Machine learning for learner English: A plea for creating learner data challenges

Article *in* International Journal of Learner Corpus Research · April 2020

DOI: 10.1075/ijlcr.18012.bal

7 authors, including:

Nicolas Ballier
Paris Diderot University
61 PUBLICATIONS **79** CITATIONS
SEE PROFILE

Stephane Canu
Institut National des Sciences Appliquées de Rouen
137 PUBLICATIONS **2,997** CITATIONS
SEE PROFILE

Gilles Gasso
Institut National des Sciences Appliquées de Rouen
56 PUBLICATIONS **741** CITATIONS
SEE PROFILE

Carlos Balhana
University of Cambridge
4 PUBLICATIONS **16** CITATIONS
SEE PROFILE

Machine Learning for learner English
"a plea for creating learner data challenges"

Nicolas Ballier (1), Stéphane Canu (2), Caroline Petitjean (2), Gilles Gasso (2), Carlos Balhana (3), Theodora Alexopoulou (3), Thomas Gaillat (4)
(1) Université Paris Diderot Paris 7 , France, (2) Normandie Univ, INSA Rouen, UNIROUEN, UNIHAVRE, LITIS, France, (3) University of Cambridge, UK, (4) Université de Rennes 1, France

Authors' addresses:
(1) CLILLAC-ARP, 5 rue Thomas Mann, 75205 Paris Cedex 13. (2) LITIS, Avenue de l'Université, 76801 Saint-Étienne-du-Rouvray, France. (3) Linguistics Section, Faculty of Modern and Medieval Languages, 9 West Road University of Cambridge Sidgwick Avenue CAMBRIDGE CB3 9DA. (4) SCELVA 263 avenue du Général Leclerc 35042 Rennes CEDEX.

Summary

This paper discusses machine learning techniques for the prediction of Common European Framework of Reference (CEFR) levels in a learner corpus. We summarise the CAp 2018 Machine Learning (ML) competition, a classification task of the six CEFR levels, which maps linguistic competence in a foreign language onto six reference levels. The goal of this competition was to produce a machine learning system to predict learners' competence levels from written productions comprising between 20 and 300 words and a set of characteristics computed for each text extracted from the French component of the EFCAMDAT data (Geertzen *et al.*, 2013). Together with the description of the competition, we provide an analysis of the results and methods proposed by the participants and discuss the benefits of this kind of competition for the learner corpus research (LCR) community. The main findings address the methods used and lexical bias introduced by the task.

Keywords

natural language processing (NLP), machine learning, learners of English, EFCAMDAT corpus, CEFR, language proficiency.

1. Introduction

Learner corpus research (LCR) has increasingly taken into account advances in natural language processing (NLP) (Callies & Paquot, 2015; Granger *et al.*, 2007; Meurers, 2015; Wisniewski, 2017), such as native language identification (Jarvis & Paquot, 2015), grammatical error correction (Leacock *et al.*, 2010) and automated scoring (Higgins *et al.*, 2015). Recent progress in machine learning (ML) techniques (see for instance Goldberg, 2017) suggests that this type of approach can be used to solve many challenges in analyzing learner corpus data. However, the question of choosing the best-suited machine learning methods for developing dedicated techniques to learner level identification remains open. To this end, a learner data challenge was organised in a machine learning conference to compare participants' skills and algorithms. This data challenge, sponsored by Nvidia, took place in Rouen during CAp2018 [1], the yearly French conference on machine learning that gathers specialists in artificial intelligence. The aim was to encourage cross-fertilisation between linguists and specialists of Artificial Intelligence (AI) and Natural Language Processing.

The domains of AI and NLP have a long history in data challenges, or shared tasks, especially in the areas of corpus annotation (Magerman, 1995), semantic labelling such as named entity recognition (Sang & De Meulder, 2003), opinion mining, sentiment analysis (Liu, 2012) etc. The challenges usually consist in automatically predicting specific labels for a given task (such as annotation). A training set includes the labels to be predicted whilst a test set does not. The winning technique is the one which predicts the labels of the test set with the best accuracy (see Paroubek *et al.*, 2007; Nissim *et al.*, 2017, and included references). Many workshops have been organised as satellite events of computational linguistics conferences such as the Conference on Computational Natural Language Learning (CoNNL). One of the most popular tasks is the analysis of syntactic and semantic dependencies in multiple languages [2] with a specific data format, known as CoNNL-U [3]. Shared tasks have been instrumental in opening avenues for empirical research using NLP techniques.

We intend to show the benefits of similar undertakings to learner corpus research. Section 2 contextualises the data challenges in NLP and second language acquisition (SLA). Section 3 introduces the aims of the competition. Section 4 describes the features that were included in the data sets and how the data was split

[1] cap2018.litislab.fr
[2] ufal.mff.cuni.cz/conll2009-st/task-description.html
[3] universaldependencies.org/format.html

into a training set and a test set. Section 5 describes the cost matrix that was used to rank submissions. Section 6 reports the methods and results, discusses the lessons that can be drawn from this competition for corpus linguists and suggests some next steps for this line of research. We conclude in Section 7.

2. Learner corpora in shared tasks

In second language acquisition, several types of shared tasks have relied on learner corpora. The first type corresponds to learner corpora exploited in native language identification tasks. Malmasi *et al.* (2017) report the latest competition at the BEA12 workshop which included two data sets made up of speech and written data from learners of English. The task was to automatically identify the native language of the learners. Performance results were reported in terms of precision, recall and F_1-Score. These measures are valued within a [0,1] range and indicate accuracy, F_1-Score being the mean of recall and precision measures. Best results reported a 0.88 F_1-score, which indicates a high level of accuracy.

The second type of shared tasks have focused on building error detection systems. A number of these tasks have been organized in recent years. The HOO Pilot Shared tasks initiated the move with the detection of all error types in a corpus of research article fragments from the ACL Anthology (Dale & Kilgarriff, 2011). The 2012 edition focused on preposition and determiner error detection (Dale *et al.*, 2012). The CoNLL-2014 shared task on grammatical error correction (Ng *et al.*, 2014) made use of the NUCLE corpus (Dahlmeier *et al.*, 2013) and extended its 2013 edition by targeting all error types in the NUCLE corpus. As a satellite workshop to the 40th Annual Conference of the Cognitive Science Society (CogSci 2018), the Duolingo shared task on second language acquisition modeling (SLAM) (Settles *et al.*, 2018; Tetreault *et al.*, 2018) was proposed and has already resulted in several papers written for the competition (e.g. Rich *et al.*, 2018; Hopman *et al.*, 2018). The Duolingo[4] data set was built with extractions from the Duolingo app[5], retracing initial learner productions. The data set was complex, using learner input for three tasks (reverse translation and reverse translation with provided words, including distractors and translation of heard phrases) and three data sets from different learner groups were created for the competition: Learners of English (with Spanish as L1), learners of Spanish (with English as L1), and learners of French (with English as L1). Participants had to identify errors from a longitudinal sam-

[4] sharedtask.duolingo.com

[5] www.duolingo.com

pling of learner productions presented in the CONNL format. Results at CONNL-2014 showed 0.3733 $F_{0.5}$ while HOO 2012 showed 0.4147 F_1-score. The most recent Duolingo shared task reported results of 0.561 F_1-score. These results show that multi-class (such as error categories) prediction tasks still have a large error margin (around 50%) and will benefit from more research in error-related features and classification methods.

The third type of shared tasks with learner data concerns automatic essay scoring, as in the 2019 Building Educational Applications workshop (Yannakoudakis *et al.*, 2019) and Automatic Scoring Systems including Automatic Essay Scoring (AES) tasks. The first AES for open-ended questions emerged from Page's PEG-IA system (Page, 1968) and focused on native English. More recently, Automatic Scoring Systems have focused on learner language data (Shermis *et al.*, 2010; Cushing Weigle, 2010), which has raised the need to use learner corpora to train models (Barker *et al.*, 2015; Higgins *et al.*, 2015).

To the best of our knowledge, few Automatic Scoring Systems shared tasks have made use of learner corpora for the purpose of scoring. The two editions of the Spoken CALL shared Task (Baur *et al.*, 2017, 2018) focused on the distinction between linguistically correct and incorrect short open-ended responses in Swiss German learners' speech. Despite the scarcity of shared tasks of this third type, many studies have been conducted on automatic level scoring in learner English (Attali & Burstein, 2006; Yannakoudakis *et al.*, 2011) and also in other languages such as Estonian (Vajjala & Loo, 2014) and Swedish (Volodina *et al.*, 2016). All papers report on different methods that use n-grams, errors, syntactic and lexical features to rank learner texts. They may focus on scoring specific language aspects such as text coherence or global proficiency levels of the learners. Some of these approaches are deployed in commercial products[6].

In terms of Automatic Scoring Systems performance, depending on the types of scores (continuous or categorical values), several types of evaluation metrics were used ranging from correlation coefficients[7] to precision, recall and F_1-Score. These differences in test scores make comparisons difficult. In the case of numeric scores (Cushing Weigle, 2010) compared the e-rater system's scores with human scores and concluded to the same level of correlation (r=0.81 between e-rater and two humans). Yannakoudakis *et al.* (2011) reported a Spearman's correlation co-

[6] For instance, see the IntelligentEssayAssessor™ developed at Pearson Knowledge Technologies; the IntelliMetric™EssayScoringSystem developed by Vantage Learning and e-rater® developed by Educational Testing Service (ETS).

[7] Correlation coefficients are within the [-1;+1] interval, with 0 indicating no association and 1 perfect association. Polarity signs indicate the direction of the association (same or opposite)

efficient of 0.773 for their Support Vector Machine (SVM) model. In the case of categorical scores. Vajjala's Common Europeran Framework of Reference (CEFR) classification approach resulted in a 0.74 F_1-Score and (Volodina *et al.*, 2016) reported a 0.66 F_1-Score. Both studies showed higher levels of accuracy than error detection tasks due to their inherent limited number of categories to predict.

So far, learner essay scoring has been the result of independent projects which did not share the same data. As a result, methods and results were not easily comparable. The CAp 2018 "My tailor is rich" shared task intended to fill this gap by providing a single unique data set with identical features and CEFR classified writings.

3. Aims of the competition

The CEFR maps linguistic competence in a foreign language onto six reference levels to be shared by European countries: A1, A2, B1, B2, C1 and C2. Level A1 denotes the lowest proficiency and level C2 the highest. As the framework is common to all European languages, it describes competencies in terms of functional *can-do* statements regarding the set of communicative goals learners ought to be able to achieve at each level, rather than linguistic features *per se* (Council of Europe, 2001a). The CEFR also provides a level of granularity in terms of writing or comprehension skills. (Council of Europe, 2001b, 23).

It is clear that any teaching curriculum working with CEFR descriptors will aim for a variety of tasks that can meet the various communicative goals and can-do statements of each CEFR level. As can be observed, the descriptors are not language specific. A critical question then for learners, teachers and assessors is which linguistic features or grammatical knowledge is necessary for meeting the functional goals of CEFR descriptors. Various language-specific projects have sought to identify the properties of the learner grammars or interlanguage underpinning each CEFR level [8]. The English Profile project has taken a learner corpus approach, aiming to identify a set of linguistic features that are characteristic of each level and hence assumed as "criterial features" for each CEFR level (Hawkins & Buttery, 2010; Hawkins & Filipović, 2012; O'Keeffe & Mark, 2017).

The properties of these levels have been investigated more thoroughly by the learner corpus research community, including the investigation of errors as "criterial

[8] see the 2018 Companion volume with new descriptors (Council of Europe, 2018) and the Reference level descriptions, retrievable from `https://www.coe.int/en/web/common-european-framework-reference-languages/reference-level-descriptions-rlds-developed-so-far`

features" (Thewissen, 2015). Both theory-driven and data-driven approaches have been adopted. Hawkins & Filipović (2012) and O'Keeffe & Mark (2017) illustrate hypothesis-driven investigations while Alexopoulou *et al.* (2013) involve a data-driven approach in which the most discriminative features of a system classifying B1 exams are interpreted as criterial.

Following up on this background, in May 2018, the CAp 2018 conference hosted a machine learning competition. Within an NLP/AI perspective, the CEFR levels represent six classes to predict[9] and a potential proxy for language proficiency [10]. The goal of this competition was to produce, by means of machine learning methods, a system to predict the level of the written productions, each comprising between 20 and 300 words, and a set of characteristics (see Section 4) computed from each text.

4. Data set description

The data were derived from extracts of EFCAMDAT, an open access corpus developed by linguists at Cambridge University in collaboration with Education First (EF), an international school of English as a foreign language (Geertzen *et al.*, 2013). The corpus[11] consists of written assignments submitted to Englishtown, the online school of EF Education First, summing 1,180,309 individual scripts by 174,743 learners. The curriculum of Englishtown[12] covers all proficiency levels, from CEFR A1 to C2 organised along 16 EF teaching levels. Each EF level contains 8 teaching units ending with an open-ended writing task, yielding 128 distinct writing assignments. Writing tasks cover a variety of communicative functions, e.g. writing a complaint, contributing to a blog discussion, completing a movie story, making a job application, writing a brochure for a product etc. Such tasks cover a variety of genres, e.g. descriptive, argumentative, narrative. Writing tasks and types are broadly aligned with the teaching goals at different proficiency levels, and, therefore, tasks in higher levels are more variable and potentially more com-

[9] Compared with NLP tasks, not so many gold standards (Paroubek *et al.*, 2007) are available to the Learner Corpus Research Community.

[10] By simulating human decision on the basis of observable evidence, the task is a formalization of human decision. It could be argued that this kind of competitions using human judgments on learner productions could be interpreted as a way to flesh out the CEFR levels with real observations in terms of regularly observed errors for given levels.

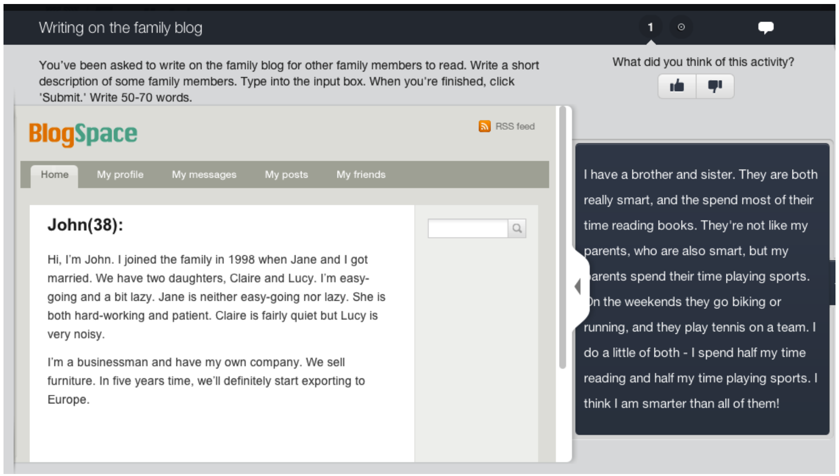
[11] Learning and test data were selected and manipulated independently of the participation of the Cambridge and Education First research teams. It can be accessed at `corpus.mml.cam.ac.uk/efcamdat2/public_html/`.

[12] The curriculum description corresponds to an earlier version of the teaching materials withdrawn in 2013 and no longer in use.

plex, aiming to elicit longer texts than tasks in lower levels, in line with the CEFR descriptors reviewed in the previous section. Thus, texts vary in length from 20-40 words (lower levels) to 150-180 words (higher levels). Information on learner nationality is used as a proxy to estimate L1 background.

Regarding the process of CEFR alignment, Englishtown learners are allocated to a teaching level by EFL teachers, after a placement test. Learners are always allocated at the first level of a stage, that is, levels 1, 4, 7 and 10. As the data come from a real life learning environment, the data are more variable and potentially dynamic as learners progress from lesson to lesson and as individual performance might vary from lesson to lesson. In addition, more variation is expected in the 'placement' levels 1, 4, 7 and 10. EFCAMDAT then differs in this respect from corpora involving examination scripts, e.g. Cambridge Learner Corpus or Merlin Corpus (Boyd *et al.*, 2014). Nevertheless, (Murakami, 2014, 80) presents a comparison between EFCAMDAT and the Cambridge Learner Corpus and concludes in favour of a good CEFR correspondence between the two corpora, at least for A and B CEFR levels.

To appreciate the variety of task prompts within proficiency levels, consider the EFCAMDAT tasks In Figures [13] 1, 2 and 3. All tasks are from the same level but differ in their demands; 6.2 asks learners to create a profile, 6.3 asks learners to provide a set of instructions while 6.5 is a narrative.



Writing on the family blog

You've been asked to write on the family blog for other family members to read. Write a short description of some family members. Type into the input box. When you're finished, click 'Submit.' Write 50-70 words.

What did you think of this activity?

John(38):

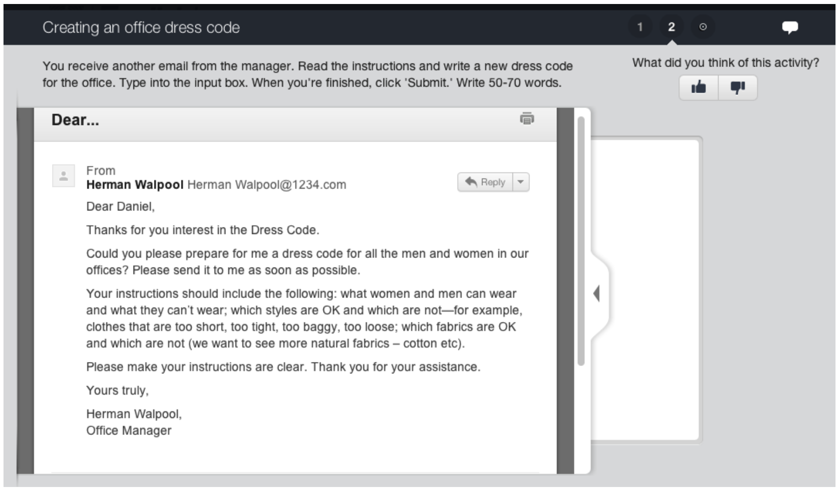
Hi, I'm John. I joined the family in 1998 when Jane and I got married. We have two daughters, Claire and Lucy. I'm easy-going and a bit lazy. Jane is neither easy-going nor lazy. She is both hard-working and patient. Claire is fairly quiet but Lucy is very noisy.

I'm a businessman and have my own company. We sell furniture. In five years time, we'll definitely start exporting to Europe.

I have a brother and sister. They are both really smart, and the spend most of their time reading books. They're not like my parents, who are also smart, but my arents spend their time playing sports. n the weekends they go biking or running, and they play tennis on a team. I do a little of both - I spend half my time reading and half my time playing sports. I think I am smarter than all of them!

Figure 1 – Writing task from lesson 2 of Level 6

[13] A full list of task prompts can be found at https://corpus.mml.cam.ac.uk/efcamdat2/public_html/


Creating an office dress code
1
2
What did you think of this activity?
Dear...
Reply

You receive another email from the manager. Read the instructions and write a new dress code for the office. Type into the input box. When you're finished, click 'Submit.' Write 50-70 words.

From
Herman Walpool Herman Walpool@1234.com

Dear Daniel,

Thanks for you interest in the Dress Code.

Could you please prepare for me a dress code for all the men and women in our offices? Please send it to me as soon as possible.

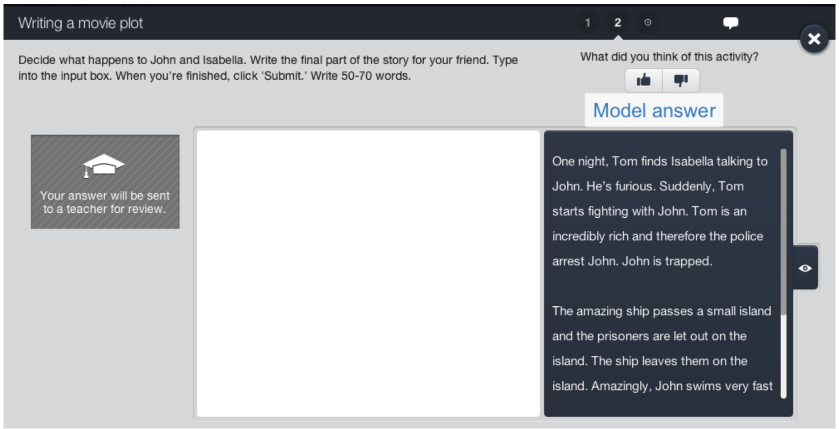
Your instructions should include the following: what women and men can wear and what they can't wear; which styles are OK and which are not—for example, clothes that are too short, too tight, too baggy, too loose; which fabrics are OK and which are not (we want to see more natural fabrics – cotton etc).

Please make your instructions are clear. Thank you for your assistance.

Yours truly,

Herman Walpool,
Office Manager

Figure 2 – Writing task from lesson 3 of Level 6



Decide what happens to John and Isabella. Write the final part of the story for your friend. Type into the input box. When you're finished, click 'Submit.' Write 50-70 words.

Your answer will be sent to a teacher for review.

One night, Tom finds Isabella talking to John. He's furious. Suddenly, Tom starts fighting with John. Tom is an incredibly rich and therefore the police arrest John. John is trapped.

The amazing ship passes a small island and the prisoners are let out on the island. The ship leaves them on the island. Amazingly, John swims very fast

Figure 3 – Writing task from lesson 5 of Level 6

The variety of task prompts is an important feature of the EFCAMDAT corpus. As machine learning systems are very dependent on the training data, the EFCAMDAT task prompts are bound to have an impact on the systems trained on this data set. It is, therefore, possible that systems trained on this test set might show diminished performance on classifying writings elicited through a different set of task prompts. At the same time, the logic of the EFCAMDAT tasks is not uncharacteristic of task-based language learning systems.

For machine learning, a data set is needed. A data set can be described as a table with one observation per line and features as columns. In our case, the data were hybrid, including a column for texts (one learner production for a given task), a column for the assigned CEFR level and other columns for the numerical values produced by linguistic tools (lexical and readability metrics). Extracting the French component of the ECFAMDAT resulted in a proposed data set that included 40,966 text examples written by French L1 learners only and a set of associated characteristics including:

- lexical complexity metrics,
- readability metrics,
- the prediction variable (the linguistic competence of the person).

4.1. Features provided

Fifty-nine feature variables are available. The first (called `fulltext`) is a text value: The full text produced by the person to be assessed (with an average length of 70 words). The other 58 variables are metrics calculated from the text using X. Lu's Lexical Complexity Analyzer (LCA) (Lu, 2014) and the koRpus R package (Michalke, 2017). They describe the degree of vocabulary diversity and the complexity of the text. Among these metrics we find: The number of sentences, words, letters, and syllables, the *Type-Token* ratio (and derived measures), readability measures calculated from the correlation between the number of words and length of words used, and lexical sophistication, which measures the richness of the lexicon using reference inventories. A precise description of the characteristics is available on the website [14] and in Appendix A.

Classification of learners' texts for lexical proficiency is usually based on features such as vocabulary size (Crossley *et al.*, 2011). Recent papers have also incorporated readability metrics (Lissón, 2017). We have not used syntactic metrics (Lu, 2010) but we have taken lexical sophistication into account. Table 9 in Appendix B gives a detailed description of each metric. Lexical sophistication gives information on how unusual the words used by a given learner are. Sophistication is defined as the percentage of low-frequency words belonging to specific morpho-syntactic categories, i.e. verbs, nouns, adjectives and adverbs. Frequency thresholds in lexicons

[14] `cap2018.litislab.fr/competition_annexes_EN.pdf`

Table 1 – Conversion between the levels estimated by EFCAMDAT and the CEFR associated with the size of each class in the training set.

EFCAMDAT	CEFR	sample number
1-3	A1	11,361
4-6	A2	7,688
7-9	B1	5,383
10-12	B2	2,337
13-15	C1	491
16	C2	50

indicate whether they belong to diverse classes. Finally lemmatised versions of words are taken into account.

Features computed with the R package koRpus (Michalke, 2017) include indices of lexical sophistication (e.g. type-token ratio, HD-D/vocd-D, MTLD) and readability metrics (Flesch-kincaid, SMOG, LIX, Dale-Chall). Readability corresponds to the ease with which a reader can process a written text. It often corresponds to grades in the US school system (primary school up to 5th grade, middle school until 8th grade, and high school up to 12th grade). Some details are given in Appendix 2.

4.2. Training labels

The classes which were to be predicted correspond to the six reference levels of the CEFR. The learner data shared on the EFCAMDAT corpus are of 16 different levels. Table 1 shows the conversion between the levels estimated by EFCAMDAT and the CEFR, and the number of each class in the training set.

From a machine learning perspective, this is an unbalanced ordinal classification problem - that is a multiclass classification problem with ranked target classes. "Unbalanced" means that the number of examples for each label is far from equal. "Multi-class" implies that more than one boundary has to be designed to separate the data into six classes (as opposed to a binary classification, which is easier). The ranking nature of the problem is due to the fact that the labels are ordered, assuming A1 $<$ A2 $<$ B1 $<$ B2 $<$ C1 $<$ C2.

4.3. Competition landmarks

The competition took place over two months during the spring of 2018, starting on March 28th and ending on May 28th. During the first month, only the training data was made available. The test data, without the labels, was released on the 8th of May.

The whole data set of 40,966 samples was randomly split into a training set and test set, containing two thirds (27,310) and one third (13,656) of the samples, respectively. Splitting was conducted so that both data sets have the same proportion by class. The labels (i.e. the proficiency levels) of the test set were hidden to the participants and known only by the organisers.

5. Evaluation

The list of metrics suitable for comparing the performances of different ordinal classification models is limited, and all the more so as we were dealing with unbalanced data. Indeed in this case, a classifier's accuracy measured in the proportion of correct classification is inadequate, since it will tend to remove small class sizes from prediction and will not take into account the natural ordering of the class: Predicting A2 instead of A1 is far less severe that predicting C2 instead of A1. To measure the ordinal association between two quantities, the Kendall rank correlation coefficient can be used, but it still fails to address the unbalanced issue. Furthermore, additional prior knowledge is available about the targeted classes. Because many universities require foreign students to achieve B2 for admission, we considered that there is some institutional interest in properly labelling B2. As a result, we deemed that more weight should be given to labelling errors between B1 and B2. A way to tackle these issues is to incorporate costs in decision-making defining fixed and unequal misclassification costs between classes representing prior knowledge about the nature of the problem. Formulaically, the performance measure E used for the competition reads:

$$E = \frac{100}{n} \sum_{i=1}^{6} \sum_{j=1}^{6} C_{ij} N_{ij}, \qquad [1]$$

where N is the confusion matrix of general term N_{ij} counting the number of items of class i classified as j, n the size of the test set and C the cost matrix defined in Table 2 [15].

[15] This section can be used as a how-to guide for setting up a usable data set in a data challenge. To foster communications between communities, we have provided a companion webpage on this github [16] with the code

Table 2 – The classification cost matrix C used to evaluate the models.

Predicted Real	A1	A2	B1	B2	C1	C2
A1	0	1	2	3	4	6
A2	1	0	1	4	5	8
B1	3	2	0	3	5	8
B2	10	7	5	0	2	7
C1	20	16	12	4	0	8
C2	44	38	32	19	13	0

The cost matrix has been designed to address two different issues: the unbalanced nature of the classes and the distance between classes related with their ordered nature. The unbalanced nature of the classes is tackled by setting a high cost to the miss-classification of the under-represented classes. To this end, following Lin *et al.* (2017), costs are set to be proportional to the class entropy $e = -\log p$ where p is the prior probability of each class. Their values are given in Table 3. We can see that based on the entropy vector e, the cost of an error when the true class is C2 is more than seven times higher than the one when the true class is A1. To model the distances between classes, we introduced in Table 3 a weight matrix W of general term $W_{ij} = |i - j| + t_{ij}$ where the correction term $t_{ij} \in \{0, 1, 2\}$ put more emphasis on the distances between classes B1 and B2 on one hand and classes C1 and C2 on the other.

Therefore to cope with the twofold purpose of dealing with unbalanced classes and distance between classes, the cost matrix C has been defined as:

$$C = diag(e)W,$$

where $diag(e)$ is a diagonal matrix with vector e on the diagonal. By construction it is not symmetric. Indeed, for instance when considering classes A1 and C2 with respectively 11,361 and 50 learners, the error at predicting C2 when the true label is A1 is more frequent and less severe than making a mistake the other way round by predicting A1 instead of C2. In that case, $C_{1,5} = 6$ and $C_{5,1} = 44$. Note that the perfect classification corresponds to the diagonal of zeros and would correspond to a penalty score of zero.

used for the evaluation (in R, Python and Matlab). It shows the details of how the matrix cost is calculated for each submission.

Table 3 – Prior knowledge used to build the cost matrix C, that is the initial classification weight matrix W, the prior probability vector p and the associated entropy e for each class

Real \ Estimated	A1	A2	B1	B2	C1	C2	p	e
A1	0	1	2	4	5	7	0.42	0.86
A2	1	0	1	3	4	6	0.28	1.28
B1	2	1	0	2	3	5	0.19	1.64
B2	4	3	2	0	1	3	0.08	2.47
C1	5	4	3	1	0	2	0.02	4.03
C2	7	6	5	3	2	0	0.002	6.31

6. Results and discussion

6.1. The leader board

The competition received 37 submissions from 14 different teams, with an average slightly above 2.5 submissions per team and a maximum of six. Table 4 shows the final leader board of all participating teams together with their affiliation and their best performance measure E (Equation [1]) computed on the test set.

No ties were observed. We provide details of the winning submission including confusion matrices in Table 8. The best score is 3.49[17]. This corresponds to an error rate of 1.41 $\%$ (only 193 misclassified examples out of 13,656 test examples), an average precision of 0.93, an average recall of 0.94 and an average F_1-score of 0.94. For C2, the most difficult class to predict, the precision is 0.76, the recall is 0.73 and the F_1-score is 0.75. With such results, we can effectively consider that the winning team has solved the problem of automatically predicting learners' competence levels as formulated in the competition goals.

To assess the relevance of the final ranking according to E, we proposed to statistically test the significance of their differences. A way to do so is by using McNemar's test for paired proportions, forcing us to reconsider the problem as a binary classification task. To this end, we chose to compare the different solutions in their ability to classify learners above and below B1, since B2 is often used as the threshold for foreign students to be admitted to universities. We therefore applied

[17] Note that the E performance measure is relevant for ranking the proposed methods but, unlike the error rate, it is not open to any interpretation.

Table 4 – Final leader board of the CAp 2018 Competition.

Rank	Team	Affiliation	Score E
1.	Balikasg	Université de Grenoble	3.49
2.	Terislepacom	Telecom Paris Tech	7.29
3.	ICSI	Université de Technologie de Compiègne	8.60
4.	Caoutchouc	Université de Technologie de Compiègne	9.45
5.	ACNK	Université de Technologie de Compiègne	10.43
6.	reciTAL team	Université Pierre et Marie Curie	11.21
7.	TAU	Université de Technologie de Compiègne	12.82
8.	Capitaine-Ad-Hoc	IRIT, Toulouse	14.17
9.	Chamlia	LIP6, Paris	17.52
10.	Haralambous + Lenca	IMT Atlantique, Brest	17.98
11.	MB	Telecom Paris Tech	31.67
12.	Rufino	Instituto Caro y Cuervo, Bogota, Colombia	33.42
13.	Team UTC	Université de Technologie de Compiègne	40.79
14.	Limsi	LIMSI, Orsay	41.52

Edwards' corrected version of McNemar's test on the best four competitors [18]. The resulting p-values were 3.69e-07 (Balikasg vs. Terislepacom), 0.005 (Terislepacom vs. ICSI), and 0.411 (ICSI vs. Caoutchouc). These p-values suggest that the results obtained by the first three competitors, namely Balikasg, Terislepacom and ICSI are significantly different while those of ICSI and Caoutchouc (competitors respectively ranked 3 and 4) are not according to this test. In other words, the first two competitors are undeniably ahead of the competition but there is a clear winner as the team's system performs best at distinguishing learners above and below B1. As to competitors three and four, even though their final scores were different, their proposed classification of learners above and below B1 is not significantly different.

6.2. Analysis of the different proposed solutions

As usual in modern machine learning, most technical implementations consist of a pipeline of different processes including feature engineering, data representation and classification. Feature engineering describes the kind of features taken into account in the analysis. We have classified the features as 'G' for the given features

[18] https://fr.mathworks.com/matlabcentral/fileexchange/189-discrim

(lexical sophistication, lexical complexity, etc.), ’O’ for other linguistic indicators, ’W’ for words, ’C’ for characters and ’S’ for syntactic features. ’Data representation’ refers to how words were considered (raw words, bag of words or word embeddings). Classification sums up the machine learning algorithm, method or family of methods adopted for classification. A summary of the choices proposed by the different teams is given in Table 5. As is often the case in machine learning challenges, the performances of a method cannot be dissociated from the context of its use. Thus, during this competition, boosting has been used both by teams having obtained the best results and by teams having obtained the worst results. Boosting is a machine learning ensemble approach that combines many ”weak” classification methods (weak in the sense of low prediction accuracy) to produce a powerful ”committee”, yielding to a ”strong” classification performance (Friedman *et al.*, 2001). In the following subsection, the solutions proposed by the competitors[19] are presented according to the choices made in terms of features, representation and classification. In Table 5, we have indicated the most relevant linguistic features used by competitors: ’G’ stands for the given features (lexical sophistication, lexical complexity, etc.), ’O’ for other linguistic indicators, ’W’ for words, ’C’ for characters and ’S’ for syntactic features.

Table 5 – Comparison of the different methods proposed by the different teams

Team	Score	Linguistic features	Representation	Classification
Balikasg	3.49	G O W S	Clustering, LDA, BoW	GB trees: LightGBM
Terislepacom	7.29	G W	BoW + SVM	GB trees: XGBoost
ICSI	8.60	G W	BoW	H2O autoML
Caoutchouc	9.45	G W	BoW	Logistic regression
ACNK	10.43	W	Transfer learning	LSTM NN
ReciTAL	11.21	G W S C	BoW + Naive Bayes	GB trees: XGBoost
TAU	12.82	G W	NastText + PCA	Random forest
Cap. Ad-Hoc	14.16	G W S	Doc2Vec, FastText, GloVe	Ensemble of CNN
Chamlia	17.51	W	None	Recurrent bi-dir. NN
Haral-Lenca	17.97	G O W S	Graph2vec	NN
Rufino	33.42	G O W C	Feature selection	kNN regressor
Team UTC	40.79	G	None	GB trees: XGBoost

[19] Note that a detailed description of the winning system "Balikasg" has been published on arXiv (Balikas, 2018).

6.2.1. *Features used*

A secondary objective of this competition was to determine what the most relevant features are for predicting a learner's English proficiency level. To this end, we offered competitors a set of 59 pre-calculated linguistic characteristics described in section 4.1. It turns out that this set was not enough for most of the participants, and many other relevant features were proposed which can be grouped into three categories.

First, some participants were not satisfied by the provided indices, and they therefore recomputed some linguistic indicators and proposed others. For instance, the winning team used spaCy [20] and textstat [21] to recompute the number of sentences and of difficult words and report a slight improvement in performance by adopting these metrics. Also, the team Haral-Lenca used 547 values provided by TAALES software [22].

Second, use of the full text allowed participants to design semantic features based on word representations, topic and language models. To this end, some usual preprocessing tools such as NLTK [23] have been used to clean the text data including capitalization, numbers and characters normalisation and other details using regular expressions. Note that for our challenge, stopword removal is not a good strategy since use of very frequent words (usually, function words such as articles or auxiliary verbs) provides some discriminative information about the English level.

Third, the winning team shows that the use of spelling mistakes and syntactic patterns found in the essays, such as typos and part-of-speech (POS) tags, improves performance. The number of typos can be determined (and thus corrected) using the GNU English dictionary. The use of a POS tagger such as the one included in spaCy enables processing of other features such as a unigram term-frequency approach on top of POS sequences, the size of the largest noun phrase, and the number of distinct POS tags in each essay.

6.2.2. *Representation*

Once features are determined, the choice of representation mode is critical for obtaining good results. Representation often comes with dimensionality reduction using principal component analysis (PCA) or feature selection (see for instance Flach, 2012). Interestingly, Rufino's team performs poorly by selecting the 13 most

[20] spacy.io
[21] github.com/shivam5992/textstat
[22] kristopherkyle.com/taales.html
[23] nltk.org

significant features, while Terislepacom reports very good results, and even an improvement, when selecting only the top 10 most relevant features.

To deal with natural language, different representations were proposed. The most popular, which also gave the best results, is the bag of words (BoW) that counts how many times a word or a bigram appears in a text. To get rid of the size effect, word counts are often replaced with term frequency (TF) and inverse document frequency (IDF) scores across the whole data set. This technique has also been applied to character ngrams and POS tag ngrams. Another way to represent a text is to use a word embedding approach that maps terms or bigrams obtained from a corpus to vectors of real numbers. Most popular approaches are Word2vec [24], FastText [25] and GloVe [26]. Samples of 17,000, 22,000, and 60,000 words and bigrams were used by different teams. The Captain Ad-Hoc team used a Doc2vec representation (Le & Mikolov, 2014) while Haral-Lenca proposed representing complex structures by learning distributed representations of graphs using Graph2vec [27] with not-so-convincing results.

The Balikasg team used embeddings to extract bag-of-clusters representations together with the Latent Dirichlet Allocation (LDA), an unsupervised model that captures text semantics through 30 to 60 latent topics [28]. On top of these classes, a Naive Bayes classifier was used by the ReciTAL team to estimate probabilities for each category.

6.2.3. *Classification methods*

In 2006, the IEEE Conference on Data Mining (ICDM) identified the top 10 machine learning algorithms. Among them competitors used decision trees (random forest), boosting (GB trees), support vector machines (SVM), k nearest neighbors (kNN), Naive Bayes and logistic regression (for details see for instance (Murphy, 2012) or (Abney, 2007)). The well-known "no free lunch theorem" in machine learning states that no single algorithm works best for every problem. As a result, many different algorithms should be tested, while using a hold-out “validation set” of data to evaluate performance and select the most relevant option. More than the method itself, it is the choice of input and the tuning of the hyperparameters that make the difference.

Deep neural networks are missing from the IEEE ICDM list because in 2006 they were not yet as widespread, while today they often establish the state of the

[24] mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model
[25] fasttext.cc/docs/en/english-vectors.html
[26] nlp.stanford.edu/projects/glove/
[27] github.com/MLDroid/graph2vec_tf
[28] github.com/balikasg/topicModelling

art in the field of text classification. More precisely, the generic term "deep neural networks" refers to different types of models used by competitors such as standard feedforward neural networks (referred to as NN), convolutional neural networks (CNN), long short-term memory neural networks (LSTM NN) and the recurrent bi-directional network with attention (Recurrent bi-dir. NN). For more details see for instance (Goldberg, 2017) or the on-line material of the syllabus of the Stanford course CS224d on Deep Learning for natural language processing[29].

Interestingly, on this particular problem, deep neural networks did not deliver the best results and were overtaken by GB trees and logistic regression. However, it is worth mentioning the remarkable performance achieved by the ACNK team. Only using texts written by learners as input, they demonstrate that applying long short-term memory neural networks (LSTM NN) together with transfer learning as implemented in the universal language model fine-tuning (ULMFiT)[30] can be very effective. Transfer learning refers to the knowledge transfer between tasks (Thrun & Pratt, 1998), that is, the use of a model solving one problem (such as language modelling on a WikiText data set) to tackle another related task (such as text classification) using significantly less data than if training from scratch. A common transfer learning strategy is to specialise and fine-tune the first model using data from the application of interest. This popular approach in image classification is now fully lifted into NLP tasks (see ULMFiT for instance).

Finally, at least one competitor (ICSI) used an AutoML tool, an off-the-shelf machine learning method that automatically performs algorithm selection and hyperparameter tuning, obtaining good performances at a low implementation cost. Such an AutoML function is now available in some machine learning frameworks such as H2O[31] (the one used by ICSI) and auto-sklearn[32].

6.2.4. *The software used*

For this competition, Python was the most popular programming language. The tools the participants used are described in Table 6. The software components[33] range from libraries to preprocess natural language to core machine learning frameworks, including tools for word/sentence/text embedding or language model query.

[29] cs224d.stanford.edu/syllabus.html

[30] nlp.fast.ai/category/classification.html

[31] docs.h2o.ai/h2o/latest-stable/h2o-docs/automl.html

[32] automl.github.io/auto-sklearn

[33] The code produced for the competition by Terislepacom team is available at github.com/AlexouGarcia/CAP2018_SharedTask_2nd

Table 6 – Summary of frameworks and libraries the participants used.

Framework/Library	Usage within the competition
	Natural language preprocessing tools
Spacy	Number of words/sentences, Unigram TF, POS...
TextStat	Essay statistics to determine complexity and grade level
NLTK	Text capitalization, numbers and characters normalization...
TAALES	Tool for the Automatic Analysis of Lexical Sophistication
	Text representation learning
Word2Vec, FastText, Glove	Word embedding
Doc2vec	Sentence and essay embedding
Graph2vec	Graph embedding of complex representation of the essays
Balikasg's topicModelling	Latent Dirichlet Allocation for topic modeling
Kenlm	Bigram language modelling with modified Kneser-Ney smoothing
	Machine learning
Sk-learn, Dmlc XGboost, Microsoft LightGBM	Classification pipeline
Keras	High-level NN application programming interface (API)
ULMFiT	LSTM NN training for NLP with transfer learning
H20	Automatic machine learning

6.2.5. *Methodological concerns*

First, as pointed out by Igor Axinti, Yannis Haralambous and Philippe Lenca, it was possible to download the EFCAMDAT corpus in full and retrieve all the texts from test data sets and thus the correct associated labels. For this reason, details of the algorithm used by each participant were requested.

Second, as reviewed in Section 3, the emphasis of the CEFR on the communicative functions that a learner ought to be able to perform at each proficiency level potentially introduces a correlation between task types and proficiency. Indeed, task effects have been demonstrated on elicited language in EFCAMDAT by Alexopoulou *et al.* (2017). Since each EFCAMDAT task comes with a `topic` variable, it was possible that competitors might use this variable to predict proficiency. To prevent the use of this variable and encourage identification of proficiency, irrespective of task/topic, we removed this variable information from the data set. However, in some cases where a specific task prompt elicited very specific lexical items (e.g. proper nouns, such as ’Tour Eiffel’), some competitors were able to retrieve words that were strongly correlated with the given task prompts. This suggests that we should have removed entries containing text related to some specific task prompts.

Note also that competitors suggested metrics which are relevant for distinguishing early stages (e.g. A1, A2) based on repetition and spelling error scores, with no use of the topic.

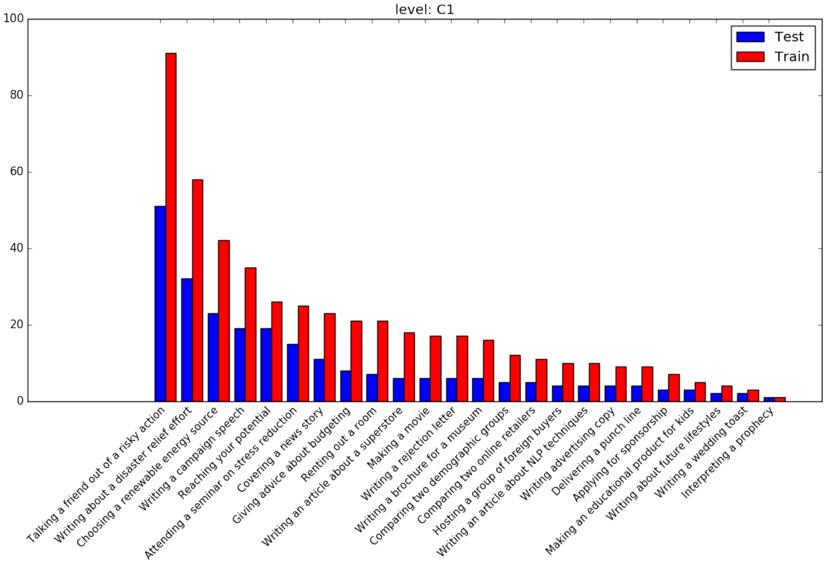


Figure 4 – Distribution of topics for training and test sets in C1

Finally, we realised during the competition that we made the mistake of not removing the `text` variable (the 46th additional variable). This variable (not to be confused with `fulltext`) included the ID numbers of the texts. It was created to help us manipulate the data and was actually correlated with the labels. For this reason we asked the two winning teams to run their solutions without the `text` variable. Results reported in Table 7 show a clear deterioration in terms of performance but not in ranks since the score of the winning team without the `text` variable still outperforms all others.

Table 7 – Comparison of the results obtained by the two winning teams with and without the `text` variable.

	With `text` variable		Without `text` variable	
	Score	Error	Score	Error
1. Balikasg	3.49	1.41%	8.25	2.43%
2. Terislepacom	7.29	2.23%	11.83	3.57%

6.3. Analysis of the results

Table 8 presents the confusion matrix of the best results obtained. The largest confusion rates are observed in consecutive classes, which is normal. However, there is a large gap between C2 and the rest of the classes, and the maximum confusion rate (bold font in the table) is obtained for the C2 samples. In one case out of 5, the system predicts these to be C1. This can be explained by the fact that the C2 class has a very small number of training samples and that the boundary between C1 (level 13-15 in EFCAMDAT) and C2 (level 16) is difficult to establish or somehow fuzzy. Note that among these five misclassified samples two of them are short sentences, which is unusual for the C2 sample.

Table 8 – Confusion matrices by Balikasg in number of examples (left) and percentage (right).

		Predicted						Predicted					
		A1	A2	B1	B2	C1	C2	A1	A2	B1	B2	C1	C2
Real	A1	5,622	45	11	1	2	0	99.0	0.8	0.2	0.0	0.0	0
	A2	19	3,776	45	1	0	3	0.5	98.2	1.2	0.0	0	0.1
	B1	3	10	2,669	4	5	1	0.1	0.4	99.1	0.1	0.2	0.0
	B2	0	3	5	1,138	20	2	0	0.3	0.4	97.4	1.7	0.2
	C1	0	0	2	4	239	1	0	0	0.8	1.6	97.1	0.4
	C2	0	0	0	1	5	19	0	0	0	4.0	**20.0**	76.0

More costly errors also occurred, such as mislabelled samples whose true label is several classes apart. This is the case, for example, for two samples labelled A1 and predicted as C1, and three others, labelled A2 and predicted as C2. In the first case, we note the use of several proper nouns in a single long sentence. In the second case, the sentences are also quite lengthy and tokenisation issues led to an over-interpretation of the proper nouns as rare words. The use of quotes artificially elevated lexical sophistication metrics as well.

All the available EFCAMDAT scripts for advanced learners (C2) of French nationality were included, but the small number of texts for this proficiency level is a clear limitation to the effectiveness of machine learning techniques.

Similarly, instructions for C2 topics account for many findings of the Terislepacom Team. Topics such as "Attending a robotics conference" and "Writing about a symbol of your country" resulted in making words "robot" and "Eiffel" artificially effective predictors. Surprisingly, "UFOs" was also detected as a good predictor for class C2. It is an artifact of only one text but used four times, for the topic "Researching a legendary creature".

Interestingly, analysing this method evidenced the lexical bias introduced by the task prompts used in the EFCAMDAT data. Tasks for early stages are heavily based on self-presentation, so that "task description" is almost a criterial feature of its own. This is partly due to the fact that EFCAMDAT reflects a teaching curriculum from beginner to advanced levels where certain topics and associated lexis are characteristic of specific proficiency levels. Even though we had cleared the essay topic from the competition data set, instructions such as "Introducing yourself by email" or "Writing an online profile" are likely to trigger specific lexical units, which in turn are correlated with proficiency.

6.4. Lessons learnt

As expected, the best results were achieved through the use of many different types of characteristics including the features provided with their re-engineered derivations, the raw text, and syntactic patterns. It was sensitivity to these syntactic and stylistic (repetitions) properties that made the difference between the first and second places in the competition. But the use of all these variables is not necessary, and it has been shown that very good results can be achieved by cleverly selecting only 10 (very well chosen) variables (including the number of sentences, the number of words, the number of letters, the average sentence length) as evidenced by the system of Terislepacom. An important aspect from a linguistic standpoint was the inclusion of stopwords in the analysis, contrarily to standard machine learning techniques. Unexpectedly from a machine learning perspective, deep learning coupled with word embeddings were not able to give the best results. In our opinion, this is due to the amount of data available which is not sufficient to effectively train deep neural networks. Nevertheless, for a classification task using only raw natural text as an input, deep learning together with transfer learning approaches (learning whereby big data is used to model the domain-specific data under scrutiny) can be trained very quickly and efficiently by using tools such as the universal language model fine-tuning (ULMFiT)[34].

Receiving submissions from competitors with a strong mathematical background allowed a range of submissions that were not necessarily infused with natural language processing techniques. Some submissions made the most of packages reputed for their success on language tasks. The strong classification performance of transfer learning methods suggest that these kinds of techniques could be applied to learner data, even for A1 / A2 levels.

However rudimentary the task might seem, this competition may have been the first ever learner data challenge based on the French component of the EFCAMDAT corpus. Perhaps more interesting tasks and formats could be designed. A more refined version of the competition could be organised, taking into account several tracks for several tasks. In one of the tracks, several first languages could be mixed (French was the only one used with this data set) in order to see if specific features could help indentify L1-based errors. For example, the Duolingo challenge had three tracks (three competition subsets) for three language learning trajectories (English to French, English to Spanish, Spanish to English). Last, more linguistic features could be used in the competition. We had not included syntactic features (Lu, 2010) among the linguistic features supplementing the text. More than 400 features can now be automatically computed in the Web-based tool CTAP[35] (Chen & Meurers, 2016) for automatic complexity analysis. Other corpora could be analysed, as well as other languages, such as using the MERLIN data on German, Czech or Italian (Boyd *et al.*, 2014).

7. Conclusions

This challenge based on learner data was a success at the CAp2018 Machine Learning Conference and generated real interest among the machine learning community. The solution suggested by the winner solved the task as it was formulated, with the proviso that the trained model probably suffers from overfitting in the sense that the lexicon used is too heavily dependent on classes. There are 128 topics in the data set (see full distribution in Figure 4). The topic distribution in the test set was similar to the topic distribution in the training set (see Figure 4). With hindsight, we consider that we should have used different topics in the training and test sets since, as Murakami (2016) pointed out, there is a correlation between the level of the learner

[34] nlp.fast.ai/category/classification.html
[35] http://www.ctapweb.com/

and the task instruction in the EFCAMDAT. It is worth noting that the variety of 128 tasks in EFCAMDAT contrasts with corpora like the MERLIN corpus[36] which involve writings from exams and where CEFR levels A and B involve a writing task of the same genre (writing a letter). This means that machine learning systems could not exploit the task or topic characteristics for CEFR alignment, at least not as readily as with our EFCAMDAT task. However, since CEFR descriptors require performance on a variety of tasks, it seems more appropriate to set competitions where systems are trained on one (sub)set of tasks and are able to 'generalise' to a different (sub)set of tasks.

We have highlighted some of the benefits for the learner corpus research community, but they might be made more obvious if the data sets were universally available and potentially processed and discussed by other scientific communities. The full availability of data sets on repositories like Kaggle has been instrumental in the evolution of machine learning. The Duolingo challenge data set was uploaded to a public repository[37] with a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0) and standard bibliographical credits (Settles, 2018). The Kaggle repository includes data sets on sign language[38] or on short text language identification, a task where participants have to predict the language of a Twitter post[39]. The complete availability of the competition data sets in repositories such as Kaggle should be the real order of the day to develop a data culture in learner corpus research. Data sets and notebooks which serve scientific blogs facilitating the implementation of the method (the script and its comments) are part and parcel of the ML community. Such a step within the learner corpus community needs to be balanced against the need to build corpora that can support wider research in language learning and the need to obtain informed consent from learners regarding the use of their data. In the case of EFCAMDAT, the terms of use have taken into account a wider range of academic users in the SLA and LCR communities, while ensuring use in accordance with obtained informed consent inevitably leads to managed access. So far, the data culture has been somewhat hindered in the learner corpus research community due to copyright restrictions or the lack of precise metadata about learner levels. Learner corpora collected in the early days of learner corpus research did not use placement tests systematically, resulting in publicly available learner data sets which often lack a crucial variable assessing learner levels. Learner corpora have been used with data-mining techniques (Jarvis, 2011)

[36] https://merlin-platform.eu/index.php
[37] https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/8SWHNO
[38] https://www.kaggle.com/kumawatmanish/deep-learning-sign-language-dataset
[39] https://www.kaggle.com/c/shorttext-langid/data

and can be used as material for data sets calling for more elaborate methods for automatic treatment and analysis of learner corpus data (Díaz-Negrillo *et al.*, 2013), provided labels are assigned for supervised classification tasks. For the learner corpus research community, money and time should be invested in analyzing properly scored essays on specifically described tasks. Discoverable, accessible and retrievable data is key in the FAIR paradigm (Mons, 2018). Calling for more reproducible research with quantitative methods is consistent with Paquot & Plonsky (2017) plea for more quantitative research in learner corpus research.

References

Abney, S. 2007. *Semisupervised Learning for Computational Linguistics*. London: Chapman and Hall/CRC.

Alexopoulou, T., Michel, M., Murakami, A. & Meurers, D. 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.

Alexopoulou, T., Yannakoudakis, H. & Salamoura, A. 2013. "Classifying intermediate learner English: a data-driven approach to learner corpora". In *Twenty Years of Learner Corpus research: Looking Back, Moving Ahead*, 11–23, Belgium: Presses Universitaires de Louvain.

Attali, Y. & Burstein, J. 2006. "Automated essay scoring with e-rater® v.2". *The Journal of Technology, Learning and Assessment*, **4**(3).

Balikas, G. 2018. "Lexical bias in essay level prediction". *ArXiv e-prints*.

Barker, F., Salamoura, A. & Saville, N. 2015. "Learner corpora and language testing". In S. Granger, G. Gilquin & F. Meunier, Eds., *The Cambridge Handbook of Learner Corpus Research*, Cambridge Handbooks in Language and Linguistics. Cambridge University Press, 511–534.

Baur, C., Caines, A., Chua, C., Gerlach, J., Qian, M., Rayner, M., Russell, M., Strik, H. & Wei, X. 2018. "Overview of the 2018 spoken CALL shared task". In *Interspeech 2018*, 2354–2358, Geneva: ISCA.

Baur, C., Chua, C., Gerlach, J., Rayner, E., Russel, M., Strik, H. & Wei, X. 2017. "Overview of the 2017 spoken CALL shared task". In *Workshop on Speech and Language Technology in Education (SLaTE).*, Stockholm, Sweden.

Boyd, A., Hana, J., Nicolas, L., Meurers, D., Wisniewski, K., Abel, A., Schöne, K., Stindlová, B. & Vettori, C. 2014. "The MERLIN corpus: Learner language and the CEFR". In *LREC*, 1281–1288, Reykjavik, Iceland.

Callies, M. & Paquot, M. 2015. "Learner corpus research: An interdisciplinary field on the move". *International Journal of Learner Corpus Research*, **1**(1), 1–6.

Chen, X. & Meurers, D. 2016. "CTAP: A web-based tool supporting automatic complexity analysis". In *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity (CL4LC)*, 113–119.

Council of Europe 2001a. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment*. Strasbourg, Language Policy Division: Cambridge University Press.

Council of Europe 2001b. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment. Structured Overview of all CEFR Scales*. Strasbourg, Language Policy Division: Cambridge University Press.

Council of Europe 2018. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment; Companion Volume with New Descriptors*. Strasbourg, Language Policy Division: Cambridge University Press.

Crossley, S. A., Salsbury, T., McNamara, D. S. & Jarvis, S. 2011. "Predicting lexical proficiency in language learner texts using computational indices". *Language Testing*, **28**(4), 561–580.

Cushing Weigle, S. 2010. "Validation of automated scores of TOEFL iBT tasks against non-test indicators of writing ability". *Language Testing*, **27**(3), 335–353.

Dahlmeier, D., Ng, H. T. & Wu, S. M. 2013. "Building a large annotated corpus of learner English: The NUS corpus of learner English". In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, 22–31: Association for Computational Linguistics. event-place: Atlanta, Georgia.

Dale, R., Anisimoff, I. & Narroway, G. 2012. "HOO 2012: A report on the preposition and determiner error correction shared task". In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, NAACL HLT '12, 54–62, Stroudsburg, PA, USA: Association for Computational Linguistics. event-place: Montreal, Canada.

Dale, R. & Kilgarriff, A. 2011. "Helping our own: The HOO 2011 pilot shared task". In *Proceedings of the 13th European Workshop on Natural Language Generation*, ENLG '11, 242–249, Stroudsburg, PA, USA: Association for Computational Linguistics. event-place: Nancy, France.

Díaz-Negrillo, A., Ballier, N. & Thompson, P. 2013. *Automatic Treatment and Analysis of Learner Corpus Data*. Amsterdam and Philadelphia: John Benjamins.

Flach, P. 2012. *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge: Cambridge University Press.

Friedman, J., Hastie, T. & Tibshirani, R. 2001. *The Elements of Statistical Learning*, volume 1. New York: Springer series in statistics.

Geertzen, J., Alexopoulou, T. & Korhonen, A. 2013. "Automatic linguistic annotation of large scale L2 databases: The EF-Cambridge open language database (EFCAMDAT)". In *Proceedings of the 31st Second Language Research Forum.*, Somerville, MA: Cascadilla Proceedings Project.

Goldberg, Y. 2017. *Neural Network Methods for Natural Language Processing*. Synthesis Lectures on Human Language Technologies. San Rafael (CA): Morgan & Claypool Publishers.

Granger, S., Kraif, O., Ponton, C., Antoniadis, G. & Zampa, V. 2007. "Integrating learner corpora and natural language processing: A crucial step towards reconciling technological sophistication and pedagogical effectiveness". *ReCALL*, **19**(3), 252–268.

Hawkins, J. A. & Buttery, P. 2010. "Criterial features in learner corpora: Theory and illustrations". *English Profile Journal*, **1**(01).

Hawkins, J. A. & Filipović, L. 2012. *"Criterial features in L2 English: Specifying the reference levels of the Common European Framework"*, volume 1 of *English Profile Studies*. United Kingdom: Cambridge University Press.

Higgins, D., Ramineni, C. & Zechner, K. 2015. "Learner corpora and automated scoring". In S. Granger, G. Gilquin & F. Meunier, Eds., *The Cambridge Handbook of Learner Corpus Research*, Cambridge Handbooks in Language and Linguistics. Cambridge University Press, 587–604.

Hopman, E., Thompson, B., Austerweil, J. & Lupyan, G. 2018. "Predictors of L2 word learning accuracy: A big data investigation". In *the 40th Annual Conference of the Cognitive Science Society (CogSci 2018)*, 513–518.

Jarvis, S. 2011. "Data mining with learner corpora". In F. Meunier, S. De Cock, G. Gilquin & M. Paquot, Eds., *A Taste for Corpora: In Honour of Sylviane Granger*, volume 45. Amsterdam and Philadelphia: John Benjamins, 127–154.

Jarvis, S. & Paquot, M. 2015. "Learner corpora and native language identification". In S. Granger, G. Gilquin & F. Meunier, Eds., *The Cambridge Handbook of Learner Corpus Research*. Cambridge: Cambridge University Press, 605–628.

Le, Q. V. & Mikolov, T. 2014. "Distributed representations of sentences and documents". *ArXiv: 1405.4053*.

Leacock, C., Chodorow, M., Gamon, M. & Tetreault, J. 2010. "Automated grammatical error detection for language learners". *Synthesis Lectures on Human Language Technologies*, **3**(1), 1–134.

Lin, T.-Y., Goyal, P., Girshick, R., He, K. & Dollár, P. 2017. "Focal loss for dense object detection". In *Proceedings of the IEEE International Conference on Computer Vision*, 2980–2988.

Lissón, P. & Ballier, N. 2018. "Investigating learners' progression in french as a foreign language: vocabulary growth and lexical diversity". CUNY Student Research Day. Poster.

Lissón, P. 2017. "Investigating the use of readability metrics to detect differences in written productions of learners: a corpus-based study". *Bellaterra Journal of Teaching & Learning Language & Literature*, **10**(4), 68–86.

Liu, B. 2012. *Sentiment Analysis and Opinion Mining*. San Rafael, Calif.: Morgan & Claypool Publishers.

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

Lu, X. 2014. *Computational methods for corpus annotation and analysis*. New York: Springer.

Magerman, D. M. 1995. "Statistical decision-tree models for parsing". In *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics*, 276–283, Stroudsburg, PA, USA: Association for Computational Linguistics.

Malmasi, S., Evanini, K., Cahill, A., Tetreault, J., Pugh, R., Hamill, C., Napolitano, D. & Qian, Y. 2017. "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*, 62–75, Copenhagen, Denmark: Association for Computational Linguistics.

Meurers, D. 2015. "Learner corpora and natural language processing". In S. Granger, G. Gilquin & F. Meunier, Eds., *The Cambridge Handbook of Learner Corpus Research*. Cambridge: Cambridge University Press, 537–566.

Michalke, M. 2017. *koRpus: An R Package for Text Analysis*. (Version 0.10-2), Available at: https://reaktanz.de/?c=hacking&s=koRpus (accessed October 2018).

Mons, B. 2018. *Data Stewardship for Open Science: Implementing FAIR Principles*. London: Chapman and Hall/CRC.

Murakami, A. 2014. *Individual variation and the role of L1 in the L2 development of English grammatical morphemes: Insights from learner corpora*. PhD thesis, University of Cambridge.

Murakami, A. 2016. "Modeling systematicity and individuality in nonlinear second language development: The case of English grammatical morphemes". *Language Learning*, **66**(4), 834–871.

Murphy, K. P. 2012. *Machine Learning. A Probabilistic Perspective. Adaptive Computation and Machine Learning*. Cambridge (MA): MIT Press.

Ng, H. T., Wu, S. M., Briscoe, T., Hadiwinoto, C., Susanto, R. H. & Bryant, C. 2014. "The CoNLL-2014 shared task on grammatical error correction". In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, 1–14, Baltimore, Maryland: Association for Computational Linguistics.

Nissim, M., Abzianidze, L., Evang, K., van der Goot, R., Haagsma, H., Plank, B. & Wieling, M. 2017. "Sharing is caring: The future of shared tasks". *Computational Linguistics*, **43**(4), 897–904.

O'Keeffe, A. & Mark, G. 2017. "The English grammar profile of learner competence". *International Journal of Corpus Linguistics*, **22**(4), 457–489.

Page, E. B. 1968. "The use of the computer in analyzing student essays". *International Review of Education / Internationale Zeitschrift für Erziehungswissenschaft / Revue Internationale de l'Education*, **14**(2), 210–225.

Paquot, M. & Plonsky, L. 2017. "Quantitative research methods and study quality in learner corpus research". *International Journal of Learner Corpus Research*, **3**(1), 61–94.

Paroubek, P., Chaudiron, S. & Hirschman, L. 2007. "Principles of evaluation in natural language processing". *Traitement Automatique des Langues*, **48**(1), 7–31.

Rich, A., Popp, P. O., Halpern, D., Rothe, A. & Gureckis, T. 2018. "Modeling second-language learning from a psychological perspective". In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 223–230.

Sang, E. F. & De Meulder, F. 2003. "Introduction to the conll-2003 shared task: Language-independent named entity recognition". *arXiv preprint cs/0306050*, 142—-147.

Settles, B. 2018. "Data for the 2018 Duolingo shared task on second language acquisition modeling (SLAM)". Available at: https://doi.org/10.7910/DVN/8SWHNO. (accessed October 2018).

Settles, B., Brust, C., Gustafson, E., Hagiwara, M. & Madnani, N. 2018. "Second language acquisition modeling". In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 56–65.

Shermis, M. D., Burstein, J., Higgins, D. & Zechner, K. 2010. "Automated essay scoring: Writing assessment and instruction". In P. Peterson, E. Baker & B. McGaw, Eds., *International Encyclopedia of Education (Third Edition)*. Oxford: Elsevier, 20–26.

Tetreault, J., Burstein, J., Kochmar, E., Leacock, C. & Yannakoudakis, H. 2018. *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*. New Orleans, Louisiana: Association for Computational Linguistics.

Thewissen, J. 2015. *Accuracy Across Proficiency Levels: A Learner Corpus Approach*. Louvain: Presses universitaires de Louvain.

Thrun, S. & Pratt, L. 1998. *Learning to Learn*. Norwell, MA, USA: Kluwer Academic Publishers.

Vajjala, S. & Loo, K. 2014. "Automatic CEFR level prediction for Estonian learner text". In *NEALT Proceedings Series*, volume 22, 113–128.

Volodina, E., Pilán, I. & Alfter, D. 2016. "Classification of Swedish learner essays by CEFR levels". *CALL Communities and Culture–Short Papers from EUROCALL*, **2016**, 456–461.

Wisniewski, K. 2017. "Empirical learner language and the levels of the Common European Framework of Reference". *Language Learning*, **67**(S1), 232–253.

Yannakoudakis, H., Briscoe, T. & Medlock, B. 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 - Volume 1*, HLT '11, 180–189, Stroudsburg, PA, USA: Association for Computational Linguistics.

Yannakoudakis, H., Kochmar, E., Leacock, C., Madnani, N., Pilán, I. & Zesch, T. 2019. *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*. Florence, Italy: Association for Computational Linguistics.

Appendix 1: The list of textual metrics used as given features in the competition

1. sentences: Number of sentences.
2. words: Number of words.
3. letters: Named vector with total number of letters ("all") and possibly several entries
4. punct: Number of punctuation characters.
5. avg.sentc.length: Average sentence length (number of words per sentence)
6. avg.word.length: Average word length (number of characters per word)
7. avg.syll.word: Average number of syllables per word
8. sntc.per.word: Number of sentences per word.

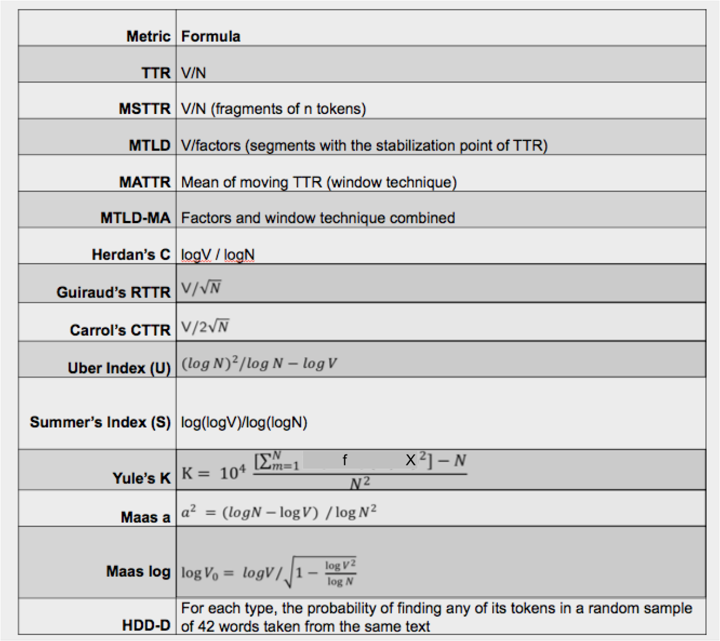
9. TTR: type to token ratio
10. ARI: Automated Readability Index. It takes into account the number of token words divided by the number of syllables and the number of prepositions in the text.
11. Bormuth: Bormuth readability index. It gives an estimation of the grade required to understand the text. The computation is based on the most frequent 3 000 words in English (Dale-Chall list).
12. Coleman.C1: Readability Formulas, taking into account monosyllabic words
13. Coleman.C2: variant also taking into account the number of words divided by the number of sentences
14. Coleman.C3 variant also taking into account the proportion of pronouns among words
15. Coleman.C4 variant also taking into account the proportion of prepositions among words
16. Coleman.Liau : readability index, proportional to the number of letters and sentences (every 100 words)
17. Dale.Chall : readability index (1995), which reflects the degree of familiarity of the lexicon, compared to the 3,000 most frequents words in English (Dale-Chall list).
18. Danielson.Bryan.DB1 & Danielson.Bryan.DB2 : two readability formulas based on the number of characters (space included).
19. Dickes.Steiwer: readability for German that takes into account values proportional to the number of words, characters and TTR.
20. DRP: (*Degrees of Reading Power*) measure readibility from Bormuth index.
21. ELF : (*Easy Listening Formula*): number of polysyllabic words divided by the number of sentences.
22. Farr.Jenkins.Paterson: a simplified version of Flesch, where the number of one syllable words per 100 words replaces the number of syllables per 100 words.
23. Flesch : the English values for this language-dependent metrics have been used. The index takes into account the number of syllables. It ranges between 100 (easy texts) and 0 (very difficult texts).
24. Flesch.Kincaid : this metric was developed with Vietnam draftees to assess the US school grade corresponding to the difficulty level of a text.
25. FOG : readability index suggested in the 1950's. It measures the number of years of study (school grade) required to understand a text on its first reading.

It takes into account the number of words per sentence and the proportion of words with three syllables or more.

26. FORCAST : (FORCAST = Patrick FORd, John CAylor and Thomas STicht) a method implemented with Vietnam draftees, which is based on word length.
27. Fucks : a stylistic feature proposed by W. Fucks. The number of characters divided by the number of words is multiplied by the number of words divided by the number of sentences.
28. Linsear.Write : readability index that takes into account the number of words of three syllables or more, the number of words and the number of sentences.
29. LIX : this readability index was first proposed for Swedish, it takes into account the proportion of words of seven letters or more. Texts with a 25 index are supposed to be easy to read, "normal" texts are around 40 and texts above 50 are considered to be difficult to read.
30. nWS1 to nWS4 : these readability indices proposed in the 80's for the analysis of German (Neue Wiener Sachtextformeln), take into account -in variable proportions- words of three syllables or more and words of six letters or more.
31. RIX : adaptation for English of the LIX index. It takes into account the number of six letters or more divided by the number of sentences.
32. SMOG : *Simple Measure of Gobbledygook* (SMOG). Readability Index based on the square root of the number of polysyllabic words computed at the beginning, middle and end of the text.
33. Spache : readability index based on the number of words of a text that is not in Spache reference inventory of words.
34. Strain : readability index for medias proposed in 2006, which takes into account the number of syllables.
35. Traenkle.Bailer.TB1 & Traenkle.Bailer.TB2 : readability indices taking into account the proportion of prepositions (Traenkle.Bailer.TB1) and conjunctions (Traenkle.Bailer.TB2).
36. TRI (Kuntzsch's Text-Redundanz-Index) readability index initially suggested for German newspapers, it takes into account the number of punctuation symbols and foreign words.
37. Tuldava: a supposedly language-independent readability index that takes into account the logarithm of the number of words divided by the number of sentences.
38. Wheeler.Smith: readability index proposed in the 1650s that takes into account words of two syllables ore more.
39. CTTR : algorithm proposed by Carroll to smooth TTR.

40. HD-D (vocd-D): lexical sophistication index based on the likelihood to find a given word in a 42 word window.
41. Herdan's C : log(V) / log(N), where V is the number of types and N the number of tokens.
42. Maas & lgV0 : indices of lexical complexity suggested in 1972, which take into account logarithms of types and tokens '
43. MATTR: (*Moving Average of TTR*), computed by means of a mobile window. Returns "NA" if the text has less than 400 words.
44. MSTTR (Mean Segmental Type-Token Ratio): averages TTR over several segments.
45. MTLD (Measure of Textual lexical sophistication): corrected measure of the TTR
46. Root TTR : rooted square TTR
47. Summer: lexical sophistication index, log(log(V)) / log(log(N))
48. TTR.1 : rounded Type-Token ratio
49. Yule's K : lexical sophistication index proposed by Yule in 1944.
50. level: the European level (from A1 to C2) of the learner that we try to predict

Appendix 2: Definition of some metrics used as features



Metric	Formula
TTR	V/N
MSTTR	V/N (fragments of n tokens)
MTLD	V/factors (segments with the stabilization point of TTR)
MATTR	Mean of moving TTR (window technique)
MTLD-MA	Factors and window technique combined
Herdan's C	logV / logN
Guiraud's RTTR	$V/\sqrt{N}$
Carrol's CTTR	$V/2\sqrt{N}$
Uber Index (U)	$(log\ N)^2/log\ N - log\ V$
Summer's Index (S)	log(logV)/log(logN)
Yule's K	$K = 10^4 \frac{[\sum_{m=1}^{N} f \quad X^2] - N}{N^2}$
Maas a	$a^2 = (logN - \log V)\ / \log N^2$
Maas log	$\log V_0 = logV/\sqrt{1 - \frac{\log V^2}{\log N}}$
HDD-D	For each type, the probability of finding any of its tokens in a random sample of 42 words taken from the same text

Table 9 – Definition of some metrics used as features. In the table, N denotes the number of words, V the number of type of words, X the frequency vector for each type and f the frequency vector for each X (adapted from (Lissón & Ballier, 2018)).