A Recurrent and Compositional Model for Personality Trait Recognition from Short Texts

Fei Liu[♠]*, **Julien Perez**[♡] **and Scott Nowson**[♣]*
[♠]The University of Melbourne, Victoria, Australia
[♡]Xerox Research Centre Europe, Grenoble, France
[♣]Accenture Centre for Innovation, Dublin, Ireland
fliu3@student.unimelb.edu.au julien.perez@xrce.xerox.com
scott.nowson@accenture.com

Abstract

Many methods have been used to recognise author personality traits from text, typically combining linguistic feature engineering with shallow learning models, e.g. linear regression or Support Vector Machines. This work uses deep-learning-based models and atomic features of text, the characters, to build hierarchical, vectorial word and sentence representations for trait inference. This method, applied to a corpus of tweets, shows state-of-the-art performance across five traits compared with prior work. The results, supported by preliminary visualisation work, are encouraging for the ability to detect complex human traits.

1 Introduction

Techniques falling under the umbrella of "deep-learning" are increasingly commonplace in the space of Natural Language Processing (NLP) (Manning, 2016). Such methods have been applied to a number of tasks from part-of-speech-tagging (Ling et al., 2015) to sentiment analysis (Socher et al., 2013). Essentially, each of these tasks is concerned with learning representations of language at different levels. The work we outline here is no different in essence, though we choose perhaps the highest level of representation – that of the author of a given text rather than the text itself. This task, modelling people from their language, is one built on the long-standing foundation that language use is known to be influenced by sociodemographic characteristics such as gender and personality (Tannen, 1990; Pennebaker et al., 2003). The study of personality traits in particular is supported by the notion that they are considered temporally stable (Matthews et al., 2003), and thus our modelling ability is enriched by the acquisition of more data over time.

Computational personality recognition, and its broader applications, is becoming of increasing interest with workshops exploring the topic (Celli et al., 2014; Tkalčič et al., 2014). The addition of personality traits in the PAN Author Profiling challenge at CLEF in 2015 (Rangel et al., 2015) is further evidence. Much prior literature in this field has used some variation of enriched bag-of-words; e.g. the "Open vocabulary" approach (Schwartz et al., 2013). This is understandable as exploring the relationship between word use and traits has delivered significant insight into aspects of human behaviour (Pennebaker et al., 2003). Different levels of representation of language have been used such as syntactic, semantic, and higher-order such as the psychologically-derived lexica of the Linguistic Inquiry and Word Count (LIWC) tool (Pennebaker et al., 2015). One drawback of this bag-of-linguistic-features approach is that considerable effort can be spent on feature engineering. Another is an unspoken assumption that these features, like the traits to which they relate, are similarly stable: the same language features always indicate the same traits. However, this is not the case. As Nowson and Gill (2014) have shown, the relationship between language and personality is not consistent across all forms of communication and that it is more complex. In order to better explore this complexity in this work we propose a novel deep-learning feature-engineering-free modelisation of the problem of personality trait recognition. The task is framed

*Work carried out at Xerox Research Centre Europe

Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media, pages 20–29, Osaka, Japan, December 12 2016.

as one of supervised sequence regression based on a joint atomic representation of the text: specifically on the character and word level. In this context, we are exploring short texts. Typically, classification of such texts tends to be particularly challenging for state-of-the-art BoW based approaches due, in part, to the noisy nature of such data (Han and Baldwin, 2011). To cope with this we propose a novel recurrent and compositional neural network architecture, capable of constructing representations at character, word and sentence level. The paper is structured as follows: after we consider previous approaches to the task of computational personality recognition, including those which have a deep-learning component, we describe our model. We report on two sets of experiments, the first of which demonstrates the effectiveness of the model in inferring personality for users, while the second reports on the short text level analysis. In both settings, the proposed model achieves state-of-the-art performance across five personality traits.

2 Related Work

Early work on computational personality recognition (Argamon et al., 2005; Nowson and Oberlander, 2006) used SVM-based approaches and manipulated lexical and grammatical feature sets. Today, according to the organisers (Rangel et al., 2015) "most" participants to the PAN 2015 Author Profiling task still use a combination of SVM and feature engineering. Data labelled with personality data is sparse (Nowson and Gill, 2014) and there has been more interest in reporting novel feature sets. In the PAN task alone[1] there were features used from multiple levels of representation on language. Surface forms were present in word, lemma and character n-grams, while syntactic features included POS tags and dependency relations. There were some efforts of feature curation, such as analysis of punctuation and emoticon use, along with the use of latent semantic analysis for topic modelling. Another popular feature set is the use of external resources such as LIWC (Pennebaker et al., 2015) which, in this context, represents over 20 years of psychology-based feature engineering. When applied to tweets, however, LIWC requires further cleaning of the data (Kreindler, 2016).

Deep-learning based approaches to personality trait recognition are, unsurprisingly given the typical size of data sets, relatively few. The model detailed in Kalghatgi et al. (2015) presents a neural network based approach to personality prediction of users. In this model, a Multilayer Perceptron (MLP) takes as input a collection of hand-crafted grammatical and social behavioral features from each user and assigns a label to each of the 5 personality traits. Unfortunately no evaluation of this work, nor details of the dataset, were provided. The work of Su et al. (2016) describes a Recurrent Neural Network (RNN) based system, exploiting the turn-taking of conversation for personality trait prediction. In their work, RNNs are employed to model the temporal evolution of dialog, taking as input LIWC-based and grammatical features. The output of the RNNs is then used for the prediction of personality trait scores of the participants of the conversations. It is worth noting that both works utilise hand-crafted features which rely heavily on domain expertise. Also the focus is on the prediction of trait scores on the user level given all the available text from a user. In contrast, not only can the approach presented in this paper infer the personality of a user given a collection of short texts, it is also flexible to predict trait scores from a single short text, arguably a more challenging task considering the limited amount of information.

The model we present in Section 3.2 is inspired by Ling et al. (2015), who proposed a character-level word representation learning model under the assumption that character sequences are syntactically and semantically informative of the words they compose. Based on a widely used RNN named long short-term memory network (LSTM) (Hochreiter and Schmidhuber, 1997), the model learns the embeddings of characters and how they can be used to construct words. Inspired by this, Yang et al. (2016) introduced Hierarchical Attention Networks where the representation of a document is hierarchically built up. The work of (Ling et al., 2015) provides a way to construct words from their constituent characters (Character to Word, `C2W`) while Yang et al. (2016) describe a hierarchical approach to building representations of documents from words to sentences, and eventually to documents (Word to Sentence to Document, `W2S2D`). In this work, inspired by the above works, we present a hierarchical model situated between the above two models, connecting characters, words and sentences, and ultimately personality traits

[1] Due to space consideration we are unable to cite the individual works.

(Character to Word to Sentence for Personality Trait, C2W2S4PT).

3 Proposed Model

To motivate our methodology, we review a commonly-used approach to representing sentences and discuss some of its limitations and motivation. Then, we propose the use of a compositional model to tackle the identified problems.

3.1 Current Issues and Motivation

One classical approach for applying deep learning models to NLP problems involves word lookup tables where words are typically represented by dense real-valued vectors in a low-dimensional space (Socher et al., 2013). In order to obtain a sensible set of embeddings, a common practice is to train on a large corpus in an unsupervised fashion, e.g. Word2Vec (Mikolov et al., 2013). Despite the success in capturing syntactic and semantic information with such word vectors, there are two practical problems with such an approach (Ling et al., 2015). First, due to the flexibility of language, previously unseen words are bound to occur regardless of how large the unsupervised training corpus is. The problem is particularly serious for text extracted from social media platforms such as Twitter and Facebook due to the noisy nature of user-generated text – e.g. typos, ad hoc acronyms and abbreviations, phonetic substitutions, and even meaningless strings (Han and Baldwin, 2011). Second, the number of parameters for a model to learn is overwhelmingly large. Assume each word is represented by a vector of d dimensions, the total size of the word lookup table is $d \times |V|$ where $|V|$ is the size of the vocabulary which tends to scale to the order of hundreds and thousands. Again, this problem is even more pronounced in noisier domain such as short text generated by online users. To address the above issues, we adopt a compositional character to word model described in the next section.

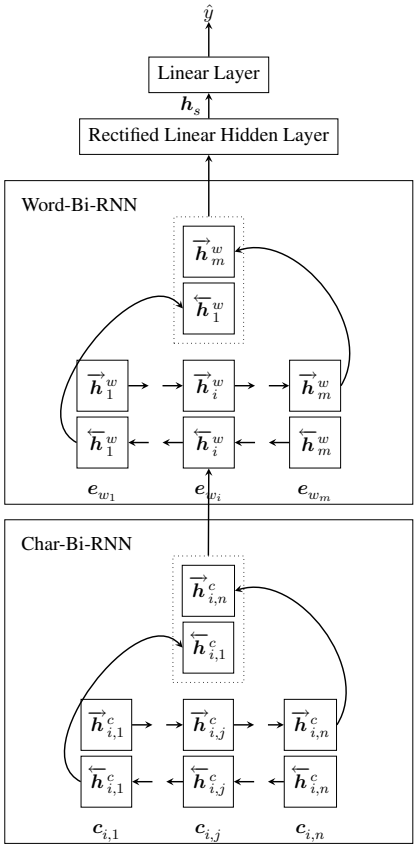

Figure 1: Illustration of the C2W2S4PT model. Dotted boxes indicate concatenation.

From the personality perspective, character-based features have been widely adopted in trait inference, such as character n-grams(González-Gallardo et al., 2015; Sulea and Dichiu, 2015), emoticons (Nowson et al., 2015; Palomino-Garibay et al., 2015), and character flooding (Nowson et al., 2015; Giménez et al., 2015). Motivated by this and the issues identified above, we propose in the next section a compositional model that operates hierarchically at the character, word and sentence level, capable of harnessing personality-sensitive signals buried as deep as the character level.

3.2 Character to Word to Sentence for Personality Traits

To address the problems identified in Section 3.1, we propose to extend the compositional character to word model first introduced by Ling et al. (2015) wherein the representation of each word is constructed, via a character-level bi-directional RNN (Char-Bi-RNN), from its constituent characters. The constructed word vectors are then fed to another layer of word-level Bi-RNN (Word-Bi-RNN) and a sentence is represented by the concatenation of the last and first hidden states of the forward and backward Word-RNNs respectively. Eventually, a feedforward neural network takes as input the representation of a sentence and returns a scalar as the prediction for a specific personality trait. Thus, we name the model C2W2S4PT (Character to Word to Sentence for Personality Traits) which is illustrated in Figure 1. Specifically, suppose we have a sentence s consisting of a sequence of words $\{w_1, w_2, \ldots, w_i, \ldots, w_m\}$.

We define a function $\mathrm{c}(w_i, j)$ which takes as input a word w_i, together with an index j and returns the one-hot vector representation of the j^{th} character of the word w_i. Then, to get the embedding $\boldsymbol{c}_{i,j}$ of the character, we transform $\mathrm{c}(w_i, j)$ by: $\boldsymbol{c}_{i,j} = \boldsymbol{E}_c \mathrm{c}(w_i, j)$ where $\boldsymbol{E}_c \in \mathbb{R}^{d \times |C|}$ and $|C|$ is the size of the character vocabulary. Next, in order to construct the representation of word w_i, the sequence of character embeddings $\{\boldsymbol{c}_{i,1}, \ldots, \boldsymbol{c}_{i,n}\}$ is taken as input to the Char-Bi-RNN (assuming w_i is comprised of n characters). In this work, we employ gated recurrent unit (GRU) (Cho et al., 2014) as the recurrent unit in the Bi-RNNs, given that recent studies indicate that GRU achieves comparable, if not better, results to LSTM (Chung et al., 2014).[2] Concretely, the forward pass of the Char-Bi-RNN is carried out using the following:

$$\overrightarrow{\boldsymbol{z}}^c_{i,j} = \sigma(\overrightarrow{\boldsymbol{W}}^c_z \boldsymbol{c}_{i,j} + \overrightarrow{\boldsymbol{U}}^c_{hz} \overrightarrow{\boldsymbol{h}}^c_{i,j-1} + \overrightarrow{\boldsymbol{b}}^c_z) \tag{1}$$

$$\overrightarrow{\boldsymbol{r}}^c_{i,j} = \sigma(\overrightarrow{\boldsymbol{W}}^c_r \boldsymbol{c}_{i,j} + \overrightarrow{\boldsymbol{U}}^c_{hr} \overrightarrow{\boldsymbol{h}}^c_{i,j-1} + \overrightarrow{\boldsymbol{b}}^c_r) \tag{2}$$

$$\overrightarrow{\tilde{\boldsymbol{h}}}^c_{i,j} = \tanh(\overrightarrow{\boldsymbol{W}}^c_h \boldsymbol{c}_{i,j} + \overrightarrow{\boldsymbol{r}}^c_{i,j} \odot \overrightarrow{\boldsymbol{U}}^c_{hh} \overrightarrow{\boldsymbol{h}}^c_{i,j-1} + \overrightarrow{\boldsymbol{b}}^c_h) \tag{3}$$

$$\overrightarrow{\boldsymbol{h}}^c_{i,j} = \overrightarrow{\boldsymbol{z}}^c_{i,j} \odot \overrightarrow{\boldsymbol{h}}^c_{i,j-1} + (1 - \overrightarrow{\boldsymbol{z}}^c_{i,j}) \odot \overrightarrow{\tilde{\boldsymbol{h}}}^c_{i,j} \tag{4}$$

where $\odot$ is the element-wise product, $\overrightarrow{\boldsymbol{W}}^c_z, \overrightarrow{\boldsymbol{W}}^c_r, \overrightarrow{\boldsymbol{W}}^c_h, \overrightarrow{\boldsymbol{U}}^c_{hz}, \overrightarrow{\boldsymbol{U}}^c_{hr}, \overrightarrow{\boldsymbol{U}}^c_{hh}$ are the parameters for the model to learn, and $\overrightarrow{\boldsymbol{b}}^c_z$, $\overrightarrow{\boldsymbol{b}}^c_r$, $\overrightarrow{\boldsymbol{b}}^c_h$ the bias terms. The backward pass, the hidden state of which is symbolised by $\overleftarrow{\boldsymbol{h}}^c_{i,j}$, is performed similarly, although with a different set of GRU weight matrices and bias terms. It should be noted that both the forward and backward Char-RNN share the same character embeddings. Ultimately, w_i is represented by the concatenation of the last and first hidden states of the forward and backward Char-RNNs: $\boldsymbol{e}_{w_i} = [\overrightarrow{\boldsymbol{h}}^c_{i,n}; \overleftarrow{\boldsymbol{h}}^c_{i,1}]^\top$. Once all the word representations $\boldsymbol{e}_{w_i}$ for $i \in [1, n]$ have been constructed from their constituent characters, they are then processed by the Word-Bi-RNN, similar to Char-Bi-RNN but on word level with word rather than character embeddings:

$$\overrightarrow{\boldsymbol{z}}^w_i = \sigma(\overrightarrow{\boldsymbol{W}}^w_z \boldsymbol{e}_{w_i} + \overrightarrow{\boldsymbol{U}}^w_{hz} \overrightarrow{\boldsymbol{h}}^w_{i-1} + \overrightarrow{\boldsymbol{b}}^w_z) \tag{5}$$

$$\overrightarrow{\boldsymbol{r}}^w_i = \sigma(\overrightarrow{\boldsymbol{W}}^w_r \boldsymbol{e}_{w_i} + \overrightarrow{\boldsymbol{U}}^w_{hr} \overrightarrow{\boldsymbol{h}}^w_{i-1} + \overrightarrow{\boldsymbol{b}}^w_r) \tag{6}$$

$$\overrightarrow{\tilde{\boldsymbol{h}}}^w_i = \tanh(\overrightarrow{\boldsymbol{W}}^w_h \boldsymbol{e}_{w_i} + \overrightarrow{\boldsymbol{r}}^w_i \odot \overrightarrow{\boldsymbol{U}}^w_{hh} \overrightarrow{\boldsymbol{h}}^w_{i-1} + \overrightarrow{\boldsymbol{b}}^w_h) \tag{7}$$

$$\overrightarrow{\boldsymbol{h}}^w_i = \overrightarrow{\boldsymbol{z}}^w_i \odot \overrightarrow{\boldsymbol{h}}^w_{i-1} + (1 - \overrightarrow{\boldsymbol{z}}^w_i) \odot \overrightarrow{\tilde{\boldsymbol{h}}}^w_i \tag{8}$$

where $\overrightarrow{\boldsymbol{W}}^w_z, \overrightarrow{\boldsymbol{W}}^w_r, \overrightarrow{\boldsymbol{W}}^w_h, \overrightarrow{\boldsymbol{U}}^w_{hz}, \overrightarrow{\boldsymbol{U}}^w_{hr}, \overrightarrow{\boldsymbol{U}}^w_{hh}$ are the parameters for the model to learn, and $\overrightarrow{\boldsymbol{b}}^w_z$, $\overrightarrow{\boldsymbol{b}}^w_{wr}$, $\overrightarrow{\boldsymbol{b}}^w_h$ the bias terms. In a similar fashion to how a word is represented, we construct the sentence embedding by concatenation: $\boldsymbol{e}_s = [\overrightarrow{\boldsymbol{h}}^w_m; \overleftarrow{\boldsymbol{h}}^w_1]^\top$. Lastly, to estimate the score for a particular personality trait, we top the Word-Bi-RNN with an MLP which takes as input the sentence embedding $\boldsymbol{e}_s$ and returns the estimated score $\hat{y}_s$: $\boldsymbol{h}_s = \mathrm{ReLU}(\boldsymbol{W}_{eh} \boldsymbol{e}_s + \boldsymbol{b}_h)$ and then $\hat{y}_s = \boldsymbol{W}_{hy} \boldsymbol{h}_s + b_y$ where ReLU is the REctified Linear Unit defined as $\mathrm{ReLU}(x) = \max(0, x)$, $\boldsymbol{W}_{eh}, \boldsymbol{W}_{hy}$ the parameters for the model to learn, $\boldsymbol{b}_h, b_y$ the bias terms, and $\boldsymbol{h}_s$ the hidden representation of the MLP. All the components in the model are jointly trained with *mean square error* being the objective function: $L(\theta) = \frac{1}{n} \sum_{i=1}^{n} (y_{s_i} - \hat{y}_{s_i})^2$ where y_{s_i} is the ground truth personality score of sentence s_i and θ the collection of all embedding and weight matrices and bias terms for the model to learn.

3.2.1 Multitask Learning

While the dimensions of personality in any single model are designed to be independent of one another, there are often strong correlations between traits (Matthews et al., 2003). Understanding that such correlations exist, we ask whether it is beneficial to train a model capable of simultaneously predicting multiple highly correlated personality traits. To support this, we report the Pearson correlations of our dataset (see section 4.1) in Table 1 where EXT, STA, AGR, CON and OPN are abbreviations

[2] We performed additional experiments which confirmed this finding. Therefore due to space considerations, we do not report results using LSTMs here.

	EXT	STA	AGR	CON	OPN
EXT		0.295***	0.257**	0.216**	0.057
STA	0.295***		0.351***	0.091	0.045
AGR	0.257**	0.351***		0.035	0.039
CON	0.216**	0.091	0.035		0.174*
OPN	0.057	0.045	0.039	0.174*	

Note: *** $p \leqslant 0.001$, ** $p \leqslant 0.01$, * $p \leqslant 0.05$

Table 1: Pearson correlations for the five personality traits

for Extroversion, Emotional Stability (the inverse of Neuroticism), Agreeableness, Conscientiousness and Openness respectively. This gives us confidence that there are at least linear relationships between individual traits which could potentially be exploited by multitask learning (Caruana, 1997). Inspired by this and building on top of the compositional model, we propose a multitask learning model which shares the Char-Bi-RNN and Word-Bi-RNN components but has personality-trait-specific final layers, to predict multiple correlated personality traits simultaneously. Concretely, while Char-Bi-RNN and Word-Bi-RNN remain the same as described in Section 3.2, we utilise a collection of personality-trait-specific final layers: $\boldsymbol{h}_{p,s} = \text{ReLU}(\boldsymbol{W}_{peh}\boldsymbol{e}_s + \boldsymbol{b}_{ph})$ and then $\hat{y}_{p,s} = \boldsymbol{W}_{phy}\boldsymbol{h}_{p,s} + b_{py}$ where $p \in \{\text{EXT, STA, AGR, CON, OPN}\}$, $\boldsymbol{W}_{peh}, \boldsymbol{W}_{phy}, \boldsymbol{b}_{ph}, b_{py}$ are the trait-specific weight matrices and bias terms, and loss functions: $L_p(\theta_p) = \frac{1}{n}\sum_{i=1}^{n}(y_{p,s_i} - \hat{y}_{p,s_i})^2$ where $L_p(\theta_p)$ is the loss function for a specific personality trait p. Note that, apart from the Bi-RNN embedding and weight matrices and bias terms, θ_p now also includes the trait-specific weight matrices $\boldsymbol{W}_{peh}, \boldsymbol{W}_{phy}$ and bias terms $\boldsymbol{b}_{ph}, b_{py}$. The model is then jointly trained using the sum of the loss functions: $L(\theta) = \sum_{p \in P} L_p(\theta_p)$ where P is a collection of (correlated) personality traits and $\theta = \bigcup_{p \in P} \theta_p$.

4 Experiments and Results

We report two sets of experiments: the first a comparison at the user level between our feature-engineering-free approach and current state-of-the-art models which rely on linguistic features; the second designed to evaluate the performance of the proposed model against other feature-engineering-free approaches on individual short texts. We show that in both settings, i.e., against models with or without feature engineering, our proposed model achieves better results across all personality traits.

4.1 Dataset

We use the English data from the PAN 2015 Author Profiling task dataset (Rangel et al., 2015), collected from Twitter and consisting of $14,166$ tweets and 152 users. For each user there is a set of tweets (average $n = 100$) and gold standard personality labels. The five trait labels – scores between -0.5 and 0.5 – are calculated following the author's self-assessment responses to the short Big 5 test, BFI-10 (Rammstedt and John, 2007) which is the most widely accepted and exploited scheme for personality recognition and has the most solid grounding in language (Poria et al., 2013).

In our experiments, each tweet is tokenised using Twokenizer (Owoputi et al., 2013), in order to preserve hashtag-preceded topics and user mentions. Unlike the majority of the language used in a tweet, URLs and mentions are used for their targets, and not their surface forms. Therefore each text is normalised by mapping these features to single characters (e.g., *@username* → @, *http://t.co/* → ^). Thus we limit the risk of modelling, say, character usage which was not directly influenced by the personality of the author.

4.2 Evaluation Method

Due to the unavailability of the test corpus – withheld by the PAN 2015 organisers – we compare the k-fold cross-validation performance ($k = 5$ or 10) on the available dataset. Performance is measured using Root Mean Square Error (RMSE) on either the tweet level or user level depending on the granularity of

the task: $RMSE_{tweet} = \sqrt{\frac{\sum_{i=1}^{T}(y_{s_i} - \hat{y}_{s_i})^2}{T}}$ and $RMSE_{user} = \sqrt{\frac{\sum_{i=1}^{U}(y_{user_i} - \hat{y}_{user_i})^2}{U}}$ where T and U are the total numbers of tweets and users in the corpus, y_{s_i} and $\hat{y}_{s_i}$ the true and estimated personality trait score of the i^{th} tweet, similarly y_{user_i} and $\hat{y}_{user_i}$ are their user-level counterparts. Each tweet in the dataset inherits the same five trait scores as assigned to the author from whom they were drawn. $\hat{y}_{user_i} = \frac{1}{T_i}\sum_{j=1}^{T_i}\hat{y}_{s_j}$ where T_i refers to the total number of tweets of $user_i$. In Section 4.3 and 4.4, we present the results measured at the user and tweet level using $RMSE_{user}$ and $RMSE_{tweet}$ respectively. It is important to note that, to enable direct comparison, we use exactly the same dataset and evaluation metric $RMSE_{user}$ as in the works of (Sulea and Dichiu, 2015; Mirkin et al., 2015; Nowson et al., 2015).

4.3 Personality Trait Prediction at User Level

We test the proposed models on the dataset described in Section 4.1 and train our model to predict the personality trait scores based purely on the text with no additional features supplied. To demonstrate the effectiveness of the proposed model, we evaluate the performance on the user level against models incorporating linguistic and psychologically motivated features. This allows us to directly compare the performance of current state-of-the-art models and `C2W2S4PT`. For 5-fold cross-validation, we compare to the tied-highest ranked (under evaluation conditions) of the PAN 2015 submissions (Sulea and Dichiu, 2015).[3] For 10-fold cross-validation, we similarly choose the work by ranking and metric reporting (Nowson et al., 2005). As here, these works predicted scores on text level, and averaged for each user. Therefore, we include subsequent work which reports results on concatenated tweets – a single document per user (Mirkin et al., 2015). We also show the most straightforward baseline `Average Baseline` which assigns the average of all the scores to each user. `C2W2S4PT` is trained with Adam (Kingma and Ba, 2014) and hyper-parameters: $\boldsymbol{E}_c \in \mathbb{R}^{50\times|C|}$, $\overrightarrow{\boldsymbol{h}}^{c}_{i,j}$ and $\overleftarrow{\boldsymbol{h}}^{c}_{i,j} \in \mathbb{R}^{256}$, $\overrightarrow{\boldsymbol{h}}^{w}_{i}$ and $\overleftarrow{\boldsymbol{h}}^{w}_{i} \in \mathbb{R}^{256}$, $\boldsymbol{W}_{eh} \in \mathbb{R}^{512\times256}$, $\boldsymbol{b}_h \in \mathbb{R}^{256}$, $\boldsymbol{W}_{hy} \in \mathbb{R}^{256\times1}$, $b_y \in \mathbb{R}$, dropout rate to the embedding output: 0.5, batch size: 32. Training is performed until 100 epochs are reached. The $RMSE_{user}$ results are shown in Table 2.

RNN-based models outperform the previous state of the art In the 5-fold cross-validation group, `C2W2S4PT - Multitask All` is superior to the baselines, achieving better performance in three traits (tying the remaining traits). This is worth noting considering the model is trained jointly on all five traits. Even greater improvement is attained by training on fewer personality traits with state-of-the-art performance achieved mostly by `C2W2S4PT`. In terms of the performance measured by 10-fold cross-validation, the dominance of the RNN-based models is even more pronounced with `C2W2S4PT` outperforming the two selected baseline systems across all personality traits. Overall, in comparison to the previous state-of-the-art models in both groups, `C2W2S4PT` not only outperforms them – by a significant margin in the case of 10-fold cross-validation – but it also achieves so without any hand-crafted features, underlining the soundness of the approach.

4.4 Personality Trait Prediction at Single Tweet Level

Although user-level evaluation is the common practice, we choose tweet-level performance to study the models' capabilities to infer personality at a lower granularity level. To support our evaluation, a number of baselines were created. To facilitate fair comparison, the only feature used is the surface form of the text. `Average Baseline`, the most straightforward baseline, assigns the average of all the scores to each tweet. Also, two BoW systems, namely, `Random Forest` and `SVM Regression`, have been implemented for comparison. For these two BoW-based baseline systems, we perform grid search to find the best hyper-parameter configuration. For `SVM Regression`, the hyper-parameters include: kernel $\in \{\text{linear}, \text{rbf}\}$ and $C \in \{0.01, 0.1, 1.0, 10.0\}$ whereas for `Random Forest`, the number of trees is chosen from the set $\{10, 50, 100, 500, 1000\}$.

Additionally, two simpler RNN-based models, namely `Bi-GRU-Char` and `Bi-GRU-Word`, which only work on character and word level respectively but share the same structure of the final MLP classifier ($\boldsymbol{h}_s$ and $\hat{y}_s$), have also been presented in contrast to the more sophisticated character to word composi-

[3] Cross-validation $RMSE_{user}$ performance is not reported for the other top system (Álvarez-Carmona et al., 2015).

k	Model	EXT	STA	AGR	CON	OPN
—	Average Baseline	0.166	0.223	0.158	0.151	0.146
5	Sulea and Dichiu (2015)	0.136	0.183	0.141	0.131	0.119
	C2W2S4PT	**0.131**	**0.171**	**0.140**	**0.124**	**0.109**
	C2W2S4PT - Multitask STA&AGR	×	0.172	0.140	×	×
	C2W2S4PT - Multitask AGR&CON	×	×	**0.138**	**0.124**	×
	C2W2S4PT - Multitask All	0.136	0.177	0.141	0.128	0.117
10	Mirkin et al. (2015)	0.171	0.223	0.173	0.144	0.146
	Nowson et al. (2015)	0.153	0.197	0.154	0.144	0.132
	C2W2S4PT	**0.130**	**0.167**	**0.137**	**0.122**	**0.109**
	C2W2S4PT - Multitask STA&AGR	×	0.168	0.140	×	×
	C2W2S4PT - Multitask AGR&CON	×	×	0.138	0.123	×
	C2W2S4PT - Multitask All	0.136	0.175	0.140	0.127	0.115

Table 2: $RMSE_{user}$ across five traits. **Bold** highlights best performance. × indicates N/A.

Model	EXT	STA	AGR	CON	OPN
Average Baseline	0.163	0.222	0.157	0.150	0.147
SVM Regression	0.148	0.196	0.148	0.140	0.131
Random Forest	0.144	0.192	**0.146**	0.138	0.132
Bi-GRU-Char	0.150	0.202	0.152	0.143	0.137
Bi-GRU-Word	0.147	0.200	**0.146**	0.138	0.130
C2W2S4PT	**0.142**	**0.188**	0.147	**0.136**	**0.127**
C2W2S4PT - Multitask STA&AGR	×	**0.189**	**0.146**	×	×
C2W2S4PT - Multitask AGR&CON	×	×	**0.146**	**0.136**	×
C2W2S4PT - Multitask All	**0.142**	0.191	**0.146**	0.137	0.127

Table 3: $RMSE_{tweet}$ across five traits level. **Bold** highlights best performance. × indicates N/A.

tional model C2W2S4PT. For training, C2W2S4PT inherits the same hyper-parameter configuration as described in Section 4.3. For Bi-GRU-Char and Bi-GRU-Word, we set the character and word embedding size to 50 and 256 respectively. Due to time constrains, we did not perform hyper-parameter fine-tuning for the RNN-based models and C2W2S4PT. The $RMSE_{tweet}$ of each effort, measured by 10-fold stratified cross-validation, is shown in Table 3.

C2W2S4PT achieves comparable or better performance with SVM Regression and Random Forest C2W2S4PT is state of the art in almost every trait with the exception of AGR. This demonstrates that C2W2S4PT generates at least reasonably comparable performance with SVM Regression and Random Forest in the feature-engineering-free setting on the tweet level and it does so without exhaustive hyper-parameter fine-tuning.

C2W2S4PT outperforms the RNN-based models This success can be attributed to the model's capability of coping with arbitrary words while not forgetting information due to excessive lengths as can arise from representing a text as a sequence of characters. Also, given that C2W2S4PT does not need to maintain a large vocabulary embedding matrix as in Bi-GRU-Word, there are much fewer parameters for the model to learn (Ling et al., 2015), making it less prone to overfitting.

Multitask learning provides little benefits to performance Surprisingly, the model jointly trained on the weakest correlated pair, namely AGR&CON, achieves even better results than the one trained on the strongest correlated pair (STA&AGR). In fact, despite the noise introduced by training on non-correlated personality traits, there is little impact on the performance of the multitask-learning models and the model jointly trained on all 5 personality traits generates equally competitive performance.

4.5 Visualisation

To further investigate into the learned representations and features, we choose the C2W2S4PT model trained on a single personality trait and visualise the sentences with the help of PCA (Tipping and Bishop, 1999). We also experimented with t-SNE (Van der Maaten and Hinton, 2008) but it did not produce an interpretable plot. 100 tweets have been randomly selected (50 tweets each from either end of the EXT spectrum) with their representations constructed by the model. Figure 2 shows the scatter plot of the representations of the sentences reduced to a 2D space by PCA for the trait of Extraversion (EXT), selected as it is the most commonly studied and well understood trait. The figure shows clusters of both positive and negative Extraversion, though the former intersect the latter. For discussion we consider three examples as highlighted in Figure 2:

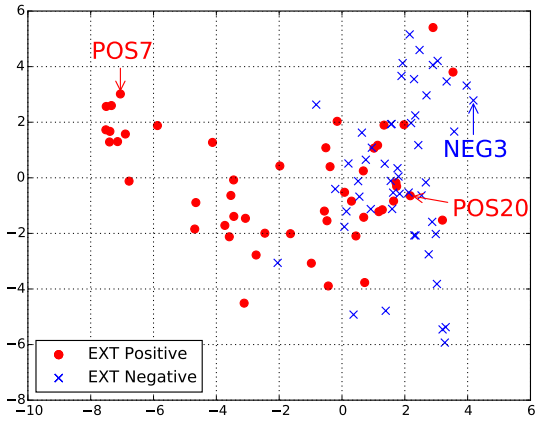

Figure 2: Scatter plot of sentence representations processed by PCA.

- POS7: *"@username: Feeling like you're not good enough is probably the worst thing to feel."*
- NEG3: *"Being good ain't enough lately."*
- POS20: *"o.O Lovely."*

The first two examples (POS7 and NEG3) are drawn from largely distinct areas of the distribution. In essence the semantics of the short texts are the same. However, they both show linguistic attributes commonly understood to relate to Extraversion (Gill and Oberlander, 2002): POS7 is longer and, with the use of the second person pronoun, is more inclusive of others; NEG3 on the other hand is shorter and self-focused, aspects indicative of Introversion. The third sentence, POS20, is a statement from an Extravert which appears to map to an Introvert space. Indeed, while short, the use of "Eastern" style, non-rotated emoticons (such as *o.O*) has also been shown to relate to Introversion on social media (Schwartz et al., 2013). This is perhaps not the venue to consider the implications of this further, although one explanation might be that the model has uncovered a flexibility often associated with Ambiverts (Grant, 2013). However, it is important to consider that the model is indeed capturing well-understood dimensions of language yet with no feature engineering.

5 Discussion and Future Work

Overall, the results in the paper support our methodology: C2W2S4PT not only provides state-of-the-art results on the user level, but also performs reasonably well when adapted to the short text level compared to other widely used models in the feature-engineering-free setting. However, interpretation of the performance of the multitask experiments is less straightforward. At text level (as per Table 3) the results are almost identical whether modelling traits individually, all together, or with differing prior relationships. Perhaps it is the case that simple linear correlations do not adequately explain the relationships between traits when mediated via language use. It could also be that our model captures a more complex, non-linear relationship or some notion of latent variables. It is clear that this requires further investigation, though this will likely require an additional dataset, as with only 150 authors, the distribution of scores is somewhat limited. One advantage of our approach which requires validation is that lack of feature engineering should support language independence. Preliminary tests on the Spanish data from the PAN 2015 Author Profiling dataset show promising results. To further examine this property of the proposed model, we plan to adopt TwiSty (Verhoeven et al., 2016), a recently introduced corpus consisting of 6 languages and labelled with MBTI type indicators (Myers and Myers, 2010). However, due to time constraints, we leave this exercise for future work.

References

Miguel A. Álvarez-Carmona, A. Pastor López-Monroy, Manuel Montes y Gómez, Luis Villaseñor-Pineda, and Hugo Jair Escalante. 2015. INAOE's participation at PAN'15: Author Profiling task—Notebook for PAN at CLEF 2015. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*.

Shlomo Argamon, Sushant Dhawle, Moshe Koppel, and James W. Pennebaker. 2005. Lexical predictors of personality type. In *Proceedings of the 2005 Joint Annual Meeting of the Interface and the Classification Society of North America*.

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28(1):41–75.

Fabio Celli, Bruno Lepri, Joan-Isaac Biel, Daniel Gatica-Perez, Giuseppe Riccardi, and Fabio Pianesi. 2014. The workshop on computational personality recognition 2014. In *Proc. ACMMM*, pages 1245–1246, Orlando, USA.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Alastair J. Gill and Jon Oberlander. 2002. Taking Care of the Linguistic Features of Extraversion. In *Proc. CogSci*, pages 363–368, Fairfax, USA.

Maite Giménez, Delia Irazú Hernández, and Ferran Pla. 2015. Segmenting Target Audiences: Automatic Author Profiling Using Tweets—Notebook for PAN at CLEF 2015. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*.

Carlos E. González-Gallardo, Azucena Montes, Gerardo Sierra, J. Antonio Núñez-Juárez, Adolfo Jonathan Salinas-López, and Juan Ek. 2015. Tweets Classification Using Corpus Dependent Tags, Character and POS N-grams—Notebook for PAN at CLEF 2015. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*.

Adam M. Grant. 2013. Rethinking the extraverted sales ideal: The ambivert advantage. *Psychological Science 24(6)*, 24(6):1024–1030.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a #twitter. In *Proc. ACL*, pages 368–378, Portland, Oregon, USA.

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

Mayuri Pundlik Kalghatgi, Manjula Ramannavar, and Nandini S. Sidnal. 2015. A neural network approach to personality prediction based on the big-five model. *International Journal of Innovative Research in Advanced Engineering (IJIRAE)*, 2(8):56–63.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jon Kreindler. 2016. Twitter psychology analyzer api and sample code. `http://www.receptiviti.ai/blog/twitter-psychology-analyzer-api-and-sample-code/`. Accessed: 2016-09-30.

Wang Ling, Chris Dyer, Alan W Black, Isabel Trancoso, Ramon Fermandez, Silvio Amir, Luis Marujo, and Tiago Luis. 2015. Finding function in form: Compositional character models for open vocabulary word representation. In *Proc. EMNLP*, pages 1520–1530, Lisbon, Portugal.

Christopher D Manning. 2016. Computational linguistics and deep learning. *Computational Linguistics*.

Gerald Matthews, Ian J. Deary, and Martha C. Whiteman. 2003. *Personality Traits*. Cambridge University Press, second edition. Cambridge Books Online.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proc. NIPS*, pages 3111–3119, Stateline, USA.

Shachar Mirkin, Scott Nowson, Caroline Brun, and Julien Perez. 2015. Motivating personality-aware machine translation. In *Proc. EMNLP*, pages 1102–1108, Lisbon, Portugal.

Isabel Myers and Peter Myers. 2010. *Gifts differing: Understanding personality type*. Nicholas Brealey Publishing.

Scott Nowson and Alastair J. Gill. 2014. Look! Who's Talking? Projection of Extraversion Across Different Social Contexts. In *Proceedings of WCPR14, Workshop on Computational Personality Recognition at ACMM (22nd ACM International Conference on Multimedia)*.

Scott Nowson and Jon Oberlander. 2006. The Identity of Bloggers: Openness and gender in personal weblogs. In *AAAI Spring Symposium, Computational Approaches to Analysing Weblogs*.

Scott Nowson, Jon Oberlander, and Alastair J. Gill. 2005. Weblogs, genres and individual differences. In *Proc. CogSci*, pages 1666–1671.

Scott Nowson, Julien Perez, Caroline Brun, Shachar Mirkin, and Claude Roux. 2015. XRCE Personal Language Analytics Engine for Multilingual Author Profiling. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proc. NAACL*, pages 380–390, Atlanta, USA.

Alonso Palomino-Garibay, Adolfo T. Camacho-González, Ricardo A. Fierro-Villaneda, Irazú Hernández-Farias, Davide Buscaldi, and Ivan V. Meza-Ruiz. 2015. A Random Forest Approach for Authorship Profiling—Notebook for PAN at CLEF 2015. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*.

James W Pennebaker, Kate G Niederhoffer, and Matthias R Mehl. 2003. Psychological aspects of natural language use: Our words, our selves. *Annual Review of Psychology*, 54:547–577.

J. W. Pennebaker, R. L. Boyd, K. Jordan, and K. Blackburn. 2015. The development and psychometric properties of LIWC2015.

Soujanya Poria, Alexandar Gelbukh, Basant Agarwal, Erik Cambria, and Newton Howard, 2013. *Common Sense Knowledge Based Personality Recognition from Text*, pages 484–496.

Beatrice Rammstedt and Oliver P. John. 2007. Measuring personality in one minute or less: A 10-item short version of the big five inventory in english and german. *Journal of Research in Personality*, 41(1):203–212.

Francisco Rangel, Fabio Celli, Paolo Rosso, Martin Potthast, Benno Stein, and Walter Daelemans. 2015. Overview of the 3rd Author Profiling Task at PAN 2015. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*.

H Andrew Schwartz, Johannes C Eichstaedt, Margaret L Kern, Lukasz Dziurzynski, Stephanie M Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin E P Seligman, and Lyle H Ungar. 2013. Personality, Gender, and Age in the Language of Social Media: The Open-Vocabulary Approach. *PLOS ONE*, 8(9).

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. EMNLP*, Seattle, USA.

Ming-Hsiang Su, Chung-Hsien Wu, and Yu-Ting Zheng. 2016. Exploiting turn-taking temporal evolution for personality trait perception in dyadic conversations. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(4):733–744.

Octavia-Maria Sulea and Daniel Dichiu. 2015. Automatic profiling of twitter users based on their tweets. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*.

Deborah Tannen. 1990. *You Just Dont Understand: Women and Men in Conversation*. Harper Collins, New York.

Michael E Tipping and Christopher M Bishop. 1999. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622.

Marko Tkalčič, Berardina De Carolis, Marco de Gemmis, Ante Odić, and Andrej Košir. 2014. Preface: Empire 2014. In *Proceedings of the 2nd Workshop Emotions and Personality in Personalized Services (EMPIRE 2014)*. CEUR-WS.org, July.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85.

Ben Verhoeven, Walter Daelemans, and Barbara Plank. 2016. TwiSty: a multilingual twitter stylometry corpus for gender and personality profiling. In *Proc. LREC*, pages 1632–1637, Portorož, Slovenia.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proc. NAACL*, pages 1480–1489, San Diego, USA.