Proceedings of the Ninth International AAAI Conference on Web and Social Media

Distant Supervision for Tweet Classification Using YouTube Labels

Walid Magdy, Hassan Sajjad, Tarek El-Ganainy, Fabrizio Sebastiani*
Qatar Computing Research Institute, Qatar Foundation, Doha, QA
Email: {wmagdy,hsajjad,telganainy,fsebastiani}@qf.org.qa

Abstract

We study an approach to tweet classification based on distant supervision, whereby we automatically transfer labels from one social medium to another. In particular, we apply classes assigned to YouTube videos to tweets linking to these videos. This provides for free a virtually unlimited number of labelled instances that can be used as training data. The experiments we have run show that a tweet classifier trained via these automatically labelled data substantially outperforms an analogous classifier trained with a limited amount of manually labelled data.

Introduction

Interest in classifying microblogs has increased with the widespread use of microblogging platforms such as Twitter. A major challenge in tweet classification is the fact that manually annotated data are needed to train an effective classifier, which is an expensive and time-consuming task, especially when a large number of classes is used, since a sufficient number of examples per class are required. We here present a novel method for automatically generating a large number of training examples for tweet classification. We do this by leveraging manually labelled data that we automatically obtain from another social medium, YouTube, and using them for training a tweet classifier; in the literature, this is usually called *distant supervision* (Go, Bhayani, and Huang 2009). Specifically, we collect a large set of tweets linking to YouTube videos. Since each such video is manually assigned to one of a predefined set of 18 broad classes at the time of posting, we may attach the class assigned to a video to the tweets that link to it; this automatically creates a large set of labelled tweets that we can then use for training a tweet classifier, which can then be applied to any tweet (i.e., not necessarily containing links to YouTube). The benefits of this method stem from the practically unlimited availability of such training instances. Our experimental results show that our distant-supervision method outperforms common supervised methods that make use of a limited number of manually annotated data.

*Fabrizio Sebastiani is on leave from Consiglio Nazionale delle Ricerche, Italy.

Related Work

Distant supervision has been proposed in the literature for various applications, such as sentiment classification (Go, Bhayani, and Huang 2009), relation extraction (Mintz et al. 2009), topical classification of blogs (Husby and Barbosa 2012), and tweet classification (Zubiaga and Ji 2013). Most such works used distant supervision in order to obtain annotated data for their task from some other annotated dataset. For instance, (Go, Bhayani, and Huang 2009) used the emoticons occurring in tweets as "silver" labels (i.e., as labels with more uncertain status that the ones found in usual "gold" standards) for tweet sentiment analysis. For relation extraction, (Mintz et al. 2009) used textual features extracted from Freebase relations in order to train a relation classifier. (Husby and Barbosa 2012) also used Freebase to obtain labels of Wikipedia articles, and used them for blog post classification by topic. (Zubiaga and Ji 2013) used distant supervision for tweet classification. Their approach consists in assuming that a tweet where a webpage URL occurs is on the same topic as that of the webpage; this is similar to our assumption about tweets linking YouTube. They consider tweets linking to webpages classified under human-edited webpage directories. However, the shortcoming of their approach is that it depends on a human-edited directory which is limited in size and not necessarily up to date. Our proposed method is more robust, since it is not dependent on any manually maintained resource.

Most previous work on tweet classification uses manually annotated training data, which is both expensive and time-consuming (Becker, Naaman, and Gravano: 2011; Kinsella, Passant, and Breslin 2011; Kothari et al. 2013). Moreover, classifiers may need to be updated over time, so as to cope with concept drift and the dynamic nature of social media (Magdy and Elsayed 2014). Therefore, methods that overcome the need for extensive manual annotation are to be preferred.

Distant Supervision for Tweet Classification

More than 4 million tweets in different languages linking to some YouTube video are tweeted everyday[1]. Every video on YouTube is assigned one of 18 pre-defined classes by the user who uploads it. Our approach for collecting labelled

[1]http://topsy.com/analytics?q1=site:youtube.com

tweets is based on the hypothesis that a tweet linking to a YouTube video can be reasonably assigned the same class that the video has been assigned. To validate this hypothesis, we have assigned labels to tweets linking to YouTube videos and used them to train a tweet classifier. We have used the Twitter API[2] with the string "youtube lang:en" to query the stream of English tweets with links to YouTube videos[3]. We have thus collected a set of $\approx$ 19.5 million tweets with hyperlinks to $\approx$ 6.5 million different YouTube videos in a period of 40 days between the end of March and the beginning of May 2014; it is often the case that multiple tweets link to the same video. We have then used the YouTube API[4] to extract the titles and classes of these videos, and have assigned these video classes as labels to the tweets linking them.

The number of tweets per class ranges from 1668 to more than 7 million. Only three classes (Movies, Trailers, and Shows) contain fewer than 100k tweets. To avoid data sparseness, we have merged them with Film&Animation, since these three classes are topically similar. People&Blogs is the default class of YouTube, and is automatically assigned to a video when no class is specified by the user who uploads it; we thus decided to drop this class, since we expect it to be noisy. This process led to 14 classes with $>$100k tweets per class.

We have noticed that the collected tweet set contains large number of retweets and duplicate tweets, i.e., tweets with the same text. We have thus filtered out all the tweets that are retweets or have duplicate text, so as to keep at most one occurrence of each tweet in the dataset; this has the effect of avoiding to train the classifier with repeated examples, which may lead to bias. Moreover, duplicate tweets often contain automatically generated text (e.g., "Just watched video ..."), which can act as noise when training the classifier. This step reduced our dataset size from $\approx$ 19.5 million to $\approx$ 9.2 million tweets only. In the end, the smallest class in our data contains $\approx$ 62k unique tweets.

Model Generation

In the tweet classification literature various types of features have been used for training a classifier. These include Twitter-specific features (Kothari et al. 2013), social network features (Lee et al. 2011), hyperlink-based features (Kinsella, Passant, and Breslin 2011), and standard bag-of-words features, which are the most commonly used (Becker, Naaman, and Gravano: 2011; Lee et al. 2011; Sankaranarayanan et al. 2009). Since feature design is not our main focus in this paper we simply apply a bag-of-words (BOW) approach, where each feature represents a term and the feature value is binary, denoting presence or absence of the term in the tweet. Nonetheless, in the following we discuss two methods for text enrichment that attempt to improve the performance of the BOW approach.

Since tweets are very short and the information contained in them is thus limited, we have applied two different feature enrichment methods. The first method enriches the tweet text in the training data with the title of the linked video. This method is only applicable to our automatically obtained training tweets, since they all link to YouTube, but is not applicable in general to the unlabelled tweets we want to classify, since these may not link to any YouTube video. The second method duplicates the hashtags contained in the tweets and removes the hash character "#" from the second copy, so to allow the terms contained in the hashtags to increase the robustness of the term counts in the texts.

In all our experiments, we applied simple text normalization, which includes case folding, elongation resolution (e.g., "cooooool" $\rightarrow$ "cool"), and hyperlinks filtration. Neither stemming nor stop word removal were applied. We have then applied feature selection, by scoring all features via information gain (IG). All features are ranked according to their IG value for the class, after which a round-robin mechanism (Forman 2004) is applied in which the top n features are selected from each class-specific ranking and then merged to form the final feature space. We select the top 10,000 terms for each class; for 14 classes the theoretically maximum size of the feature space is thus 140,000 features, but the feature space is actually smaller since there is some overlap between the term sets selected for different classes. As a learning algorithm we have used support vector machines; in particular, SVM^{light}.

Experimental Setup

In our experimental setup we have focused on testing the effectiveness of our method at classifying generic tweets, regardless of the fact that they link or not to a YouTube video. We created two test sets: 1) an automatically labelled test set, harvested in the same manner as our training set (the "silver standard"); and 2) a manually labelled test set, consisting of tweets that do not necessarily have links to YouTube videos (the "gold standard").

Silver-Standard Training and Test Sets

From our dataset of automatically labelled tweets (described above) we randomly pick out for testing 1000 tweets for each class, for a total of 14,000 tweets evenly distributed across 14 classes. We refer to this test set as $test_S$ (S standing for "silver"). We consider $test_S$ as a "silver standard", since labels are not verified manually. For the rest of the automatically labelled tweets, we opted to balance the number of tweets in each class by randomly selecting 100,000 tweets from each class, so as to match the number of tweets in the smallest class (Pets&Animals), which contains 98,855 tweets. The final training set thus contains $\approx$ 1.4 million tweets; however, after applying duplicate and retweet filtering, this number reduced to $\approx$ 913k tweets (each class having 60k to 70k examples), which is three orders of magnitude larger than typical training sets used in the tweet classification literature. We dub this dataset $train_S$. We trained SVMs on $train_S$ using a linear kernel; this required a couple of hours on a standard desktop machine.

Gold-Standard Training and Test Sets

We created a second test set (the "gold standard") consisting of manually labelled generic tweets; we dub this test

[2]http://twitter4j.org/en/index.html

[3]This also captures tweets with shortened YouTube urls

[4]http://developers.google.com/youtube/

set $test_G$ (G standing for "gold")[5]. There are two important reasons to have a manually labelled test set. First, our $test_S$ silver standard may be biased in favour of the system trained on $train_S$, because both datasets were sampled from the same distribution (i.e., they were labelled in the same automatic manner) and both consist of only tweets that link to YouTube; instead, the tweets in $test_G$ do not necessarily contain a link to a YouTube video. The second reason is that $test_G$ gold standard can be used for cross-validation experiments, as described below. This will provide a solid baseline for the classifier trained using $train_S$.

To create a manually labelled set, it is difficult to randomly collect tweets covering all 14 classes, since some classes are rare and do not come up often in practice. In order to choose the tweets to label, we thus performed a guided search for each class by using the Twitter API to stream tweets that contain hashtags similar to class names. This was done in the same month in which we collected our automatically labelled training dataset. For example, for the class Autos&Vehicles we collected tweets containing hashtags #autos or #vehicles. This helped us collect a set of tweets that, with high likelihood, had a substantial number of representatives for each of our classes of interest. We randomly selected 200 tweets for each class (based on hashtags), removed the hashtags that relate them with their possible class, and submitted them to a crowdsourcing platform for annotation. For every tweet, we asked at least three annotators if the displayed tweet matches the assumed class or not. Out of 2800 tweets representing 14 classes, only 1617 were assessed by all annotators as matching the assumed class; the number of tweets per class after validation ranged from 84 to 148. This number of training examples is comparable to the numbers used in other studies from the literature (Becker, Naaman, and Gravano: 2011; Kothari et al. 2013; Lee et al. 2011; Sankaranarayanan et al. 2009).

Classification Runs

We have built the following classifiers for our experimentation:

- C_S: trained via distant supervision using $train_S$, which includes $\approx$ 913k automatically-labelled tweets.
- $C_{S(v)}$: same as C_S, with tweet enrichment using the title of the linked video.
- $C_{S(h)}$: same as C_S, with tweet enrichment obtained by adding the terms contained in the hashtags to the text.
- $C_{S(vh)}$: same as C_S, with tweet enrichment obtained by both heuristics above.

The S subscript indicates that all these classifiers have been trained on "silver" labels.

Further to this, we have run 10-fold cross-validation (10FCV) experiments on the 1617 manually labelled tweets in $test_G$. We will then compare the results obtained by C_S and its variants on $test_G$, with the ones obtained by the classifiers generated in these 10FCV experiments; specifically, we will look at the results of

[5]This test set is available for download at http://alt.qcri.org/~wmagdy/resources.htm.

	P	**R**	F_1	**A**
C_S	**0.583**	0.573	0.564	0.574
$C_{S(v)}$	0.574	0.567	0.560	0.568
$C_{S(h)}$	0.582	**0.575**	**0.568**	**0.576**
$C_{S(vh)}$	0.576	0.569	0.562	0.571

Table 1: Classification results on the silver-standard test set ($test_S$). **Boldface** indicates the best performer.

- C_G: this is not actually a single classifier but 10 different classifiers, as generated within the 10FCV; that is, the results of applying C_G to $test_G$ will be the union of the 10 folds, each of them classified within one of the 10 experiments;
- $C_{G(h)}$: similar to C_G, but with tweet enrichment obtained by adding the terms contained in the hashtags to the text. Enrichment using the title of the linked video is not applicable, since most of the tweets in $test_G$ do not link to YouTube.

Here, the G subscript indicates that all these classifiers have been trained on "gold" labels.

The main objective of our experiments was to examine if any of the C_S classifiers can achieve comparable (or even better) results with respect to the C_G classifiers, which would support our hypothesis and would also show the value of freely available labelled data. Different setups of the C_S classifier were examined for both test sets to find the optimal configuration that achieves the best results.

Evaluation

The evaluation measures we used in this task are "macroaveraged" precision (**P**), recall (**R**), F_1 (popularly known as the "F-measure"), and accuracy (**A**). That is, all of these measures were calculated for each class separately, after which the average was computed across the 14 classes. Since our test sets contain fairly balanced numbers of examples from each class, these macroaveraged figures are very similar to the corresponding "microaveraged" ones (where classes more frequent in the test set weigh more), which are then not reported explicitly. Moreover, accuracy is indeed a reasonable measure of classification effectiveness (this is unlike the cases of severe imbalance, when accuracy is unsuitable).

Results

Table 1 and Table 2 report the classification results obtained on the "silver" test set $test_S$ and on the "golden" test set $test_G$. All results in both tables display a relatively good effectiveness for a single-label 14-class classification task, where random classification would achieve (given the approximately balanced nature of our test sets) an expected classification accuracy of $\approx 7\%$.

Table 1 shows that the "enhanced" setups of the C_S classifier did not lead to noticeable improvement. Enriching the training tweets with the title of the linked video even led to a small degradation in performance, while enriching the representation of the tweets by duplicating hashtags achieved only slightly better results.

	P	R	F_1	A
C_G	0.511	0.506	0.507	0.518
$C_{G(h)}$	0.541	0.534	0.537	0.546
C_S	**0.619**	**0.588**	**0.579**	**0.611**
$C_{S(v)}$	0.570	0.566	0.548	0.586
$C_{S(h)}$	0.600	0.583	0.573	0.605
$C_{S(vh)}$	0.578	0.567	0.551	0.588

Table 2: Classification results on the gold-standard test set ($test_G$). **Boldface** indicates the best performer.

The results in Table 1 suggest that our idea of using YouTube labels for training a tweet classifier is a reasonable one. Nevertheless, the main experiments are those reported in Table 2, which reports results obtained on a truly gold standard. Here, all different setups of C_S achieved better performance than all different setups of C_G, which confirms that our method for inexpensively acquiring large numbers of automatically annotated training examples is more effective than the (more expensive) method of labelling a limited number of training examples.

Regarding the best setup for the training data, we noticed that hashtag term duplication improved the performance in the case of C_G, but did not lead to any improvement for C_S. The limited number of training examples used for generating C_G can be the reason for this result: here some enrichment to the representation of the training examples seems to help, unlike in the case of C_S, which was trained via a large number of training examples and does thus not require further enrichment. The best result achieved for C_S and its variants was $A = 0.611$ and $F_1 = 0.579$ (which was obtained for C_S itself), which is substantially higher than the best result achieved for C_G and its variants ($A = 0.546$ and $F_1 = 0.537$, which was obtained for $C_{G(h)}$).

Conclusion

We have experimentally demonstrated the effectiveness of a "distant supervision" approach to tweet classification, consisting in automatically obtaining labelled data from one social media platform (YouTube) and using them for training a classifier for another such platform (Twitter). This generates a large amount of freely available labelled training data, thus overcoming the need for manual annotations.

An extended version of this paper (Magdy et al. 2015) discusses further experiments aimed at testing the robustness of our approach (a) with a smaller number of more general classes, (b) with resource-poor languages, and (c) with respect to time drift. Furthermore, it explores the minimum size of silver training data that could be used to outperform the manually labeled one (Magdy et al. 2015).

References

Becker, H.; Naaman, M.; and Gravano:, L. 2011. Beyond trending topics: Real-world event identification on Twitter. In *ICWSM 2011*.

Forman, G. 2004. A pitfall and solution in multi-class feature selection for text classification. In *ICML 2004*.

Go, A.; Bhayani, R.; and Huang, L. 2009. Twitter sentiment classification using distant supervision. Technical report, Stanford University.

Husby, S. D., and Barbosa, D. 2012. Topic classification of blog posts using distant supervision. In *Proceedings of the EACL Workshop on Semantic Analysis in Social Media*.

Kinsella, S.; Passant, A.; and Breslin, J. G. 2011. Topic classification in social media using metadata from hyperlinked objects. In *ECIR 2011*.

Kothari, A.; Magdy, W.; Darwish, K.; Mourad, A.; and Taei, A. 2013. Detecting comments on news articles in microblogs. In *ICWSM 2013*.

Lee, K.; Palsetia, D.; Narayanan, R.; Patwary, M. M. A.; Agrawal, A.; and Choudhary, A. 2011. Twitter trending topic classification. In *OEDM 2011*.

Magdy, W., and Elsayed, T. 2014. Adaptive method for following dynamic topics on Twitter. In *ICWSM 2014*.

Magdy, W.; Sajjad, H.; El-Ganainy, T.; and Sebastiani, F. 2015. Bridging social media via distant supervision. In *CoRR abs/1503.04424*.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *ACL/IJCNLP 2009*.

Sankaranarayanan, J.; Samet, H.; Teitler, B. E.; Lieberman, M. D.; and Sperling, J. 2009. TwitterStand: News in tweets. In *GIS 2009*.

Zubiaga, A., and Ji, H. 2013. Harnessing Web page directories for large-scale classification of tweets. In *WWW 2013*.