# Keystroke logging in second language writing

## Exploring the writing process through real-time production metrics

## Introduction and motivation

Extracting information from written text is a challenging task. In the past, research was based on the final output, losing all the information produced during the writing process itself. A form of data that is now more widely available is the *keystroke log* associated with the text that records the exact timestamps of every key press.

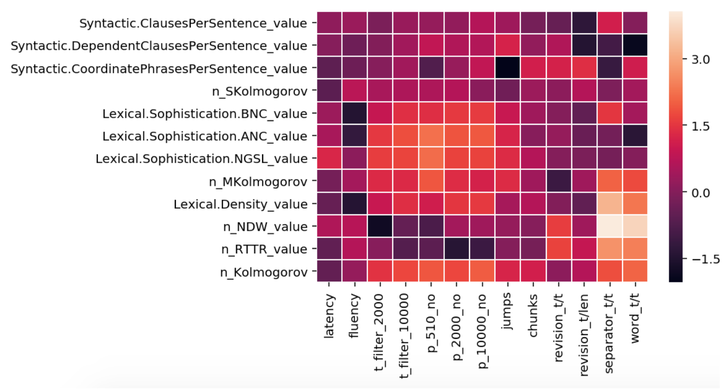
For this project, we built tools to analyze keystroke logs for large corpora. We then explore the texts produced during English classes at RWTH Aachen. To assess the quality of the final product, we compute sentence complexity metrics using a tool developed by the Linguistics department, *CoCoGen*. By exploring the relationships between keystroke and linguistical metrics, we obtain novel insights about the writing process and the cognitive process that lie behind it, which in turn can be valuable to linguists, cognitive psychologists, etc.


```
{"text":"\n","attribs":"|1+1"}}},{"changeset":"Z:1>1*0+1$L","meta":
{"author":"a.SOsOoMqkqdc2tepz","timestamp":1524218645504}},{"changeset":"Z:
2>2=1*0+2$ea","meta":{"author":"a.SOsOoMqkqdc2tepz","timestamp":1524218646014}},
{"changeset":"Z:4>4=3*0+4$rnin","meta":{"author":"a.SOsOoMqkqdc2tepz","timestamp":
1524218646529}},{"changeset":"Z:8>2=7*0+2$g ","meta":
{"author":"a.SOsOoMqkqdc2tepz","timestamp":1524218647043}},
{"changeset":"Z:a>1=9*0+1$J","meta":{"author":"a.SOsOoMqkqdc2tepz","timestamp":
1524218647545}},{"changeset":"Z:b>3=a*0+3$our","meta":
```


Sample of keystroke log in the initial compressed JSON format

## Linguistic complexity

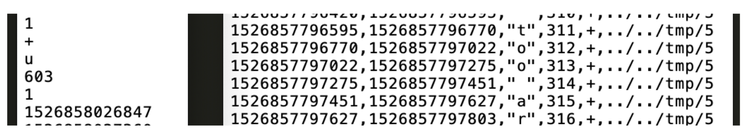
We found that our metrics are good at predicting the linguistic complexity of the final sentence.




Correlation between keystroke/linguistic metrics. Brighter = better predictor

## Preprocessing and data cleaning

First, we build a node.js script that parses the JSON, throws away unnecessary information, and outputs the changesets in separate lines in a new file.




Then, we removed contaminated data. Our solution consisted of several filters and conditions that sentences had to satisfy in order not to be marked as copy-pasted and removed. These included being written with uneven spacings, or occasional deletions, along with other subtler criteria.

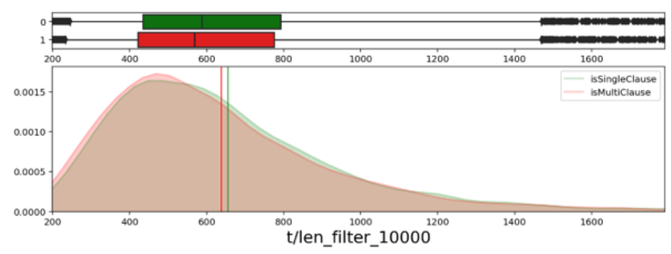
### Running CoCoGen

Our last step was obtaining linguistical metrics about the final output, sentence by sentence. To split our files, we used the Java implementation of the Stanford-CoreNLP library. We requested access to CoCoGen, a tool developed by the Linguistics department here at RWTH Aachen We ran it on all our samples to obtain more than 30 different complexity scores for each sentence.

| | |
|---|---|
| Syntactic.ClausesPerSentence_value | No. of clauses per sentence |
| Lexical.Sophistication.BNC_value | Average frequency of words in the British National Corpus |
| Lexical.Diversity.TTR_value | Type-Token ratio |
| KolmogorovDeflate_value | Information theoretical metric: how compressible is the text |

### Number of clauses

We find that for a given user and length, single-clause sentences take longer to write (p = 2 x 10-9). This might be counterintuitive at first glance, but it is often more challenging to write a long, continuous sentence rather than breaking it up in multiple chunks.

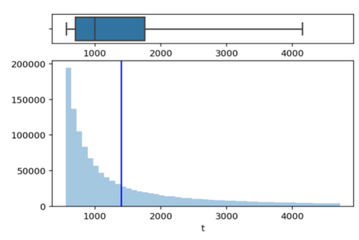



```
Sample size:  13941    Mean diff:  16.3281
stds:  0.0496   p-value:  2.32270E-09
```

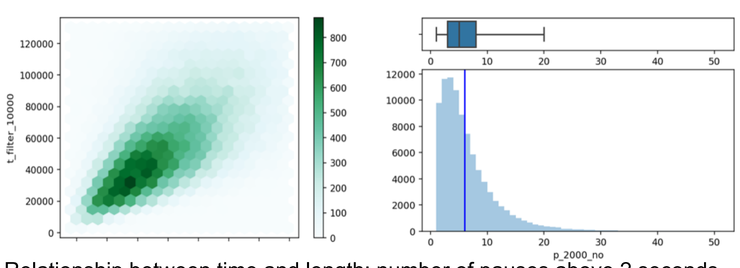
## Metrics from keystroke data

Our final dataset consists of 14,966,923 keystrokes, forming 1,831,361 words and 95,354 sentences over 3,466 text files. Each file was ~530 words in length and was written at around ~20 words per min.

| char | freq | speed | deleted |
|---|---|---|---|
| | 14.41 % | 424.3 | 16.493 % |
| e | 9.63 % | 255.6 | 14.730 % |
| t | 6.67 % | 386.2 | 14.232 % |
| a | 6.01 % | 361.3 | 14.818 % |
| i | 5.27 % | 352.1 | 14.840 % |
| s | 5.11 % | 342.3 | 15.492 % |
| n | 5.08 % | 265.7 | 14.107 % |
| o | 4.81 % | 351.0 | 14.772 % |
| r | 4.60 % | 282.1 | 15.000 % |
| h | 3.31 % | 294.0 | 15.650 % |
| l | 2.82 % | 315.2 | 13.924 % |
| c | 2.60 % | 405.7 | 14.152 % |



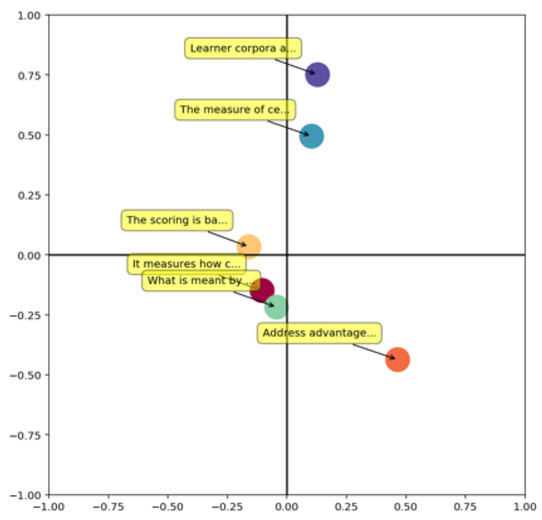

List of the most common keystrokes; histogram of pause length distribution




Relationship between time and length; number of pauses above 2 seconds

### Fluency and latency

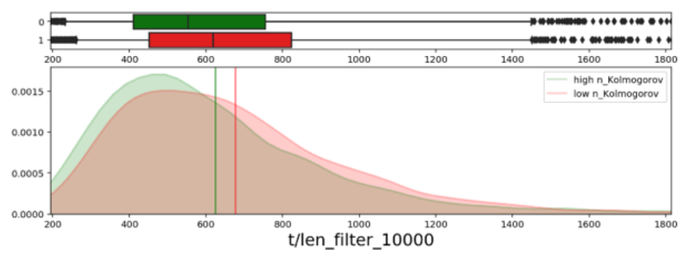
We defined two new metrics to capture the spectrum of pause length distributions. Together, they describe the process through which a sentence was written.



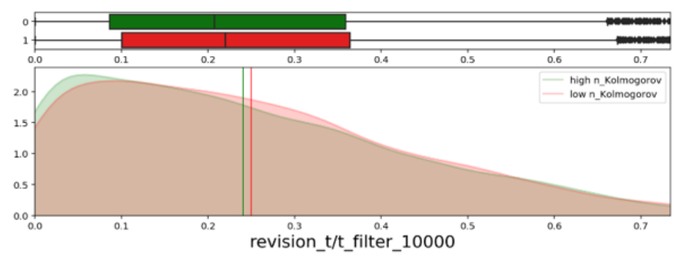

Fluency-Latency in the x-y plane for selected groups of sentences

### Kolmogorov complexity

Numerous computer-generated metrics assess complexity without human knowledge about language. Kolmogorov Deflate roughly measures by what factor the given text can be compressed through standard algorithms. We found that this metric is correlated with most of our keystroke-derived measures.




```
Sample size:  3711    Mean diff:  52.1265
stds:  0.1618   p-value:  3.24403E-23
```




```
Sample size:  3711    Mean diff:  0.0090
stds:  0.0486   p-value:  0.001549
```