Using Linear Classifiers for the Automatic Triage of Posts in the 2016 CLPsych Shared Task

Juri Opitz
Computational Linguistics
Heidelberg University
69120 Heidelberg, Germany
opitz@cl.uni-heidelberg.de

Abstract

The 2016 CLPsych Shared Task was to automatically triage posts from a mental health forum into four categories: *green* (everything is fine), *amber* (a moderator needs to look at this post), *red* (a moderator urgently needs to look at this post) and *crisis* (the person might hurt himself or others). The final results for the task revealed that this problem was not an easy task. I chose to treat the problem as a text categorization task using a system composed of different Support Vector Machines (SVMs) in a one-vs-rest setting. This approach was straight-forward and achieved good performance in the final evaluation. The major difficulty was to find suitable features and feature combinations.

1 Approach

Treating the problem as a multi-class text categorization problem motivated the usage of linear SVMs. SVMs promise good regularization in high dimensional spaces (as this and most other text spaces are) and have demonstrated empirical success for many kinds of text categorization problems (Joachims, 1998), (Manevitz and Yousef, 2002).

To map the posts into vector space, suitable features had to be chosen. I experimented with three different types of features. First and foremost the traditional *bag-of-ngram features*: In information retrieval and document classification tasks, documents are often treated as bag-of-words or bag-of-ngrams. One distinct dimension represents each distinct n-gram. These features simply assumed a boolean value of 1 at index i in the feature vector, when the document in question contained the n-gram represented by i, and 0 otherwise. I created 1,2 and 3 grams based on the available data and discarded those which appeared in less than twelve documents. This resulted in a maximum number of 65287 features. The ngrams were drawn from the tokenized main message text and the title (if the title did not contain "Re:", indicating that the title-text might be from another user).

The second category of features were *user features*: These described i.a. the ratios of *green*, *amber*, *crisis* and *red* labels in a user's history and the label of his last post (if there was one). Motivation were assumptions like: given a user posted a *crisis* post, chances are higher that the next post of this user is also a *crisis* post.

Also manually created were *post features*: these features described the number of kudos and the time of the post (in a categorical way). Motivation: Was the post created very late in the night? This could indicate sleep problems, which again could indicate a *crisis* or *red* label.

For each label a different SVM was trained. The best feature combination for each of the four SVMs was searched on 250 development posts (these were cut off at the end of the 947 training posts). Of course, it was intractable to validate all different possible combinations of features. I chose to focus on the following options:

1. all features
2. 1-grams
3. 1,2-grams

Proceedings of the 3rd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pages 162–165,
San Diego, California, June 16, 2016.

4. 1,2,3-grams

5. 1,2 grams using 1k,5k..., 40k of the 2 grams

6. 1,2,3 grams using the best of 5. and 1k,5k,..., 15k of the 3 grams

7. The best of the above combinations with user and/or post features

Taking also the 20 different options for the SVM regularization parameter into account, more than 400 parameter combinations for each label were checked. The four SVMs representing the labels achieving the best label-wise F1-measure were chosen for the multi-class classification.

The decision for the final label was based on the soft outputs of the decision functions (dot-product of weights and feature values) of the four one-vs-rest classifiers. Here I chose to experiment with two options: 1., argmax and 2., train another classifier (used AdaBoost) on the output scores of the four SVMs as a "meta-classifier".

2 Results

Table 1 shows the F1-scores on the development and test set of the best combination of parameters found on the development set. For each single label (vs. rest) and for the final multi-class classification where the single binary classifiers were combined to make a final decision. The evaluation measure of the Shared Task was Macro F1, averaged over *amber*, *red* and *crisis*. The argmax decisions of the soft

label	feature option	F1 Dev	F1 Test
green	1,2 grams	0.88	0.89
amber	1,2 grams	0.60	0.62
red	$1{,}2_{1k}{,}3_{1k}$ grams	0.48	0.48
crisis	$1{,}2_{1k}{,}3_{1k}$ grams	0.37	0.0
all, argmax	-	0.44	**0.37**
all, AdaBoost	-	0.34	0.31

Table 1: Results of the best parameter options found on the development set. In the final multi class classification, F1 means Macro F1 averaged over all labels but *green*.

SVM-outputs outperformed the AdaBoost decisions by 10% on development and 6% on test. For *green* an F1 score of 0.89 was achieved. All labels but "crisis" yielded better scores on test set. The significant drop in performance from the development data (44% Macro F1) to the test data (37% Macro F1) mainly originated in the could-not-be-worse performance for finding the *crisis* posts (37% F1 development, 0% test).

3 Analysis

3.1 Why the total fail at labelling *crisis*?

Achieving 0.0 F1 for the label *crisis* had a very negative impact on the final Macro F1 measure. A possible explanation of the bad performance for *crisis* is indicated by Figure 1. With respect to the ratio of *crisis*, both train- and development set are not representative for the held-out test set. Indeed, in the test set, there was only one *crisis* in 241 test samples. As the final evaluation measure was Macro F1,

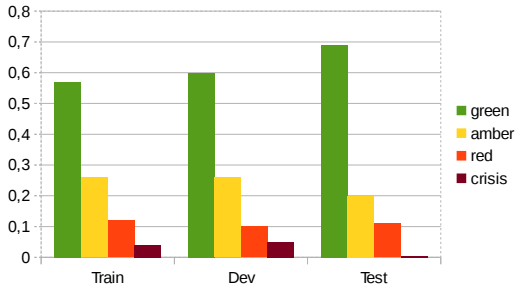

Figure 1: Label distirbutions in the different data sets used. From left to right: *green*, *amber*, *red* and *crisis*.

this was the major reason for the heavy drop in performance on the test data. Finding the only positive sample out of 241 negative ones without making to many guesses is very difficult. This again makes it very likely for recall (and hence F1) to be zero. With more guesses, the chances of finding this one sample may be still small while the precision (and hence F1) for *crisis* drops (and probably also the scores of the three other labels). With $F1 = 0$ for one out of three labels, the Macro F1 was already bounded by 0.67. The best of my systems (Macro F1 $= 0.37$) fired once on *crisis* and missed (the true *crisis* post was labelled *red* - maybe not the worst of an error). Another system I submitted fired 12 times on *crisis*, but missed it every time. In fact, none of the five systems I submitted was able to find this needle in a haystack.

Two things I find important to conclude from that:

1. Macro F1 was an evaluation measure bringing a "harsh" punishment for mislabelling one specific sample.

2. a not-so-good F1 Macro score does not necessarily imply a not-so-good system. As the F1 for one out of three labels was always zero, classifying the other two non-*green* labels worked better (*amber* 0.62 F1 and *red* 0.48 F1).

3.2 The manually designed features did not work well

In the first section I proposed two types of feature sets, which intuitively made sense for me.

These features were designed manually and originated from motivations like: A user who posted a *crisis* post before might be more likely to have another *crisis* in his next post. *Post features*, also i.a. described the time a post was submitted. However, as it turned out, these features led to over-fitting problems as indicated by figure 2. The figure describes functions of the SVM regularization parameter C with regard to 1. a feature vector containing only uni-grams, 2. post and user features appended and 3., only user features appended and 4., only post features append. It is clearly visible that the usage of the hand-crafted features led to problems on the unseen development data.

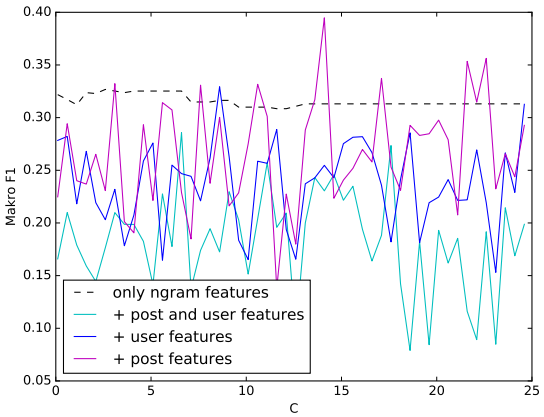

Figure 2: Performance of different feature combinations on the unseen development set. Severe over-fitting problems occurred when including the manually designed features.

3.3 Phrases with high weights assigned

The ranking of features by their respective squared weights can be interpreted as metric of feature relevance (Guyon et al., 2002). High weights (their squared value to take negative weights into account) influence the output of the decision function by tendency more than low weights.

Table 2 displays, for each possible label, the phrases with the highest weights (positive and negative). This analysis i.a. shows that emoticons were of importance for the discrimination of posts. For example,
:-) is negatively correlated with *crisis*, *red* and *amber* and positively correlated with *green*. 72% of 65026 posts contained emoticons. Further interpretation of the high weighted phrases is left to the reader.

4 Conclusion and Outlook

The approach I proposed was to train one SVM for each label and do the final vote between the four SVMs with an argmax of the soft-outputs of their respective decision functions. The system led to above-median performance in the final evaluation of the Shared Task. The approach is also straight forward, the major difficulty being the search for good features and feature combinations there are very many of these possible. The best performing features turned out to be bag-of-phrase features: unigrams plus partially bi- and tri-grams. The SVMs appeared to cope well with high dimensions (as expected), but not so well with the manually designed features (as not expected). These features led to problems on unseen data. It is very likely that there exist features or sets of features which are able to further enhance the automatic triage of posts, making the SVM approach all in all a promising technique for this task.

As the results of all systems of all participants on the held-out test set show, the automatic triage of posts in a psychology forum is not an easy task. I think that deciding whether a post is to be labelled *red* (a moderator needs to look at the post as soon as possible and take action) or *crisis* (the author might hurt himself) is often not only difficult to decide for machines, but also for humans themselves (maybe even for psychological experts, especially without knowing the author in person). Thus, for further ex-

label	phrases
green (-)	don't, cant, just, I'm, negative, want, help, don't know, feeling, not, everything, do, scared, know, anymore, help me, guess, feel, don't want, has, nothing, :-(
green (+)	be lonely, you, :-), your, :-D, awesome, proud, you are, love, 1, we, you can, good, for, hope, well, you're, if you, by, hey, morning, for you, how, 2, some, there
amber (-)	:-), be lonely, your, you are, there, 1, day, I'm so, can, love, well, hope, anymore, will, :-D, 3, sorry, hey, out, how, if you, into, you have, awesome, coming, you can, friend
amber (+)	don't, me, help, think, but, other, not, thanks, about, I'm, all, yeah, just, help me those, have, put, negative, services, thank, anxious, lot, there's, don't have, thank you, isn't, guess
red (-)	for, thanks, you, about, :-), hope, too, good, proud, :-D, an, put, think, one, awesome, still, me but, thought, but don't, make, phone, week, other, sitting
red (+)	breathe, :-(, passed, empty, ..., family, worse, should, feeling so, hospital, anymore, things are, disappointment, incapable, shit, afraid, please, cant, practically, through this, identical, can not, failed
crisis (-)	you, my, your, I've, :-), some, was, been, with, its, people, things, all, would, have, we, are, them, love, see, there, said, much, after, not, good, someone, thing
crisis (+)	can't, life, just, for me, just want, back, negative, home, want, I'm so, thought about, me, sorry for, anymore, worth, everything, feel like, die, harm, sorry, self, bad, unsafe, don't know, tips, useless

Table 2: Features with the highest positive (+) and negative (-) weights for each label. Emoticons: :-) = happy emoticon, :-D = very happy emoticon, :-(= sad emoticon.

amination and comparison of systems in this classification problem, I would suggest to also consider other evaluation measures, which take not only *error yes-no* into account, but also the severity of an error. With respect to a real world application, a *crisis* post labelled *red* should not be as severe of an error as handing out a *green* label: *red* and *crisis* (by definition) are very close neighbors, *crisis* and *green* are opposites.

References

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. 2002. Gene selection for cancer classification using support vector machines. *Mach. Learn.*, 46(1-3):389–422, March.

Thorsten Joachims. 1998. Text categorization with suport vector machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning*, ECML '98, pages 137–142, London, UK, UK. Springer-Verlag.

Larry M. Manevitz and Malik Yousef. 2002. One-class svms for document classification. *J. Mach. Learn. Res.*, 2:139–154, March.