Predicting Psychological Health from Childhood Essays
The UGent-IDLab CLPsych 2018 Shared Task System

Klim Zaporojets, Lucas Sterckx, Johannes Deleu, Thomas Demeester and **Chris Develder**
IDLab, Ghent University - imec
firstname.lastname@ugent.be

Abstract

This paper describes the IDLab system submitted to Task A of the CLPsych 2018 shared task. The goal of this task is predicting psychological health of children based on language used in hand-written essays and socio-demographic control variables. Our entry uses word- and character-based features as well as lexicon-based features and features derived from the essays such as the quality of the language. We apply linear models, gradient boosting as well as neural-network based regressors (feed-forward, CNNs and RNNs) to predict scores. We then make ensembles of our best performing models using a weighted average.

1 Introduction

The goal of the CLPsych 2018 shared task is to predict the psychological health of children based on essays and socio-demographic control variables. The provided data stems from the National Child Development Study (NCDS) which followed a number of people born in a single week of March 1958 in the UK (Power and Elliott, 2005). The psychological health of this group of individuals was monitored in intervals of several years. At the age of 11, participants were asked to write an essay describing where they saw themselves at age 25. Simultaneously, their psychological health was evaluated by their teachers based on metrics defined by the Bristo Social Adjustment Guides (BSAG) (Shepherd, 2013).

Given the written essays and social control variables (gender and social class), CLPsych participants are to predict three types of BSAG scores: (i) `total` BSAG score, (ii) the `depression` BSAG score, and (iii) the `anxiety` BSAG score. In order to predict these scores, participants are allowed to use the social control variables next to the features extracted from the essays themselves.

Our system uses several types of features: bag-of-word and bag-of-character features, features derived from lexicons and term lists, and features based on text statistics (see Section 3.2 for more details). Using these features, we apply several types of regressors: linear models, gradient boosting and neural-network based models. For each of the regressors, we explore different combinations of features to predict each of the BSAG scores. Subsequently, these models are combined using weighted average ensembling. Two sets of predictions were made: the first one is based on the single best models, a second uses an ensemble of models for each of the three scores (`depression`, `anxiety` and `total` BSAG scores).

Our ensemble of models gives a competitive result, positioning our system on the second place with only 0.01 points under the winner of this shared task. We think that this good performance is mostly due to the different nature of our individual models which complement each other when ensembled.

The remainder of this paper is organized as follows: Section 2 describes the shared task in more detail. Section 3 presents the features used by the regressors. Section 4 describes regressors and the general methodology of our approach. Section 5 describes results we obtained during development on our internal validation set and on the real test set. Finally, we summarize our findings and present future directions in Section 6.

2 Task and Data

Input for task A consists of essays written by 11-year-old children describing where they see themselves at age 25, as well as several social control variables:

1. **Gender**: gender of the participant child.

Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic, pages 119–125
New Orleans, Louisiana, June 5, 2018.

2. **Social Class**: the job hierarchy of the father of the participant child. The domain comprises 6 values representing different job categories: starting with professional and managerial occupations and ending with unskilled occupations.

3. **Essay**: content of the essay written by the participant child. Originally, the essays were hand-written and later transcribed in digital format. The average length of the essays is 225 characters.

The goal of shared task A is to predict the current psychological health of the children. Psychological health is measured using scores assigned by teachers of the children following metrics defined in the BSAG. These guides score the total psychological health using 12 different syndromes (depression, anxiety, hostility, etc.). CLPsych shared task A requires participants to predict three scores:

1. **Total**: the sum of all the BSAG scores of all the different syndromes.

2. **Depression**: the BSAG score related to the depression syndrome.

3. **Anxiety**: the BSAG score related to the anxiety syndrome.

Participants are given a training set consisting of essays from 9,217 children with corresponding input variables and BSAG scores.

3 Features

In this section, we present features used by our models, and experiment with a number of different categories of feature extraction.

3.1 Lexical features

We use bag-of-n-gram features both on word- and character-level. The latter provides robustness to the spelling variation found in children's writing. For word-level we experiment with n-grams for n ranging from 1 to 4. At character-level, we experiment with 3- up to 6-grams. These one-hot encodings are weighted using TF-IDF.

3.2 Feature Engineering

Next to the sparse bag-of-n-grams representations of the essays, we apply several manually designed features.

Social control features These features are given as input in the data and consist of the *gender* and *social class* of the participants. In order to be used in regressors, we encode these features as one-hot vectors.

Lexicon-based features We experiment with features based on two lexicons: the Linguistic Inquiry Word Count (LIWC) described in (Pennebaker et al., 2015) and the DepecheMood (Staiano and Guerini, 2014). The LIWC is a psycholinguistic lexicon that allows to measure the emotional health of individuals by providing a set of term categories related to different mental states. In our experiments we use all 73 (partly overlapping) psychological word categories found in the LIWC dictionary.

Similarly, DepecheMood is a lexicon consisting of 37k different words (verbs, nouns, adjectives and adverbs). Each of the words has weights associated to the following 8 mental states: afraid, amused, angry, annoyed, don't care, happy, inspired and sad. In our experiments, we calculate the average of TF-IDF weights for these categories. These TF-IDF weights are already given inside DepecheMood lexicon and are originally calculated on articles from `rappler.com` based on Rappler's *Mood Meter* crowdsourcing.

Textual statistics features We extract a number of features describing several characteristics of the essays:

- Total number of words
- Average sentence length
- Average word length
- Ratio of spelling mistakes
- Ratio of different words
- Number of words not recognized (illegible) when transcribing the essays from hand-written to digital form.

Sentiment features We reason that the participants' psychological health can partially be detected by evaluating the essay in a positive-negative sentiment spectrum. We use the pre-trained sentiment classifier from (Cagan et al., 2014).[1] We hypothesize that individuals with good psychological health will tend to use more

[1]The python library can be found at: `https://pypi.python.org/pypi/sentiment_classifier`

positive expressions than individuals with high scores in any of BSAG syndromes.

Language model features Coming from the intuition that mental state may be related to the development of language skills, we include two language model features. Our primary language model feature is the average perplexity of the essays, as it is an often used metric to score the general language quality and coherence of the texts. As a secondary feature, we include the fraction of out-of-vocabulary tokens over the entire essay, with respect to the Penn Treebank data. We use the word-level `AWD_LSTM` language model trained on the Penn Treebank, presented by Merity et al. (2017).

4 Models Description

We train a variety of different regression models predicting the three aforementioned BSAG scores. We include simple linear models as well as gradient boosted trees and neural network-based models. Our best performing models are subsequently combined using ensembling. As a general rule, we try to select different model function types in order to achieve lower correlation between predictions from the different types of models.

4.1 Linear Models

We experiment with two types of linear regressors: support vector machines (SVMs) and ridge regression. Linear models are trained on two sets of features.

1. *Lexical features* based purely on the text of the essays (see Section 3.1). Here we use TF-IDF weighted bag-of-word features as well as character features.

2. *Designed features* through feature engineering (see Section 3.2).

To avoid overfitting, we tune the regularization parameter α on a validation set. For SVM models this parameter corresponds to squared L2 penalty. For ridge models, it corresponds to the strength of L2 regularization term. We experiment with selecting models based on lowest RMSE error as well as the ones with highest disattenuated Pearson correlation score.

4.2 Gradient Boosting

We apply gradient boosted tree regressors using XGBoost (Chen and Guestrin, 2016) trained on the *designed features* (see Section 3.2). To train XGBoost models, we use early stopping by evaluating on a validation set with 10,000 estimators and a logarithmic scale grid search of learning rate from $10\mathrm{e}{-5}$ to $10\mathrm{e}{+5}$. We experiment with RMSE as well as disattenuated Pearson correlation scores as criterion to perform early stopping.

4.3 Feed-Forward Neural Networks

As a second type of non-linear models, we use feed-forward neural networks (FFNNs). We train FFNNs on our *designed features* (see Section 3.2) expecting that the introduced non-linearity will complement the results of previous models. Our FFNN architecture consists of 3 hidden layers with $\tanh$ activation units. We apply dropout regularization of 0.5 between each of the layers. The network has a total of 223 input features in the first layer and 256 neurons in each of the three intermediate hidden layers. We experiment with optimizing for three loss functions:

1. **Mean squared error (MSE)**: this is our default choice used for most of the regressors.

2. **Huber**: Huber loss is less sensitive to outliers which are present in BSAG scores (high BSAG scores for few individuals).

3. **Pearson correlation**: we experiment with correlation loss because it is directly related to the metric used to evaluate the model performance by organizers of shared task A.

4.4 Neural Sequence Encoders

We include two types of models based on neural networks which encode the essays to a low dimensional representation, after which a score is predicted using a feed-forward layer. Essays are encoded using two of the most prevalent neural network architectures for modeling of sequences, convolutional neural networks (CNN) and recurrent neural networks (RNN).

Pretrained Embeddings The first layer of NN architectures embeds the one-hot token representations into a vector space of lower dimensionality, which it then fine-tuned through backpropagation. We initialize the embedding layer using embeddings from dedicated word embedding techniques Word2Vec (Mikolov et al., 2013) and Glove (Pennington et al., 2014). This proved to be essential for good performance of the neural sequence models.

CNNs We apply the architecture proposed by Kim (2014) which consists of a single convolutional layer with multiple filter sizes, followed by one feed-forward layer over the three-dimensional score vector. We use filters of size 3, 4, 5, 6 and 7 and vary the amount from 64 to 512 filters for each size.

RNNs We experiment with two types of RNNs to encode the essays, long short-term memory networks (LSTM) (Hochreiter and Schmidhuber, 1997) and gated recurrent units (GRU) (Cho et al., 2014). After encoding the essay in forward and backward direction, we use the concatenated sequences of hidden states to predict scores. To reduce the dimensionality of this representation, we use max-pooling and self-attention to obtain the final essay encodings (Lin et al., 2017). We experiment with single-layer bidirectional RNNs with hidden state vectors of 64, 128 and 256 dimensions. A fully connected layer of 32 and 64 nodes is used to predict scores.

4.5 Model Ensembling

To produce weighted averages of predictions, we use the *forward model selection* algorithm that greedily selects the combination of models that maximizes the disattenuated Pearson correlation on the evaluation set. We use 100 iterations and choose the best model if there is no improvement after 30 iterations on the evaluation set.

5 Experiments

5.1 Training Details

We divide the training set of 9,217 individual evaluations into two parts: (i) a *train set* consisting of 7,835 examples, and (ii) an *evaluation set* consisting of the rest (1,382 examples). For SVM, Ridge and XGBoost models, we select the best models on our evaluation set using two metrics: (i) models with the lowers RMSE score, and (ii) models with the highest disattenuated Pearson correlation score. For feed-forward neural nets we experiment with three loss functions: (i) MSE, (ii) Huber, and (iii) disattenuated Pearson correlation. Finally, for neural sequence encoders, we use MSE as a loss function. In order to build an ensemble of models, we further subdivide our evaluation set in two equal parts:

1. **Validation set**: the validation set is used to choose the best combination of models using forward model selection (see Section 4.5).
2. **Test set**: the test set is used to verify that a given model combination does not overfit the evaluation set.

Before extracting features from the text of input essays, we perform basic text preprocessing functions: lowercasing, removal of punctuation and extra spaces. For TF-IDF and embedding lexical features we also remove the stop words. Additionally, we use TextBlob (`https://textblob.readthedocs.io/`) in order to correct the spelling mistakes.

Feed-forward neural networks are trained for 100 epochs with learning rate of $1e{-}5$. We also apply a weight decay (L2 penalty) of $1e{-}6$ on the Adam optimizer. Most of the models converge after training approximately for 20 epochs with a batch size of 8.

CNN and RNN models are trained with Adam and early stopping based on disattenuated Pearson correlation. Models converge after training for approximately 10 epochs, with batch size 32. For RNN models we apply a dropout with probability 0.3 on the embedding layer and the output layer. For both CNN and RNN models we apply dropout on the fully connected layer with probability 0.15.

5.2 Results

Table 1 summarizes results for different models on our validation set. For linear models, we notice that SVM models are sensitive to optimizing towards RMSE or disattenuated correlation score. We also observe that SVM models have lower disattenuated correlation scores for the anxiety BSAG metric. For feed-forward neural nets, use of the Huber loss obtains the best performance. We speculate that this is because this method is not as influenced by outliers as other loss functions. The rest of the models has approximately similar performance.

A large boost in performance is observed when creating ensembles of models. We gain between 0.02 and 0.04 points on our validation set for the disattenuated correlation metric. We don't see this improvement on RMSE and MAE metrics since our ensemble is greedily built to optimize for Pearson correlation between predicted and ground truth results.

Table 2 shows the weight combinations of our ensemble for all three objectives to predict. We only add best RMSE models for Ridge, SVM and XGBoost regressors. The reason is that adding

	Anxiety			Depression			Total		
	RMSE	MAE	Diss. R	RMSE	MAE	Diss. R	RMSE	MAE	Diss. R
Development									
Ridge RMSE (lex. feat.)	1.222	0.784	0.2100	1.460	1.076	0.3493	8.356	6.472	0.4532
+Diss. R (lex. feat.)	1.225	0.782	0.2160	1.497	1.138	0.4046	8.643	7.043	0.4783
+RMSE (des. feat.)	1.218	0.773	0.2136	1.446	1.073	0.3781	8.272	6.280	0.4719
+Diss. R (des. feat.)	1.218	0.773	0.2136	1.446	1.073	0.3781	8.272	6.280	0.4719
SVM RMSE (lex. feat.)	1.260	0.690	0.1129	1.517	1.046	0.2542	8.643	5.940	0.4526
+Diss. R (lex. feat.)	1.360	**0.573**	0.1220	1.811	1.007	0.4094	9.047	6.091	0.4624
+RMSE (des. feat.)	1.241	0.723	0.1227	1.470	1.005	0.3736	8.683	6.920	0.3418
+Diss. R (des. feat.)	1.352	**0.573**	0.1026	1.897	1.694	0.3508	8.449	6.019	0.4473
XGBoost RMSE (des. feat.)	1.221	0.769	0.1982	1.452	1.081	0.3624	8.302	6.257	0.4600
+Diss. R (des. feat.)	1.225	0.768	0.1997	1.458	1.073	0.3579	8.312	6.343	0.4557
CNN RMSE loss	1.221	0.772	0.2053	1.473	1.128	0.3863	8.390	6.488	0.4556
RNN RMSE loss	1.228	0.769	0.1630	1.444	1.070	0.3938	8.271	6.206	0.4805
FFNN MSE loss (des. feat.)	**1.216**	0.775	0.2253	1.445	1.073	0.3837	**8.219**	6.310	0.4945
+Huber loss (des. feat.)	1.246	0.697	0.2294	1.483	0.997	0.3921	8.486	**5.884**	0.5000
+Diss. R loss (des. feat.)	1.288	0.616	0.2010	1.675	**0.959**	0.3488	11.556	7.743	0.4290
Ensemble	1.223	0.743	**0.2660**	**1.435**	1.035	**0.4246**	8.252	6.047	**0.5191**
Test Runs									
Submission 1 (Ensemble)	1.119	**0.476**	**0.1946**	**1.393**	**1.004**	**0.4536**	**7.843**	5.691	**0.5667**
Submission 2 (Single Model)	**1.022**	0.697	0.1760	1.403	1.019	0.4192	8.134	**5.688**	0.5140

Table 1: Results on internal evaluation set for best individual models; "lex. feat." refers to the lexical features (see section 3.1), whereas "des. feat." are the designed features (see section 3.2).

	Anxiety	Depression	Total
Ridge RMSE (lex. feat.)	0.2698	0.0625	0.1825
Ridge RMSE (des. feat.)	-	-	-
SVM RMSE (lex. feat.)	-	-	-
SVM RMSE (des. feat.)	0.0688	0.1563	0.0584
XGBoost RMSE (des. feat.)	0.2646	0.0469	0.0949
CNN RMSE loss	0.0423	0.1250	-
RNN RMSE loss	-	0.3281	0.2993
FFNN MSE loss (des. feat.)	-	0.2813	0.0365
FFNN Huber loss (des. feat.)	0.3545	-	0.3285
FFNN Diss. R loss (des. feat.)	-	-	-

Table 2: Weights of the ensemble components.

models that had the best performance on Pearson disattenuated correlation score decreased significantly the RMSE and MAE scores of the ensemble. How these models can still be added without producing this drop in performance is left for future work.

The bottom rows of Table 1 show the results of our two submissions on the official CLPsych test collection. We obtain a considerable improvement using ensembles of models with respect to our single best model submission, resulting in the overall second best submission. We speculate that this is because of different score distributions produced by dissimilar models used in this work. This generates low correlation of individual model predictions, which results in better ensembles. We were surprised to see that disattenuated correlation score was several points higher in depression and total BSAG predictions than on our internal validation set. The anxiety score, on the other hand, is considerably lower. Further analysis is needed to understand these differences, and to investigate the impact of the individual types of hand-designed features.

6 Conclusion and Future Work

In this paper we briefly described the Ghent University – IDLab submission to the CLPsych 2018 shared task A. We found that linear models, gradient boosting as well as neural network based models perform similarly but produce different models that, when combined, can increase the performance on the test set considerably.

For future work, we plan to conduct a careful error analysis (e.g. ablation tests) and examine the best ways to design our train-validation splits in order to decrease the score difference between the validation and test sets. We also plan to experiment with more sophisticated ways of ensembling and stacking techniques.

We consider that in the end, most of the success of this task comes down to designing a good

set of features. In particular, one of the features we didn't explore is topic modeling. Additional features can be obtained from topic model distributions as they provide positive results on similar tasks described in (Resnik et al., 2015) and (Cohan et al., 2016).

Finally, another direction we want to explore consists of using word and phrase embeddings, pre-trained on a corpus of individuals with psychological disorders. Some work has already been done to gather this kind of corpus from online resources (Twitter and Reddit in particular) (Yates et al., 2017) and (Coppersmith et al., 2015). We hypothesize that we can get a significant improvement by initializing our CNN and RNN models with these embeddings.

Acknowledgments

We are grateful to Giannis Bekoulis for fruitful discussions on model cross-validation and for providing resources, support and encouragement.

Human Subjects Review

This study was evaluated by the Ethics Committee of the faculty of Psychology and Educational Sciences of Ghent University, which concluded that ethical approval was not needed for the research conducted for this manuscript.

References

Tomer Cagan, Stefan L Frank, and Reut Tsarfaty. 2014. Generating subjective responses to opinionated articles in social media: an agenda-driven architecture and a turing-like test. In *Proceedings of the Joint Workshop on Social Dynamics and Personal Attributes in Social Media*, pages 58–67.

Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA. ACM.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Arman Cohan, Sydney Young, and Nazli Goharian. 2016. Triaging mental health forum posts. In *Proceedings of the Third Workshop on Computational Lingusitics and Clinical Psychology*, pages 143–147.

Glen Coppersmith, Mark Dredze, Craig Harman, Kristy Hollingshead, and Margaret Mitchell. 2015. CLPsych 2015 shared task: Depression and PTSD on Twitter. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 31–39.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2017. Regularizing and Optimizing LSTM Language Models. *arXiv preprint arXiv:1708.02182*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. 2015. The development and psychometric properties of LIWC2015. Technical report.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Chris Power and Jane Elliott. 2005. Cohort profile: 1958 British birth cohort (national child development study). *International journal of epidemiology*, 35(1):34–41.

Philip Resnik, William Armstrong, Leonardo Claudino, and Thang Nguyen. 2015. The University of Maryland CLPsych 2015 shared task system. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 54–60.

Peter Shepherd. 2013. Bristol social adjustment guides at 7 and 11 years. *Centre for Longitudinal Studies*.

Jacopo Staiano and Marco Guerini. 2014. DepecheMood: a lexicon for emotion analysis from crowd-annotated news. *arXiv preprint arXiv:1405.1605*.

Andrew Yates, Arman Cohan, and Nazli Goharian. 2017. Depression and self-harm risk assessment in online forums. *arXiv preprint arXiv:1709.01848*.