

bot.zen at LangLearn: regressing towards interpretability

Egon W. Stemle^{1,2,*}, Martina Tebaldini^{3,1}, Francesca Bonanni^{3,1}, Filippo Pellegrino^{3,1}, Paolo Brasolin¹, Greta H. Franzini¹, Jennifer-Carmen Frey¹, Olga Lopopolo^{1,4} and Stefania Spina^{1,4}

¹Institute for Applied Linguistics, Eurac Research, Viale Druso, 1, 39100 Bolzano (BZ), Italy

²Faculty of Informatics, Masaryk University, Czech Republic

³University of Bolzano, Italy

⁴Università per Stranieri di Perugia, Italy

Abstract

This article describes the bot.zen system that participated in the Language Learning Development (LangLearn) shared task of the EVALITA 2023 campaign. We developed a simple machine learning system with good interpretability for later use, and used the shared task as an opportunity to provide Master's students with hands-on training and practical experience in NLP.

Keywords

system description, langlearn, evalita, shared task, regression, MALT-IT2, bot.zen

1. Introduction

There has been an increasing interest in using Natural Language Processing (NLP) tools and machine learning techniques to analyse writing development in first (L1) and second language (L2) acquisition settings. The topic has been explored in Second Language Acquisition (SLA), Learner Corpus Research (LCR) (e.g. [1]), Corpus Linguistics and in writing development research (e.g. [2]), and its goal is to understand how specific features can reflect writing quality and development.

The analysis of language learner data typically spans sociolinguistic metadata (information about the author), linguistic data (information extracted from the text) and textual metadata (information about the text). According to [3], metadata such as reading time, geographic factors, and parents' occupation level can have an impact on language skill development, whereas [4] finds writing quality and development to be influenced by both text length and linguistic features including lexical density,

diversity and sophistication, as well as syntactic complexity and text cohesion. Finally, a text usually includes metadata such as the author, the date of creation, the context in which it was written, and a language proficiency rating. This contextual information enhances the overall understanding of the content. All of these research strands can support NLP applications for writing evaluation and assessment, including automatic essay scoring, automatic writing evaluation systems, and automatic classification of text difficulty for learners. (For an in-depth overview and additional references, see [4].)

At EVALITA 2023 [5], the Language Learning Development (LangLearn) shared task (ST) on automatic language development assessment [6] consisted in predicting the relative order of two essays written by the same student. More specifically, the texts provided were in Italian and Spanish, and came with only a very limited set of metadata. We participated in this ST to acquire experience with this type of data, and as an opportunity to involve and train Master's students from the University of Bolzano¹ in NLP scientific work through practical internships.

Our system relies only on the data provided for the ST, generates explicit information about students' progress out of implicit information in the data and uses regression without Large Language Models (LLMs) or Neural Networks (NNs) with features from an external tool specifically designed for Italian texts. As a result, our system performed well on Italian but poorly on Spanish data.

The rest of the paper is organised as follows: Section 2 describes the system design and implementation; Section 3 describes our experiments and results; and Section 4 concludes with a short discussion.

EVALITA 2023: 8th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Sep 7 – 8, Parma, IT

*Corresponding author.

✉ egon.stemle@eurac.edu (E. W. Stemle);
martina.tebaldini@student.unibz.it (M. Tebaldini);
francesca.bonanni@student.unibz.it (F. Bonanni);
filippo.pellegrino@student.unibz.it (F. Pellegrino);
paolo.brasolin@eurac.edu (P. Brasolin); greta.franzini@eurac.edu
(G. H. Franzini); jennifercarmen.frey@eurac.edu (J. Frey);
olga.lopopolo@eurac.edu (O. Lopopolo); stefania.spina@eurac.edu
(S. Spina)

🌐 <https://iiegn.eu> (E. W. Stemle)


📄 0000-0002-7655-5526 (E. W. Stemle); 0000-0003-2471-7797

(P. Brasolin); 0000-0003-1159-5575 (G. H. Franzini);

0000-0002-7008-6394 (J. Frey); 0000-0003-0997-367X (O. Lopopolo);

0000-0002-9957-3903 (S. Spina)

© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

¹<https://www.unibz.it/en/faculties/education/master-applied-linguistics/>

2. System Design and Implementation

Our objective was to develop a simple machine learning system with good interpretability. Therefore, we prioritised a simple design that could provide transparent explanations for its decision-making process over complex implementation and high predictive performance.

2.1. Data Pre-processing

In a first processing step, we restructure the given ST data, which provides essay ids with their respective time of writing in tabular format, as shown in Figure 1.

essay_1	essay_2	order_1	order_2
5074	4666	1_1	1_2
5074	4948	1_1	1_4
5074	4872	1_1	1_5
5074	4937	1_1	2_1
5074	5128	1_1	2_2
5074	4212	1_1	2_3
5074	5361	1_1	2_4

Figure 1: Excerpt for one author from the 4-column Training_CITa.tsv file with the first two columns (essay_1, 2) representing essay ids and the last two columns (order_1, 2) their respective time of writing (YEAR_Essay-Number).

We reconstruct the data for individual authors by collecting all pairs with overlapping essays and reordering them based on the provided information, as shown in Figure 2. This new data structure contains an *absolute position* for the time of writing, which eventually becomes the target variable to learn.

With this remodelled dataset, we transform a classification task into a regression task: rather than deciding whether one text was written earlier than the other in a binary fashion, we predict the absolute position of the texts which, in turn, can be used to calculate the order in which the two essays were written.

2.2. Feature Extraction

We use spaCy [7] and MALT-IT2 [8] in order to transform the raw input data into a meaningful set of informative features, as they provide easy-to-use and reliable feature extraction methods.

2.2.1. spaCy

spaCy is an open-source NLP library in Python providing tools for many tasks and pre-trained models for several

order:essay	abs.position
1_1:5074	1
1_2:4666	2
1_4:4948	4
1_5:4872	5
2_1:4937	7
2_2:5128	8
2_3:4212	9
2_4:5361	10

Figure 2: Excerpt from our restructured Training_CITa.tsv file representing all essays written by one author with their time of writing and inferred abs(olute) positions. 2_4:5361 means that essay:5361 was the fourth essay to be written in YEAR:2 by this author, which corresponds to the overall tenth position for this author relative to all other authors (because the author missed order positions 1_3 and 1_6 in the first year).

languages, including Italian and Spanish².

After tokenisation, we collect 1- to 3-grams of the word forms and the part-of-speech tags. We additionally collect 2-grams of the morphological analyses of the words and 1-grams of a word’s dependency relation. Overall, this amounts to roughly 17,000 features per document.

2.2.2. MALT-IT2

MALT-IT2 is an automatic classification system for measuring the complexity level of Italian texts, and it is based on experiments that compared ten machine learning models on a dataset of 692 texts and 139 linguistic features. The features were divided into the following six broad categories [9]:

1. Raw Text Features are the most elementary features and they are based on simple counting procedures executed on the tokenised text.
2. Lexical Features are computed by considering: (i) lemmas, POS tags and morphological annotations; (ii) external resources for the Italian language, for example, the vocabulary rate in the “new basic Italian vocabulary”.
3. Morphological Features consider the morphological complexity index (MCI) computed for two word classes: verbs and nouns.
4. Morpho-Syntactic Features are computed on the basis of POS tagging, morphological analysis, and syntactic parsing.
5. Syntactic Features reflect the main characteristics and the structure of the syntactic constituents and the dependency relations of the sentences in a text.

²We use the it_core_news_lg and es_core_news_lg models.

- Discursive Features take into account the cohesive structure of a text.

MALT-IT2 has to be invoked externally to process text files into a comma-separated values (CSV) file containing one line per document within its feature space; the CSV file is subsequently ingested by our system without any additional interaction or knowledge of MALT-IT2. This means that we can swap out MALT-IT2 with a different system or add another system capable of producing a document-feature-matrix in CSV format.

2.2.3. CTAP

We also experimented with a version of the Common Text Analysis Platform (CTAP) [10] adapted for Italian text [11]. Much like MALT-IT2, CTAP is a linguistic complexity measurement tool offering various statistics and features to analyse text complexity in terms of length, lexical, syntactic and morpho-syntactic aspects. Unfortunately, we encountered some problems while processing the entire dataset. Very short texts, for instance, caused CTAP to end prematurely with no error message, leaving us with no choice but to exclude CTAP features from our system. CTAP is capable of producing a document-feature-matrix in CSV format and could have been easily integrated into our system.

2.3. Processing Pipeline

Our data processing pipeline has been implemented in Python and makes use of the pandas and scikit-learn libraries³.

pandas [12] is an open-source library for data manipulation and analysis that integrates well with other libraries in the Python ecosystem, making it a versatile tool for data analysis and preparation.

Our system uses pandas for internal data representations, manipulations and calculations during data pre-processing (Section 2.1) and the processing of CSV files.

scikit-learn [13] is an open-source machine learning library for Python providing a wide range of algorithms and tools for various tasks, including classification and dimensionality reduction. With a user-friendly and consistent interface, extensive documentation and an established user base, scikit-learn makes it easy to implement machine learning workflows.

Our system uses scikit-learn for the main processing, as illustrated in Figure 3.

The processing pipeline requires a document feature matrix that represents all texts as vectors in our feature

```
pipeline = Pipeline(steps=[
    ('combined_features',
     combined_features),
    ('scaler', StandardScaler()),
    ('redux', TruncatedSVD(125)),
    ('estimator', HistGradientRegressor(
        loss='squared_error'))
])
```

Figure 3: The core of the processing pipeline. From top to bottom, the combined features are scaled, reduced and processed with only minimal parameterisation.

space. This space is the combination (concatenation) of all different tools after feature extraction (Section 2.2), totalling around 17,200 features. To standardise the data, we use the `StandardScaler()`, which removes the mean and scales it to unit variance. We also reduce the linear dimensions using the `TruncatedSVD()` method⁴. As a result, our processed dataset consists of 125 features. Finally, we use the `HistGradientRegressor()` for learning, which is an ensemble method⁵.

In order to use our system for the ST, we perform data post-processing. We convert the output of our regression model for individual texts into a binary label for pairs of texts that indicated which of the two was written first.

2.4. Optimisation

The different parts of our system were optimised towards our target variable (absolute position) via an ad-hoc grid search in 3-fold cross validation (CV) runs.

The parts we optimised were: the types of spaCy information to collect⁶; n-gram ranges and the minimum document frequencies for the spaCy collectors; the type of dimensionality reduction⁷ and the number of dimensions to use; the regression algorithm to use⁸.

3. Experiments and Results

3.1. Shared Task (ST)

The Language Learning Development (LangLearn) ST [6] consisted in predicting the relative order of two essays: given a randomly ordered pair (Essay 1, Essay 2) written

⁴We perform feature reduction to remove noise or irrelevant information, and highlight important aspects of the data, enabling the model to make more accurate predictions.

⁵Ensemble methods combine and aggregate predictions of multiple models to improve predictive performance.

⁶`token.text`, `token.lemma_`, `token.pos_`, `token.morph`, `token.dep_`

⁷`PCA()`, `TruncatedSVD()`

⁸ `DecisionTreeRegressor()`, `SVR()`, `KernelRidge()`, `HistGradientRegressor()`

³We used Python 3.8.16, pandas 2.0.1, scikit-learn 1.2.2, and spacy 3.5.3 + it-core-news-lg 3.5.0 for processing.

by the same student, the task was to predict whether Essay 1 had been written before Essay 2.

3.2. Shared Task Data

The LangLearn ST data contains essays from two different corpora, namely CItA [14] and COWS-L2H [15], with texts in Italian and Spanish, respectively.

Training data includes information on pairs of texts written by the same student at different times. Each entry represents the sequence of two essays, and by considering multiple entries with overlapping text-ids we are able to recreate the sequence of all texts for each student (see Section 2.1). The data also contains the texts themselves but no additional (meta)information beyond this.

CItA The CItA corpus (Corpus Italiano di Apprendenti L1) is a collection of Italian essays written by students learning their first language in seven different lower secondary schools in Rome over the course of two years (2012-2013 and 2013-2014). The students were asked to write different types of essays, namely reflexive, narrative, descriptive, expository and argumentative. The ST data contains 834 of the total 1,352 essays written but does not provide any information about the type of text.

COWS-L2H The COWS-L2H corpus (Corpus of Written Spanish of L2 and Heritage Speakers) is a collection of texts created by students of Spanish as a second language enrolled at a North American university. The students were asked to write multiple compositions at different times throughout the academic quarters, and the essays were collected over the course of two years, from 2017 to 2020. The essays were written by the same students, and the ST data contains 1,426 of the original 3,498 essays.

3.3. Results

The performance of our system on the two datasets (as reported by the ST organisers) was:

		acc	f-score
CItA	bot.zen	0.83	0.84
	best	0.93	0.93
	baseline	0.55	0.55
		acc	f-score
COWS-L2H	bot.zen	0.50	0.52
	best	0.75	0.75
	baseline	0.66	0.66

The baseline scores were calculated by training a LinearSVM using the number of tokens per document and the Type-Token-Ratio (TTR) of the first 100 tokens in each document as input features.

3.4. Analysis

We also analysed the CItA-part of the ST dataset independent of our system’s performance. To this end, we used the original data with texts in *Set 1* always written before texts in *Set 2*. We then used CTAP to calculate feature values for all texts in both sets. Afterwards, we conducted a (paired) t-test to detect features that differed in their means (as a starting point for later research).

We found some evidence that *Set 1* had a higher number of ‘basic vocabulary’ words, whereas *Set 2* had a higher number of imageability words. *Set 1* also had higher TTR and HDD (Hypergeometric Distribution D) measures, but since *Set 2* generally had longer texts, length effects certainly come into play [16]. Also, *Set 1* used more auxiliary verbs, possibly due to a higher presence of past participle verbs. The use of connectives was higher in *Set 2*, especially for additive and consequence connectives. The number of dependent clauses per sentence did not differ significantly between the sets. Finally, *Set 2* contained more sentences and more punctuation marks but sentence length remained constant.

4. Discussion

Our system (see Section 2) was relatively simple. Neither LLMs nor recurrent neural networks (RNNs) were integrated, nor did we use any data other than those provided by the organisers. While our results for the Italian data were satisfactory, we performed very poorly on the Spanish data, as expected: MALT-IT2, our main processing component, was designed for Italian texts only, which had a negative impact on our system when processing Spanish data, and despite the baseline system information also being encoded in our features, the presence of too much irrelevant data hampered the overall performance.

Nevertheless, the ST served as a great opportunity for Master’s students to gain practical project work experience: running into all-too-common data processing, encoding and decoding difficulties whilst navigating the intricacies of understanding, analysing and evaluating the data for the task at hand. With the help of the literature suggestions provided by the organisers, the students were able to develop relevant ideas and provide target-oriented answers to emerging questions. Although the internship was only 150 hours long and did not include the implementation of a functional application⁹, the students had the opportunity to familiarise themselves with the crucial stages of a scientific project, documenting all steps into a project report, which was partially incorporated in this paper.

⁹Eurac Research took over the task of implementing a functional application.

Acknowledgments

We would like to thank our colleagues Arianna Bienati, Francesco Fernicola and Lionel Nicolas for their support during the project.

References

- [1] S. A. Crossley, D. S. McNamara, Does writing development equal writing quality? A computational investigation of syntactic complexity in L2 learners, *Journal of Second Language Writing* 26 (2014) 66–79. doi:10.1016/j.jslw.2014.09.006.
- [2] P. Durrant, M. Brenchley, L. McCallum, *Understanding Development and Proficiency in Writing: Quantitative Corpus Linguistic Approaches*, 1st ed., Cambridge University Press, 2021. doi:10.1017/9781108770101.
- [3] A. Barbagli, F. Dell’Orletta, G. Venturi, P. Lucisano, S. Montemagni, Il ruolo delle tecnologie del linguaggio nel monitoraggio dell’evoluzione delle abilità di scrittura: primi risultati (2015) 105–123. URL: <https://journals.openedition.org/ijcol/326>.
- [4] S. A. Crossley, Linguistic features in writing quality and development: An overview, *Journal of Writing Research* 11 (2020) 415–443. doi:10.17239/jowr-2020.11.03.01.
- [5] M. Lai, S. Menini, M. Polignano, V. Russo, R. Sprugnoli, G. Venturi, EVALITA 2023: Overview of the 8th evaluation campaign of natural language processing and speech tools for Italian, in: *Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023)*, CEUR.org, Parma, Italy, 2023.
- [6] C. Alzetta, D. Brunato, F. Dell’Orletta, A. Miaschi, K. Sagae, C. H. Sánchez-Gutiérrez, G. Venturi, LangLearn at EVALITA 2023: Overview of the language learning development task, in: *Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023)*, CEUR.org, Parma, Italy, 2023.
- [7] M. Honnibal, I. Montani, spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing, 2017. URL: <https://spacy.io/>.
- [8] L. Forti, G. Grego Bolli, F. Santarelli, V. Santucci, S. Spina, MALT-IT2: A new resource to measure text difficulty in light of CEFR levels for Italian L2 learning, in: *Proceedings of the Twelfth Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2020*, pp. 7204–7211. URL: <https://aclanthology.org/2020.lrec-1.890>.
- [9] V. Santucci, F. Santarelli, L. Forti, S. Spina, Automatic classification of text complexity, *Applied Sciences* 10 (2020) 7285. doi:10.3390/app10207285.
- [10] X. Chen, D. Meurers, CTAP: A web-based tool supporting automatic complexity analysis, in: *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity (CL4LC), The COLING 2016 Organizing Committee, Osaka, Japan, 2016*, pp. 113–119. URL: <https://aclanthology.org/W16-4113>.
- [11] N. Okinina, J.-C. Frey, Z. Weiss, CTAP for Italian: Integrating components for the analysis of Italian into a multilingual linguistic complexity analysis tool, in: *Proceedings of the Twelfth Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2020*, pp. 7123–7131. URL: <https://aclanthology.org/2020.lrec-1.880>.
- [12] The pandas development team, pandas-dev/pandas: Pandas 2.0.1, Zenodo, 2020. URL: <https://pandas.pydata.org/>. doi:10.5281/zenodo.3509134.
- [13] 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, E. Duchesnay, Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research* 12 (2011) 2825–2830. URL: <https://scikit-learn.org/>.
- [14] A. Barbagli, P. Lucisano, F. Dell’Orletta, S. Montemagni, G. Venturi, CltA: an L1 Italian learners corpus to study the development of writing competence, in: *Proceedings of the Tenth International Conference on Language Resources and Evaluation, European Language Resources Association, Portorož, Slovenia, 2016*, pp. 88–95. URL: <https://aclanthology.org/L16-1014>.
- [15] A. Yamada, S. Davidson, P. Fernández-Mira, A. Carando, K. Sagae, C. Sánchez-Gutiérrez, COWS-L2H: A corpus of Spanish learner writing, *Research in Corpus Linguistics* 8 (2020) 17–32. doi:10.32714/ricl.08.01.02.
- [16] M. Stills, *Language Sample Length Effects on Various Lexical Diversity Measures: An Analysis of Spanish Language Samples from Children*, Technical Report, Portland State University, 2016. doi:10.15760/honors.250.

A. Online Resources

- The bot.zen system for the EVALITA 2023 LangLearn shared task (on GitHub)