punctuation(e.g. when complaining). Several marks are chosen from the tweets.

- Exclamation mark.
- Question mark.
- Ellipsis mark("...").

We use the frequency of each type of punctuation as a new feature, since more than one mark could occur in one single tweet.

6.2 Emotional icons

Likewise, people use emotional icons(such as :(or :D) to describe their emotion. I collect many emotional icons from the tweets, and divide

them into two sets, which respectively represent the positive and the negative emotion. So, we divide the tweets into three parts, "Happy", "Unhappy" and "Neutral"(H, U, N). By this way, we create a new nominal feature(H, U, N).

6.3 ADR lexicon

There are some ADR related terms in the tweets, such as some drug side effects(e.g. insomnia), and when these terms occur, obviously, it will be more likely to find an ADR. So, we use the frequency of ADR related terms as the new feature. The ADR lexicon is cited from Nikfarjam et al. (2015), including information from SIDER, CHV(Consumer Health Vocabulary) and DIEGO_lab.

6.4 Sentiment analysis

Many tweets contain sentiment, which may be related to ADR. For example, the first instance, "Glad I'm taking olanzapine...", presents positive sensiment, which may suggest there is no ADR. We have already extracted a sentiment related feature, the emotion icon, but we still want to consider the sentiment of the whole tweet text. By using python package TextBlob(Loria, 2013), which has a method to quantify the sentiment polarity, from -1 to 1, we extract this value from tweets as a new feature.

6.4.1 Subjectivity

Another paramter given by TextBlob package is the subjectivity, which evaluates whether the given text is subjective or objective, also with the range from -1 to 1. We also use it as a new feature, since the subjectivity of tweet may have some relation with the ADR.

6.5 Polynomial features

More features could be generated by polynomial combinations of the original ones, and may give us some better performance. We used python package scikit-learn(Pedregosa et al., 2011) to create the 3-degree polynomial features among punctuation features and among lexicon and sentiment features.

6.6 Topic model

Tweets may contain different topics, and the ones which contain ADRs may have similar topics. Based on this idea, we extract the topic features from tweets, using Stanford Topic Modeling Toolbox(Ramage and Rosen, 2009), which generates weights of topics according to the text, and we use these weights as new features.

6.7 Doc2Vec

Based on the words and their contexts, text could be converted into a N-dimension vector. This method is called Doc2Vec. By using python package gensim(Řehůřek and Sojka, 2010), we genenrate a 100-dimension vector for each tweet and use vector values as features.

7 Evaluation

Feature	Accu	F1/N	F1/Y	F1
Baseline	89.4052%	0.944	0.0	0.844
Basic features	89.777%	0.945	0.286	0.875
Basic + punctuation	89.684%	0.944	0.343	0.880
Basic + punc (poly)	89.9628%	0.946	0.349	0.882
Basic + emotion	89.684%	0.944	0.343	0.880
Basic + lexicon	90.0558%	0.946	0.301	0.878
Basic + sentiment	89.8699%	0.946	0.278	0.875
Basic + subjectivity	89.777%	0.945	0.345	0.881
Basic + lss[2](poly)	89.8699%	0.945	0.323	0.879
Basic + topic	89.777%	0.945	0.329	0.879
Basic + doc2vec	89.8699%	0.945	0.297	0.877
Basic + All	89.9628%	0.946	0.341	0.882
Basic + All(poly)	90.2416%	0.947	0.371	0.886

Table 1: Comparison of different features

To evaluate, we choose the baseline which predicts all tweets contain no ADR, with the accuracy of 89.4052%. F-Measure and the accuracy are used as evaluation metrics. Cost parameter(c) is an important parameter for SVM, describing the penalty for mismatch. We try different values of c, and only show results with the best performance.

[2]lss means the combination of lexicon, sentiment and subjectivity

As the result, our all-feature model with polynominal combinations has the best performance(see Table 1). The accuracy and F-Measure are both improved, and if we dig into this improvement, we can find that the main effort comes from the Recall of positive ADR, increased from 0.193 to 0.272, which means now we can detect positive ADRs better.

8 Conclusions

In this report, we establish the model for predicting ADR in tweets, and we focus more on the feature extraction, using the punctuation, emotion icon, ADR lexicon, sentiment, topic model and doc2vec model. As a result, the performance of the new model is increased effectively. More methods of feature extraction and classification may be discussed in the future.

References

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November.

Steven Loria. 2013. TextBlob. `https://github.com/sloria/TextBlob`.

Azadeh Nikfarjam, Abeed Sarker, Karen O'Connor, Rachel Ginn, and Graciela Gonzalez. 2015. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, 22(3):671–681.

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. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Daniel Ramage and Evan Rosen. 2009. Stanford topic modeling toolbox.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA. `http://is.muni.cz/publication/884893/en`.

Abeed Sarker and Graciela Gonzalez. 2015. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *Journal of Biomedical Informatics*, 53(Supplement C):196 – 207.