Automated Classification of Written Proficiency Levels on the CEFR-Scale through Complexity Contours and RNNs

Elma Kerz[1] **Daniel Wiechmann**[2] **Yu Qiao**[1] **Emma Tseng**[3] **Marcus Ströbel**[1]
[1]RWTH Aachen University, [2]University of Amsterdam, [3]University of Washington
elma.kerz@ifaar.rwth-aachen.de, d.wiechmann@uva.nl
yu.qiao@rwth-aachen.de, eftseng@uw.edu
marcus.stroebel@rwth-aachen.de

Abstract

Automatically predicting the level of second language (L2) learner proficiency is an emerging topic of interest and research based on machine learning approaches to language learning and development. The key to the present paper is the combined use of what we refer to as 'complexity contours', a series of measurements of indices of L2 proficiency obtained by a computational tool that implements a sliding window technique, and recurrent neural network (RNN) classifiers that adequately capture the sequential information in those contours. We used the EF-Cambridge Open Language Database (Geertzen et al., 2014) with its labelled Common European Framework of Reference (CEFR) levels (Council of Europe, 2018) to predict six classes of L2 proficiency levels (A1, A2, B1, B2, C1, C2) in the assessment of writing skills. Our experiments demonstrate that an RNN classifier trained on complexity contours achieves higher classification accuracy than one trained on text-average complexity scores. In a secondary experiment, we determined the relative importance of features from four distinct categories through a sensitivity-based pruning technique. Our approach makes an important contribution to the field of automated identification of language proficiency levels, more specifically, to the increasing efforts towards the empirical validation of CEFR levels.

1 Introduction

The Common European Framework of Reference (CEFR) is an internationally recognized standard for describing language proficiency based on six reference levels – A1, A2, B1, B2, C1 and C2 – the same letter pairs corresponding to a three level distinction between beginner, intermediate and advanced (Council of Europe, 2018). Each proficiency level is related to specific linguistic features and skills, establishing a progression from rudimentary language to varied and sophisticated language. The CEFR descriptors, available for all four fundamental language skills (receptive skills: reading & listening and productive skills: writing & speaking), describe the expected competencies in terms of functional *can-do* statements. For example, a learner at a vantage or upper intermediate B2 level in the domain of writing is expected to have "a sufficient range of language to be able to give clear descriptions, express viewpoints and develop arguments without much conspicuous searching for words, using some complex sentence forms to do so" (Council of Europe, 2018: 131). The *can-do* descriptors formulated for each of the six CEFR proficiency levels are typically vague and subjective and are only useful for orientation purposes. Thus, there is an urgent need for research on empirical validation of CEFR levels at the interface between areas of language learning, testing and assessment, and natural language processing and machine learning (Wisniewski, 2017).

A closely related line of research has been directed toward automated essay scoring (AES) (for overviews see (Higgins et al., 2015; Ke and Ng, 2019; Klebanov and Madnani, 2020)). This line of research has benefited from the increasing availability of publicly accessible learner corpora of L2 writing, such as CLC-FCE (Yannakoudakis et al., 2011), TOEFL11 (Blanchard et al., 2013), MERLIN (Boyd et al., 2014) and EFCAMDAT (Geertzen et al., 2014). Supervised approaches to AES have recast the task as (1) a regression task aimed at predicting the score of an essay (Yannakoudakis et al., 2011; Klebanov and Flor, 2013), (2) a classification task aimed at classifying a text as belonging to one of a specified number of classes, e.g. the three score levels (low, medium or high) in the TOEFLL11 corpus or the six CEFR levels (A1-C2) in the MERLIN corpus (Hancke and Meurers, 2013; Pilán et al., 2016; Vajjala and Rama,

Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications, pages 199–209

2018; Weiß and Meurers, 2018; Caines and Buttery, 2020)[1] or (3) a ranking task aimed at ranking two or more texts based on their quality (Yannakoudakis and Briscoe, 2012; Taghipour and Ng, 2016). Previous work on AES has taken both feature-based approaches and neural approaches (see (Ke and Ng, 2019) for a discussion of (dis)advantages of these two approaches). The features employed are diverse, ranging from the use of descriptive metrics of the text related to word or sentence length to more abstract features related to proficiency development in the area of (second) language learning ((Vajjala, 2018) for a recent overview). The existing studies that have used a feature-based approach have typically relied on text averages of a given feature. However, the use of such aggregate scores may obscure the considerable degree of variation in distribution of feature values within the text.

In this paper, we present experiments geared towards the automated assessment of written language proficiency of non-native learners (L2) of English. For the experiments, we take advantage of the EF-Cambridge Open Language Database (EFCAMDAT, (Geertzen et al., 2014)), a large-scale learner corpus consisting of 1.8 million texts labeled with the six CEFR proficiency levels (A1-C2). The aim of the paper is twofold: (1) to apply a sliding window technique in a feature-based modeling approach to automated proficiency classification and (2) to determine what features contribute the most to the classification accuracy. The features employed in this paper are derived from numerous studies in the field of L2 acquisition centering around the notion of 'complexity'[2] (see e.g. (Lu, 2010a, 2012; Bulté and Housen, 2012)). The inclusion of such features is further motivated by the fact that, according to the CEFR descriptors, learners are expected to acquire the ability to produce increasingly varied and sophisticated written language, as they progress through the six proficiency levels. Such diverse and sophisticated language use should be evident not only in vocabulary growth, but also in the choice of individual words and multi-word phrases, and in the complexity of sentence, clause, and phrase structures. Through the sliding window technique we obtain a series of measurements for a given feature tracking the progression of complexity within a text in a sentence-by-sentence fashion. We refer to such series of measurements as 'complexity contours'. These contours are then fed into recurrent neural network (RNN) classifiers – adequate to take into account the sequential information in the contours – to perform grade-level classification tasks. We demonstrate the utility of the approach by comparing the performance of 'contour-based' RNN models against those of 'means-based' RNN models trained on text-average performance scores.

In a second step, we determine what features drive classification accuracy through a Sensitivity-Based Pruning (SBP) technique. The approach taken in this paper was already successfully applied in the area of first language (L1) writing development. Kerz et al. (2020) showed that RNN classifiers trained on complexity contours achieve higher classification accuracy in predicting secondary school children's grade levels in both English and German (second-, sixth-, ninth- and eleventh-grade in English schools and fifth- and ninth-grade in German). Here we set out to extend the approach to automated proficiency classification in L2 English. The remainder of the paper is organized as follows: In Section 2, we provide a concise overview of related work. Section 3 presents the dataset and Section 4 the features used in the experiments. Section 5 describes the sliding-window approach to generating complexity contours. Sections 6 presents the architecture of the RNNs and the training procedure. Sections 7 introduces the SBP method used to determine the relative feature importance. Section 8 reports the results before conclusions are drawn in Section 9 along with indications of future research directions.

2 Related work

In this section, we present two types of previous work: L2 studies that have investigated the relationship between certain linguistic features and proficiency levels, and those that have used supervised machine learning approaches to predict learner proficiency on the CEFR scale.

Numerous studies reveal that syntactic complexity can be considered as one of the key skills that strongly influence L2 proficiency (see e.g. (Ortega, 2003) for a synthesis of twenty-five studies, see

[1]The latter paper has been published in the context of a recent shared task on Language Proficiency Scoring at the LREC 2020 – REPROLANG Task D.2 https://lrec2020.lrec-conf.org/en/reprolang2020/selected-tasks/

[2]Complexity – commonly defined as "the range of forms that surface in language production and the degree of sophistication of such forms" (Ortega 2003:492) – is one of the three dimensions of the 'Complexity–Accuracy–Fluency' triad that has emerged as a prominent conceptual framework for L2 assessment (see e.g. (Wolfe-Quintero et al., 1998; Larsen-Freeman, 2006)

also (Kuiken et al., 2019) for a recent special issue). These studies have measured this multidimensional construct along both global features, such as length measures and subordination ratios, as well as more specific features pertaining to the usage of particular structures. For example, Lu (2011) conducted an evaluation of 14 features of syntactic complexity in a corpus of English essays written by Chinese L1 students and found that the complexity measures that best discriminated between proficiency levels were the number of complex nominals per sentence and the mean sentence length. Another series of studies indicate the importance of lexical complexity (aka lexical richness) subsuming its three sub-dimensions (lexical density, sophistication and variation) in the assessment of L2 proficiency (see (Lu, 2012)). For example, Kyle and Crossley (2014) showed 47.5% of the variance in holistic scores of lexical proficiency and 48.7% of the variance in holistic scores of speaking proficiency can be explained using a range of lexical sophistication indices. This study also introduced the use of multi-word sequences (MWS) as an indicator of language proficiency, operationalized in terms of register-specific n-gram measures (bigrams and trigrams). The inclusion of such features reflects the growing interest of MWS in language learning and development. This interest stems from an extensive body of evidence demonstrating that both child and adult populations, including adult second-language learner populations, can develop the sensitivity to the statistics of MWS and rely on knowledge of such statistics to facilitate their language processing and boost their acquisition (for overviews see e.g. (Shaoul and Westbury, 2011; Christiansen and Arnon, 2017)). Garner et al. (2020), for instance, investigated the relationship of the usage of MWS and human judgments of writing proficiency based on the CEFR-graded Yonsei English Learner Corpus (Rhee and Jung, 2014) and found that essays from higher CEFR levels include a greater proportion of frequent academic trigrams and more strongly associated spoken trigrams. Finally, in recent years the L2 literature has introduced information-theoretic features based on Kolmogorov complexity (Ehret, 2016; Ehret and Szmrecsanyi, 2019). Ehret and Szmrecsanyi (2019) investigated essays written by advanced learners of English from International Corpus of Learner English (Granger et al., 2002) and showed that more advanced learners use considerably more complex texts than beginner learners, although this tendency is not always reflected in a clear, linear relationship between proficiency and complexity.

Studies that have employed supervised machine learning approaches to predict proficiency on the CEFR scale for different L2s have used numerous linguistic features in combination with a host of classifiers (Hancke and Meurers, 2013; Volodina et al., 2016; Vajjala and Rama, 2018; Vajjala and Lõo, 2014; Ballier et al., 2019; Caines and Buttery, 2020). The classification accuracy reported in these studies ranged between 62.7% and 83.8%. Hancke and Meurers (2013) reached a classification accuracy of 62.7% in predicting five (out of six) CEFR levels of professionally rated free text essays from the MERLIN database comprising CEFR exams taken by second language learners of German based on a total of 3821 lexical, morphological, and syntactic features using the Sequential Minimal Optimization (SMQ) algorithm implemented in WEKA. Using the same SMQ algorithm, Volodina et al. (2016) achieved an accuracy of 67% in correctly identifying the CEFR level of L2 Swedish learner essays on the basis of 61 count-based, lexical, syntactic, morphological, and semantic features extracted from the linguistic annotation available in the SweLL corpus[3]. Ballier and Gaillat (2019) achieved 70% accuracy in predicting CEFR-levels of L1 French and Spanish L2 English users on the basis of manually annotated errors in the L1 French and Spanish subsets of the EFCAMDAT corpus. Vajjala and Lõo (2014) reported a classification accuracy of 79% in an experiment on the Estonian Interlanguage Corpus[4].

In another study, Vajjala and Rama (2018) performed experiments with cross-lingual and multilingual classifiers on individual language classification. The data used in their study included 2,286 manually graded texts (five levels, A1 to C1) from the MERLIN learner corpus (German, 1,029 texts; Italian, 803 texts, and Czech, 434 texts). Trained on a wide range of feature, such as word and POS n-grams, task-specific word and character embeddings, dependency n-grams, features pertaining to lexical richness and error features, their classification models obtained an accuracy of 0.68 for German, 0.84 for Italian and 0.73 for Czech for monolingual classification. For multilingual classification, their models reached classification accuracy up to 0.73. The data set and findings ob-

[3] https://spraakbanken.gu.se/eng/research/icall/swellcorpus

[4] http://evkk.tlu.ee/?language=en

tained in this study have served as the baseline for the REPROLANG 2020 shared task on 'Language proficiency scoring'[5]. In the context of this task, Caines and Buttery (2020) reproduce and extend the finding described in (Vajjala and Rama, 2018) reaching a classification accuracy of up to 83.8% for the Italian component. Their results indicate that feature-based approaches perform better than neural network classifiers for text datasets of the given size.

3 Data

The data come from the EFCAMDAT, an open access corpus compiled at Cambridge University in collaboration with EF Education First, an international school of English as a second/foreign language (Geertzen et al., 2014). The corpus consists of writing assignments submitted to the Englishtown, the online school of EF Education First, summing up to a total of 1,180,309 individual writing samples by 174,743 L2 learners. The curriculum of Englishtown covers all six proficiency levels, from CEFR A1 to C2 organized along 16 EF teaching levels with each level subsuming 8 teaching units and ending with an open-ended writing task (128 distinct writing assignments). The length of the writing samples in the corpus increases monotonically with Englishtown levels, ranging from an average of 30.1 words at the lowest level (1) to an average of 170 words at the highest level (16). Since one of our main aims is to demonstrate the usefulness of the sliding window technique and the inclusion of a set of measurements per individual feature (complexity contours), we filtered the original dataset to obtain texts containing at least 100 words. This resulted in a total of 163,657 writing samples. In addition, we removed texts that had received exceptionally low scores on writing performance, since their inclusion would add bias and variance and may skew the results (Bøvelstad et al., 2017). Specifically, texts whose writing score fell more than 1.5 times the interquartile range below the first quartile, corresponding to a threshold score of 75%, were removed, which resulted in a loss of 7% of the data. The final dataset comprised a total of 152,314 individual learner texts. whose distributions across CEFR levels along with associated text length statistics are shown in Table 1.

[5] http://www.lrec-conf.org/proceedings/lrec2020/index.html

Table 1: Distribution of texts by CEFR proficiency level and text length statistics (in words)

CEFR	N texts	Mean length	SD
A1	8313	132.24	42.62
A2	19587	119.70	27.22
B1	61396	118.86	24.94
B2	48535	142.89	35.03
C1	12831	174.08	37.36
C2	1652	180.79	63.49

4 Features

The 57 features used in this paper fall into four distinct groups: (1) measures of syntactic complexity, (2) measures of lexical richness, (3) measures pertaining to the usage of multi-word sequences (MWS) and (4) information-theoretic measures. The first group consists of 16 features used in the past to measure syntactic complexity in writing and its relation to writing proficiency reviewed in Section 2. These features are implemented based on descriptions in Lu (2010b) and using the Tregex tree pattern matching tool (Levy and Andrew, 2006) with syntactic parse trees for extracting specific patterns. The second group subsumes 13 features pertaining to lexical richness: five measures of lexical variation, one measure of lexical density, seven measures of lexical sophistication. The operationalizations of these measures follow those described in Lu (2012) and (Ströbel, 2014). The third group includes 25 n-gram frequency features that are derived from the five register sub-components of the Contemporary Corpus of American English (COCA, (Davies, 2008)): spoken, magazine, fiction, news and academic language[6]. Our frequency n-gram measures differ from those used in the earlier studies reviewed in Section 2. Instead of using only bigrams and trigrams, we extend them to include longer word combinations (four- and five-grams) and use a more nuanced definition to operationalize the usage of such combinations given in equation (1):

$$\text{Norm}_{n,s,r} = \frac{|C_{n,s,r}| \cdot \log \left[\prod_{c \in |C_{n,s,r}|} freq_{n,r}(c)\right]}{|U_{n,s}|} \quad (1)$$

Let $A_{n,s}$ be the list of n-grams ($n \in [1,5]$) appearing within a sentence s, $B_{n,r}$ the list of n-

[6] The Contemporary Corpus of American English is the largest genre-balanced corpus of American English, which at the time the measures were derived comprised of 560 million words.

gram appearing in the n-gram frequency list of register r ($r \in \{\text{acad}, \text{fic}, \text{mag}, \text{news}, \text{spok}\}$) and $C_{n,s,r} = A_{n,s} \cap B_{n,r}$ the list of n-grams appearing both in s and the n-gram frequency list of register r. $U_{n,s}$ is defined as the list of unique n-gram in s, and $freq_{n,r}(a)$ the frequency of n-gram a according to the n-gram frequency list of register r.

A total of 25 measures results from the combination of (a) a 'reference list' containing the top 100,000 most frequent n-grams and their frequencies from one of five register subcomponents of the COCA corpus and (b) the size of the n-gram ($n \in [1, 5]$). The fourth group includes three information-theoretic measures that are based on Kolmogorov complexity. These measures use the Deflate algorithm (Deutsch, 1996) to compress a text and obtain complexity scores by relating the size of the compressed file to the size of the original file (for the operationalization and implementation of these measures see (Ströbel, 2014)).

5 A Sliding-Window Approach

Text complexity of the writing samples was automatically assessed using the CoCoGen, a computational tool that implements a sliding-window technique to generate a series of measurements for a given complexity measure (CM) (Ströbel 2014). In contrast to the standard approach that represents text complexity as a single score, providing a 'global assessment' of the complexity of a text, the use of a sliding-window technique enables a 'local' (sentence-level) assessment of complexity within a text. A sliding window can be conceived of as a window of size ws, which is defined by the number of sentences it contains. The window is moved across a text sentence-by-sentence, computing one value per window for a given CM. The series of measurements generated by CoCoGen captures the progression of linguistic complexity within a text for a given CM and is referred here to as a 'complexity contour' (see Figure 1). To compute the complexity score of a given window, a measurement function is applied to each sentence in the window. The size of the window (ws) is user-defined parameter whose optimal value depends on the goals of the analysis: When complexity is measured for each sentence, i.e. $ws = 1$, the resulting complexity contour will typically exhibit many sharp turns. By increasing the window size, i.e. the number of sentences in a window, the complexity contour can be smoothened akin to a moving average technique.[7] In this paper, the window size parameter was set to one sentence, meaning that no smoothing of the curve was performed. Figure 1 illustrates complexity contours on three randomly selected texts across three CEFR levels (A2, B2 and C2) for eight selected complexity measures. CoCoGen uses the Stanford CoreNLP suite (Manning et al., 2014) for performing tokenization, sentence splitting, part-of-speech tagging, lemmatization and syntactic parsing (Probabilistic Context Free Grammar Parser (Klein and Manning, 2003)).

6 Classification Models

We used a Recurrent Neural Network (RNN) classifier, specifically a dynamic RNN model with Gated Recurrent Unit (GRU) cells (Cho et al., 2014). A dynamic RNN was chosen as it can handle sequences of variable length[8]. As shown in Figure 2, the input of the contour-based model is a sequence $X = (x_1, x_2, \ldots, x_l, x_{l+1}, \ldots, x_n)$, where x_i, the output of CoCoGen for the ith window of a document, is a 57 dimensional vector, l is the length of the sequence, $n \in \mathbb{Z}$ is a number, which is greater or equal to the length of the longest sequence in the data and $x_{l+1}, \cdots, x_n$ are padded $\mathbf{0}$-vectors. The input of the contour-based model was fed into a RNN that consists of two layers of GRU cells with 200 hidden units for each. To predict the class of a sequence, the last output of the RNN, i.e. the output of the RNN right after the feeding of x_l, is transformed through a feed-forward neural network. The feed-forward neural-network consists of three fully connected layers, whose output dimensions are 512, 256, and 6 respectively. The Rectifier Linear Unit (ReLU) was used as an activation function. Before the final output, a softmax layer was applied. For the mean-based model, we used the same neural network as in the contour-based model, except that the network was trained with vectors of text-average complexity scores. The models were implemented using PyTorch (Pytorch, 2019).

We evenly split our data into 10 folds and applied a 10-fold cross validation, i.e. each time a fold (10% of data) is taken out as test set and the rest (90% of data) are used as the training set. In both

[7] When the window size is specified to be greater than 1, CoCoGen returns complexity scores for a given measures as fractions (wn_m/wd_m). In this case, the denominators and numerators of the fractions from the first to the last sentence in the window are added up to form the denominator and numerator of the resulting complexity score of a given window.

[8] The lengths of the feature vector sequences depends on the number of sentences of the texts in our corpus.

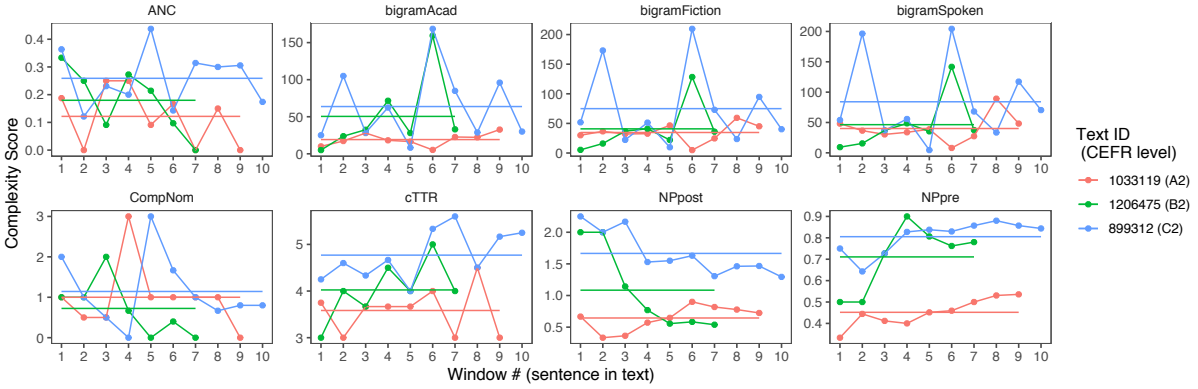

Figure 1: Complexity contours (window size = 1 sentence) for eight selected measures of complexity for three random texts from CEFR levels A2, B2 and C2.

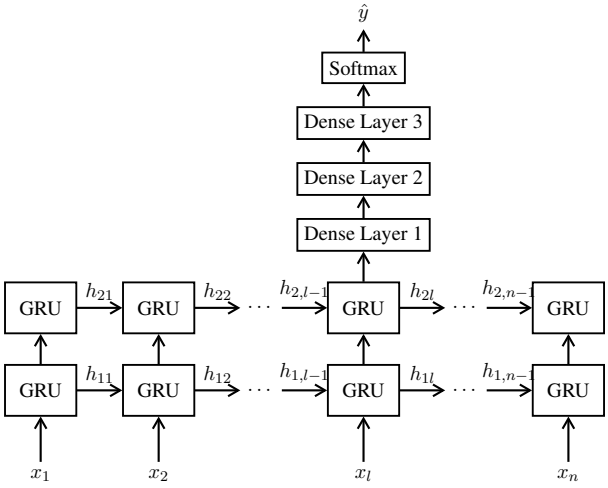

Figure 2: Roll-out of the contour-based RNN model based on complexity contours

datasets, the distributions of classes were identical. As the loss function for training cross entropy was used:

$$\mathcal{L}(\hat{Y}, c) = -\sum_{i=1}^{C} p(y_i) \log(p(\hat{y}_i))$$

in which c is the true class label of the current observation, C is the number of classes, $(p(y_1), \ldots, p(y_C))$ is a one-hot vector with

$$p(y_i) = \begin{cases} 1 & i = c \\ 0 & \text{otherwise} \end{cases}$$

and $\hat{Y} = (p(\hat{y}_1), p(\hat{y}_2), \ldots, p(\hat{y}_C))$ is the output vector of the softmax layer, which can be viewed as the predicted probabilities of the observed instance falling into to each of the classes. Since the EFCamDat dataset is hightly imbalanced, we additionally assigned weights to the classes for the cross entropy function, such that a_k is weight for class k:

$$a_k = \frac{N}{100 N_k}$$

where N is the total number of instance in the dataset and N_k is the number of instance in the dataset with label k. For optimization, we used Stochastic Gradient Descent (SGD) with a learning rate $\eta = 0.01$ $momentum = 0.9$ and a learning rate decay factor of 0.1. The minibatch size is 32, which was shown as a reasonable value for modern GPU (Masters and Luschi, 2018).

7 Feature Ablation

To determine the relative importance of the individual features, we conducted feature ablation experiments for the contour-based RNN. Classical forward or backward sequential selection algorithms that proceed by sequentially adding or discarding features require a quadratic number of model training and evaluation in order to obtain a feature ranking (Langley, 1994). In the context of neural network model training a quadratic number of models can become prohibitive. To alleviate this problem, we used an adapted version of the iterative sensitivity-based pruning algorithm proposed by Díaz-Villanueva et al. (2010). This algorithm ranks the features based on a 'sensitivity measure' (Moody, 1994; Utans and Moody, 1991) and removes the least relevant variables one at a time. The classifier is then retrained on the resulting subset and a new ranking is calculated over the remaining features. This process is repeated until all features are removed (see Algorithm 1). In this fashion, rather than training $\frac{n(n+1)}{2}$ required for sequential algorithms, the number of models trained is reduced to $\frac{n}{m}$, where m is the number of features that can be removed at each step. We report the results obtained with m = 1, i.e. the removal of a single feature at each step. The procedure of finding the rank order of feature importance is de-

scribed as following. To increase the robustness of the feature importance rank order, k-fold cross-validation is applied. At step t, neural network models $M_{t,n}, n \in \{1, \ldots, k\}$ are trained on the training sets of a k-fold cross-validation, where n is the fold ID. The training sets at step t consist of instances with feature set $F_t = \{f_1, f_2, \ldots, f_{D_t}\}$ where $f_1, \ldots f_{D_t}$ are the remaining features at the current step, whose importance rank is to be determined. We define $X_{t,n}$ as the test set of the nth fold with feature set F_t and $X^i_{t,n}$ as the same dataset as $X_{t,n}$ except we set the i^{th} feature f_i of each instance within the dataset to its average. Furthermore, we define $g(X)$ as the classification accuracy of $M_{t,n}$ for a dataset X. The sensitivity of feature f_i on the nth fold at step t is obtained from:

$$S_{i,t,n} = g(X_{t,n}) - g(X^i_{t,n})$$

The final sensitivity for a feature f_i at step t is:

$$S_{i,t} = \frac{1}{k} \sum_{n=1}^{k} S_{i,t,n}$$

The most important feature at step t can be found by:

$$f_{\hat{i}} : \hat{i} = \arg\max_{i: f_i \in F_t} (S_{i,t})$$

Then we set the rank for feature $f_{\hat{i}}$:

$$Rank_{\hat{i}} = t$$

In the end, feature $f_{\hat{i}}$ is dropped from F_t and the corresponding columns in training and test dataset are also dropped simultaneously:

$$F_{t+1} = F_t - \{f_{\hat{i}}\}$$

This procedure is repeated, until $|F_{t'}| = 1$.

Algorithm 1: Feature ablation algorithm

Input: N training instances with feature set $F = \{f_1, \ldots, f_D\}$
Input: m features to remove at each step
Result: $list$ containing the feature importance rank order

```
begin
    t ← 0
    list ← []
    while |F| > 0 do
        Train a classifier with |F| input features;
        Compute S_{i,t}, i ∈ F;
        Find f_{i_1}, ..., f_{i_m}, where S_{i_1,t}, ..., S_{i_m,t} are m largest among all S_{i,t}(i ∈ F) in descending order;
        list ← list.append([f_{i_1}, ..., f_{i_m}]);
        F ← F − {f_{i_1}, ..., f_{i_m}};
        t ← t + 1;
    return list
```

8 Results and Discussion

An overview of the performance statistics of the models in terms of precision, recall and F1 scores is presented in Table 2. The classification accuracy results indicated that the inclusion of complexity contours led to an increase in overall classification accuracy of 9.3% from 66.1% for the means-based RNN model to 75.4% for the contour-based RNN model. Classification performance of the contour-based RNN model was consistently higher than those of the means-based RNN model across all six CEFR proficiency levels. Its performance was higher for the beginner and intermediate CEFR proficiency levels (A1 to B2) with F1 scores ranging between 0.73 and 0.81 compared to the advanced levels (C1 and C2) with F1 scores dropping to 0.61 for the C1 level and 0.42 for the C2 level. The confusion matrix of the contour-based RNN model is presented in Table 3. As is evident in this table, most classification errors appeared in adjacent categories, with few classification errors occurring between distant categories.

The top 20 features that contributed most to the classification accuracy of the contour-based RNN model are shown in Table 2 (see the column 'Acc after Del'). The results of the feature ablation experiments revealed that classification accuracy was mainly driven by frequency n-grams measures pertaining to the usage of multiword sequences (MWS). The twelve of the top 20 features are uni-, bi-, and trigram frequency measures from all five register sub-components of the COCA corpus. Writing samples from higher CEFR levels exhibit higher scores for all five unigram measures. A similar pattern can be observed for bigram scores from the academic register. A more differentiated pattern is apparent in trigram measures: For example, trigram scores from the fiction register show a U-shaped progression, such that they first increase up to the B2 level and then decrease (see Table 5 and Figure 3 in the Appendix for an overview). Overall, these findings indicate the importance of including n-gram frequency measures for the task of automated language performance classification. Moreover, they are consistent with results reported

Table 2: Performance statistics of RNN classifiers (left) and results of feature ablation (right). Values in '()' indicate standard deviations. 'Base Mod Acc'= Accuracy of baseline model; 'Acc after Del' = Accuracy of model after deletion of feature (only top-20 features are shown).

Performance statistics	**Means-based** RNN Model	**Contour-based** RNN Model
Accuracy train	0.938 (0.012)	0.976 (0.012)
Accuracy test	0.661 (0.004)	**0.754** (0.004)
Precision_{A1}	0.677 (0.021)	0.784 (0.016)
Precision_{A2}	0.640 (0.013)	0.745 (0.007)
Precision_{B1}	0.710 (0.005)	0.795 (0.005)
Precision_{B2}	0.657 (0.007)	0.739 (0.004)
Precision_{C1}	0.479 (0.008)	0.632 (0.014)
Precision_{C2}	0.436 (0.050)	0.505 (0.043)
Recall_{A1}	0.677 (0.021)	0.784 (0.016)
Recall_{A2}	0.640 (0.013)	0.745 (0.007)
Recall_{B1}	0.710 (0.005)	0.795 (0.005)
Recall_{B2}	0.657 (0.007)	0.739 (0.004)
Recall_{C1}	0.479 (0.008)	0.632 (0.014)
Recall_{C2}	0.436 (0.050)	0.505 (0.043)
F1_{A1}	0.634 (0.010)	0.744 (0.010)
F1_{A2}	0.626 (0.009)	0.725 (0.007)
F1_{B1}	0.713 (0.004)	0.803 (0.002)
F1_{B2}	0.671 (0.006)	0.751 (0.003)
F1_{C1}	0.467 (0.007)	0.614 (0.008)
F1_{C2}	0.374 (0.044)	0.419 (0.038)

Feature importance		
CM	Base Mod Acc	Acc after Del
Bigram fic	0.754 (0.004)	0.684 (0.005)
Bigram acad	0.748 (0.003)	0.686 (0.005)
ANC	0.744 (0.006)	0.691 (0.006)
MLWs	0.732 (0.003)	0.689 (0.002)
MLWc	0.728 (0.004)	0.675 (0.005)
Bigram spok	0.720 (0.002)	0.664 (0.004)
Unigram acad	0.712 (0.005)	0.659 (0.004)
Trigram fic	0.712 (0.004)	0.665 (0.005)
BNC	0.706 (0.004)	0.669 (0.004)
Bigram news	0.695 (0.004)	0.661 (0.004)
Unigram fic	0.691 (0.004)	0.660 (0.006)
NGSL	0.687 (0.005)	0.655 (0.005)
LD	0.681 (0.004)	0.646 (0.003)
Unigram spok	0.667 (0.004)	0.631 (0.006)
Trigram news	0.670 (0.005)	0.634 (0.006)
Unigram mag	0.666 (0.005)	0.634 (0.006)
Unigram news	0.667 (0.004)	0.604 (0.005)
Bigram mag	0.661 (0.004)	0.613 (0.005)
KolDef	0.655 (0.003)	0.626 (0.004)
KolDefMor	0.653 (0.003)	0.622 (0.005)

Table 3: Confusion matrix of the contour-based RNN model (sum across 10-fold cross validation). The $C_{i;j}$ value is the number of predictions known to be in group i and predicted to be in group j.

	A1	A2	B1	B2	C1	C2
A1	5885	460	935	683	327	23
A2	428	13849	3754	1317	212	27
B1	526	2923	49842	7125	899	81
B2	428	1097	6951	37078	2715	266
C1	226	207	1115	3443	7652	188
C2	20	57	131	552	298	594

in numerous studies indicating that the knowledge of MWS is a key component of both L1 and L2 writing and speaking skills (see e.g. (Christiansen and Arnon, 2017; Garner et al., 2020; Saito, 2020). Another group of features that figures prominently in the top 20 list are five measures of lexical sophistication: Higher CEFR levels are characterized by higher proportions of unusual/advanced words and words of greater surface length (compare *same* vs. *equal* vs. *identical* vs. *tantamount*). These results replicate and extend the findings reported in Durrant and Brenchley (2019) and (Kerz et al., 2020). Both studies found that measures of lexical sophistication are good predictors of children's L1 writing development. And finally, the top-20 list includes two of the three information-theoretic measures, indicating that more advanced learners produce considerably more complex (i.e. informationally denser) texts than beginner learners. The fact that two measures from the smallest subset of CMs were ranked among the top-20 most important features is an indication of their usefulness in research on automated proficiency classification. As discussed in detail in Ehret (2016) and Ehret and Szmrecsanyi (2019), CMs based on Kolmogorov complexity have the potential to avoid some of the known problems of traditional metrics that are based on different measures of unit length and that involve frequencies of various types of forms, which gives rise to 'concept reductionism' (Ortega, 2012, 128).

9 Conclusion and Outlook

In this paper, we applied a sliding window technique in a feature-based modeling approach to automated classification of written proficiency levels on the CEFR-scale (A1-C2 levels). We made use of 'complexity contours' obtained through this technique to represent the distribution of scores per linguistic feature within a text in combination

with RNN classifiers that exploit the sequential information in those contours. We demonstrated that an RNN classifier trained on complexity contours achieved higher classification accuracy across all six CEFR proficiency levels compared to one trained on text-average scores with an increase in performance of up to 14% in terms of precision and 15% in terms of recall. We also showed that iterative sensitivity-based pruning approach is a viable way of assessing relative feature importance in text classification tasks performed with neural network models. This approach taken in our paper has the potential to provide a valuable contribution to increasing efforts to identify 'criterial features', i.e. features that are characteristic and indicative of language proficiency at each level (Hawkins and Filipović, 2012). In our future work, we intend to include additional sets of features of language use based on crowd-sourced language metrics entitled word prevalence (Johns et al., 2020) as well as LIWC-style features that relate language use with behavioral and self-reported measures of personality, social behavior, and cognitive styles (Tausczik and Pennebaker, 2010). We also intend to take into account the effects of task type on the features of language use investigated in this paper (Alexopoulou et al., 2017).

References

Theodora Alexopoulou, Marije Michel, Akira Murakami, and Detmar Meurers. 2017. Task effects on linguistic complexity and accuracy: A large-scale learner corpus analysis employing natural language processing techniques. *Language Learning*, 67(S1):180–208.

Nicolas Ballier and Thomas Gaillat. 2019. Investigating the scope of textual metrics for learner level discrimination and learner analytics. In *Learner Corpus Research Conference*.

Nicolas Ballier, Thomas Gaillat, Andrew Simpkin, Bernardo Stearns, Manon Bouyé, and Manel Zarrouk. 2019. A supervised learning model for the automatic assessment of language levels based on learner errors. In *European Conference on Technology Enhanced Learning*, pages 308–320. Springer.

Daniel Blanchard, Joel Tetreault, Derrick Higgins, Aoife Cahill, and Martin Chodorow. 2013. Toefl11: A corpus of non-native english. *ETS Research Report Series*, 2013(2):i–15.

Hege Marie Bøvelstad, Einar Holsbø, Lars Ailo Bongo, and Eiliv Lund. 2017. A standard operating procedure for outlier removal in large-sample epidemiological transcriptomics datasets. *BioRxiv*, page 144519.

Adriane Boyd, Jirka Hana, Lionel Nicolas, Detmar Meurers, Katrin Wisniewski, Andrea Abel, Karin Schöne, Barbora Stindlová, and Chiara Vettori. 2014. The MERLIN corpus: Learner language and the CEFR. In *LREC*, pages 1281–1288. Reykjavik, Iceland.

Bram Bulté and Alex Housen. 2012. Defining and operationalising l2 complexity. *Dimensions of L2 performance and proficiency: Complexity, accuracy and fluency in SLA*, pages 23–46.

Andrew Caines and Paula Buttery. 2020. Reprolang 2020: Automatic proficiency scoring of czech, english, german, italian, and spanish learner essays. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5614–5623.

KyungHyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *CoRR*, abs/1409.1259.

Morten H Christiansen and Inbal Arnon. 2017. More than words: The role of multiword sequences in language learning and use. *Topics in Cognitive Science*, 9(3):542–551.

Mark Davies. 2008. The corpus of contemporary american english (coca): 560 million words, 1990-present.

Peter Deutsch. 1996. Deflate compressed data format specification version 1.3. *IETF RFC 1951*.

Wladimiro Díaz-Villanueva, Francesc J Ferri, and Vicente Cerverón. 2010. Learning improved feature rankings through decremental input pruning for support vector based drug activity prediction. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 653–661. Springer.

Philip Durrant and Mark Brenchley. 2019. Development of vocabulary sophistication across genres in English children's writing. *Reading and Writing*, 32(8):1927–1953.

Council of Europe. Council for Cultural Co-operation. Education Committee. Modern Languages Division. 2018. *Common European Framework of Reference for Languages: learning, teaching, assessment*. Cambridge University Press.

Katharina Ehret. 2016. *An information-theoretic approach to language complexity: variation in naturalistic corpora*. Ph.D. thesis, Universität.

Katharina Ehret and Benedikt Szmrecsanyi. 2019. Compressing learner language: An information-theoretic measure of complexity in sla production data. *Second Language Research*, 35(1):23–45.

James Garner, Scott Crossley, and Kristopher Kyle. 2020. Beginning and intermediate l2 writer's use of n-grams: an association measures study. *International Review of Applied Linguistics in Language Teaching*, 58(1):51–74.

Jeroen Geertzen, Theodora Alexopoulou, and Anna Korhonen. 2014. Automatic linguistic annotation of large scale L2 databases: The EF-Cambridge Open Language Database(EFCamDat).

Sylviane Granger, Estelle Dagneaux, and Fanny Meunier. 2002. The international corpus of learner english. handbook and cd-rom.

Julia Hancke and Detmar Meurers. 2013. Exploring CEFR classification for German based on rich linguistic modeling. *Learner Corpus Research*, pages 54–56.

John A Hawkins and Luna Filipović. 2012. *Criterial features in L2 English: Specifying the reference levels of the Common European Framework*, volume 1. Cambridge University Press.

Derrick Higgins, Chaitanya Ramineni, and Klaus Zechner. 2015. Learner corpora and automated scoring. In Sylviane Granger, Gaetanelle Gilquin, and Fanny Meunier, editors, *The Cambridge Handbook of Learner Corpus Research*, pages 587–604. Cambridge University Press.

Brendan T Johns, Melody Dye, and Michael N Jones. 2020. Estimating the prevalence and diversity of words in written language. *Quarterly Journal of Experimental Psychology*, 73(6):841–855.

Zixuan Ke and Vincent Ng. 2019. Automated essay scoring: A survey of the state of the art. In *IJCAI*, pages 6300–6308.

Elma Kerz, Yu Qiao, Daniel Wiechmann, and Marcus Ströbel. 2020. Becoming linguistically mature: Modeling english and german children's writing development across school grades. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 65–74.

Beata Beigman Klebanov and Michael Flor. 2013. Word association profiles and their use for automated scoring of essays. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1148–1158.

Beata Beigman Klebanov and Nitin Madnani. 2020. Automated evaluation of writing–50 years and counting. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7796–7810.

Dan Klein and Christopher D Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430. Association for Computational Linguistics.

Folkert Kuiken, Ineke Vedder, Alex Housen, and Bastien De Clercq. 2019. Variation in syntactic complexity: Introduction. *International Journal of Applied Linguistics*, 29(2):161–170.

Kristopher Kyle and Scott A Crossley. 2014. Automatically assessing lexical sophistication: Indices, tools, findings, and application. *Tesol Quarterly*, 49(4):757–786.

Pat Langley. 1994. Selection of relevant features in machine learning. In *Proceedings of the AAAI Fall symposium on relevance*, pages 1–5.

Diane Larsen-Freeman. 2006. The emergence of complexity, fluency, and accuracy in the oral and written production of five chinese learners of english. *Applied linguistics*, 27(4):590–619.

Roger Levy and Galen Andrew. 2006. Tregex and tsurgeon: tools for querying and manipulating tree data structures. In *LREC*, pages 2231–2234. Citeseer.

Xiaofei Lu. 2010a. Automatic analysis of syntactic complexity in second language writing. *International Journal of Corpus Linguistics*, 15(4):474–496.

Xiaofei Lu. 2010b. Automatic analysis of syntactic complexity in second language writing. *International journal of corpus linguistics*, 15(4):474–496.

Xiaofei Lu. 2011. A corpus-based evaluation of syntactic complexity measures as indices of college-level esl writers' language development. *Tesol Quarterly*, 45(1):36–62.

Xiaofei Lu. 2012. The relationship of lexical richness to the quality of esl learners' oral narratives. *The Modern Language Journal*, 96(2):190–208.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Dominic Masters and Carlo Luschi. 2018. Revisiting small batch training for deep neural networks. *arXiv preprint arXiv:1804.07612*.

John Moody. 1994. Prediction risk and architecture selection for neural networks. In *From statistics to neural networks*, pages 147–165. Springer.

Lourdes Ortega. 2003. Syntactic complexity measures and their relationship to l2 proficiency: A research synthesis of college-level l2 writing. *Applied linguistics*, 24(4):492–518.

Lourdes Ortega. 2012. Interlanguage complexity. *Linguistic complexity: Second language acquisition, indigenization, contact*, 13:127.

Ildikó Pilán, Elena Volodina, and Torsten Zesch. 2016. Predicting proficiency levels in learner writings by transferring a linguistic complexity model from expert-written coursebooks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2101–2111.

Pytorch. 2019. Pytorch: Tensors and dynamic neural networks in Python with strong GPU acceleration. https://github.com/pytorch/pytorch.

Seok-Chae Rhee and Chae Kwan Jung. 2014. Compilation of the yonsei english learner corpus (yelc) 2011 and its use for understanding current usage of english by korean pre-university students. *The Journal of the Korea Contents Association*, 14(11):1019–1029.

Kazuya Saito. 2020. Multi-or single-word units? the role of collocation use in comprehensible and contextually appropriate second language speech. *Language Learning*, 70(2):548–588.

Cyrus Shaoul and Chris Westbury. 2011. Formulaic sequences: Do they exist and do they matter? *The mental lexicon*, 6(1):171–196.

Marcus Ströbel. 2014. *Tracking complexity of l2 academic texts: A sliding-window approach*. Master thesis. RWTH Aachen University.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1882–1891.

Yla R Tausczik and James W Pennebaker. 2010. The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54.

Joachim Utans and John Moody. 1991. Selecting neural network architectures via the prediction risk: Application to corporate bond rating prediction. In *Proceedings First International Conference on Artificial Intelligence Applications on Wall Street*, pages 35–41. IEEE.

Sowmya Vajjala. 2018. Automated assessment of non-native learner essays: Investigating the role of linguistic features. *International Journal of Artificial Intelligence in Education*, 28(1):79–105.

Sowmya Vajjala and Kaidi Lõo. 2014. Automatic CEFR level prediction for Estonian learner text. In *Proceedings of the third workshop on NLP for computer-assisted language learning*, pages 113–127, Uppsala, Sweden. LiU Electronic Press.

Sowmya Vajjala and Taraka Rama. 2018. Experiments with universal CEFR classification. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 147–153, New Orleans, Louisiana. Association for Computational Linguistics.

Elena Volodina, Ildikó Pilán, and David Alfter. 2016. Classification of Swedish learner essays by CEFR levels. *CALL communities and culture–short papers from EUROCALL*, 2016:456–461.

Zarah Weiß and Detmar Meurers. 2018. Modeling the readability of German targeting adults and children: An empirically broad analysis and its cross-corpus validation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 303–317, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Katrin Wisniewski. 2017. Empirical learner language and the levels of the common european framework of reference. *Language Learning*, 67(S1):232–253.

Kate Wolfe-Quintero, Shunji Inagaki, and Hae-Young Kim. 1998. *Second language development in writing: Measures of fluency, accuracy, & complexity*. 17. University of Hawaii Press.

Helen Yannakoudakis and Ted Briscoe. 2012. Modeling coherence in esol learner texts. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 33–43.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.