Generating domain specific sentiment lexicons using the Web Directory

Akshay Minocha[1] and Navjyoti Singh[2]

[1]Center for Exact Humanities, International Institute of Information Technology Hyderabad, Hyderabad, India.
akshay.minocha@students.iiit.ac.in
[2]Center for Exact Humanities, International Institute of Information Technology Hyderabad, Hyderabad, India.
navjyoti@iiit.ac.in

ABSTRACT –

In this paper we aim at proposing a method to automatically build a sentiment lexicon which is domain based. There has been a demand for the construction of generated and labeled sentiment lexicon. For data on the social web (E.g., tweets), methods which make use of the synonymy relation don't work well, as we completely ignore the significance of terms belonging to specific domains. Here we propose to generate a sentiment lexicon for any domain specified, using a twofold method. First we build sentiment scores using the micro-blogging data, and then we use these scores on the ontological structure provided by Open Directory Project [1], to build a custom sentiment lexicon for analyzing domain specific micro-blogging data.

KEYWORDS –

Sentiment Analysis, Domain Specific Ontology, Ontology, Twitter, Micro-Blogging, Sentiment Lexicon.

1. INTRODUCTION –

Sentiment classification [2][3] is the process of identifying the opinion (e.g. negative or positive) of a given document. With an ever expanding corpus of micro-blogging data available on the web, it becomes increasingly demanding for us to classify this data. We use dataset formed from tweets on twitter, mainly because it covers a large deal of domains from product reviews, general opinions, and personal as well as political thought.

We first aim at labeling words from the micro-blogging dataset according to the valency of the emotions they convey (positive or negative). This process involves a lot of processing of the data because the opinions conveyed by the micro-blogging data are not very highly detailed, and since this is not gold standard, corpus processing becomes necessary as an attempt to reduce the noise somewhat. The data collected here is not specific to a particular domain but is randomly fetched to build generic scores.

It is very difficult to build a sentiment lexicon which would yield consistent results in all the domains, mainly because of the lack of knowledge about what terms are more important to a particular domain than the others. There may be an approach by manually building such lexicons for domains, but indeed this would be difficult to make, as this procedure requires a great deal of knowledge about the specific domain by the people building it. Instead we propose

a general approach to build a lexicon which would be highly domain specific, using the relational hierarchies already provided to us by the open directory project. This ontology based structure of categories is built using many publicly available resources which are posted on the web. This initiative of adding in the Open Directory Project [1] is manually done by about 96,000 editors who have successfully classified things into over 800,000 categories. Work advances in this field have been done to some extent, where researchers have proposed the use of domain ontology to identify features used to determine the sentiment [4]. Some have used statistical learning methods to extract domain specific sentiments [5]. Here we mainly will focus on the building up of the domain specific sentiment lexicon using the ontology provided and combining it with the general rated terms to determine the sentiment.

We match the scores collected from the corpus in the first stage and then map it to the nodes of the categorical structure provided to us by the open directory project. By this technique many of the nodes are matched but still there are a lot that remain unknown. It is now we adapt a method of pseudo labels on the unknown nodes of this hierarchy by taking a few assumptions (like likelihood of a child node being more prominent in the case of a known parent, or major positive. Negative siblings if the parent is unknown, etc.) Into consideration and generate the complete labeled structure.

After this phase of labeling is done, we developed an interface where you can enter a domain of choice, if it exists in the tree structure then a customized lexicon is created in which the nodes of this domain are more prominent than the rest, along with the most contributing 2000 n-grams are also included. Generally the other approaches don't perform well on specific domains when the data is small or when it comes from a different domain [6] [7], or time period [7]. By this approach we are able to tag many important words and terms altogether.

2. PROCEDURE

2.1 Data Collection

The data was assumed to be classified by the labels of emoticons attached to them. This automatic collection of training data has been a common and reliable practice. [8] [9]
The following shows how we did distinguish between the positive and the negative dataset:

Positive data had emoticons - " :) :-) :D :-D "
The negative data had emoticons - " :(:-(:'(:-'("

The language used by the users of the micro-blogging communities do not follow the general grammar guidelines given to the restriction in size (140 characters) by websites like twitter. For the same reason it was seen that parts-of-speech features were not useful for sentiment analysis in this domain, instead the emoticons proved to be really helpful [10].
We chose (44823) tweets from the twitter dataset available [11]. Here we extracted (27538) positive and (17285) negative tweets out of the whole dataset to help us build our initial corpus.

2.2 Processing the Corpus

The data was processed through a number of stages so as to reduce the noise –

i. Removal of hyperlinks – These hyperlinks do not contribute to our lexicon.

ii. Removal of "RT" – For data collected from twitter people tend to re-tweet the tweets posted by another user. This re-tweet is done along with a "RT" tag at the beginning of the tweet. This tag was removed.

iii. Removal of @xyz – for data collected from twitter people use the "@" sign followed by the username of the person they are referring to. The user-name as well as the "@" symbol do not contribute anything to our lexicon so they are removed as well.

iv. Removal of other extra punctuation marks – this is done in order to better tokenize the words so that their frequency features can be compared.

The distribution of the word frequencies follows Zipf's law [8].So as a basic test model for our hypothesis we build a sentiment lexicon which makes use of the frequencies of the n-grams in the dataset. The sentiment scores, to each of them are given as – $[X_p, X_n]$ where $X_p+X_n =1$, and X_p is the positive score and X_n is the negative score. Hence a neutral term will have the score of [0.5, 0.5].

2.3 Ontology Tree Generation –

Web directories such as Open directory Project [1] and Yahoo directory [12], divide web pages into categories. Here we use the categorical data available by the Open Directory Project[1] We use the Open Directory Project [1]for the category and sub-category classification of terms. This gives us a broader ontology tree where domain specific ontological terminology can be located.

We first generate an ontology tree using the help of the Open Directory Project. There are 16 top level categories including arts, business, games, shopping, etc. and all the other terms fall into sub-categories under them within 15 levels. These categories are listed in a hierarchical ontology tree like structure.

2.4 Rating the Unknowns nodes –

After getting to know about the sentiment scores of the words in the lexicon we take the next step where our aim is to label the nodes of the Open Directory [1] Category Tree, which was created in the last step.
The nodes are labeled in the following ways –

i. We match the unigrams, bigrams and trigrams already rated from the dataset to the nodes in the tree which match perfectly with them.
ii. There will be some unknown nodes which will remain. We now make use of the definitions of some of the node-terms which have been provided, to generate a sentiment score on a similar scale (X_p, X_n). After determining the score we label the nodes with the same.
iii. We make the following assumptions to label the remaining unknown children nodes -

a. Parent nodes have major contribution to the score of their children nodes.

b. If value of the parent node is unknown then the node is more likely to resemble the siblings.

So first, we apply the algorithm -

$$\frac{(d-1)*X_{1p}+(d-2)*X_{11_p},\cdots+(d-n)X_{1\ldots 1p}n*d}{d*n}$$

Calculation Of the score of an unknown node via its parent, Here d is the depth of the node in the tree X_{1p} is the positive rating of the parent, X_{11p} is the positive score of the parent of X_{1p} and so on up to $X_{1\ldots 1pn}$ which is the score of the last known parent in the hierarchy.

iv. Out of the remaining we apply the determining method as talked about in the assumption above which involves the calculation on the basis of the siblings in the tree structure. Here we calculate the mean of the known nodes and then label them as (X_p, X_n).

$$\frac{\sum_{i=1}^{n} X_i}{n}$$

Where n is the number of known sibling nodes.

In our procedure at the end of the above mentioned labeling process there were no unknown nodes but if there would have been any, labels of a neutral score for them would have been the most appropriate, i.e, (0.5, 0.5)

2.5 Building a Customized Domain Specific Sentiment Lexicon –

There is an interface built where we input the domain of choice, and then we get the terms along with the scores related to the domain in the tree structure. It may be pointed out here, that we are using a greedy approach to choose just the nodes which would be important for the sentiment analysis of the domain. As described in the figure below, suppose our domain of inspection be "arts" then values under the specific domain will be retrieved making them more prominent over all other domains –

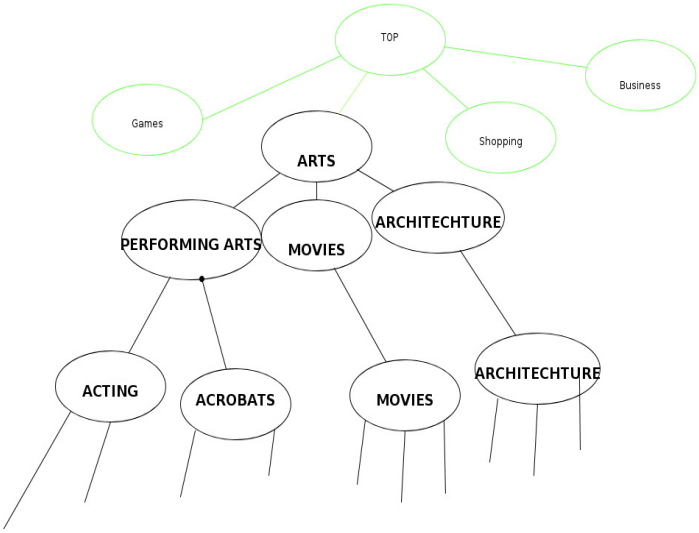

Figure 1. Shows the prominence of the domain of Arts in the hierarchy when "Arts" is selected

Apart from the above mentioned terms some general 2000 N-gram positive and negative terms are also taken, from the lexicon created using the dataset in 2.2.

3. ANALYSIS -

It is now that the whole sentiment lexicon has been created and is ready for analysis. Theoretically this lexicon should have leverage over the general analysis as the terms from the ontology have been made more prominent. It is all about the accuracy of the sentiment analysis of the corpus terms that the accuracy of the domain specific lexicon would be dependent upon. The working of the whole procedure is shown in Figure 2.

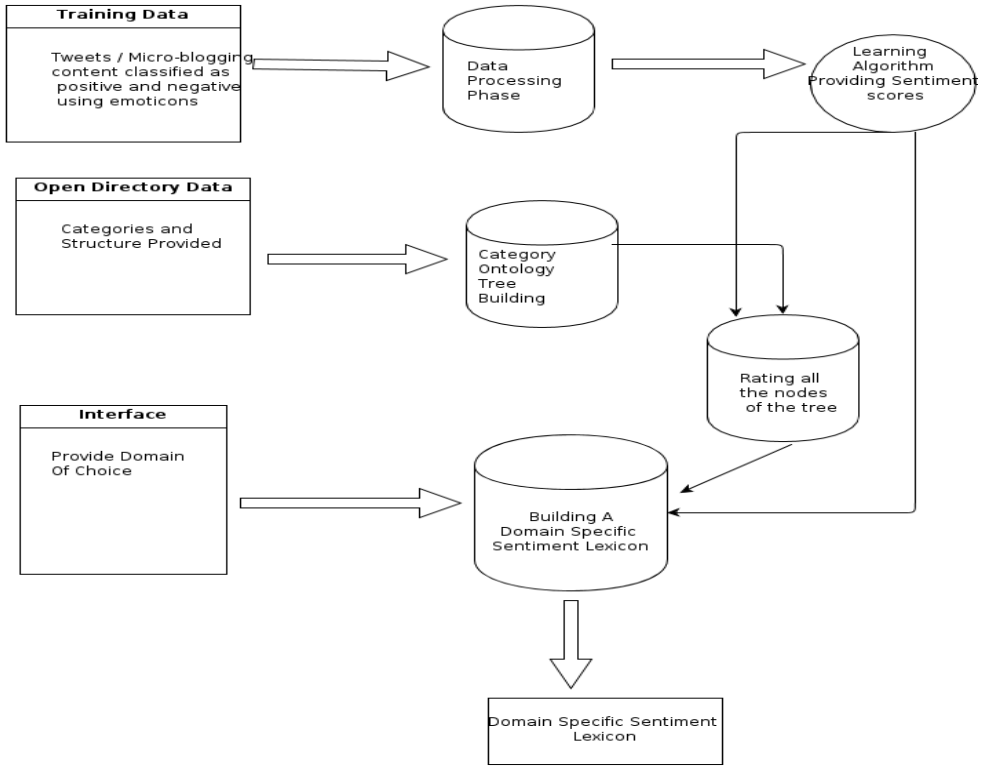

Figure 2. The working of the all the steps described.

We performed the following experiments on 5 different domains; Tweets from these domains were collected. The size of the dataset was about 2800 (1400 positive and 1400 negative tweets) . A comparison of accuracy from the new domain specific lexicon vs. the earlier sentiment scores from dataset. And we found a considerable increase in accuracy for each of them.

Table 1. Comparison of Accuracy

Domain	Accuracy % without use of Domain-specific Ontology	Accuracy % By New Domain-Specific Lexicon Created
Computers	62.03	65.21
Science	59.92	66.04
Shopping	59.42	62.53
Games	60.32	66
News	81.35	84

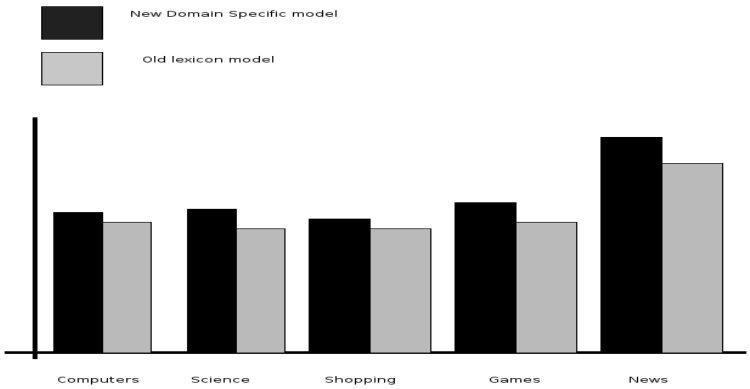

Figure 3. Comparison of Accuracy by the two models.

4. CONCLUSION

Our main aim was to build automatically an ontological based lexicon and then merge it with the sentiment scores provided by the classifiers or learning algorithm which is used to tag the dataset in 2.2. Looking at the results we can see that the customized sentiment lexicon works better than the initial learning algorithm in classifying the data related to a particular domain.

References

[1] http://dmoz.org/about.html

[2] P. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In Proceedings of ACL 2002.

[3] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In Proceedings of EMNLP, 2002.

[4] K. P. P. Shein, "Ontology based combined approach for sentiment classification," in 3rd International Conf. on Communication and Information Technology, 2009, pp. 112–115

[5] Lau, Raymond, Lai, Chun-Lam, Bruza, Peter D., & Wong, Kam-Fai (2011) Leveraging Web 2.0 data for

scalable semi-supervised learning of domain-specific sentiment lexicons. In 20th ACM Conference on
Information and Knoweledge Management, 24-28 October 2011, Crowne Plaza, Glasgow.

[6] Aue, A. and M. Gamon. 2005. Customizing sentiment classifiers to new domains: a case study. In Proceedings of the International Conference on Recent Advances in Natural Language Processing, Borovets, BG.

[7] Jonathon Read. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In Proceedings of the ACL-2005 Student Research Workshop, Ann Arbor, MI.

[8] Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. Proceedings of LREC 2010, Retrieved October 5, 2010 from: http://www.lrec-conf.org/proceedings/lrec2010/pdf/2385_Paper.pdf.

[9] Bifet, A. & Frank, E. (2010). Sentiment knowledge discovery in Twitter streaming data. In B. Pfahringer,
G. Holmes, & A. Hoiffmann (Eds.), LNAI 6332, Discovery Science, Proceedings of 13th International
Conference, DS 2010, Canberra, Australia, October 6-8, 2010 (pp. 1-15). Berlin, Germany: Springer.

[10] E. Kouloumpis, T. Wilson, and J. Moore. Twitter sentiment analysis: The good the bad and the OMG!
ICWSM, 2011.

[11] Z. Cheng, J. Caverlee, and K. Lee. You Are Where You Tweet: A Content-Based Approach to Geo-locating Twitter Users. In Proceeding of the 19th ACM Conference on Information and Knowledge Management (CIKM), Tonronto, Oct 2010.

[12] http://dir.yahoo.com

Authors

Akshay Minocha is a student pursuing a Dual degree of B.Tech in Computer Science and MS in Exact Humanities at the International Institute of Information Technology Hyderabad, India. His research interests are opinion mining, sentiment analysis, topic modeling and semantic relatedness.

Navjyoti Singh is a Professor and Head of the Center for Exact Humanities at International Institute of InformationTechnology, Hyderabad, India. After engineering education from Indian Institute of Technology Kanpur in Mechanical Engineering and Nuclear Technology, he shifted to professional research in Philosophy and history of science. His research activity has been focused on theoretical inquiry at the crossroads of Humanities, Science and Indian Analytic traditions, His publications and teaching are in the area of formal ontology, philosophy, theories of arts and cultural computing.