Measuring Text Complexity for Italian as a Second Language Learning Purposes

Luciana Forti[1], **Alfredo Milani**[2], **Luisa Piersanti**[2],
Filippo Santarelli[1], **Valentino Santucci**[1], **Stefania Spina**[1]

[1]Department of Humanities and Social Sciences,
University for Foreigners of Perugia, Perugia (Italy)
[2]Department of Mathematics and Computer Science,
University of Perugia, Perugia (Italy)
{luciana.forti,valentino.santucci,stefania.spina}@unistrapg.it
luisa.piersanti@studenti.unipg.it, alfredo.milani@unipg.it
filippo.santarelli@unicam.it

Abstract

The selection of texts for second language learning purposes typically relies on teachers' and test developers' individual judgment of the observable qualitative properties of a text. Little or no consideration is generally given to the quantitative dimension within an evidence-based framework of reproducibility. This study aims to fill the gap by evaluating the effectiveness of an automatic tool trained to assess text complexity in the context of Italian as a second language learning. A dataset of texts labeled by expert test developers was used to evaluate the performance of three classifier models (decision tree, random forest, and support vector machine), which were trained using linguistic features measured quantitatively and extracted from the texts. The experimental analysis provided satisfactory results, also in relation to which kind of linguistic trait contributed the most to the final outcome.

1 Introduction

The task of automatically classifying a text according to its different levels of complexity has had various applications in a number of different fields. It is key in mood and sentiment analysis, in the detection of hate speech, in text simplification, and also in the assessment of text readability in relation to both native and non-native readers.

Being able to select a text and compare it with others is a central concern in the field of second language learning. When choosing a text to be used in a lesson or as part of a language test, a teacher and/or language test developer will generally assess the suitability of that text on the basis of several aspects: the need to adhere to a specific syllabus and curriculum, as well as general guidelines and test specifications. Other aspects that are considered include learner-related variables such as their linguistic needs, their educational background, and their age, all elements involving other aspects such as text genre, text type, tasks to be assigned to the text, and so on.

According to the literature, there is wide consensus on specific characteristics that can influence the difficulty of a text in the context of a reading comprehension task. These characteristics have a role in terms of the cognitive demands that a text will impose on its reader (Bachman and Palmer, 2010)(Purpura, 2014). These characteristics are text length, grammatical complexity, word frequency, cohesion, rhetorical organization, genre, text abstractness, subject knowledge and cultural knowledge. All these aspects relate to readability, and are often evaluated intuitively and subjectively by individual experienced teachers, who will then use a given text deemed to be representative of a certain proficiency level in a lesson or as part of a test.

Although this kind of sensitivity to the text is extremely valuable, especially when adapting a lesson or test item to the specific needs of a group of learners, its limitations are evident for at least two reasons: the evaluation is performed by single teachers or test developers at the one time and it is not reproducible; the evaluation is conducted solely on the basis of observable qualitative features of a text.

In the context of language assessment, text selection for the purposes of a reading comprehension task has considerable implications with regard to the interpretation of test scores: a text subjectively deemed suitable for a given proficiency level, which would have objectively been deemed otherwise, will inevitably hinder the validity of the overall testing process. The same can be argued for text selection aimed at lesson planning: an in-

Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 360–368
Florence, Italy, August 2, 2019.

adequate text chosen for a given class will hinder the whole learning process.

As a result, an automatic system able to used extract objective and reproducible information about a text, combining qualitative and quantitative data, is highly desirable in the field of second language learning, though still largely lacking, especially for the Italian language and in relation to different proficiency levels. The Common European Framework of Reference for Languages (CEFR) descriptors are unable to provide this kind of support in relation to the readability of a text.

In this study, we assess the effectiveness of an automatic classification tool for the evaluation of text complexity in Italian. We used a dataset of texts used at CVCL, Centro Valutazione Certificazioni Linguistiche, one of the main Italian language testing centres with sections all over the world, based at the University for Foreigners of Perugia. Each text in the dataset was labeled by test development experts according to the CEFR descriptors. The dataset was used to train a classification model, enabling it to automatically predict the proficiency level of any text in input. The dataset was used to test three different classifiers: decision tree, random forest and support vector machine. The main difference between this study and the related work in the field that will be described in the following paragraph is that a set of linguistic features is used to distinguish texts from the perspective of CEFR levels. Therefore, linguistic features measured quantitatively and extracted from the texts are used to train the classification models that, in turn, allow to predict the proficiency level of an unseen text.

The rest of the article is organized as follows. The literature related to this work is described in Section 2. The architecture of the system is introduced in Section 3, while the definitions of the linguistic features adopted in the study are provided in Section 4. Experiments are discussed in Section 5, while the conclusions are drawn in Section 6 together with future lines of research.

2 Related Work

The assessment of text readability in relation to its complexity has been a central research topic for many decades now. In particular, advances in computational linguistics and the development of corpora, along with the availability of sophisticated language technologies, allow the capturing of a wide variety of increasingly complex linguistic features that are able to affect the readability of a text.

A number of studies aimed at developing automatic readability measures have focused on the English language, both for the simplification of administrative texts and for the purposes of first and second language learning. In more recent years, these studies have also involved other languages, such as French, Swedish, Dutch, German and Portuguese.

The texts used as a gold standard to train the classification models vary. For French, the corpus of texts was a second language coursebook corpus, containing texts developed by expert teachers and materials' designers (François and Fairon, 2012). A similar approach has been used for Swedish (Pilán et al., 2016), with the subsequent addition of a corpus of texts produced by second language learners (Pilán and Volodina, 2018). Other studies have used exams texts (Branco et al., 2014) or a combination of exam texts and native texts (Xia et al., 2011). One study on the readability of Dutch texts (Velleman and Van der Geest, 2014) uses a set of reference texts calibrated in order to represent a range of reading skills, while another one tailored for English (Vajjala and Meurers, 2016) includes a wide range graded corpora to cater for both natives' and learners' reading skills in both general and specialist language needs.

In terms of features, a number of systems have been developed, such as the Flesch-Kincaid (Kincaid and Lieutenant Robert, 1975), Coh-metrix (Graesser et al., 2004) and CTAP (Xiaobin and Meurers, 2016). In relation to the Italian language, three main approaches have been explored: the Flesch-Vacca formula, an adaptation of the Flesch-Kincaid formula for English (Franchina and Vacca, 1986), the GulpEase index (Lucisano and Piemontese, 1988), and READ-IT (Dell'Orletta et al., 2011).

In the Flesch-Kincaid formula, and its Italian counterpart, text complexity is measured with reference to the average length of words, based on syllables, and the average length of sentences, based on words. In addition to this, the formula provides an output indicating the approximate number of years spent in the education system that are necessary to comprehend a given text. The GulpEase index provides information that is similar to the Flesch-Kincaid formula, though it

is based on considerably different characteristics. First, it is created directly on and for the Italian language. Second, though it includes the average length of words as well as the average length of sentences, the former is calculated on the basis of letters, not syllables, which aids the automatic treatment of the text.

For both of these measures, values range from 1 to 100, namely the highest and lowest textual complexity levels respectively. READ-IT, on the other hand, is based on a number of raw text, lexical, morpho-syntactic, and syntactic features, based on Support Vector Machines. This set of features is used together with a training corpus in order to develop a statistical model that is able to assess the complexity of newly inputted texts. The training corpus is formed by newspaper articles and a simplified reader of newspaper articles.

The aim of these measures developed for the Italian language have so far concerned the requirements of text simplification of administrative texts or other typically complex texts, in order to meet the needs of people with low literacy levels or with mild cognitive disorders. To the best of our knowledge, this study represents the first attempt to automatically classify Italian texts on the basis of a wide set of linguistic features, and in relation to the CEFR levels. In this respect, it lays the groundwork for a new resource in the context of Italian as a second language learning and teaching.

3 The System's Architecture

The problem of automatically measuring text complexity through the CEFR proficiency levels has been cast to a supervised classification problem.

We collected a dataset of texts labeled by the experts of the CVCL center of the University for Foreigners of Perugia. This dataset is used to train a classification model that, in turn, allows to automatically predict the proficiency level of any text in input.

As it is possible to see from the system's architecture depicted in Figure 1, the classification model does not directly work with the texts in their pure form. Indeed, any text is converted to a vector of numeric features and then passed on to the classification model (both for training or level prediction).

This scheme allows, on the one hand to adopt the most common classification models available in the machine learning literature (Shalev-Shwartz and Ben-David, 2014) and, on the other hand, to build a classification model based only on the linguistic features of the text that, we think, are what discriminate texts from the point-of-view of proficiency classes.

Therefore, the most important part of our system is the component performing the "Linguistic Features Extraction" phase. This component has been implemented on top of Natural Language Processing (NLP) tools for the Italian language. In particular, we have adopted the NLP pipeline tools provided by Tint (Palmero Aprosio and Moretti, 2016), i.e., the Italian counterpart of the widely known Stanford CoreNLP tool (Manning et al., 2014). The linguistic features used in this work are detailed in Section 4.

Although most of the recently proposed works in NLP use semantically based features (Santucci et al., 2018), such as the well known word embeddings system introduced in (Mikolov et al., 2013) and (Bojanowski et al., 2016), it is worthwhile to note that here we chose lexical and syntactic features because they are the key linguistic traits for distinguishing different CEFR levels.

Regarding the classification model, we ran experiments on our system with three widely known models: decision tree (DT), random forest (RF), and support vector machine (SVM)[1]. While DT and RF provide more interpretable models, that can be analyzed ex post by linguistic scholars, we expect that SVM should reach a larger accuracy.

In our prototypical system, we have used the implementations of DT, RF and SVM available in the widely adopted "Sci-Kit Learn" library (Pedregosa et al., 2011).

4 Linguistic Features

The features used for predicting the text level are inspired by those adopted in (Dell'Orletta et al., 2011). These linguistics features have been divided into four main categories, and are described as follows.

4.1 Raw text features

Raw text features are the most elementary type of features considered here and they were computed through the tokenization of the text in input. In particular, they are:

[1] See (Shalev-Shwartz and Ben-David, 2014) for a description of the models employed here

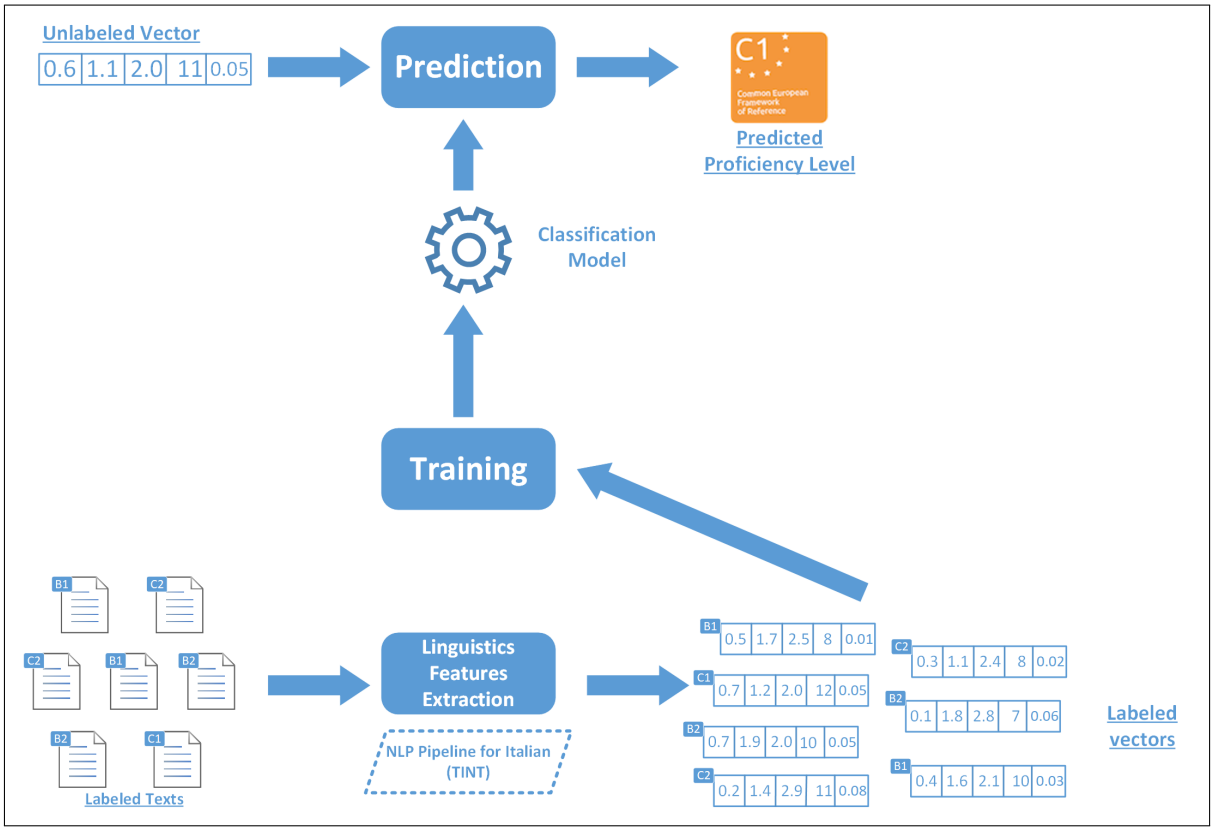

Figure 1: Architecture of the System

- *Word Length*, computed as the average number of characters per word, and
- *Sentence Length*, i.e., the average number of tokens per sentence.

4.2 Lexical Features

Lexical features are mainly computed through the lemmatization of the text's tokens and with reference to the New Basic Italian Vocabulary (NBIV) (De Mauro and Chiari, forthcoming) which includes the following three reference wordlists:

- fundamental (F) words (i.e.: the first 2000 most frequent words), such as *cane, gatto*;
- high usage (HU) words (i.e: occurring between frequency ranks 2000 and 4300) , such as *accademia, incapace, orribile*;
- high availability (HA) words (i.e.: identified in De Mauro and Chiari (forthcoming) through a native speaker judgment questionnaire), such as *affannato, mandarino, salvadanaio*.

Therefore, the considered lexical features are:

- *Lemmas in NBIV*, i.e., the percentage of text's lemmas belonging to NBIV;
- *Lemmas' distribution with respect to NBIV*, i.e., the distribution of the text's lemmas in the previous point among the three subsets F, HU and HA of NBIV (this feature is a vector of three numbers);
- *Type Token Ratio* (TTR), computed as the ratio between the number of different lemmas and the number of tokens in the text; however, since TTR is highly influenced by the text length, we restricted the computation to the first 100 tokens of every text in input.

4.3 Morpho-syntactic features

Morpho-syntactic features are computed basing on the Part-of-speech (POS) tagging and the morphological analysis performed on the text in input. In particular, we have considered the following features:

- *POS Tags Distribution*, i.e., for each POS tag p, it is computed the ratio between the number of tokens with type p and the total number of tokens in the text;

- *Lexical Density*, computed as the ratio between the number of content words (i.e., words tagged as nouns, adjectives, adverbs or verbs) and the total number of tokens in the text;

- *Verbal Moods Distribution*, i.e., the distribution of the seven verbal moods among the verbal tokens of the text.

4.4 Syntactic Features

The text in input is syntactically parsed by Tint and, for each sentence of the text, a dependency parse tree is produced. The syntactic features, described in the following, are then computed through dependency trees.

- *Dependency Types Distribution*, i.e., for each possible dependency type d, we computed the ratio between the number of dependencies with type d and the total number of dependencies, considering all the parse trees.

- *Depth of the Parse Trees*, computed as the maximum depth among all the parse trees.

- *Length of non-Verbal Chains*, i.e., the average length of the paths without verbal nodes in the parse trees.

- *Verbal Roots*, i.e., the percentage of parse trees with a verbal root.

- *Average Verbal Arity*, where the verbal arity of a verbal node v is the number of dependency links with v as a head.

- *Subordinates*, i.e., the number of subordinate clauses in the parse trees.

- *Average Length of the Dependency Links*, where the length of a dependency link between two tokens t_1 and t_2 is the distance, in terms of number of tokens, between t_1 and t_2 in the syntagmatic dimension of the sentence.

5 Experiments

Experiments have been held in order to: analyze the effectiveness and the robustness of the prototypical classification system here proposed, and gain useful insights about which features discriminate more the texts.

The rest of the section is organized as follows. Section 5.1 describes the corpus of texts and the datasets used in our experimentation. The experimental design is described in Section 5.2. The effectiveness of our system is analyzed in Section 5.3, while its robustness is discussed in Section 5.4. Finally, Section 5.5 analyzes the contribution of the different features selected for this work.

	B1	**B2**	**C1**	**C2**	**Total**
2C	0	129	0	97	226
4C	194	129	103	97	523

Table 1: Distribution of proficiency levels across datasets 2C and 4C.

5.1 Corpus and Datasets

An important preliminary step to the experiments was the creation of a reliable corpus of labeled texts. In regard to this, we collected 523 texts with CEFR levels manually marked by expert language test developers. The corpus contains texts for the four CEFR levels B1, B2, C1 and C2.

Two different datasets, namely *4C* and *2C*, have been extracted from the corpus. While *4C* (i.e., four classes) corresponds to the whole corpus, the smaller dataset *2C* (i.e., two classes) contains the subset of 226 texts belonging to the levels B2 and C2. Table 1 provides the distribution of the different levels for both datasets.

Two main reasons motivated the introduction of the smaller dataset *2C*. First, the distribution of the proficiency levels in 4C is unbalanced, therefore a smaller and more balanced dataset such as 2C can be more reliable in terms of representativeness. Second, as the classification models do not treat the four levels as part of an ordinal scale, thus ignoring the natural ordering characterizing them, choosing two levels instead of four eliminates this issue.

5.2 Experimental Design

We tested three classifier models, namely, Decision Tree (DT), Random Forest (RF) and SVM, on both the datasets *2C* and *4C*.

For each dataset, the effectiveness and robustness of each model was evaluated using the nested cross-validation scheme (Varma and Simon, 2006). Two nested cross-validation loops were performed: the outer loop aims at estimating the effectiveness of the model setting which is calibrated in the inner loop. Both loops use 5 stratified folds. The inner loop performs an exhaustive

Parameters Name	Values
Feature-wise normalization	NN, SS, L2, RS
Split quality measure (`criterion`)	GI, IE
Min. impurity decrease (`min_impurity_decrease`)	0, 2
Min. samples to split (`min_samples_split`)	2, 10, $\frac{N_S}{10}$
Min. samples per leaf (`min_samples_leaf`)	1, 5, 10
Max. number of features (`max_features`)	$\sqrt{N_F}$, N_F
Max. number of leaf nodes (`max_leaf_nodes`)	2, 5
Weights associated with classes (`class_weight`)	1, $\frac{N_S}{N_C \cdot n_k}$

Table 2: Parameters space for the Decision Tree and the Random Forest classifiers. The original name of each parameter in the Sci-Kit documentation is in typewriter font within brackets. The function used to measure the quality of a split can either be the Gini impurity (GI) or the Information entropy (IE). N_F is the number of features, N_S and N_C are the number of samples and the number of classes in the dataset, respectively; n_k is the number of samples in the k-th class of the dataset.

grid search on the hyper-parameters space, cross-validated on the training and validation sets obtained by the outer loop. Every grid search returns the setting of hyper-parameters which maximizes the (macro-averaged) F_1-score. Then, the generalization ability of the selected model setting is assessed on the test sets generated by the outer loop and using the classic metrics accuracy, precision, recall and F_1-score.

The linguistic features described in Section 4 may have different scales, hence we introduced a preprocessing step to normalize them. Four normalization methods were considered: no normalization at all (NN), L^2 normalization (L2), standardization (SS), and robust standardization (RS) (which, with respect to SS, do not consider the outlier values). Hence, the choice of the normalization method is a further hyper-parameter which is tuned by the grid search.

The calibrated hyper-parameters and their ranges are provided in Table 2 for DT and RF, and Table 3 for SVM.

Finally, in order to reduce the computational effort, we fixed the following hyper-parameters:

- the maximum tree depth of DT and RF has been set to $\lfloor \sqrt{N_F} \rfloor$, where N_F is the number of features;
- the number of trees in RF has been set to 100;
- the SVM uses the radial basis function as kernel type.

Parameters Name	Values
Penalty parameter (`C`)	0.5, 0.75, 1.0, 1.25, 1.5
Kernel coefficient γ (`gamma`)	10^{-3}, 10^{-4}
One-vs-rest or one-vs-one (`decision_function_shape`)	OvO, OvR
Weights associated with classes (`class_weight`)	1, $\frac{N_S}{N_C \cdot n_k}$

Table 3: Parameters space for the SVM classifier. The original name of each parameter in the Sci-Kit documentation is in typewriter font within brackets.

2C	A	P	R	F_1
DT	0.9292	0.9265	**0.9303**	0.9281
RF	0.9292	0.9278	0.9278	0.9278
SVM	**0.9336**	**0.9355**	0.9291	**0.9318**

4C	A	P	R	F_1
DT	0.7189	0.6908	0.6888	0.6885
RF	0.7495	0.7136	0.7186	0.7130
SVM	**0.7725**	**0.7400**	**0.7407**	**0.7398**

Table 4: Accuracy A, precision P, recall R and F_1-score for Decision Tree (D.T.), Random Forest (R.F.) and Support Vector Machine (SVM) on 2C (upper table) and 4C (lower table). Such measures are first computed for each class, then their unweighted mean is computed.

5.3 Results

For each dataset we tested all three classifiers, and we report the results, in terms of accuracy, precision, recall and F_1-score, in Table 4 for datasets 2C and 4C.

For the 2C dataset the differences between Decision Trees and Random Forests are irrelevant, indeed the precision slightly increases, the recall decreases comparably, and the F_1-score consequently remains pretty much the same. SVMs

Actual \ Predicted	B1	B2	C1	C2
B1	174	20	0	0
B2	13	95	20	1
C1	0	29	45	29
C2	0	3	16	78

Table 5: Confusion Matrix for Random Forests on 4C.

achieve better results, although the performances are satisfactory for all classifiers.

For the 4C dataset the situation is different: there is an improvement switching from Decision Trees to Random Forests, as is expected, and also from these ones to SVMs, that outperform the other models by 2.3% in terms of accuracy. However, as can be expected, the performances obtained on the 2C dataset are generally better than those obtained on 4C.

As previously reported, our models do not take into account the levels' ordering. Hence, we analyze how much this aspect influenced the performances. In particular, we analyze the results obtained by the RF model on the 4C dataset. With this aim, Table 5 shows the confusion matrix (for RF on 4C) with the levels ordered according to their natural ordering. From these data, it is evident that only four misclassified texts are two classes away from the actual class. Therefore, the classifier model does not seem to be very sensitive to the levels' ordering.

5.4 Robustness Analysis

Here we analyze the robustness of the RF model by considering the differences among the five settings of hyper-parameters obtained by the calibrations performed in the outer-loop of the nested cross-validation.

Most notably, every setting has the same assignment for the hyper-parameters `criterion`, `min_impurity_decrease` and `max_leaf_nodes`. Moreover, the hyper-parameters `class_weight`, `min_samples_leaf`, `max_features` and `min_samples_split` have been assigned to the same value in three settings out of five. The only unstable parameter is the normalization method which assumes all the possible values.

These results, despite some minor differences, shows the robustness of the proposed approach.

5.5 Analysis of Linguistic Features

The choice of Random Forests classifiers come in handy when we want to analyze how the classifier works internally, how the linguistic features are used and how they contribute to the final result.

The importance of a feature used by the Random Forests classifier can be quantified by means of the loss of Gini impurity due to each node where the splitting is performed according to that feature.

As already pointed out, a single run of nested cross-validation for Random Forests provides 5 different sets of parameters for each dataset; we analyzed them all separately, however, due to space constraints, in Figure 2, we only show the features' importance for one of them. The importance of linguistic features remains quite steady across the different 5 folds of the nested cross-validation both for 2C and 4C, thus proving the robustness of our architecture, and providing a sound assessment of the contribution of a feature to the final outcome.

6 Conclusion and Future Work

In this work we introduced an automatic classification system for assessing the proficiency level of an Italian text used for second language learning purposes.

A dataset of texts labeled by expert test developers was used to evaluate the performance of three classifier models (decision tree, random forest, and support vector machine), which were trained using linguistic features measured quantitatively and extracted from the texts.

Experiments were held in order to analyze the effectiveness and robustness of the proposed prototypical classification system, and to gain useful insight about which features contribute the more to discriminate the texts from the point of view of CEFR levels.

Overall, considering the preliminary nature of the work, the classification accuracy we obtained is satisfactory. Moreover, we derived interesting indications about the contribution of the different linguistic features we considered.

This work can be extended along several future research avenues: integrating more linguistic features, considering the natural ordering among the proficiency levels, including more classification models, and artificially augmenting the dataset of texts.

References

L. Bachman and A.S. Palmer. 2010. *Language Assessment in Practice*. Oxford University Press.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

António Branco, João Rodrigues, Francisco Costa, João Silva, and Rui Vaz. 2014. Rolling out text categorization for language learning assessment supported by language technology. *Computational Processing of the Portuguese Language*, pages 256–261.

T. De Mauro and I. Chiari. forthcoming. *Il Nuovo Vocabolario di Base della Lingua Italiana*.

Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2011. Read–it: Assessing readability of italian texts with a view to text simplification. In *Proceedings of the Second Workshop on Speech and Language Processing for Assistive Technologies*, pages 73–83, Edinburgh, Scotland, UK. Association for Computational Linguistics.

V. Franchina and R. Vacca. 1986. Adaptation of flesh readability index on a bilingual text written by the same author both in italian and english languages. *Linguaggi*, 3:47–49.

Thomas François and Cedrik Fairon. 2012. An "ai readability" formula for french as a foreign language. *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 466–477.

A.C. Graesser, D.S. McNamara, M.M. Louwerse, and Z. Cai. 2004. Coh-metrix: Analysis of text on cohesion and language. *Behavior Research Methods, Instruments, And Computers*, 36:193–202.

P.J. Kincaid and Fishburne R. Lieutenant Robert, P. 1975. Derivation of new readability formulas for navy enlisted personnel. *Research Branch Report, Millington, TN: Chiefof Naval Training*, pages 8–75.

P. Lucisano and M.E. Piemontese. 1988. Gulpease: una formula per la predizione della difficoltà dei testi in lingua italiana. *Scuola e città*, 31(3):110–124.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

A. Palmero Aprosio and G. Moretti. 2016. Italy goes to Stanford: a collection of CoreNLP modules for Italian. *ArXiv e-prints*.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Ildiko Pilán and Elena Volodina. 2018. Predicting proficiency levels in learner writings by transferring a linguistic complexity model from expert-written coursebooks. *Proceedings of the Workshop on Linguistic Complexity and Natural Language Processing*, pages 49–58.

Ildikó Pilán, Sowmya Vajjala, and Elena Volodina. 2016. A readable read: Automatic assessment of language learning materials based on linguistic complexity. *International Journal of Computational Linguistics and Applications, 7 (1)*, pages 143–159.

J.E. Purpura. 2014. Cognition and language assessment. In *The Companion to Language Assessment volume III*, pages 1,453–1,476.

Valentino Santucci, Stefania Spina, Alfredo Milani, Giulio Biondi, and Gabriele Di Bari. 2018. Detecting hate speech for italian language in social media. In *EVALITA 2018, co-located with the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018)*, volume 2263.

Shai Shalev-Shwartz and Shai Ben-David. 2014. *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Sowmya Vajjala and Detmar Meurers. 2016. Readability-based sentence ranking for evaluating text simplification. *Int. Jour. of Applied Linguistics*, pages 194–222.

Sudhir Varma and Richard Simon. 2006. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, 7(1):91.

Eric Velleman and Thea Van der Geest. 2014. Online test tool to determine the cefr reading comprehension level of text. *Procedia Computer Science*, pages 350–358.

Menglin Xia, Ekaterina Kochmar, and Ted Briscoe. 2011. Text readability assessment for second language learners. *Proc. of the 11th Works. on Innov. Use of NLP for Building Educ. Appl.*, pages 12–22.

C. Xiaobin and D. Meurers. 2016. Ctap: A web-based tool supporting automatic complexity analysis. *Proc. of the Workshop on Computational Linguistics for Linguistic Complexity*, pages 113–119.

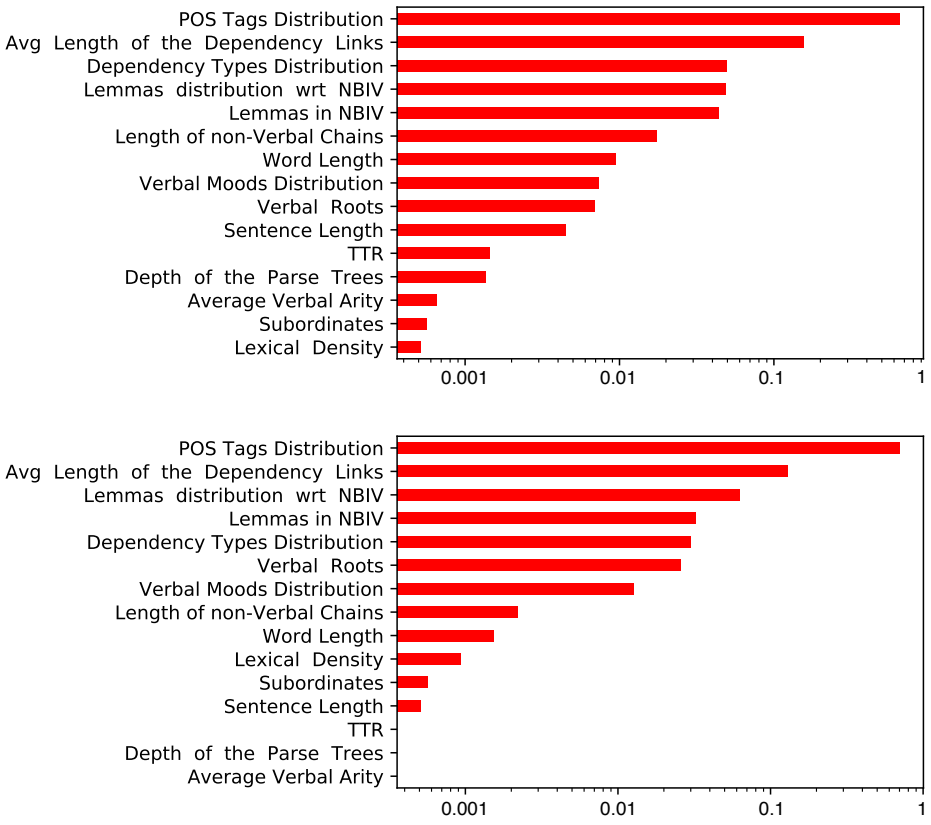

Figure 2: Relative importance of the features used by the Random Forest classifier trained on 2C (top plot) and 4C (bottom plot), in logarithmic scale.