# Text Genre Classification Based on Linguistic Complexity Contours Using A Recurrent Neural Network

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

## Abstract

Over the last years, there has been an increased interest in the combined use of natural language processing techniques and machine learning algorithms to automatically classify texts on the basis of wide range of features. One class of features that have been successfully employed for a wide range of classification tasks, including native language identification, readability assessment and text genre categorization pertain to the construct of 'linguistic complexity'. This paper presents a novel approach to the use of linguistic complexity features in text categorization: Rather than representing text complexity 'globally' in terms of summary statistics, this approach assesses text complexity 'locally' and captures the progression of complexity within a text as a sequence of complexity scores, generating what is referred to here as 'complexity contours'. We demonstrate the utility of the approach in an automatic text classification task for five genres – academic, newspaper, fiction, magazine and spoken – of the `Corpus of Contemporary American English (COCA)` [Davies, 2008] using a recurrent neural network.

## 1 Introduction

Recent years have witnessed a growing interest in the combined use of natural language processing (NLP) techniques together with machine learning algorithms to investigate the formal features of a text, rather than its content. This type of approach has been successfully applied to a range of automatic text categorization tasks, including author recognition and verification [Van Halteren, 2004], native language identification e.g. [Malmasi *et al.*, 2017; Crossley and McNamara, 2012; Kyle *et al.*, 2015], readability assessment [François and Miltsakaki, 2012] and text genre identification [Xu *et al.*, 2017]. One class of features that have been successfully employed in text classification research pertains to the multidimensional construct of 'linguistic complexity' that cuts across multiple levels of linguistic representation. Linguistic complexity is commonly defined as the "the range of forms that surface in language production and the degree of sophistication of such forms" [Ortega, 2003, p. 492]. This construct has been operationalized in terms of a number of measures that tap into different levels of linguistic analysis (e.g. lexical features, such as type-token ratio, or syntactic features, such as complex nominals per clause) and that require different NLP preprocessing steps, from tokenization to syntactic parsing (see Section 2). While previous text classification studies have combined information from multiple measures so as to cover different levels of analysis, these studies used as input for their classifiers scores representing the average complexity of a text. However, the use of such aggregate scores obscure the considerable degree of variation of complexity within a text.

In this paper we present a novel approach to the use of linguistic complexity features in the area of text classification. To this end, we employ a computational tool that implements a sliding-window technique to track the progression of complexity within a text, allowing for a 'local' - rather than a global' - assessment of complexity of a text. More precisely, we demonstrate the utility of the approach in a text genre classification task. Text genre detection is a typical classificatory task in computational stylistics that concerns "the identification of the kind of (or functional style) of the text [Stamatatos *et al.*, 2000, p. 472]. Although definitions of 'genre' remain elusive, in the broadest sense, it can be used to refer to "language use in a conventionalized communicative setting in order to give expression to a specific set of communicative goals of a disciplinary or social institution, which give rise to stable structural forms by imposing constraints on the use of lexico-grammatical as well as discoursal resources" [Bhatia, 2004, p.:23]. In this study, we start from the assumption that these constraints are reflected in the degree of linguistic complexity of a text. We then proceed to show that genres are not only distinguished by their average complexity but also in their distribution of linguistic complexity within a text.

## 2 Our approach: Measuring linguistic complexity using a sliding-window technique

The distribution of linguistic complexity within a text was measured using `Complexity Contour Generator (CoCoGen)`, a computational tool that implements a sliding-window technique to generate a series of measurements for a given complexity measure (CM), allowing for a 'local assessment' of complexity within a text [Ströbel, 2014; Ströbel *et al.*, 2016]. The approach implemented in CoCoGen stands in contrast the standard approach that represent text complexity as a single score, thus only providing a 'global assessment' of the complexity of a text. A sliding window can be conceived of as a window with a certain size 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. For a text comprising n sentences, there are $w = n - ws + 1$ windows. Given the constraint that there has to be at least one window, a text has to comprise at least as many sentences at the ws is wide $n \geq w$. To compute the complexity score of a given window $m$ ($w(m)$), a measurement function is called for each sentence in the window and returns a fraction ($wn_m/wd_m$). The denominators and numerators of the fractions from the first to the last sentence in the window are then added up to form the denominator and numerator of the resulting complexity score of a given window (see Figure 1).

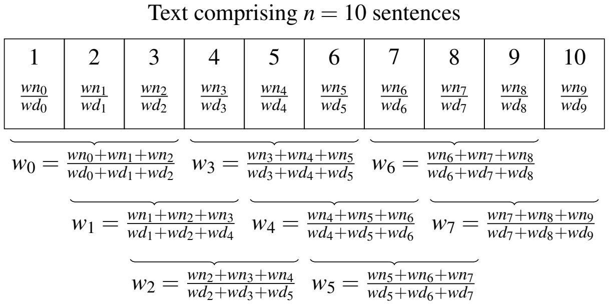
Text comprising $n = 10$ sentences
1 2 3 4 5 6 7 8 9 10
$\frac{wn_0}{wd_0}$ $\frac{wn_1}{wd_1}$ $\frac{wn_2}{wd_2}$ $\frac{wn_3}{wd_3}$ $\frac{wn_4}{wd_4}$ $\frac{wn_5}{wd_5}$ $\frac{wn_6}{wd_6}$ $\frac{wn_7}{wd_7}$ $\frac{wn_8}{wd_8}$ $\frac{wn_9}{wd_9}$
$w_0 = \frac{wn_0+wn_1+wn_2}{wd_0+wd_1+wd_2}$
$w_3 = \frac{wn_3+wn_4+wn_5}{wd_3+wd_4+wd_5}$
$w_6 = \frac{wn_6+wn_7+wn_8}{wd_6+wd_7+wd_8}$
$w_1 = \frac{wn_1+wn_2+wn_3}{wd_1+wd_2+wd_4}$
$w_4 = \frac{wn_4+wn_5+wn_6}{wd_4+wd_5+wd_6}$
$w_7 = \frac{wn_7+wn_8+wn_9}{wd_7+wd_8+wd_9}$
$w_2 = \frac{wn_2+wn_3+wn_4}{wd_2+wd_3+wd_5}$
$w_5 = \frac{wn_5+wn_6+wn_7}{wd_5+wd_6+wd_7}$

**Figure 1** Schematic illustration of how complexity measurements are obtained in CoCoGen for a text comprising ten sentences with a window size ws of three sentences.

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'. As texts vary in length, their complexity contours cannot be directly compared. To permit comparisons of such contours, CoCoGen features a scaling algorithm that divides each text into a user-defined number of approximately same-sized partitions, termed here as 'scaled windows' (see Figure 2).

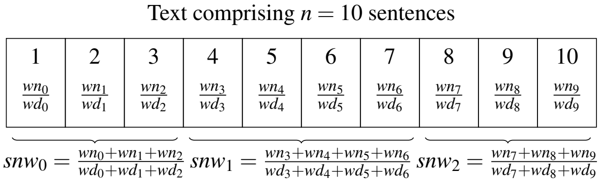
Text comprising $n = 10$ sentences
1 2 3 4 5 6 7 8 9 10
$\frac{wn_0}{wd_0}$ $\frac{wn_1}{wd_1}$ $\frac{wn_2}{wd_2}$ $\frac{wn_3}{wd_3}$ $\frac{wn_4}{wd_4}$ $\frac{wn_5}{wd_5}$ $\frac{wn_6}{wd_6}$ $\frac{wn_7}{wd_7}$ $\frac{wn_8}{wd_8}$ $\frac{wn_9}{wd_9}$
$snw_0 = \frac{wn_0+wn_1+wn_2}{wd_0+wd_1+wd_2}$
$snw_1 = \frac{wn_3+wn_4+wn_5+wn_6}{wd_3+wd_4+wd_5+wd_6}$
$snw_2 = \frac{wn_7+wn_8+wn_9}{wd_7+wd_8+wd_9}$

**Figure 2** Illustration of the complexity measurements obtained by CoCoGen for a text comprising ten sentences with the number of scaled windows set to 3.

In its current version, CoCoGen supports 32 measures of linguistic complexity. Importantly, CoCoGen was designed with extensibility in mind, so that additional complexity measures can easily be added. It uses an abstract measure class for the implementation of complexity measures. With the exception of three CMs based on an information-theoretic approach ($KolmogorovDeflate$, $SyntacticKolmogorovDeflate$, $MorphologicalKolmogorovDeflate$), the operationalizations of all CMs follow those given in [Lu, 2010] and [Lu, 2012]. For details on the operationalization of the CMs based on Kolmogorov complexity, see [Ströbel, 2014]. As preprocessing step, 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]). The set of 32 CM investigated in the present study can be binned into binned into four categories (raw text, lexical, morpho-syntactic, and syntactic) defined on the basis of the different levels of linguistic analysis automatically carried out (tokenization, lemmatization, morpho-syntactic tagging and constituency parsing).

**Raw Text Features** are derived from unanalyzed text and include CMs such as $MeanSentenceLength$, calculated as the average number of words per sentence, and $MeanLengthofWords$, calculated as the average number of characters per word. This category also subsumes the $Kolmogorov_{Deflate}$ measure (cf., [Ehret and Szmrecsanyi, 2016; Juola, 2008; Kettunen *et al.*, 2006; Li and Vitányi, 1997], for details). **Lexical Features** CMs from this category concern either the frequencies of lexical items from specific word lists, such as New Academic Word List or express vocabulary variation, i.e. measure lexical sophistication, such as the $Type - Token\ Ratio$ ($TTR$). Due to the sensitivity to sample size, two different variants of $TTR$ are also included ($Root\ TTR$, $Corrected\ TTR$). **Morpho-Syntactic Features**. An example of a CM that belongs to this category is $Lexical\ Density$ which is defined as the ratio of content words (verbs, nouns, adjectives and adverbs) to the total number of lexical tokens in a text. **Syntactic features** include make reference to syntactic dependencies. Two examples of CMs from this category are $Complex\ Nominals\ per\ T-Unit$ and $Dependent\ Clauses\ per\ Cause$. An overview of all CMs with their associated definitions, NLP categories, labels and descriptive statistics (mean text complexity and standard deviations) across complexity measures and genres is provided in Table 1.

# 3 Experiment

## 3.1 Datasets

The corpus data from the present study come from the Corpus of Contemporary American English (COCA) [Davies, 2008]. COCA is a balanced corpus of American English containing more than 560 million words of text (20 million words each year 1990-2017) equally divided among five general genres: spoken, fiction, popular magazines, newspapers, and academic texts.[1] For the

[1] The selection of the COCA over the British National Corpus (BNC) is motivated by two main reasons: (1) the COCA covers the time span from 1990 to 2012, whereas the BNC covers the time span from 1980s to 1993 (i.e. the most recent texts in the BNC are from the early 1990s, more than twenty years ago), making the COCA more representative of contemporary English and (2) the

**Table 1** Overview of complexity measures with associated definitions, NLP categories, labels and descriptive statistics (mean text complexity and standard deviations) across complexity measures and genres

| | | | | Acad | | Ficti | | Mag | | News | | Spoken | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Complexity Measure | Definition | NLP Cat. | Label | mean | SD | mean | SD | mean | SD | mean | SD | mean | SD |
| Number of diff. words / sample (cor.) | Nw diff/Nw | Lexical | CNDW | 0.865 | 0.004 | 0.888 | 0.003 | 0.897 | 0.001 | 0.89 | 0.004 | 0.896 | 0.005 |
| Corrected Type-Token Ratio | T/sqrt(2N) | Lexical | CTTR | 4.528 | 0.099 | 3.478 | 0.073 | 4.476 | 0.053 | 4.259 | 0.089 | 3.778 | 0.029 |
| Lexical Density | Nlex/N | Morphsyn | Den | 0.542 | 0.004 | 0.493 | 0.003 | 0.55 | 0.002 | 0.538 | 0.001 | 0.505 | 0.01 |
| Number of different words / sample | Nw diff | Lexical | NDW | 17.238 | 0.814 | 9.776 | 0.506 | 16.708 | 0.469 | 14.058 | 0.572 | 10.921 | 0.195 |
| Root Type-Token Ratio | T/sqrt(N) | Lexical | RTTR | 3.15 | 0.067 | 2.407 | 0.051 | 3.079 | 0.036 | 2.995 | 0.064 | 2.618 | 0.02 |
| Type-Token Ratio | T/N | Lexical | TTR | 0.86 | 0.003 | 0.874 | 0.002 | 0.897 | 0.001 | 0.876 | 0.003 | 0.896 | 0.006 |
| Mean Length Clause | NW/NC | Morphsyn | MLC | 13.083 | 0.629 | 8.3 | 0.287 | 11.347 | 0.306 | 11.868 | 0.382 | 9.061 | 0.512 |
| Mean Length T-Unit | NW/NT | Morphsyn | MLT | 23.235 | 0.654 | 12.654 | 0.611 | 19.29 | 0.359 | 19.832 | 0.76 | 24.468 | 0.549 |
| Sequences Academic Formula List | SeqN AWL | Raw text | AFL | 0.045 | 0.003 | 0.004 | 0 | 0.015 | 0.001 | 0.014 | 0.001 | 0.009 | 0.001 |
| Lexical Sophistication (ANC) | Nslex_ANC/Nlex | Raw text | ANC | 0.327 | 0.006 | 0.257 | 0.006 | 0.33 | 0.003 | 0.302 | 0.003 | 0.289 | 0.014 |
| Lexical Sophistication (BNC) | Nslex_BNC/Nlex | Raw text | BNC | 0.509 | 0.005 | 0.435 | 0.005 | 0.504 | 0.002 | 0.493 | 0.002 | 0.478 | 0.015 |
| Kolmogorov Deflate | KS2011 | Raw text | KolDef | 1.298 | 0.018 | 1.022 | 0.015 | 1.203 | 0.009 | 1.205 | 0.018 | 1.083 | 0.005 |
| Morphological Kolmogorov Deflate | KS2011 | Raw text | M_Kol | 0.734 | 0.012 | 0.887 | 0.012 | 0.787 | 0.007 | 0.786 | 0.013 | 0.882 | 0.004 |
| Mean Length Sentence | NW/NS | Raw text | MLS | 20.16 | 0.963 | 10.939 | 0.62 | 18.772 | 0.519 | 15.982 | 0.671 | 12.356 | 0.223 |
| Mean Length of Words (characters) | Nchar/Nw | Raw text | MLWC | 5.078 | 0.029 | 4.115 | 0.029 | 4.623 | 0.013 | 4.7 | 0.028 | 4.428 | 0.068 |
| Words on New Academic Word List | WN AWL | Raw text | NAWL | 0.024 | 0 | 0.008 | 0 | 0.013 | 0 | 0.01 | 0 | 0.007 | 0 |
| Words not on General Service List | WN GSL | Raw text | NGSL | 0.228 | 0.011 | 0.169 | 0.005 | 0.23 | 0.003 | 0.223 | 0.004 | 0.235 | 0.014 |
| Syntactic Kolmogorov Deflate | KS2011 | Raw text | Syn_Kol | 0.719 | 0.012 | 0.869 | 0.012 | 0.773 | 0.007 | 0.773 | 0.013 | 0.864 | 0.004 |
| Clauses per Sentence | NC/NS | Syntactic | CS | 1.68 | 0.085 | 1.367 | 0.038 | 1.831 | 0.054 | 1.581 | 0.042 | 1.478 | 0.077 |
| Clauses per T-Unit | NC/NT | Syntactic | CT | 1.834 | 0.039 | 1.53 | 0.025 | 1.805 | 0.018 | 1.868 | 0.023 | 2.805 | 0.11 |
| Complex Nominals per Clause | NCN/C | Syntactic | CNC | 1.852 | 0.088 | 0.818 | 0.052 | 1.375 | 0.031 | 1.326 | 0.057 | 1.003 | 0.072 |
| Complex Nominals per T-Unit | NCN/NT | Syntactic | CNT | 3.289 | 0.101 | 1.258 | 0.093 | 2.353 | 0.055 | 2.385 | 0.135 | 2.7 | 0.088 |
| Complex T-Units per T-Unit | NCT/NT | Syntactic | CTT | 0.476 | 0.014 | 0.298 | 0.013 | 0.444 | 0.009 | 0.466 | 0.012 | 0.404 | 0.021 |
| Coordinate Phrases per Clause | NCP/NC | Syntactic | CPC | 0.412 | 0.028 | 0.181 | 0.009 | 0.296 | 0.011 | 0.239 | 0.015 | 0.136 | 0.011 |
| Coordinate Phrases per T-Unit | NCP/NT | Syntactic | CPT | 0.713 | 0.044 | 0.272 | 0.017 | 0.483 | 0.013 | 0.424 | 0.033 | 0.362 | 0.016 |
| Dependent Clauses per Clause | NDC/NC | Syntactic | DCC | 0.369 | 0.007 | 0.251 | 0.007 | 0.351 | 0.003 | 0.356 | 0.008 | 0.38 | 0.007 |
| Dependent Clauses per T-Unit | NDC/NT | Syntactic | DCT | 0.744 | 0.012 | 0.445 | 0.019 | 0.678 | 0.008 | 0.718 | 0.021 | 1.169 | 0.057 |
| Mean Length of Words (syllables) | Nsyl/Nw | Syntactic | MLWS | 1.597 | 0.01 | 1.257 | 0.007 | 1.401 | 0.004 | 1.442 | 0.008 | 1.348 | 0.012 |
| Noun Phrase Postmodification (words) | NNP Pre | Syntactic | NPpostModW | 3.586 | 0.143 | 1.303 | 0.109 | 2.635 | 0.084 | 2.519 | 0.171 | 2.209 | 0.118 |
| Noun Phrase Premodification (words) | NNP Post | Syntactic | NPpreModW | 0.922 | 0.022 | 0.559 | 0.026 | 0.911 | 0.012 | 0.836 | 0.034 | 0.547 | 0.039 |
| T-Units per Sentence | NT/NS | Syntactic | TS | 0.895 | 0.045 | 0.866 | 0.011 | 1.006 | 0.023 | 0.846 | 0.018 | 0.601 | 0.012 |
| Verb Phrases per T-Unit | NVP/NT | Syntactic | VPT | 2.407 | 0.031 | 1.914 | 0.025 | 2.322 | 0.01 | 2.389 | 0.026 | 3.497 | 0.106 |

purposes of this study, we used a balanced subsample of the corpus comprising 10,500 texts obtained by random sampling of 2,100 texts from each of these five genres. Complexity contours were obtained using CoCoGen with a window size of 10 sentences over all 10,500 texts. For each text, we extracted a feature sequence, which consists of a series of length $n - 10 + 1$ 32 dimensional feature vectors generated at each window position, where $n$ is the number the sentences in a text. After normalization and padding of the feature sequences, the data were divided into a balanced training set of 10,000 feature sequences (2,000 texts per genre) and a balanced test set of 500 feature sequences (100 texts per genre). To determine to what extent the performance of the classification model is driven by the sequence information, i.e. by the complexity contour, rather than average text complexity, we also created a comparison dataset in which we collapsed each unnormalized feature sequences to its mean vector, so as to retain only the global feature information, and then normalized these data. We used the same COCA subset of 10,500 texts described above to train and test the classification model. Complexity contours were obtained using CoCoGen with window size of 10 sentences over all 10,500 texts. For each text, we extracted a feature sequence, which consists of a series of length $n - 10 + 1$ 32 dimensional feature vectors generated at each window position, where $n$ is the number the sentences in a text. After normalization and padding of the feature sequences, the data were divided into a balanced training set of 10,000 feature sequences (2,000 texts per genre) and a balanced test set of 500 feature sequences (100 texts per genre).

## 3.2 Model

We used a Recurrent Neural Network classifier adopting the model specification described in [Hafner, 2018]. This model was used because (1) is a dynamic RNN model that can handle sequences of variable length[2], (2) it uses Gated Recurrent Unit (GRU) cells, which have been shown yield better performance on smaller datasets [Chung *et al.*, 2014], and (3) it is a simple model.

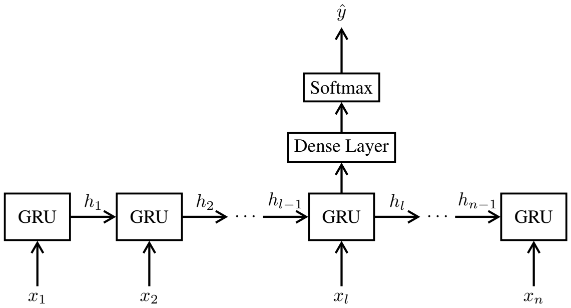

**Figure 3** Roll-out Of the RNN Model

Assume an input sequence $X = (x_1, x_2, \ldots, x_l, x_{l+1}, \ldots, x_n)$, where each of $x_i$ is a 32 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 dataset and $x_{l+1}, \cdots, x_n$ are padded **0**-vectors. As shown in Figure 3, this model consists only of GRU cells with 200 hidden units. To predict the classification, softmax was applied to the output of a fully-connected layer, where the output of the last GRU cell, i.e. whose input is $x_l$, are transformed from a 200 dimensional vector to a 5 dimensional vector. In order to make our comparison to the average-complexity approach as fair as possible, we reused the above model. However, rather than training it on sequences, it was provided only with vectors of average-complexities, i.e. the roll-out of the model consist of only one GRU cell.

## 3.3 Training

As a loss function, cross entropy was used.

$$\mathcal{L} = -\sum_{i=1}^{5} y_i \log(\hat{y}_i)$$

where $[y_1, y_2, \ldots, y_5]$ is the label of the sequence and $[\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_5]$ is the prediction of the model.

The mini-batch size was set to 100. For optimization, we compared Nesterov accelerated gradient (NAG), Adadelta and RMSprop and finally decided on NAG with a learning rate of $\eta = 0.01$ and $\gamma = 0.9$, for which we achieved the lowest error rate of our model.

## 3.4 Results and Discussion

Before turning to the results of the classification experiment, we first present the results of the CoCoGen analysis to illustrate the variation in text complexity both at the level of individual text and at the level of text genres. Note that for this illustration we used the scaled CoCoGen output with 100 scaled windows. Figure 4 visualizes the progression of complexity within a single text from the genre of academic writing for two selected measures of complexity ($corrected\ type–token–ratio$ and $Dependent\ Clauses\ per\ T - Unit$ ).

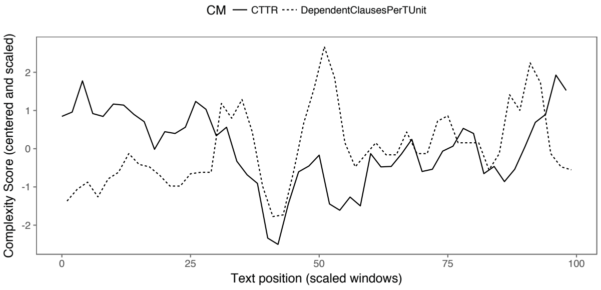

**Figure 4** Progression of complexity within a single text from the genre of academic writing for two measures of complexity ($corrected\ type–token–ratio$ and $Dependent\ Clauses\ per\ T - Unit$ )

COCA (450 million words) is more than four times as large as the BNC (100 million words).

[2] The lengths of the feature sequences depend on the number of sentences of the texts in our corpus.

As shown in Figure 4, complexity is not evenly distributed within a text but progresses through a sequence of peaks and troughs for both CMs. Furthermore, Figure 4 suggest that there is an interaction between the two measures such that higher complexity in CTTR in the beginning of the text (windows 1-30) appears to be compensated for by lower complexity in $DClperTUnit$, whereas the reverse is true for a middle part of the text (windows 40-60).

To determine to what extent text complexity varies among the genres and to see if there are tendencies for genre-specific complexity contours, we aggregated the complexity scores of all text from a given genre at each of the 100 scaled window positions. Figure 6 presents an overview of the resulting average complexity contours for the five genre compared across all CMs.

Figure 6 shows that academic writing is the most complex of the five genres investigated in this study with respect to the majority of CMs (19 out of 32 CMs). It is particularly more complex with regard to the CMs $Complex\ Nominal\ per\ Clause$, $Coordinate\ Phrases\ per\ Clause$ and $Noun\ Phrase\ Post - modifications\ per\ Clause$, but not with regard to $Dependent\ Clauses\ per\ T - Unit$ or $Verb\ Phrases\ per\ T - Unit$. Complexity scores of these latter two CMs are highest in spoken conversation. These results are consistent with findings reported in corpus based studies demonstrating that academic writing is characterized by a 'compressed' discourse style with phrasal (non- clausal) modifiers embedded in noun phrases, whereas spoken discourse is more structurally elaborated with multiple levels of clausal embedding [Biber and Gray, 2010; Biber *et al.*, 2011]. Figure 6 furthermore demonstrates that, while the averaged contours are less 'wiggly' than those of individual texts, they are typically not uniform and often nonlinear. For example, for some CMs and genres, e.g. $Coordinate\ Phrases\ per\ Clause$ in academic writing, the distribution is U-shaped, such that the beginning and end of a text are much more complex on average than its middle part. Overall, this pattern of results strongly suggest that the complexity scores are not randomly distributed across the texts of a given genre. We now turn to the results our classification experiment. Classification results of previous studies on text genre identification range from relatively low accuracy of 52% to 80% range, with results above 90% reported in some cases, depending on the number and type of genres being considered, size and difficulty of data, etc. (see, e.g [Kessler *et al.*, 1997; Dewdney *et al.*, 2001; Dell'Orletta *et al.*, 2013; Passonneau *et al.*, 2014; Yogatama *et al.*, 2017]).The performance of our RNN classifiers over 60 training epochs is presented in Figure 5. Figure 5 indicates that the sequence-based RNN displays consistently lower error rates than the average-based RNN: The average-based RNN reached a maximal performance of 91.2% at epoch 16 with an mean performance of 90% accuracy in the surrounding epochs (epochs 10-20). After that performance starts to decrease indicating overfitting. The sequence-based RNN reached a maximal accuracy of 92.8% after 10 epochs and converged on an robust average performance 91.5% after around 30 epochs. These results suggest the utility of the sequence information for the task of genre identification.

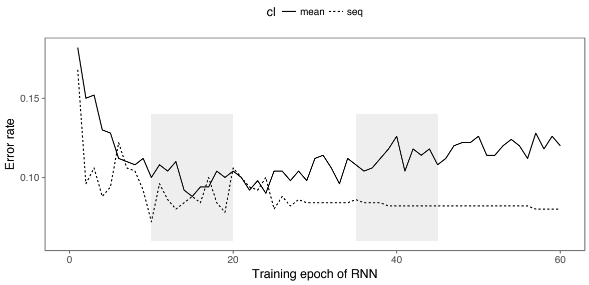

**Figure 5** Performance of our RNN classifiers over 60 training epochs

## 3.5 Conclusion

The main goal of the paper was to showcase a novel approach to the use of linguistic complexity features for purposes of automatic text classification. Using the task of text genre classification as a test case, we showed that both individual texts as well as text genres are characterized by considerable variation in within-text complexity as captured by 'complexity contours', i.e. series of measurements generated by `CoCoGen` that implements a sliding-window technique. The results of a 5-class text genre classification experiment demonstrated that the inclusion of these contours further increased the high performance ( 90%) of a GRU-RNN classifier trained on text average complexity scores. In future studies we intend to explore the utility of our approach to other tasks of text classification.

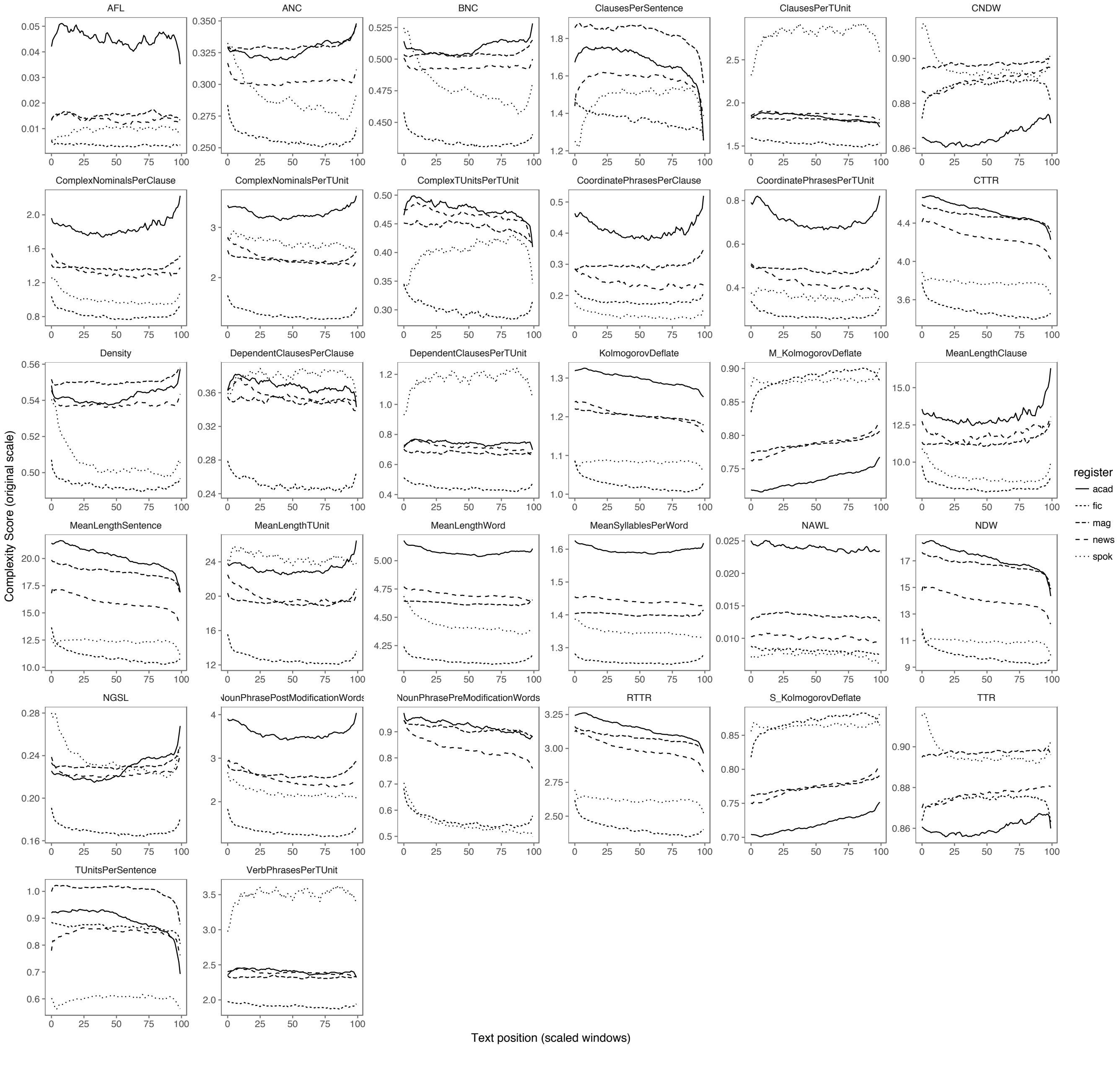

**Figure 6** Average distribution of complexity across all texts from a given genre for all measures of complexity

## References

[Bhatia, 2004] Vijay Bhatia. *Worlds of written discourse: A genre-based view*. A&C Black, 2004.

[Biber and Gray, 2010] Douglas Biber and Bethany Gray. Challenging stereotypes about academic writing: Complexity, elaboration, explicitness. *Journal of English for Academic Purposes*, 9(1):2–20, 2010.

[Biber *et al.*, 2011] Douglas Biber, Bethany Gray, and Kornwipa Poonpon. Should we use characteristics of conversation to measure grammatical complexity in l2 writing development? *Tesol Quarterly*, 45(1):5–35, 2011.

[Chung *et al.*, 2014] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014.

[Crossley and McNamara, 2012] Scott A Crossley and Danielle S McNamara. Detecting the first language of second language writers using automated indices of cohesion, lexical sophistication, syntactic complexity and conceptual knowledge. In *Approaching language transfer through text classification: Explorations in the detection-based approach*. Channel View Publications, 2012.

[Davies, 2008] Mark Davies. *The Corpus of Contemporary American English*. BYE, Brigham Young University, 2008.

[Dell'Orletta *et al.*, 2013] Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. Linguistic profiling of texts across textual genres and readability levels. an exploratory study on Italian fictional prose. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 189–197, 2013.

[Dewdney *et al.*, 2001] Nigel Dewdney, Carol VanEss-Dykema, and Richard MacMillan. The form is the substance: Classification of genres in text. In *Proceedings of the Workshop on Human Language Technology and Knowledge Management-Volume 2001*, page 7. Association for Computational Linguistics, 2001.

[Ehret and Szmrecsanyi, 2016] Katharina Ehret and Benedikt Szmrecsanyi. An information-theoretic approach to assess linguistic complexity. *Complexity and Isolation. Berlin: de Gruyter*, 2016.

[François and Miltsakaki, 2012] Thomas François and Eleni Miltsakaki. Do NLP and machine learning improve traditional readability formulas? In *Proceedings of the First Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 49–57. Association for Computational Linguistics, 2012.

[Hafner, 2018] Danijar Hafner. Variable Sequence Lengths in TensorFlow. `https://danijar.com/variable-sequence-lengths-in-tensorflow/`, 2018.

[Juola, 2008] Patrick Juola. Assessing linguistic complexity. In Matti Miestamo, Kaius Sinnemäki, and Fred Karlsson, editors, *Language Complexity: Typology, contact, change*, pages 89–108. Benjamins Amsterdam, Philadelphia, 2008.

[Kessler *et al.*, 1997] Brett Kessler, Geoffrey Numberg, and Hinrich Schütze. Automatic detection of text genre. In *Proceedings of the Eighth Conference on European Chapter of the Association for Computational Linguistics*, pages 32–38. Association for Computational Linguistics, 1997.

[Kettunen *et al.*, 2006] Kimmo Kettunen, Markus Sadeniemi, Tiina Lindh-Knuutila, and Timo Honkela. Analysis of EU languages through text compression. In *Advances in Natural Language Processing*, pages 99–109. Springer, 2006.

[Klein and Manning, 2003] Dan Klein and Christopher D Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, 2003.

[Kyle *et al.*, 2015] Kristopher Kyle, Scott A Crossley, and You Jin Kim. Native language identification and writing proficiency. *International Journal of Learner Corpus Research*, 1(2):187–209, 2015.

[Li and Vitányi, 1997] Ming Li and Paul Vitányi. *An introduction to Kolmogorov complexity and its applications*. Springer Heidelberg, 1997.

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

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

[Malmasi *et al.*, 2017] Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano, and Yao Qian. A report on the 2017 native language identification shared task. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 62–75, 2017.

[Manning *et al.*, 2014] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System demonstrations*, pages 55–60, 2014.

[Ortega, 2003] Lourdes Ortega. Syntactic complexity measures and their relationship to l2 proficiency: A research synthesis of college-level L2 writing. *Applied linguistics*, 24(4):492–518, 2003.

[Passonneau *et al.*, 2014] Rebecca J Passonneau, Nancy Ide, Songqiao Su, and Jesse Stuart. Biber redux: Reconsidering dimensions of variation in American English. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical papers*, pages 565–576, 2014.

[Stamatatos *et al.*, 2000] Efstathios Stamatatos, Nikos Fakotakis, and George Kokkinakis. Automatic text categorization in terms of genre and author. *Computational Linguistics*, 26(4):471–495, 2000.

[Ströbel *et al.*, 2016] Marcus Ströbel, Elma Kerz, Daniel Wiechmann, and Stella Neumann. Cocogen-complexity contour generator: Automatic assessment of linguistic complexity using a sliding-window technique. In *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity (CL4LC)*, pages 23–31, 2016.

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

[Van Halteren, 2004] Hans Van Halteren. Linguistic profiling for author recognition and verification. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 199. Association for Computational Linguistics, 2004.

[Xu *et al.*, 2017] Zhijuan Xu, Lizhen Liu, Wei Song, and Chao Du. Text genre classification research. In *Computer, Information and Telecommunication Systems (CITS), 2017 International Conference on*, pages 175–178. IEEE, 2017.

[Yogatama *et al.*, 2017] Dani Yogatama, Chris Dyer, Wang Ling, and Phil Blunsom. Generative and discriminative text classification with recurrent neural networks. *arXiv preprint arXiv:1703.01898*, 2017.