Crowdsourcing slang identification and transcription in twitter language

Benjamin Milde
milde@stud.tu-darmstadt.de

Abstract

This paper describes a case study for finding and transcribing slang and abbreviated words commonly used in twitter language. Using unusual words not found in a reference corpus, we create a candidate list for possible words worth exploring. We design a Human Intelligence Task (HIT) for the popular crowdsouring platform crowdflower.com and pay non-experts to categorize and transcribe our candidate words. With the collected and labeled data we use machine learning to build a detector for slang and abbreviated words. With this detector, we create a new candidate list which is then subsequently crowdsourced. We also build a dictionary of these words, which can be useful for the normalization of twitter texts in other NLP-tasks.

1 Introduction

In recent years, research efforts in understanding and processing microblogging corpora are getting very popular. Twitter, the most famous microblogging platform, is an abundant text corpus, as millions of tweets - microposts on twitter - are written daily and are easily accessible. However the informal nature of most tweets poses new challenges for natural language processing (NLP) algorithms. Slang words, abbreviations and (intended) misspellings of words are common examples of the extremely informal nature of the short texts from tweets, as users frequently abbreviate their posts to fit within the specified 140 character limit (Finin and Tseng, 2007).

In this work, we leverage the power of crowdsourcing to automate human annotation by paid non-experts, to generate a list of labeled out of vocabulary (OOV) words from a 180 million word twitter corpus. This can then be used to train machine learning algorithms to automate the task of categorizing and detecting slang words. We also collect transcriptions and corrections of misspellings for our words, to create a OOV-dictionary which can possibly be useful in twitter language normalization.

2 Related work

Finin et. al. annotated named entities from a large twitter corpus using crowdsourcing and came to the conclusion that this is a reliable and cheap way to delegate the tedious task of hand-labeling (Finin et al., 2010). Han and Baldwin analysed and categorized OOV-words from twitter, but labeled a smaller portion of the data - around 450 words - per hand (Han and Baldwin, 2011). There are also some efforts to normalize twitter mircotexts. Xue et. al. proposed a multi-channel model to normalize the text which was inspired by source channel theory (Xue et al., 2011). The performance of their model was measured in respect to 800 hand normalized sentences. Sood et. al. used a corpus of comments from a social news site and crowd sourcing to improve automated profanity detection using machine learning techniques (Sood et al., 2012). To our knowledge, there are currently no previous works on building a specific dictionary of common misspellings, slang words and other common OOV-words from twitter in significant size using crowd-sourcing.

3 Overview

To obtain a list of candidate words for slang words, abbreviations and common mis-

spellings, we first built a word frequency dictionaries for a large twitter corpus with 25 million tweets, that we collected in summer 2012 from the public twitter timeline. All twitter words not contained in a reference corpus that appear frequently have been collected as candidate words (see Section 4). We also preprocess (see Section 5) the tweets and try to resolve some common misspellings automatically. In a first stage, a Human Intelligence Task (HIT) was designed for the Crowdflower platform to categorize the candidate words by human workers and transcribe them where possible or explain them (Section 6). Using aggregated data from the platform, we train machine learning algorithms to automate some of the categorizing, i.e. we built a detector for slang words and abbreviated/misspelled words (Section 8). We then use the learned model to generate a new list of candidate words, that also include words from the reference corpus: informal words that also appear in the reference corpus and were thus not part of the first candidate list. Figure 1 illustrates this.

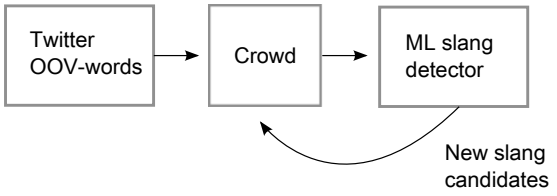

Figure 1: Schematic overview of the approach used in this paper.

4 Candidates for HIT-evaluation

We use the simple English Wikipedia as reference corpus [1]. The choice of the reference corpus will heavily influence the list of candidates: If the corpus is smaller, many brand and proper names and common interjections will also be part of the candidate list. However if the corpus is bigger, as with the simple English Wikipedia, many very well known slang terms already appear in the corpus and are thus not part of the candidate list. For example, the simple English wikipedia has a page to explain the popular slang term 'lol', including a section on variants of the term ('lolz', 'lolololol').

[1] http://simple.wikipedia.org/

Word	Freq.	Word	Freq.
lmao	70785	booze	923
ima	44999	askin	922
idk	43175	bikin	922
hahah	37092	merch	919
tweeting	31835	twitpic	919
hahahaha	29657	ganun	919
imma	28100	rusher	918
retweet	27620	blushes	916
awh	25867	mahone	915
followback	25596	macam	914
smh	22973	dieing	911
follback	22111	uber	910
directioner	20573	bitchy	908
sweetie	19311	ahhaha	906
dnt	19045	heya	906
beliebers	18331	titp	904
omfg	17588	quid	900

Table 1: Most common out of vocabulary (OOV) words from the twitter corpus vs. some less common ones (left / right)

Table 1 shows the most common out of vocabulary (OOV) words obtained in this fashion with their respective frequency from a 200 million twitter corpus and some less frequent ones. Internet and twitter slang and twitter specific vocabulary is more prevalent in higher frequent OOVs (e.g. tweeting, retweet, followback, lmao, omfg, ...), while everyday slang words appear less frequent (e.g. booze,quid,bitchy...)

5 Preprocessing

We prepare the texts from twitter quite a bit to provide good context sentences for the workers. We remove any retweets (tweets starting with RT), to minimize duplicates in our data. We exclude most emoticons and special characters and remove all hyperlinks in the tweets. We filter tweets with less than five English words, as they are possible written in a different language.

Twitter users often repeat ending characters of words, or also consonants in words to emphasize words. This is what Xue et.al. call emotional emphasis (Xue et al., 2011), as these words are intentionally misspelled. For example: loveeeeee → love, nervessss → nerves, welllllllcome → welcome. We resolve OOV-

words with repeating characters by removing one of the repeating characters at a time and looking it up again in the dictionary, until either a word from the reference corpus is found or the original word is labled as OOV-word. Also if the word ends with the letter 's', we remove it and see if the possible singular form of it is in our reference corpus.

We resolved roughly 2.9 million words from our whole twitter corpus to words contained in our reference corpus this way and eliminated quite a few bogus candidate words. This shows how prevelant the usage of this form of alternate spelling on twitter is. Manual inspection showed that most words were resolved correctly in this fashion.

We further remove also all candidate words which contain numbers, concentrating our efforts on slang words with purely alphabetic characters. However, informal words containing numbers can still later be detected with the slang classifier, see discussion in Section 8.

6 HIT Design

We use the Crowdflower [2] platform for our HITs. We designed a relatively big task in which workers had to identify a matching category for a word and additionally transcribe the word for us. Each task had five words to categorize and transcribe, solving one of them as a worker is called 'judgement'.

Context is given in form of four sentences from twitter where the word occurred, because for most words it is quite hard to infer the meaning without context. The word in question is marked with asterisk, unfortunately Crowdflower didn't support to mark words in the csv data with html tags, which would be useful to display the words in bold instead of putting them in asterisks. Figure 2 shows one such judgement from an actual task, as presented to the workers.

We are mainly interested in common misspellings and alternate spellings (gonna → going to, yr → your, ineed → i need) and slang words (slackling, bloke, belieber, dmed ...) [3] which can sometimes be twitter related like 'dmed' for 'direct messaged' or are also used in every day language like the British slang term 'bloke' for 'guy'.

6.1 Gold data

After drafting the task, we let it run for 100 items. We used the standard setting of three judgements per data item, so that there is a certain redundancy. We converted suitable data items to gold items, which are questions which all three workers got right. We used gold items later on as quality control, because they are question for which we already know the answer to. For each gold item we provided an explanation as to why we think the answer is correct, which is displayed when a worker provides a different answer. Crowdflower uses gold question automatically to track the performance of workers and rate their answers with a trust score. If a worker misses to many gold questions, all his judgements become 'untrusted' and are not included in the final results. Vice versa, all judgements from workers which continuously provide correct answers to gold questions are then trusted judgements.

We also made the description a bit clearer, and stated multiple times that shorter answers for the transcription text field are preferred over longer ones, something which many workers did wrong. (ex. they provided the transcription 'Misspelling of laughing' instead of just 'laughing')

7 HIT results

We uploaded the 5000 most frequent candidates in random order from our wordlist. At 7 cent per five judgements we collected 3,700 trusted judgements and 730 untrusted judgements for $150 in total over the course of one week. This resulted in roughly 1900 finished items.

7.1 Agreement between Workers

Crowdflower also provides analytics for the agreement between workers. The agreement on the category is 79.8%, while the agreement on the transcription is 58.4%. This is to be expected, because a freeform textbox leaves a lot more room for variations. For example 'jedhead', the word from example the task

[2] `http://crowdflower.com`

[3] For the curious, 'slackling' is a term for 'to not work for long periods of time' but can also be related to the trend sport slacklining. 'Belieber' is a slang word for the fans of Justin Bieber.

Category	Percent
Abbreviation / Alt. spelling	58%
Slang word	12%
Something else / not sure	12%
Different language	10%
Proper name	6%
Interjection	2%

Table 2: Distribution of categories for the candidate words.

in Figure 2, can be transcribed as 'jedward fans', 'fans of jedward' or 'fans of the Irish pop duo jedward'. Albeit the last one is a bit long, these are all plausible transcriptions for the slang term and mean the same thing, but Crowdflower would see them as being different from each other.

7.2 Category distribution

Table 2 presents the distribution of aggregated categories from workers judgements. It is worth noting, that Han et.al. (Han and Baldwin, 2011) reports a smiliar rate for slang terms in OOV-words in twitter. Abbreviation / Alternate spelling + Slang words make up 70% of our candidate list, which are mainly the words we are interested in to train a classifier later on.

7.3 Costs and worker time

Crowdflower reports an mean judgement time of 29 seconds for trusted judgements and a mean time of 36 seconds for untrusted judgements. We were expecting a lower judgement time and were paying 7 cents for one task with 5 judgements. The real average worker time would equate to a hourly pay of $1.68.

Judgements that had transcriptions with garbage (like 'adfadsfasdf') were rare, untrusted judgements often had very long transcriptions, albeit the clear instruction that a one word transcription is preferred were possible. This could explain the longer average time for untrusted judgements. An example is the word 'ineed', which from context is just a misspelling version of 'i need'. One of the wrong answers, although funny, was that ineed is 'the desperate desire or need to have an apple product'.

term(s)	Probability for class 'slang'
lol	99.9999992%
gonna	1.53%
sup, u	99.99958%, 99.33%
google, beatles	4.27e-08%, 5.78e-08%
about2, gn8, 2getha	98.97%, 84.02%, 99.999910%
1998	99.99999998%
university, research	1.50e-09%, 8.24e-10%
study, slang	1.35%, 5.55e-05%

Table 3: Some example classifications with the slang term classifier.

8 Automated categorization

We built a second word list with words that are contained in our reference corpus and twitter. We filtered out words that did not appear in a smaller corpus of Gutenberg texts [4] and dictionary words. We also filtered words that did not appear in a similar frequency span in the twitter corpus as the candidate slang words so far. In a first experiment we trained several machine learning (ML) algorithms using a two class problem: decide if a word is 'normal' or 'slang + misspelled or abbreviated'. We use the words from the method above as normal words, and all words categorized from workers as slang, misspelled or abbreviated as second category. We also excluded some candidate words in cases where workers had a high disagreement on the category.

As features for the classifier we use TF-IDF vectors of character n-grams directly from the words. Already with the limited data of single words, a significant learning score above baseline (78%) can be achieved. We build a second classifier which uses the context of the words, i.e. 100 randomly chosen tweets where they appear in, with either character n-grams or word n-grams. Figure 3 plots the number of feature vectors used for learning against achieved F1-score. We used 10-fold crossvalidation for all data points. As expected, using context tweets to train the classifier improves performance (92.5% F1 score using all collected data).

Using Logistic Regression (sometimes also called MaxEnt classifier) and context tweets, a well performing slang classifier can be build. Using the probabilistic score for the classifiers

[4] www.gutenberg.org

Word	Freq.	Proba.
lol	637444	0.999999991675
x	329343	0.999999963706
best	281389	0.999932982857
yeah	272439	0.999999569676
xx	250735	0.99999994069
amazing	214343	0.999999672903
omg	197570	0.999999998845
were	152404	0.9999874413
pretty	125081	0.999999993511
babe	109469	0.999999630948
?!	96779	0.999999619389
followers	96596	0.999999998096
followed	94211	0.99999999844
awww	81060	0.99997855482
niall	80221	0.999999999977
lmao	71448	0.999978506868
fucking	61809	0.999999999922
dm	50457	0.999998836712

Table 4: Most frequent informal words from the twitter corpus, as classified with high probability with the slang term classifier.

decision, it can be estimated how likely the classifiers decision is going to be right. This also allows to trade accuracy against recall after the classifier has been trained, by choosing a higher decision boundary on the slang probability.

Table 3 shows some example probabilities for some randomly chosen words. Note that words containing numbers can also be decided on, since the classifier's decision relies solely on the context tweets the word appeared in. Furthermore new words can also be ranked according to their probability from the classifier. We constructed a list of 1000 words that way, containing words that occurred 10^3 to 10^6 times in our twitter corpus and that didn't occur in the first candidate list. Under these words, the most frequent slang term is 'lol', see also Table 4. We then used Crowdflower again to annotate the new candidate list. The final dictionary contains then a mix of pure OOV-words that are informal and frequently used and words that the classifier predicts as being slang while also appearing very frequently. Only entries built from the latter word list contain very common internet slang terms like 'lol' and 'omg'.

9 Conclusion

Crowdsourcing is a cheap alternative to hand-labeling NLP datasets. We showed that the task of identifying and transcribing misspellings and slang words from twitter can be successfully mapped to a crowdsource task, which can be done by non-experts without training. The big advantage over hand-labeling is that it can be scaled to significant data sizes that are otherwise usually not possible. This is important for machine learning algorithms those performance usually also scales with the amount of training data available. This is also the case with our data, when using ML-methods to automate some of the categorisation. The collected data might also be useful to measure the performance of an attempt to automate transcription of OOV-words. Generally, twitter language normalization might profit from a dictionary of common twitter OOV-words.

References

Tim Finin and Belle Tseng. 2007. Why We Twitter : Understanding Microblogging.

Tim Finin, Will Murnane, Anand Karandikar, Nicholas Keller, Justin Martineau, and Mark Dredze. 2010. Annotating Named Entities in Twitter Data with Crowdsourcing. 2010(June):80–88.

Bo Han and Timothy Baldwin. 2011. Lexical Normalisation of Short Text Messages : Makn Sens a # twitter. pages 368–378.

Sara Owsley Sood, Judd Antin, and Elizabeth F Churchill. 2012. Using Crowdsourcing to Improve Profanity Detection. Technical report.

Zhenzhen Xue, Dawei Yin, and Brian D Davison. 2011. Normalizing Microtext.

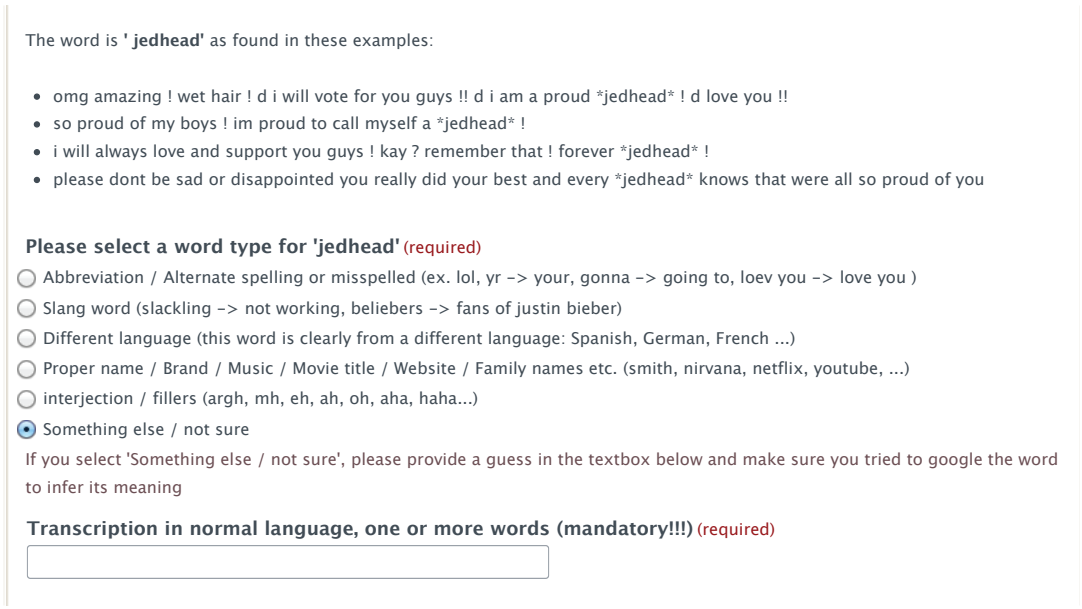

Figure 2: Screenshot of an example task as presented to the workers

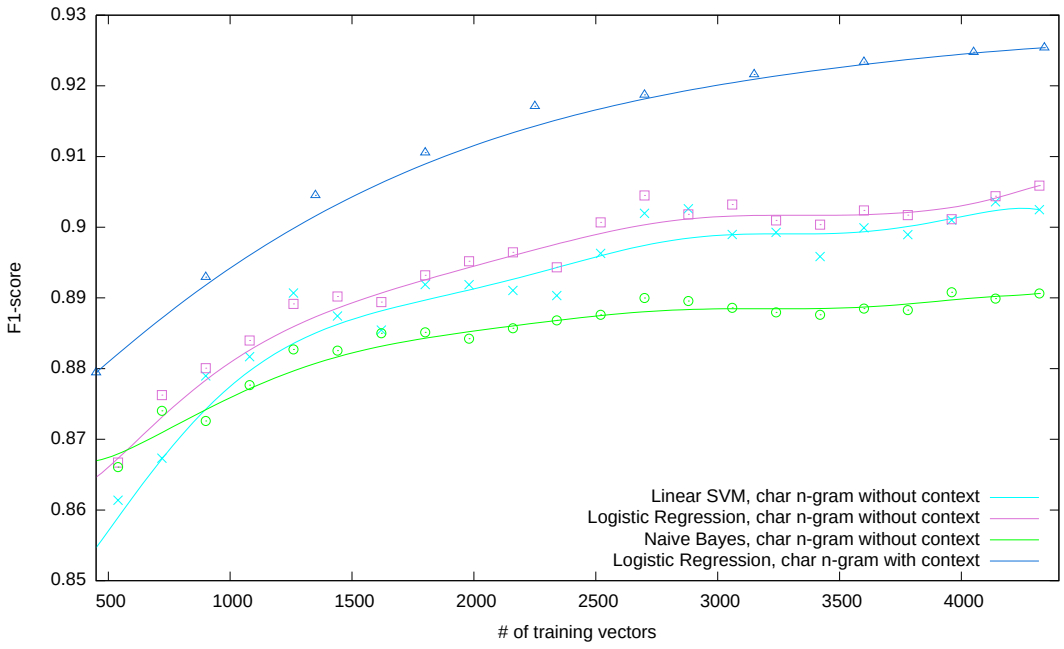

Figure 3: Number of feature vectors used for learning and achieved cross-validated F1-score for the slang word detection. All 3 tested learning algorithms perform similarly well. Naturally, providing context to the model in form of example tweets improves performance (dark blue line). The baseline is roughly 78%.