Linguistic Metadata Augmented Classifiers at the CLEF 2017 Task for Early Detection of Depression

FHDO Biomedical Computer Science Group (BCSG)

Marcel Trotzek[1], Sven Koitka[1,2], and Christoph M. Friedrich[1]

[1] University of Applied Sciences and Arts Dortmund (FHDO)
Department of Computer Science
Emil-Figge-Str. 42, 44227 Dortmund, Germany
mtrotzek@stud.fh-dortmund.de, sven.koitka@fh-dortmund.de, and
christoph.friedrich@fh-dortmund.de
[2] TU Dortmund University
Department of Computer Science
Otto-Hahn-Str. 14, 44227 Dortmund, Germany

Abstract. Methods for automatic early detection of depressed individuals based on written texts can help in research of this disorder and especially offer better assistance to those affected. *FHDO Biomedical Computer Science Group* (BCSG) has submitted results obtained from five models for the CLEF 2017 eRisk task for early detection of depression that are described in this paper. All models utilize linguistic meta information extracted from the texts of each evaluated user and combine them with classifiers based on *Bag of Words* (BoW) models, *Paragraph Vector*, *Latent Semantic Analysis* (LSA), and *Recurrent Neural Networks* (RNN) using *Long Short Term Memory* (LSTM). BCSG has achieved top performance according to $ERDE_5$ and F_1 score for this task.

Keywords: depression, early detection, linguistic metadata, paragraph vector, latent semantic analysis, long short term memory

1 Introduction

This paper describes the participation of *FHDO Biomedical Computer Science Group* (BCSG) at the *Conference and Labs of the Evaluation Forum* (CLEF) 2017 eRisk pilot task for early detection of depression [22, 23]. BCSG submitted results obtained from four different approaches and a fifth, additionally optimized variation of one model for late submission. These models as well as the findings concerning the dataset are described in this paper and an outlook on possible improvements and future research is given.

2 Related Work

It is known that depression often leads to a negative image of oneself, pessimistic views, and an overall dejected mood [2]. Accordingly, previous studies have shown that depression can have certain effects on the language used by patients. A study among depressed, formerly-depressed, and never-depressed students [36] came to the conclusion that depressed individuals more frequently used the word "I" as well as negatively connoted adjectives. Similarly, an analysis of Twitter messages has shown that users suffering from depression used the words "my" and "me" much more frequently than others [29], while a Russian speech study found an increased usage of past tense verbs and pronouns in general. Findings like these have been used, for example, to create the *Linguistic Inquiry and Word Count* (LIWC) tool [39] that allows to analyse the psychological and social state of an individual based on written texts.

A similar task using Twitter posts was organized at the CLPsych 2015 conference [9] without the early detection aspect: Participants were asked to distinguish between users with depression and a control group, users with *Post Traumatic Stress Disorder* (PTSD) and a control group, as well as between users with depression and users with PTSD. Promising results were reported using topic modeling [35] and rule-based approaches [31]. It was also investigated how a set of user metadata features can be utilized and combined with a variety of document vectorizations in an ensemble [33].

3 Dataset

The dataset presented in the CLEF 2017 eRisk pilot task consists of text contents written by users of www.reddit.com, which is a widely used communication platform for creating communities called *subreddits* that cover all kinds of topics[3]. Specifically, there is a very active community in the subreddit /r/depression[4] for people struggling with depression and similar subreddits for other mental disorders exist as well. The registration of a free account using a valid mail address and a public user name is necessary to create content, while reading is possible without registration, depending on the subreddit. Users can post content as link (using a title and either a URL or an image), as text content (using a title and optional text), or as comment (using only the text field and no title).

The given dataset contains all three kinds of content written by 887 users and 10 up to 2,000 documents per user. Table 1 gives a summary of some basic characteristics of the training and test split. The task's goal is to classify which of these users show indications of depression by reading as few of their posts as possible in chronological order. Each document contains a timestamp of publication, the title, and the text content, while title or text can be empty. There also exist 91 cases of documents with both an empty title and text. The URL or image of link entries is not provided in the dataset. The number of unique n-grams

[3] https://redditblog.com/2014/07/30/how-reddit-works-2/, Accessed on 2017-04-24
[4] http://www.reddit.com/r/depression, Accessed on 2017-04-24

contains all tokens with more than one alphabetical character (and the word "I") that occur in at least two documents, also including numbers, emoticons, and words that contain hyphens or apostrophes.

Table 1. Characteristics of the training and test datasets.

	Training	Test
Users	486	401
Depressed/Non-depressed	83/403	52/349
Documents	295,023	236,371
Comments (empty title)	198,731	168,708
Links/One-liners (empty text)	84,288	57,561
Empty documents	46	45
Avg. documents per user	607.04	589.45
Avg. characters per document (title + text)	172.19	182.29
Avg. unigrams per document (title + text)	29.80	32.27
Unique unigrams	73,501	69,587
Unique bigrams	606,608	526,380
Unique trigrams	778,694	673,048

3.1 Corpus Analysis

After examining the general characteristics of the given dataset, a detailed analysis of the text contents is necessary to get an insight into promising features and the specific properties of the domain. In order to find the most interesting n-grams of the given corpus, *Information Gain* (IG) or expected *Mutual Information* (MI) was calculated. In case of binary classification tasks, the information contained in each feature is given as [25, p. 272]:

$$I(U;C) = \sum_{e_t \in \{0,1\}} \sum_{e_c \in \{0,1\}} P(U = e_t, C = e_c)\, log_2 \frac{P(U = e_t, C = e_c)}{P(U = e_t)P(C = e_c)} \quad , \tag{1}$$

with the random variable U taking values $e_t = 1$ (the document contains term t) and $e_t = 0$ (the document does not contain term t) and the random variable C taking values $e_c = 1$ (the document is in class c) and $e_c = 0$ (the document is not in class c).

Similar to the previously described selection of unigrams in the corpus, IG was calculated without stopword removal for all uni-, bi-, and trigrams that can be found in at least two documents. The obtained scores were then used to find the 100 features with the highest IG of the corpus as well as the 100 features with the highest IG that occur more often in the *depressed* class, which is both shown in Fig. 1.

Fig. 1. Uni-, bi-, and trigrams with highest information gain for the whole corpus (left) and after excluding words that occur more often in the *non-depressed* class (right). In both text clouds, larger font size corresponds to higher information gain.

Both analyses give an interesting insight into the corpus that confirm previous research results described in the related work section. Comparing the two word clouds shows that the first person singular pronouns *I*, *me*, and *my*, which are frequently contained in documents of both classes, have the highest IG seen individually and are then found in some of the most important bi- and trigrams of the *depressed* class. The most important features of this class are, as could be expected, centered around depression and anxiety, while especially relationships (e.g. *boyfriend*, *husband*, *partner*, *best friend*), treatment (e.g. *therapist*, *psychiatrist*, *medication*), and look (e.g. *acne*, *skin*, *makeup*, *alpha hydrox*) can easily be identified as frequent topics and are often combined with personal or possessive pronouns. Interestingly, although the sad emoticon *:-(* is part of the top features in the *depressed* class, the happy emoticons *:-)* and *:)* occur even more frequently in this class and have a higher IG. The frequent combinations as in "*thank you :)*" point to the conclusion that this is often a reaction to thoroughly helpful conversations.

When examining the text data further, it becomes evident that the posts sometimes include quotes taken from messages of other users. This could be misleading for classification tasks since the quoted user might show indications of depression, while the actual author of this message might not or vice versa. Luckily, quotes seem to be unfrequent and can be identified to some extend because they are always indented by a single space, do not contain line breaks, and are preceded and followed by an empty line. There is no way to distinguish them from similarly indented one-line paragraphs by the actual author. By using a regular expression, 4,266 quotes can be found in the training data and 4,423 in the test data. For all models described in this paper, the prefix *quote_* was added to each token within a quote to make them distinguishable from the same words written by the actual author.

3.2 Hand-crafted User Features

In addition to different document vectorization methods, a set of hand-crafted features has been derived from the text data and was used in all approaches. Several text statistics have been calculated and compared between the class of depressed and non-depressed users in the given dataset. The most promising features are displayed in Fig. 2 as box plot for each class. All features have been calculated as mean over all texts of the same user. In addition to the already mentioned counts of personal and possessive pronouns, past tense verbs, and the word *I* in particular, four standard measures for text readability have been calculated for the text content, namely *Gunning Fog Index* (FOG) [14], *Flesch Reading Ease* (FRE) [12], *Linsear Write Formula* (LWF)[5] [8], and *New Dale-Chall Readability* (DCR) [10, 7]. Interestingly, while FOG, LWF, and DCR calculate a higher complexity for texts by depressed users (with values based on school years in the United States), FRE also calculates a higher score, corresponding to lower complexity in this case.

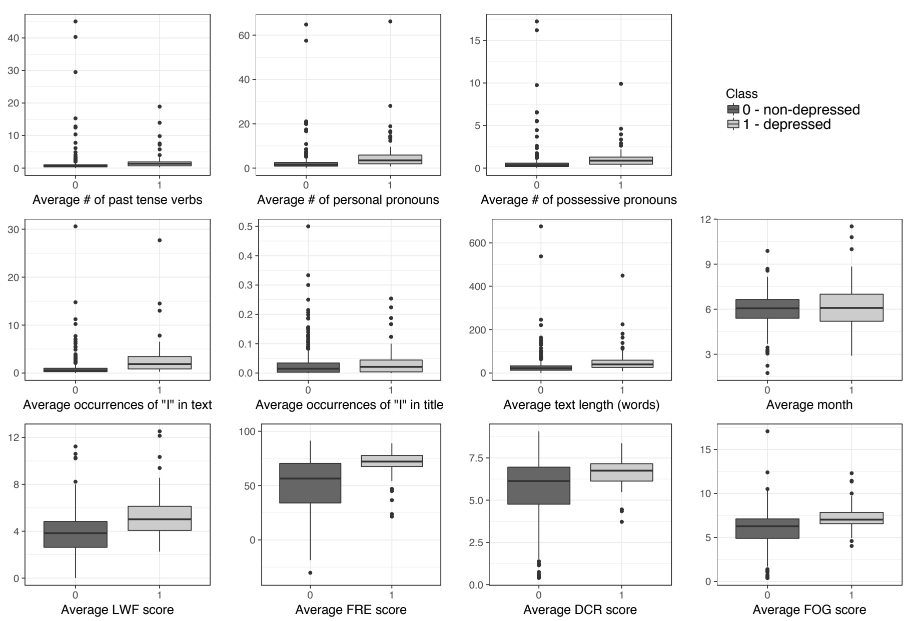

Fig. 2. Boxplots of text features for both classes per user in the eRisk training dataset.

[5] originally developed by the U.S. Air Force without any available references

The average of the months in which all texts of a user have been submitted was included based on the hypothesis that depressive symptoms can be intensified in the winter months. This is difficult to observe in the given dataset, since the age of the available texts depends on how frequently a user has posted due to the limitation to the last 2000 writings per user. Users with many and frequent writings therefore tend to have more samples from early summer 2015 (when the collection was created), while less frequent writers provide a more uniform distribution of texts over all months. Additionally, five features have been created for the users that simply count the occurrences of some very specific n-grams in all their documents. This ensures that some of the strongest indicators of depression can still be identified easily even when using averaged document vectors or just a large amount of documents. These features were used in boolean form by all described models and count the following terms without regard to case:

- The chemical and brand names of common antidepressants available in the United States (e.g.: *Sertraline* or *Zoloft*) obtained from WebMD[6]
- Explicit mentions of a diagnosis including the word *depression* (e.g.: "*I was diagnosed with depression*" or "*I've been diagnosed with anxiety and depression*")
- The term "*my depression*"
- The term "*my anxiety*"
- The term "*my therapist*"

The mentioned terms have been picked carefully only from the training documents and have been designed to capture only statements referring to the personal situation of the author with the exception of the antidepressants. They could be extended for future research to include a more comprehensive list of medications or more general expressions of diagnosis (e.g. also including the terms "major depressive disorder" or "MDD"). Although the selected terms are not primarily helpful for early predictions, they are strong indicators to find already diagnosed individuals, which is important for the given task as well. It would also be interesting to include additional statistical features like the number of adjectives, adverbs, noun phrases, positive and negative emotions, and similar, as done for example by LIWC. Figure 3 displays the correlation of all user features without scaling and also includes the label information, where a higher value corresponds to the *depressed* class. It shows that all features are at least slightly correlated to the information whether a user is depressed or non-depressed.

[6] http://www.webmd.com/depression/guide/depression-medications-antidepressants - Accessed on 2017-05-07

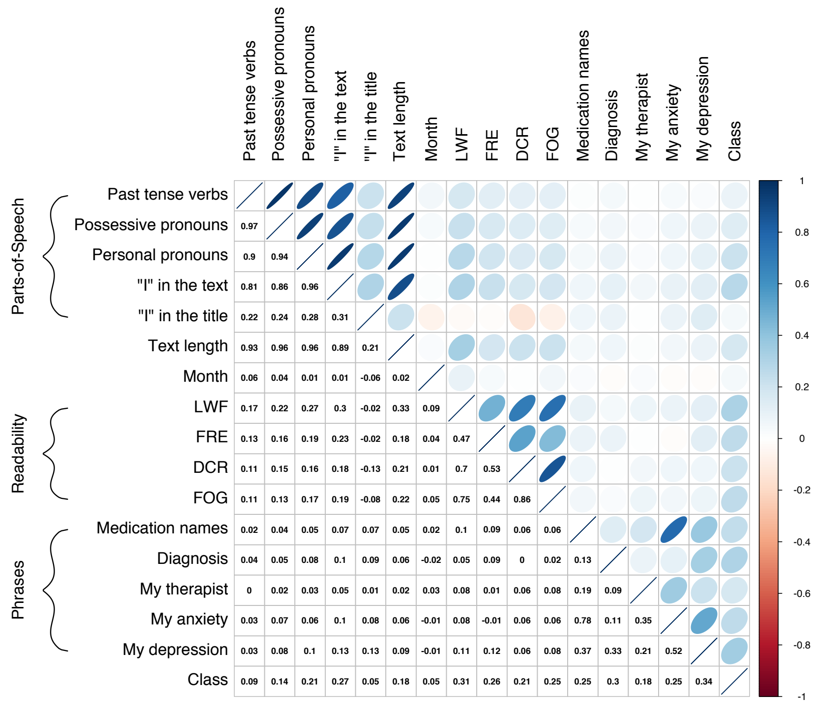

Fig. 3. Correlation matrix of all user features including the class information (non-depressed/depressed).

The findings for this specific dataset confirm that texts by individuals suffering from depression indeed contain more pronouns and especially the word "I". Their texts are also slightly longer and more complex according to three of the four text complexity measures. This likely represents the difference between average users, who often post a large amount of short statements, and those who discuss problems and may even be looking for help.

4 Chosen Models

Two conventional document vectorization models as well as three models utilizing *Long Short Term Memory* (LSTM) [16], a layer architecture for *Recurrent Neural Networks* (RNN) [13] specialized on sequences of data, have been used for the given task. One of these models also employs *Latent Semantic Analysis* (LSA) [11] as dimensionality reduction step. All models have been optimized by 5-fold cross validation on the training data using F_1 score before the submission for the first chunk of test data and were not modified at a later point. The same applies for the described prediction thresholds that were also chosen by cross validation to submit predictions each week. The only exception is the final model BCSGE, which was used to get more time for optimization: For the first nine weeks, no predictions were submitted for this model, so only the predictions using all documents at once in the last week were scored. All models use a concatenation of the text and title field of each document as input and do not treat text and title separately. Identified quotes within text contents have been modified by adding a prefix to each quoted word as described earlier, while the tokenization step includes words, numbers, and emoticons as described in section 3.

4.1 Bag of Words Ensemble - BCSGA

The first model utilizes an ensemble of *Bag of Words* (BoW) classifiers with different term weightings and n-grams. The term weighting for bags of words can generally be split into three components: a *term frequency component* or local weight, a *document frequency component* or global weight, and a *normalization component* [37]. A general term weighting scheme can therefore be given as [40]:

$$t_{t,d} = l_{t,d} \cdot g_t \cdot n_d \quad , \tag{2}$$

where $t_{t,d}$ is the calculated weight for term t in document d, $l_{t,d}$ is the local weight of term t in document d, g_t is the global weight of term t for all documents, and n_d is the normalization factor for document d. A common example would be using the *term frequency* (tf) as local weight and the *inverse document frequency* (idf) as global weight, resulting in tf-idf weighting [37].

All ensemble models use cosine normalization (l^2-norm) for n_d but varying local and global weights. The first one uses a combination of uni-, bi-, tri-, and 4-grams obtained from the training data: the 200,000 $[1-4]$-grams with the highest IG as given by Equation 1 are selected and their raw term frequency is used as local weight, while their IG score is used as global weight. The second BoW utilizes a modified version of tf, namely *augmented term frequency* (atf) [40], multiplied by idf:

$$atf\text{-}idf(t,d) = \left(a + (1-a)\frac{tf_t}{\max(tf)}\right) \cdot \log\frac{n_d}{df(d,t)} \quad , \tag{3}$$

with $\max(tf)$ being the maximum frequency of any term in the document, the total number of documents n_d, and the smoothing parameter a, which is set to 0.3 for this model. This BoW, as well as the third one, contains all unigrams of the training corpus. The local weight of the third model consists of the *logarithmic term frequency* ($logtf$) [30] and the global weight is given by *relevance frequency* (rf) [20], which can be combined as:

$$logtf\text{-}rf(t,d) = (1 + \log(tf)) \cdot \log_2\left(2 + \frac{df_{t,+}}{\max\left(1, df_{t,-}\right)}\right) \quad , \tag{4}$$

where $df_{t,+}$ and $df_{t,-}$ is the number of documents in the *depressed*/*non-depressed* class that contain the term t. The final model of this ensemble uses the handcrafted user features described in section 3.2.

All three bags of words and the hand-crafted features were each used as input for a separate logistic regression classifier. Due to the imbalanced class distribution, a modified class weight was used for these classifiers similar to the original task paper [22] to increase the cost of false negatives. It was calculated for the *non-depressed* class as $1/(1+w)$ and for the *depressed* class as $w/(1+w)$, with $w_1 = 2$, $w_2 = 6$, $w_3 = 2$, and $w_4 = 4$ in the order as the different models have been described above. The final output probabilities were calculated as unweighted mean of all four logistic regression probabilities. Each week, this ensemble predicted any user with a probability above 0.5 as *depressed* and users below 0.15 as *non-depressed*, while in the final week all users with a probability equal to or less than 0.5 were predicted as *non-depressed*.

4.2 Paragraph Vector - BCSGB

The second model is based on document vectorization by using *Paragraph Vector* [21], sometimes referred to as *doc2vec*, similar to the previously published *word2vec* [26, 27] on which it is based. While *word2vec* is used to train embedded word vectors from a large text corpus, *Paragraph Vector* learns vector representations for sentences, paragraphs, or whole documents. It was also found that *Paragraph Vector* can work better for smaller corpora than *word2vec*, which potentially makes it a viable option for this task. The two neural network architectures for each of these methods are all based on the probabilistic *Neural Network Language Model* [3].

For the Paragraph Vector classification of eRisk users, two separate models have been trained based on the training documents using the Python implementation in *gensim 1.0.1* [34]:

1. A *Distributed Bag of Words* model with 100 dimensional output, 10 training epochs, a context window of 10 words, negative sampling with 20 noise words, no downsampling, a learning rate from 0.025 to 1e−4, and all words contained in the documents.
2. A *Distributed Memory* model using the sum of input words with 100 dimensional output, 10 training epochs, a context window of 10, hierarchical softmax, downsampling of high-frequency words with 1e−4, a learning rate from 0.025 to 1e−4, and all words contained in the documents.

The output vectors of these two models were concatenated, as recommended by the developers [21], resulting in a 200 dimensional vector per document. Text content and title of the documents have again been concatenated and each of the resulting texts was used as separate input to Paragraph Vector. Test documents were vectorized by using an inference step that only outputs a new document vector and leaves all network weights fixed.

Finally, the average of all documents by each user was calculated to obtain the average topic of everything the user has written. Figure 4 shows a two-dimensional representation of the averaged training document vectors calculated by t-SNE [24]. Even after a reduction to only two dimensions, there is at least one clearly visible cluster of *non-depressed* users and a rather noisy cluster of *depressed* users.

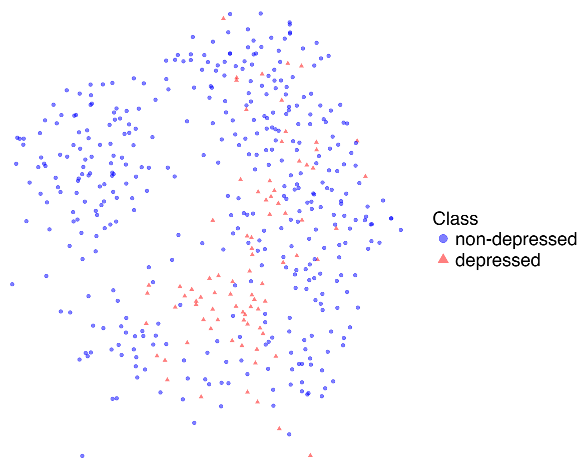

Fig. 4. Plot of the t-SNE reduced averaged document vectors per user for the Paragraph Vector model (BCSGB).

A logistic regression classifier was trained on the 200-dimensional averaged document vectors, using the same class weight equation as in the previous model with $w = 4$. The calculated class probabilities were again averaged with the probabilities obtained from the logistic regression based on the hand-crafted user features. Since this model depends more on the number of documents it has been trained on, the final predictions were based on the probability as well as the number of documents written by the user to prevent too many false positives. *Depressed* predictions were submitted for probabilities from 0.6 with at least 20 documents, 0.7 with at least 10 documents, and all probabilities above 0.9, while *non-depressed* predictions required a probability below 0.1 with at least 20 documents, 0.05 with at least 10 documents, or a probability below 0.01.

4.3 LSTM with LSA Vectors - BCSGC

This and the following two models are based on a Tensorflow [1] neural network approach using an LSTM layer. By using sequences of text documents as input, the LSTM network allows to learn a general context of each user's documents while processing them in chronological order. All three LSTM models also use the hand-crafted user features as an additional meta data input and merge them with the LSTM output in a fully connected layer. This again ensures that these features are not lost after document vectorization and averaging. A final softmax layer was used to produce the actual output probabilities, the softsign function [4] was chosen as activation for the LSTM cell, and dropout was added to prevent overfitting. The training steps of this and the following two LSTM models utilized Adam [19] to minimize the cross-entropy loss.

For this first LSTM approach, LSA was used to reduce the BoW vectorized documents to a viable number of dimensions based on *Singular Value Decomposition* (SVD). All documents were first transformed into a BoW by selecting only the 10,000 unigrams with the highest IG and using their term frequency multiplied by their IG as term weighting. LSA was then used to reduce these document vectors to 100 dimensions, which retained 90.32% of the original variance in the training dataset. To obtain an equal sequence length for all users that is viable as network input, the document sequences were modified to have a length of 25 documents: For users with fewer documents, zero vectors were appended, while two randomly selected consecutive document vectors were averaged for longer sequences, until the maximum length was reached. Adam was then used with a fixed learning rate of 1e−4, 64 units were added to the LSTM cell, a dropout keep probability of 80% was applied, and the network was trained for 300 epochs.

Similar to the previous model, prediction thresholds were based on the network's output probability and the number of documents. *Depressed* predictions required a probability above 0.5 and at least 20 documents, above 0.7 and at least five documents, or above 0.9, while *non-depressed* predictions were submitted for probabilities below 0.05.

4.4 LSTM with Paragraph Vectors - BCSGD

This fourth model utilized the same LSTM network as described for the previous one with identical parameters, except for a number of 128 hidden units in the LSTM cell and a training duration of 170 epochs. For the input sequences, documents were vectorized based on the two concatenated Paragraph Vector models of the second approach. Again, the resulting sequences of 200-dimensional document vectors were modified to have a unified length of 25. The model was configured to submit *depressed* predictions for any user with a probability above 0.3 and at least 50 documents, above 0.4 and at least 20 documents, or above 0.7, while probabilities below 0.01 resulted in a *non-depressed* prediction.

4.5 Late LSTM with Paragraph Vectors - BCSGE

To have some additional time for model optimization and to compare the impact on the ERDE score, the fifth model was not used to submit any predictions until the last week. It is identical to the fourth model but uses two new, 200-dimensional Paragraph Vector models that were trained on both training and test documents. This is an unsupervised method that uses only text documents without any label information. Also, this model uses a second fully connected layer before the softmax layer, *Rectified Linear Unit* (ReLU) activation [15] for both fully connected layers, a weight decay factor of 0.001 for all weights in the network, exponential learning rate decay from 1e−4 to 1e−5, a dropout keep probability of 70% for LSTM outputs, 128 hidden units in the LSTM, and was trained using batches of 100 users over 130 epochs. The document sequence length was again unified to 25 and a minority oversampling that duplicates each *depressed* user in the training input was used to counter the class imbalance. The final network architecture for this model is displayed in Fig. 5, where m_u represents the meta data for a single user u and $x_{u,t}$ is the sequence of input documents written by this user. In the final week, predictions obtained from this model were submitted based on the same thresholds that were used for the previous one.

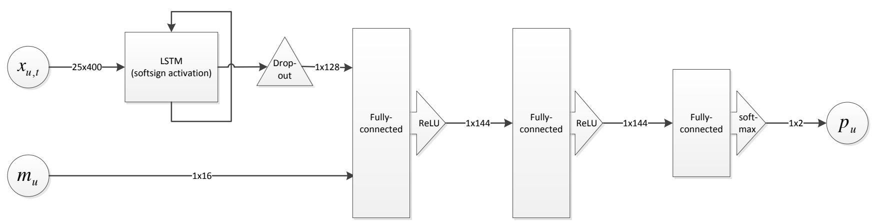

Fig. 5. Network architecture of the final LSTM model for BCSGE.

5 Results

Before discussing the official task results, analyzing the amount of correctly classified depressed individuals using the five BCSG models can give a first insight into the classification performance. The cumulative number of *depressed* predictions and actual true positives per model and week is shown in Fig. 6. A horizontal line marks the total number of 52 *depressed* samples in the test set for reference. It becomes evident that there is still a lot of room for improvements. Although each model is able to detect a growing number of depressed users over the ten weeks, the proportion of false positives is large and the number of total true positives ranges between 24 and 38 of the 52 depressed users in the test set. Most true positives were found by the fifth model but at the cost of nearly as much false positives. This could at least partially be influenced by finding better prediction thresholds.

The final submissions to the CLEF 2017 early risk detection pilot task were scored using the $ERDE_5$ and $ERDE_{50}$ score for early detection tasks defined by the organizers as well as F_1 score. The scores and the underlying precision and recall values of all models have been published [23] and are visualized in Fig. 7. It shows the evaluation results of all eight participants and their up to five different models. The highlighted models of BCSG consistently achieved positions in the first ranks and even the fifth model was ranked in the top half according to both $ERDE$ scores by only submitting a prediction in the last week. The achieved $ERDE$ scores for this task cannot be compared to the previously published results by the organizers [22], since the documents had to be processed in weekly chunks for the task and it was not possible to submit predictions before processing a complete chunk. The best results of BCSG could be achieved by using the BoW model BCSGA (first in F_1 and second in $ERDE_{50}$) and the Paragraph Vector model BCSGB (first in $ERDE_5$), with the LSTM models close behind.

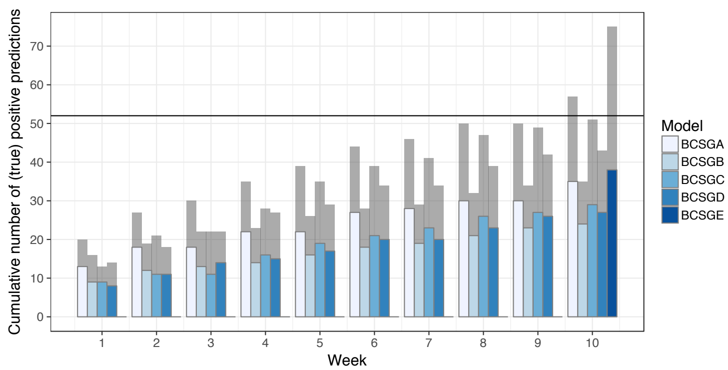

Fig. 6. Cumulative number of *depressed* predictions (blue plus gray bars) and proportion of true positives (blue bars only) per model after each week of the task. A horizontal line marks the 52 *depressed* samples in the test data.

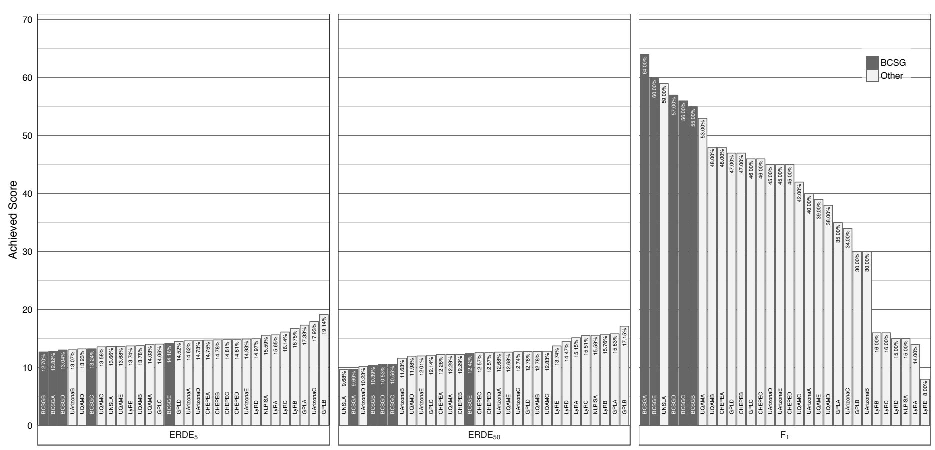

Fig. 7. Official results of the eRisk pilot task in terms of $ERDE_5$, $ERDE_{50}$ and F_1 score. The results of BCSG are highlighted. This plot is best viewed in electronic form.

Since results for BCSGE are only available for the last week, it was evaluated again for all weeks after the golden truth file was published. For this ex post analysis, separate Paragraph Vector models were trained using the training data and already released test data for each week. If BCSGE had been used from the first week, the results would have been 16.01% in $ERDE_5$, 9.78% in $ERDE_{50}$, and 0.46 in F_1. While this $ERDE_{50}$ score would have been the third best overall, the other scores show that this is still not well optimized and there are too many false positives. Future work will be used to examine the effect of hand-crafted features and preprocessing methods on the prediction results. A quick ex post analysis using the first two models BCSGA and BCSGB has shown that the selected hand-crafted features at least had a slightly positive effect (13.04% in $ERDE_5$, 9.75% in $ERDE_{50}$, and 0.63 in F_1 for BCSGA without hand-crafted features), with the exception of the $ERDE_5$ score for BCSGB, which would have been marginally better without hand-crafted features in contrast to a much worse F_1 score (12.67% in $ERDE_5$, 10.76% in $ERDE_{50}$, and 0.37 in F_1 for BCSGB).

6 Conclusions

The pilot task for early detection of depression has highlighted a variety of challenges posed by this area of research. These challenges are not limited to the task of distinguishing actual clinical depression from normal dejected mood as well as other, more or less related mental disorders like anxiety disorders, PTSD, or bipolar disorder. In the context of online platforms, there are also several other frequent false positives that could be observed in this task: relatives of depressed individuals and therapists offering advice can easily be mistaken for *depressed* cases when giving too much weight to single words or phrases. Drug users (which might indeed be an accompanying factor of depression [18, 6]) and authors posting fictional stories could regularly be spotted as false positives. On the other hand, there are cases of individuals who post hundreds of very ordinary comments but suddenly start expressing their feelings and talk about their depression. Such cases would be easier to predict by models that treat each document separately instead of using the whole history of a user.

The final results show that all chosen approaches are generally suitable for early detection of depression and all of them are of interest for future research. Due to the promising results using Paragraph Vector, optimizing these models and applying similar word and document embedding methods like fastText [5, 17] and GloVe [32] could be a priority for future work. The introduced neural network approaches with LSTM cells have been shown to be viable as well and allow for a variety of possible extensions and optimizations. Better prediction thresholds optimized based on $ERDE$ scores or more specific signals for *depressed* predictions could help in making earlier predictions without too many false positives. Finally, the collected meta information on the user base can be extended to utilize emotion lexica [28], psychological and social insights obtained for example from LIWC, and additional statistical text features.

References

1. Abadi M., Agarwal A., Barham P., Brevdo E., Chen Z., Citro C., Corrado G.S., Davis A., Dean J., Devin M., Ghemawat S., Goodfellow I., Harp A., Irving G., Isard M., Jozefowicz R., Jia Y., Kaiser L., Kudlur M., Levenberg J., Mané D., Schuster M., Monga R., Moore S., Murray D., Olah C., Shlens J., Steiner B., Sutskever I., Talwar K., Tucker P., Vanhoucke V., Vasudevan V., Viégas F., Vinyals O., Warden P., Wattenberg M., Wicke M., Yu Y., and Zheng X.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. (2015) Software available from tensorflow.org
2. Beck, A.T., Alford, B.A.: Depression: Causes and Treatment. Second Edition. University of Pennsylvania Press (2009)
3. Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C.: A Neural Probabilistic Language Model. Journal of Machine Learning Research, Vol. 3(Feb), pp. 1137–1155 (2003)
4. Bergstra, J., Desjardins, G., Lamblin, P., Bengio, Y.: Quadratic Polynomials Learn Better Image Features. Technical Report 1337, Département d'Informatique et de Recherche Opérationnelle, Université de Montréal (2009)
5. Bojanowski, P.,Grave, E., Joulin, A., Mikolov, T.: Enriching Word Vectors with Subword Information. arXiv preprint arXiv:1607.04606 (2016)
6. Brook D.W., Brook J.S., Zhang C., Cohen P., Whiteman M.: Drug Use and the Risk of Major Depressive Disorder, Alcohol Dependence, and Substance Use Disorders. Arch Gen Psychiatry, Vol. 59(11), pp. 1039–1044 (2002)
7. Chall, J.S., Dale, E.: Readability Revisited: The New Dale-Chall Readability Formula. Brookline Books (1995)
8. Christensen, J.G.: Readability Helps the Level. (2000) Available from: http://www.csun.edu/~vcecn006/read1.html - Accessed on 2017-04-21
9. Coppersmith, G., Dredze, M., Harman, C., Hollingshead, K., Mitchell, M.: CLPsych 2015 Shared Task: Depression and PTSD on Twitter. Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pp. 31–39 (2015)
10. Dale, E., Chall, J.S.: A Formula for Predicting Readability: Instructions. Educational Research Bulletin, pp. 37–54 (1948)
11. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, Vol. 41(6), pp. 391–407 (1990)
12. Flesch, R.: A New Readability Yardstick. Journal of Applied Psychology, Vol. 32(3), pp. 221–233 (1948)
13. Goodfellow I., Bengio Y., Courville A.: Deep Learning. MIT Press (2016)
14. Gunning, R.: The Technique of Clear Writing. McGraw-Hill (1952)
15. Hahnloser, R.H., Seung, H.S., Slotine, J.J.: Permitted and Forbidden Sets in Symmetric Threshold-Linear Networks. Neural Computation, Vol. 15(3), pp. 621–638 (2003)
16. Hochreiter, S., Schmidhuber J.: Long Short-Term Memory. Neural Computation, Vol. 9(8), pp. 1735–1780 (1997)
17. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of Tricks for Efficient Text Classification. arXiv preprint arXiv:1607.01759 (2016)
18. Khantzian, E.J.: The Self-Medication Hypothesis of Addictive Disorders: Focus on Heroin and Cocaine Dependence. The American Journal of Psychiatry, Vol. 142(11), pp. 1259–1264 (1985)
19. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization. Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, arXiv preprint arXiv:1412.6980 (2015)

20. Lan, M., Tan, Chew L., Low, H.-B.: Proposing a New Term Weighting Scheme for Text Categorization. Proceedings of the 21st National Conference on Artifical Intelligence (AAAI-06), Vol. 6, pp. 763–768 (2006)
21. Le, Q.V., Mikolov, T.: Distributed Representations of Sentences and Documents. Proceedings of the 31st International Conference on Machine Learning (ICML), Vol. 14, pp. 1188–1196 (2014)
22. Losada, D.E., Crestani, F.: A Test Collection for Research on Depression and Language Use. Experimental IR Meets Multilinguality, Multimodality, and Interaction: 7th International Conference of the CLEF Association, pp. 28–39. CLEF 2016, Évora, Portugal (2016)
23. Losada, D.E., Crestani, F., Parapar, J.: eRISK 2017: CLEF Lab on Early Risk Prediction on the Internet: Experimental Foundations. Proceedings Conference and Labs of the Evaluation Forum CLEF 2017, Dublin, Ireland (2017)
24. Maaten, L.v.d., Hinton, G.: Visualizing Data Using t-SNE. Journal of Machine Learning Research, Vol. 9(Nov), pp. 2579–2605 (2008)
25. Manning, C.D., Raghavan, P., Schütze, H.: An Introduction to Information Retrieval. Online Edition. Cambridge University Press (2009) Available from: https://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf - Accessed on 2017-04-21
26. Mikolov, T., Chen, K., Dean, J., Corrado, G.: Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at International Conference on Learning Representations ICLR 2013, arXiv preprint arXiv:1301.3781 (2013)
27. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed Representations of Words and Phrases and their Compositionality. Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
28. Mohammad, S.M., Turney, P.D.: Crowdsourcing a Word-Emotion Association Lexicon. Computational Intelligence, Vol. 29(3), pp. 436–465 (2013)
29. Nadeem, M., Horn, M., Coppersmith, G., Sen, S.: Identifying Depression on Twitter. arXiv preprint arXiv:1607.07384 (2016)
30. Paltoglou, G., Thelwall, M.: A Study of Information Retrieval Weighting Schemes for Sentiment Analysis. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pp. 1386–1395. Association for Computational Linguistics (2010)
31. Pedersen, T.: Screening Twitter Users for Depression and PTSD with Lexical Decision Lists. Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pp. 46–53. Association for Computational Linguistics (2015)
32. Pennington J., Richard S., Manning C.D.: GloVe: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
33. Preoţiuc-Pietro, D., Sap, M., Schwartz, H.A., Ungar, L.: Mental Illness Detection at the World Well-Being Project for the CLPsych 2015 Shared Task Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pp. 40–45. Association for Computational Linguistics (2015)
34. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pp. 45–50 (2010)
35. Resnik, P., Armstrong, W., Claudino, L., Nguyen, T.: The University of Maryland CLPsych 2015 Shared Task System. Proceedings of the 2nd Workshop on Com-

putational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pp. 54–60. Association for Computational Linguistics (2015)
36. Rude, S., Gortner, E.-M., Pennebaker, J.: Language Use of Depressed and Depression-Vulnerable College Students. Cognition & Emotion, Vol. 18(8), pp. 1121–1133 (2004)
37. Salton, G., Buckley, C.: Term-Weighting Approaches in Automatic Text Retrieval. Information Processing & Management, Vol. 24(5), pp. 513–523 (1988)
38. Smirnova, D., Sloeva, E., Kuvshinova, N., Krasnov, A., Romanov, D., Nosachev, G.: Language Changes as an Important Psychopathological Phenomenon of Mild Depression. European Psychiatry, Vol. 28 (2013)
39. Tausczik, Y.R., Pennebaker, J.W.: The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods. Journal of Language and Social Psychology, Vol. 29(1), pp. 24–54 (2010)
40. Wu, H., Gu, X.: Reducing Over-Weighting in Supervised Term Weighting for Sentiment Analysis. The 25th International Conference on Computational Linguistics (COLING 2014), pp. 1322–1330 (2014)