A Language-independent 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

There have been many attempts at automatically recognising author personality traits from text, typically incorporating linguistic features with conventional machine learning models, e.g. linear regression or Support Vector Machines. In this work, we propose to use deep-learning-based models with atomic features of text – the characters – to build hierarchical, vectorial word and sentence representations for the task of trait inference. On a corpus of tweets, this method shows state-of-the-art performance across five traits and three languages (English, Spanish and Italian) compared with prior work in author profiling. The results, supported by preliminary visualisation work, are encouraging for the ability to detect complex human traits.

1 Introduction

Deep-learning methods are becoming increasingly applied to problems in the area of Natural Language Processing (NLP) (Manning, 2016). Such techniques can be applied to tasks such as part-of-speech-tagging (Ling et al., 2015; Huang et al., 2015) and sentiment analysis (Socher et al., 2013; Kalchbrenner et al., 2014; Kim, 2014). At their core, these tasks can be seen as learning representations of language at different levels. Our work reported here is no different, though we choose a less commonplace level of representation – rather than the text itself, we focus on the author behind the text. Automatically modelling individuals from their language use is a task founded on the long-standing understanding that language use is influenced by sociodemographic characteristics such as gender and personality (Tannen, 1990; Pennebaker et al., 2003). The study of personality traits in particular is considered reliable as such traits are generally temporally stable (Matthews et al., 2003). As such, our ability to model our target – the author – is enriched by the acquisition of more data over time.

The volume of literature on computational personality recognition, and its broader applications, has grown steadily over the last decade. There have also been a number of dedicated workshops (Celli et al., 2014; Tkalčič et al., 2014) and shared tasks (Rangel et al., 2015) on the topic occurring in recent years. A significant portion of this prior literature has used some variation of enriched bag-of-words; e.g. the Open vocabulary approach (Schwartz et al., 2013). This is, theoretically speaking, entirely understandable as study of the relationship between word use and traits has delivered significant insight into human behaviour (Pennebaker et al., 2003). Language has been represented at a number of different levels in this work such as syntactic, semantic, and categorical - for example the psychologically-derived lexica of the Linguistic Inquiry and Word Count (LIWC) tool (Pennebaker et al., 2015).

These bag-of-linguistic-features approaches, however, require considerable feature engineering effort. In addition, many linguistic techniques and features are language-dependent, e.g. LIWC (Pennebaker et al., 2015), making the adaptation of models to multi-lingual scenarios more challenging. Another concern is a common assumption that these features, like the traits with which their use correlates, are similarly stable: the same language features always indicate the same traits. However, this is not the case: the relationship between language and personality is not consistent across all forms of communication, it is more complex (Nowson and Gill, 2014).

*Work carried out at Xerox Research Centre Europe

Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pages 754–764,
Valencia, Spain, April 3-7, 2017.

In order to address these challenges we propose a novel feature-engineering-free, deep-learning-based approach to the problem of personality trait recognition, enabling the model to work in various languages without the need to create language-specific linguistic features. We frame the problem as a supervised sequence regression task, taking only the joint atomic representation of the text: hierarchically on the character and word level. In this work, we focus on short texts. As pointed out by Han and Baldwin (2011), classification of such texts can often be challenging for even state-of-the-art BoW based approaches, which is, in part, caused by the noisy nature of such data. In this work, we address this by proposing a novel hierarchical neural network architecture, free of feature engineering and, once trained, capable of inferring personality across five traits and three languages.

The paper is structured as follows: we consider previous approaches to computational personality recognition, including those few which have a deep-learning component, and subsequently describe our model. We report two sets of experiments, the first to demonstrate the effectiveness of the model in inferring personality compared to the current state-of-the-art models, while the second reports analysis against other feature-engineering-free models. In both settings, the proposed model achieves state-of-the-art performance across five personality traits and three languages.

2 Related Work

Early work in computational personality recognition (Argamon et al., 2005; Nowson and Oberlander, 2006) were mainly SVM-based approaches, relying on syntactic and lexical features. A decade later, still "most" participants of the PAN 2015 Author Profiling task use SVM with feature engineering, according to the organisers (Rangel et al., 2015). Ensemble methods have been proposed (Verhoeven et al., 2013), but the base model was still SVM – the ensemble came from the combination of data from different sources as opposed to models. Data – not just text – labelled with personality traits is sparse (Nowson and Gill, 2014) and most work has focused on reporting novel feature sets rather than techniques. In the PAN task alone[1], there were features, in the form of surface forms of text, present on multiple levels of language representation, ranging from lexical features (e.g., word, lemma and character n-grams) to syntactic ones (e.g., POS tags and dependency relations). Some, on the other hand, focused on feature curation, analysing the correlation between personality and the use of punctuation and emoticon, along with the use of the topic modelling method: latent semantic analysis. In addition, external resources, such as LIWC (Pennebaker et al., 2015), constructed over 20 years of psychology-based feature engineering, are another often-used set of features. When applied to tweets, however, LIWC requires further cleaning of the data (Kreindler, 2016).

Approaches to personality trait recognition based on deep-learning are few, which is not surprising given the relatively small scale of the data sets used. Kalghatgi et al. (2015) employed a neural network based approach. In this model, a Multilayer Perceptron (MLP) takes as input a number of carefully hand-crafted syntactic and social behavioural features from each user and attempts to predict a label for each of the 5 personality traits. However, the authors reported neither evaluation of this work, nor details of the dataset. The work of Su et al. (2016), on the other hand, employs a Recurrent Neural Network (RNN), exploiting the turn-taking nature of conversation for personality trait prediction. In their work, the RNN processes the evolution of a dialogue over time, taking as input LIWC-based and grammatical features, the output of which is then fed into the classifier for the prediction of personality trait scores of each participant of the conversations. It should be noted that both works take manually-designed features, heavily relying on domain expertise. Also, the focus is on the prediction of trait scores on the author level based on modelling all available text from a user. In contrast, not only does our approach infer the personality of a user given a collection of short texts, it is also flexible enough to predict trait scores from a single short text, arguably a more challenging task considering the limited amount of information.

In Section 3.2, we propose a model inspired by the work of Ling et al. (2015) where representations are hierarchically constructed from characters to words. This is based on the assumption that character sequences are syntactically and semantically informative of the words they compose. Their model – a widely used RNN vari-

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

ant Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) – learns how to construct word embeddings via its constituent characters. When applied to the tasks of language modelling and part-of-speech tagging, the model successfully improves the accuracy upon traditional baselines, performing particularly well in morphologically rich languages. Not only does the model achieve better performance on both tasks, it also has significantly fewer parameters to learn compared to a word look-up table based approach as the number of different characters is much fewer than the number of different words in a vocabulary. Moreover, the model is able to generate a sensible representation for previously unseen words. Following this, Yang et al. (2016) took it further to the document level, introducing Hierarchical Attention Networks where two bi-directional Gated Recurrent Units (GRUs) are used to process the sequence of words and then sentences respectively with the sentence-level GRU taking as input the output of the word-level GRU and returning the representation of the document. While Yang et al. (2016) describe a means to hierarchically build representations of documents from words to sentences and eventually to documents (Word to Sentence to Document, `W2S2D`), the work of (Ling et al., 2015) is positioned at a more fine-grained level, incorporating information from the sequence of characters (Character to Word, `C2W`). In this paper, the model we propose is situated between `C2W` and `W2S2D` – connecting characters, words and sentences, and ultimately personality traits (Character to Word to Sentence for Personality Trait, `C2W2S4PT`).

3 Model

In this section, we first identify some current issues and limitations associated with a commonly-used approach to representing text to motivate our methodology. Then, we detail the elements of the proposed language-independent, compositional model to address the problems.

3.1 Issues with the Current Approach

When applying deep learning models to NLP problems, a commonly used approach is to map words to dense real-valued vectors in a low-dimensional space with word lookup tables. Typically, for this approach to work well, one needs to train on a large corpus in an unsupervised fashion, e.g. `word2vec` (Mikolov et al., 2013a; Mikolov et al., 2013b) and `GloVe` (Pennington et al., 2014), in order to obtain a sensible set of embeddings. While this approach has demonstrated its strong capabilities of capturing syntactic and semantic information and been successfully applied to a number of NLP tasks (Socher et al., 2013; Kalchbrenner et al., 2014; Kim, 2014), as identified by Ling et al. (2015), there are two practical problems with it. First, given that language is flexible, previously unseen words are bound to occur regardless of the size of the unsupervised training corpus. This problem is even more pronounced when dealing with user-generated text, such as from social media (e.g. Twitter and Facebook) due to the noisy nature of such platforms – e.g. typos, ad hoc acronyms and abbreviations, phonetic substitutions, and even meaningless strings (Han and Baldwin, 2011). One simple solution is to represent all unknown words with a special UNK vector. However, this sacrifices the meaning of the unknown word which may be critical. Moreover, it is unable to generalise to made up words, for instance, *beautification*, despite the constituent words *beautiful* and *-ification* having been observed. Second, the large number of parameters for a model to learn tends to cause overfitting. Suppose a vector of d dimensions is used to represent each word and the word lookup table is therefore of size $d \times |V|$ where $|V|$ is the vocabulary size, which normally scales to the order of hundreds and thousands. Again, this problem is particularly serious in noisier domain.

In author profiling, a large array of character-based features have been explored and shown to be effective for trait inference, such as character flooding (Nowson et al., 2015; Giménez et al., 2015), character n-grams(González-Gallardo et al., 2015; Sulea and Dichiu, 2015), and emoticons (Nowson et al., 2015; Palomino-Garibay et al., 2015). This motivates our proposed model, described in the next section, where character, word and sentence representations are hierarchically constructed, independent of a specific language and capable of harnessing personality-sensitive signals buried as deep as the character level.

3.2 Character to Word to Sentence for Personality Traits

We address the identified problems in Section 3.1 by extending the compositional character to word

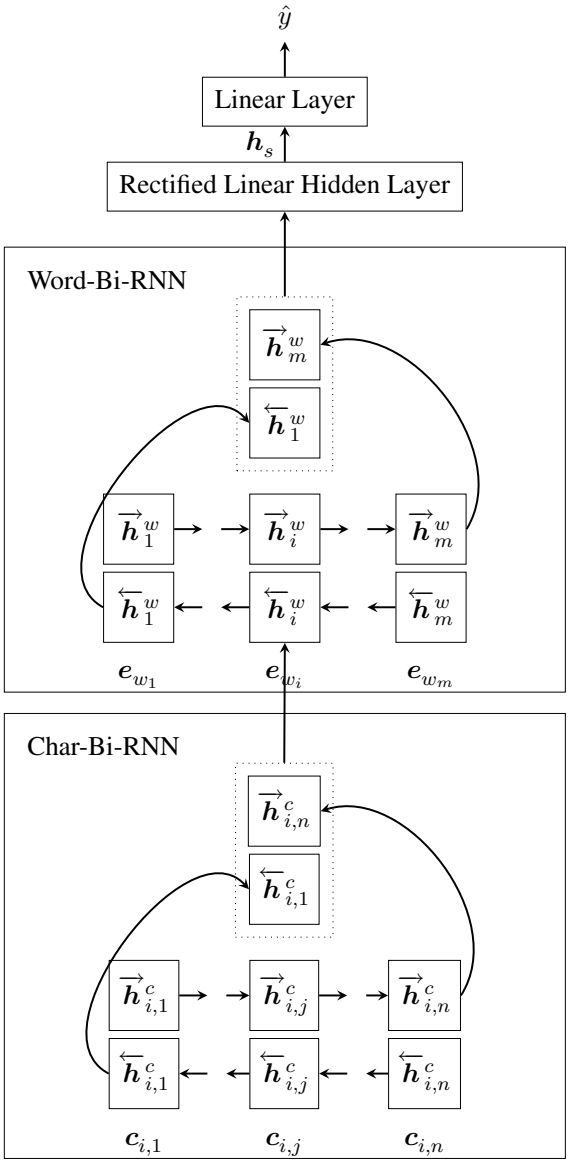

Figure 1: Illustration of the C2W2S4PT model. Dotted boxes indicate concatenation.

model (C2W) (Ling et al., 2015) wherein the constituent characters of each word is taken as input to a character-level bi-directional RNN (Char-Bi-RNN) to construct the representation of the word. A sentence is in turn represented, via another bi-directional RNN operating at the word level (Word-Bi-RNN), by the concatenation of the last and first hidden states of the forward and backward Word-RNNs respectively. Ultimately, a feedforward neural network predicts a scalar for a specific personality trait based on the input of the representation of a sentence. Given the hierarchical nature of the model, we name it C2W2S4PT (Character to Word to Sentence for Personality Traits) depicted in Figure 1. The formal definition is provided as follows where we illustrate C2W2S4PT with an example in which a sentence s is seen as a sequence of words $\{w_1, w_2, \ldots, w_i, \ldots, w_m\}$ and a word w_i is in turn a sequence of characters $c_{i,j}$ whose embedding is denoted: $\boldsymbol{c}_{i,j}$. Next, the Char-Bi-RNN takes as input the sequence of character embeddings $\{\boldsymbol{c}_{i,1}, \ldots, \boldsymbol{c}_{i,n}\}$ (assuming w_i is comprised of n characters) to construct the representation of word w_i, resulting in the word embedding $\boldsymbol{e}_{w_i}$. Here, the recurrent unit we employ in the Bi-RNNs is GRU as suggested by recent studies that GRUs achieve comparable, if not better, results to LSTM but are less demanding computationally (Chung et al., 2014; Kumar et al., 2015; Jozefowicz et al., 2015).[2] Concretely, the character embeddings are processed by the Char-Bi-RNN using the following:

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

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

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

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

where $\odot$ is the element-wise product, σ the sigmoid function, f the hyperbolic tangent function $\tanh$, $\overrightarrow{\boldsymbol{W}}^c_z, \overrightarrow{\boldsymbol{W}}^c_r, \overrightarrow{\boldsymbol{W}}^c_h, \overrightarrow{\boldsymbol{U}}^c_{hz}, \overrightarrow{\boldsymbol{U}}^c_{hr}, \overrightarrow{\boldsymbol{U}}^c_{hh}$ are the parameter matrices to learn, and $\overrightarrow{\boldsymbol{b}}^c_z$, $\overrightarrow{\boldsymbol{b}}^c_r$, $\overrightarrow{\boldsymbol{b}}^c_h$ the bias terms. In addition to the forward pass, the Char-Bi-RNN also processes the character sequence backwards (symbolised by $\overleftarrow{\boldsymbol{h}}^c_{i,j}$) with another set of GRU weight matrices and bias terms. Note that the same character embeddings are shared across the forward and backward pass. Eventually, we represent w_i as the concatenation of the last and first hidden states of the forward and backward Char-RNNs:

$$\boldsymbol{e}_{w_i} = \begin{bmatrix} \overrightarrow{\boldsymbol{h}}^c_{i,n} \\ \overleftarrow{\boldsymbol{h}}^c_{i,1} \end{bmatrix} \quad (5)$$

Sentence representations are built in a similar fashion to word representations with another Bi-RNN operating at the word level (Word-Bi-RNN) where $\boldsymbol{e}_{w_i}$ (for $i \in [1, n]$ once all the word repre-

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

sentations have been constructed from their constituent characters) are processed:

$$\overrightarrow{\boldsymbol{z}}_i^w = \sigma(\overrightarrow{\boldsymbol{W}}_z^w \boldsymbol{e}_{w_i} + \overrightarrow{\boldsymbol{U}}_{hz}^w \overrightarrow{\boldsymbol{h}}_{i-1}^w + \overrightarrow{\boldsymbol{b}}_z^w) \quad (6)$$

$$\overrightarrow{\boldsymbol{r}}_i^w = \sigma(\overrightarrow{\boldsymbol{W}}_r^w \boldsymbol{e}_{w_i} + \overrightarrow{\boldsymbol{U}}_{hr}^w \overrightarrow{\boldsymbol{h}}_{i-1}^w + \overrightarrow{\boldsymbol{b}}_r^w) \quad (7)$$

$$\overrightarrow{\tilde{\boldsymbol{h}}}_i^w = \mathrm{f}(\overrightarrow{\boldsymbol{W}}_h^w \boldsymbol{e}_{w_i} + \overrightarrow{\boldsymbol{r}}_i^w \odot \overrightarrow{\boldsymbol{U}}_{hh}^w \overrightarrow{\boldsymbol{h}}_{i-1}^w + \overrightarrow{\boldsymbol{b}}_h^w) \quad (8)$$

$$\overrightarrow{\boldsymbol{h}}_i^w = \overrightarrow{\boldsymbol{z}}_i^w \odot \overrightarrow{\boldsymbol{h}}_{i-1}^w + (1 - \overrightarrow{\boldsymbol{z}}_i^w) \odot \overrightarrow{\tilde{\boldsymbol{h}}}_i^w \quad (9)$$

where $\overrightarrow{\boldsymbol{W}}_z^w, \overrightarrow{\boldsymbol{W}}_r^w, \overrightarrow{\boldsymbol{W}}_h^w, \overrightarrow{\boldsymbol{U}}_{hz}^w, \overrightarrow{\boldsymbol{U}}_{hr}^w, \overrightarrow{\boldsymbol{U}}_{hh}^w$ are the parameter matrices to learn, and $\overrightarrow{\boldsymbol{b}}_z^w, \overrightarrow{\boldsymbol{b}}_{wr}^w, \overrightarrow{\boldsymbol{b}}_h^w$ the bias terms. The representation of the sentence is constructed, in a similar manner to how words are represented, by taking the concatenation of the last and first hidden states of the forward and backward Word-RNN:

$$\boldsymbol{e}_s = \begin{bmatrix} \overrightarrow{\boldsymbol{h}}_m^w \\ \overleftarrow{\boldsymbol{h}}_1^w \end{bmatrix} \quad (10)$$

Lastly, the score for a particular personality trait is estimated with an MLP, taking as input the sentence embedding $\boldsymbol{e}_s$ and returning the estimated score $\hat{y}_s$:

$$\boldsymbol{h}_s = \mathrm{ReLu}(\boldsymbol{W}_{eh}\boldsymbol{e}_s + \boldsymbol{b}_h) \quad (11)$$

$$\hat{y}_s = \boldsymbol{W}_{hy}\boldsymbol{h}_s + b_y \quad (12)$$

where ReLU (REctified Linear Unit) is defined as $\mathrm{ReLU}(x) = \max(0, x)$, $\boldsymbol{W}_{eh}, \boldsymbol{W}_{hy}$ the parameter matrices to learn, $\boldsymbol{b}_h, b_y$ the bias terms, and $\boldsymbol{h}_s$ the hidden representation of the MLP. All the trainable parameter/embedding matrices and bias terms are jointly optimised using *mean square error* as the objective function:

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_{s_i} - \hat{y}_{s_i})^2 \quad (13)$$

where y_{s_i} is the gold standard personality score of sentence s_i and θ the collection of all parameter/embedding matrices and bias terms for the model to learn. Note that no language-dependent component is present in the proposed model.

4 Experiments and Results

We evaluate our model in two settings, against models with or without feature engineering, to fully study the effectiveness of the proposed method. In the former, we compare – at the user level – our feature-engineering-free and language-independent model with current state-of-the-art models which make much use of linguistic features. In the latter, on the other hand, we investigate the performance against other feature-engineering-free models on individual short texts. In both settings, we show that our model achieves better results across two language (English and Spanish) and is equally competitive in Italian.

4.1 Dataset and Preprocessing

The dataset we adopt in this paper is the English, Spanish and Italian data from the PAN 2015 Author Profiling task dataset (Rangel et al., 2015), collected from Twitter and consisting of $14,166$ English (EN), $9,879$ Spanish (ES) and $3,687$ Italian (IT) tweets (from 152, 110 and 38 users respectively). Due to space constraints and the limited size of the data, the Dutch dataset is not included. Each user encompasses a set of tweets (average $n = 100$) with gold standard personality labels, the five trait labels (essentially scores between -0.5 and 0.5), calculated following the author's self-assessment responses to the short Big 5 test, BFI-10 (Rammstedt and John, 2007) which has the most solid grounding in language and is considered to be the most widely accepted and exploited scheme for personality recognition (Poria et al., 2013).

In our experiments, we tokenise each tweet with Twokenizer (Owoputi et al., 2013) to preserve user mentions and hashtag-preceeded topics. User mentions and URLs, unlike the majority of the language used in tweets, are intended for their targets, whose surface forms are deemed hardly informative. We therefore further normalise these features to single characters (e.g., *@username* → @, *http://t.co/* → ^), limiting the risk of modelling unnecessary character usage not directly influenced by nor reflecting the personality of the author.

4.2 Evaluation Metric

As the test corpus is unavailable, withheld by the PAN 2015 organisers, k-fold cross-validation is used instead to compare the performance ($k = 5$ or 10) on the available dataset. To evaluate the performance, we measure the Root Mean Square Error (RMSE) at either the tweet 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 y_{s_i}

and $\hat{y}_{s_i}$ are the gold standard and predicted personality trait score of the i^{th} tweet whereas y_{user_i} and $\hat{y}_{user_i}$ are their user-level counterparts, T and U the total numbers of tweets and users in the corpus. Note that in the dataset utilised in this work, each user is assigned a single score for a particular personality trait and every tweet collected from the same account inherits the same five personality trait assignments as its author. The predicted user level trait score is calculated: $\hat{y}_{user_i} = \frac{1}{T_i}\sum_{j=1}^{T_i}\hat{y}_{s_j}$ where T_i is the total number of tweets of $user_i$. In Section 4.3 and 4.4, the results, measured with $RMSE_{user}$ and $RMSE_{tweet}$, in the two settings, i.e. against models with and without feature-engineering, are presented respectively. Consistent with prior work in author profiling (Sulea and Dichiu, 2015; Mirkin et al., 2015; Nowson et al., 2015), we employ exactly the same evaluation metric on the same dataset to enable direct comparison.

4.3 Evaluation against State-of-the-art Models

We present the results obtained by the proposed model tested on the dataset described in Section 4.1. Note that our model is trained to predict personality trait scores, relying only on the text without any additional features. To enable direct comparison, we evaluate `C2W2S4PT` on the user level against current state-of-the-art models which incorporate linguistic features based on psychological studies.

For 5-fold cross-validation, we select the tied-highest ranked (in EN under evaluation conditions) amongst the PAN 2015 participants (Sulea and Dichiu, 2015) (also ranked 7th and 4th in ES and IT).[3] Similarly, we choose baselines by ranking and metric reporting for 10-fold cross validation (Nowson et al., 2015) (ranked 9th, 6th and 8th in EN, ES and IT). In addition to the above works which predicted scores on text level and then averaged for each user, we also include subsequent work by (Mirkin et al., 2015) who reported results on concatenated tweets (a single document per author). Also, there is the most straightforward baseline `Average Baseline` assigning the average of all the scores to each user. We train `C2W2S4PT` with Adam (Kingma and Ba, 2014) over 100 epochs with a batch size of 32 and the following hyper-parameters: $\overrightarrow{\boldsymbol{h}}^{c}_{i,j}$ and $\overleftarrow{\boldsymbol{h}}^{c}_{i,j} \in \mathbb{R}^{256}$, $\boldsymbol{E}_c \in \mathbb{R}^{50\times|C|}$, dropout rate to the embedding output: 0.5, $\overrightarrow{\boldsymbol{h}}^{w}_{i}$ and $\overleftarrow{\boldsymbol{h}}^{w}_{i} \in \mathbb{R}^{256}$, $\boldsymbol{W}_{hy} \in \mathbb{R}^{256\times1}$, $b_y \in \mathbb{R}$, $\boldsymbol{W}_{eh} \in \mathbb{R}^{512\times256}$, $\boldsymbol{b}_h \in \mathbb{R}^{256}$. The $RMSE_{user}$ results are presented in Table 1 where EXT, STA, AGR, CON and OPN are abbreviations for Extroversion, Emotional Stability (the inverse of Neuroticism), Agreeableness, Conscientiousness and Openness respectively.

`C2W2S4PT` outperforms the current state of the art in EN and ES. In the 5-fold cross-validation group, `C2W2S4PT` demonstrates its advantages, attaining superior performance to the baselines except for CON in ES. In terms of 10-fold cross validation, the superiority of our model is even more evident, supported by the dominating performance over the two selected baselines across all personality traits and two languages. In both groups, 5 or 10-fold cross validation, not only does `C2W2S4PT` outperform the baseline systems, particularly significantly in the 10-fold group, it also does so without the aid of any handcrafted features, stressing the technical soundness of `C2W2S4PT`.

On CON in ES, 5-fold cross-validation. We suspect that the surprisingly good performance of Sulea and Dichiu (2015) may likely be attributed to overfitting. Indeed, the performance on the test set on CON in ES is even inferior to Nowson et al. (2015), further confirming our speculation.

The superiority of `C2W2S4PT` is less clear in IT. This can possibly be caused by the inadequate amount of Italian data, less than 4k tweets as compared to 14k and 10k in the English and Spanish datasets, limiting the capability of `C2W2S4PT` to learn a reasonable model.

4.4 Evaluation against Other Feature-engineering-free Methods

While it is common practice in personality trait inference to evaluate at the user level, we also look into tweet-level performance to further study the models' capabilities at a more fine-grained level. A number of baselines, incorporating only the surface form of the text for the purpose of fair comparison, have been created to support our evaluation. First, we inherit the same `Average Baseline` as in Section 4.3. Next, we select two BoW-based system, `Random Forest` and `SVM Regression`, and

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

Lang.	k	Model	EXT	STA	AGR	CON	OPN
EN	—	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**
	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**
ES	—	Average Baseline	0.171	0.203	0.163	0.187	0.166
	5	Sulea and Dichiu (2015)	0.152	0.181	0.148	**0.114**	0.142
		C2W2S4PT	**0.148**	**0.177**	**0.143**	0.157	**0.136**
	10	Mirkin et al. (2015)	0.153	0.188	0.155	0.156	0.160
		Nowson et al. (2015)	0.154	0.188	0.155	0.168	0.160
		C2W2S4PT	**0.145**	**0.177**	**0.142**	**0.153**	**0.137**
IT	—	Average Baseline	0.162	0.172	0.162	0.123	0.151
	5	Sulea and Dichiu (2015)	**0.119**	0.150	0.**122**	0.101	**0.130**
		C2W2S4PT	0.124	**0.144**	0.130	**0.095**	0.131
	10	Mirkin et al. (2015)	**0.095**	0.168	0.142	0.098	0.137
		Nowson et al. (2015)	0.137	0.168	0.142	0.098	0.141
		C2W2S4PT	0.118	**0.147**	**0.128**	**0.095**	**0.127**

Table 1: $RMSE_{user}$ across five traits. **Bold** highlights best performance.

perform grid search for the best hyper-parameter setup ranging: kernel $\in$ {linear, 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 $\{10, 50, 100, 500, 1000\}$.

In addition to the above conventional machine-learning-based models, we further implement two simpler RNN-based models, Bi-GRU-Char and Bi-GRU-Word, which work only on the character and word level respectively. On top of the GRUs, both Bi-GRU-Char and Bi-GRU-Word share the same MLP classifier, $\boldsymbol{h}_s$ and $\hat{y}_s$, as in C2W2S4PT. For training, we use the same set of hyper-parameters as described in Section 4.3 for C2W2S4PT. Similarly, we set the character and word embedding size to 50 and 256 for Bi-GRU-Char and Bi-GRU-Word respectively. Hyper-parameter fine-tuning was not performed mainly due to time constraints. We present the $RMSE_{tweet}$ of each effort, measured by 10-fold stratified cross-validation, in Table 2.

C2W2S4PT is comparable with, if not superior to, the strong baselines SVM Regression and Random Forest in EN and ES. C2W2S4PT achieves state-of-the-art results in almost every trait except for two, AGR in EN and STA in ES. It is worth noting that C2W2S4PT generates this competitive performance, in the feature-engineering-free setting, against SVM Regression and Random Forest without exhaustive hyper-parameter fine-tuning.

C2W2S4PT achieves better performance than the RNN-based baselines in EN and ES. Compared with Bi-GRU-Word, C2W2S4PT is less prone to overfitting because of the relatively fewer parameters for the model to learn whereas Bi-GRU-Word needs to maintain a large vocabulary embedding matrix (Ling et al., 2015). In regards to Bi-GRU-Char, the success can be attributed to C2W2S4PT'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.

The performance of C2W2S4PT is inferior to Bi-GRU-Word in IT. Bi-GRU-Word achieves the best performance across all personality traits with C2W2S4PT coming in as a close second and tying in 3 traits. Apart from the inadequate amount of Italian data causing the fluctuation in performance as explained in Section 4.3, further investigation is needed to analyse the strong performance of Bi-GRU-Word.

Lang.	Model	EXT	STA	AGR	CON	OPN
EN	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**
ES	Average Baseline	0.171	0.204	0.163	0.187	0.165
	SVM Regression	**0.158**	**0.190**	0.157	0.171	0.152
	Random Forest	0.159	0.195	0.157	0.177	0.158
	Bi-GRU-Char	0.163	0.195	0.158	0.178	0.155
	Bi-GRU-Word	0.159	0.192	0.154	0.173	0.154
	C2W2S4PT	**0.158**	0.191	**0.153**	**0.168**	**0.150**
IT	Average Baseline	0.164	0.171	0.164	0.125	0.153
	SVM Regression	0.141	0.159	0.145	0.113	**0.141**
	Random Forest	0.140	0.161	**0.140**	0.111	0.147
	Bi-GRU-Char	0.149	0.163	0.153	0.117	0.146
	Bi-GRU-Word	**0.135**	**0.156**	**0.140**	**0.109**	**0.141**
	C2W2S4PT	0.139	**0.156**	0.143	**0.109**	**0.141**

Table 2: $RMSE_{tweet}$ across five traits level. **Bold** highlights best performance.

4.5 Visualisation

In order to investigate the features automatically learned by the models, we select C2W2S4PT trained on a single personality trait (EXT) and visualise the 2D PCA (Tipping and Bishop, 1999) scatter plot of the representations of the sentences.[4] As examples, we randomly select 100 tweets, 50 each from either extreme of the EXT spectrum - Extraversion being selected for this exercise as it is the most commonly studied and well understood trait. The text representations are automatically constructed by C2W2S4PT, with the resultant plot presented in Figure 2. Here, two clusters are easily identifiable, representing positive and negative Extraversion, with the former intersecting the latter. We consider three examples, highlighted in Figure 2, for discussion.

- 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, although essentially similar in terms of semantics,

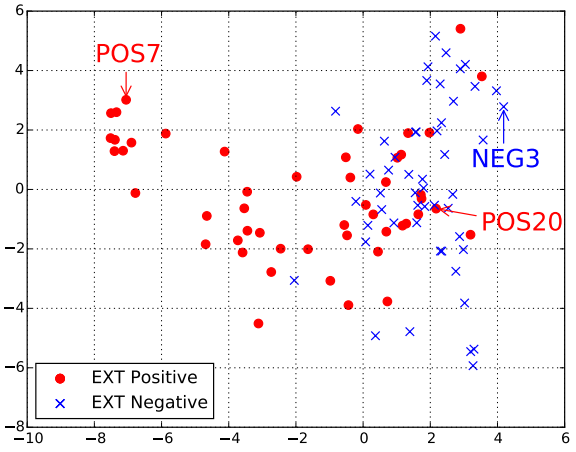

Figure 2: Scatter plot of sentence representations processed by PCA.

are placed distantly from each other at the far ends of the distribution. Despite the semantic similarities, the linguistic attributes they possess are commonly understood to be associated with Extraversion differently (Gill and Oberlander, 2002): the longer tweet, POS7, together with its use of the second person pronoun, suggests that the author is more inclusive of others while NEG3, on the other hand, is self-focused and shorter, ele-

[4]We also experimented with t-SNE (Van der Maaten and Hinton, 2008) but it did not produce an interpretable plot.

ments signifying Introversion. The third example, POS20, while appearing to be mapped to an Introvert space, is a tweet from an Extravert. Apart from being short, POS20 incorporates the use of non-rotated, "Eastern" style emoticons (*o.O*), aspects shown to be linked 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 worth noting that the model is capable of capturing, without feature engineering, well-understood dimensions of language.

5 Conclusion and Future Work

Overall, the results in this paper demonstrate the validity of our methodology: not only does `C2W2S4PT` provide state-of-the-art results compared to previous feature-engineering-heavy works, but it also performs well when compared with other widely used strong baselines in the feature-engineering-free setting. More importantly, the lack of feature engineering enables us to adapt the same model, with zero alteration to the model itself, to other languages. To further examine this property of the proposed model, we plan to explore the TwiSty dataset (Verhoeven et al., 2016), a recently introduced corpus consisting of 6 languages and labelled with MBTI type indicators (Myers and Myers, 2010).

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*.

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.

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.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. 2015. An empirical exploration of recurrent network architectures. In *Proc. ICML*, pages 2342–2350. JMLR Workshop and Conference Proceedings.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proc. ACL*, Baltimore, USA.

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.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proc. EMNLP*, Doha, Qatar.

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.

Ankit Kumar, Ozan Irsoy, Jonathan Su, James Bradbury, Robert English, Brian Pierce, Peter Ondruska, Ishaan Gulrajani, and Richard Socher. 2015. Ask me anything: Dynamic memory networks for natural language processing. *arXiv preprint arXiv:1506.07285*.

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, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proc. ICLR*, Scottsdale, USA.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013b. 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, 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. This article is published by LIWC Inc, Austin, Texas 78703 USA in conjunction with the LIWC2015 software program.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proc. EMNLP*, pages 1532–1543, Doha, Qatar.

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*, CEUR Workshop Proceedings.

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 Tom De Smedt. 2013. Ensemble Methods for Personality Recognition. In *Proceedings of WCPR13, Workshop on Computational Personality Recognition at ICWSM13 (7th International Conference on Weblogs and Social Media)*.

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.