preprint version, cite as:

Abdollahinami, S., Ducceschi, L., & Zancanaro, M. (2022). End-user Development and Closed-Reading: An Initial Investigation. In *Proceedings of the 2022 International Conference on Advanced Visual Interfaces AVI'2022*, Frascati, June https://doi.org/10.1145/3531073.3531128

End-user Development and Closed-Reading: an Initial Investigation

Sevda Abdollahinami
sevda.abdollahinami@studenti.unitn.it
University of Trento
Rovereto (TN), Italy

Luca Ducceschi
luca.ducceschi@unitn.it
University of Trento
Rovereto (TN), Italy

Massimo Zancanaro
massimo.zancanaro@unitn.it
University of Trento
Rovereto (TN), Italy

ABSTRACT

In this work, we explore the idea of designing a tool to augment the practice of closed-reading a literary text by employing end-user programming practices. The ultimate goal is to help young humanities students learn and appreciate computational thinking skills. The proposed approach is aligned with other methods of applying computer science techniques to explore literary texts (as in digital humanities) but with original goals and means. An initial design concept has been realised as a probe to prompt the discussion among humanities students and teachers. This short paper discusses the design ideas and the feedback from interviews and focus groups involving 25 participants (10 teachers in different humanities fields and 15 university students in humanities as prospective teachers and scholars).

1 INTRODUCTION

Computational thinking is a kind of analytical thinking [25] to solve problems effectively and efficiently with solutions that are reusable in different contexts [20]. It comprises a collection of mental strategies, such as recursive and procedural thinking, modeling, abstraction, decomposition, heuristic reasoning, and parallelism, which could be applied in several domains [25, 26]. Although it should imply *thinking like a computer scientist* [14], Computational Thinking does not necessarily equate to computer programming or coding [26]. Nevertheless, most of the approaches in literature use coding as a means to teach *computational thinking* [19, 21]. The main risk with this approach is to disengage the students who are less attracted to programming and, in some sense, less likely to acquire computational skills in their academic and professional life. In this paper, we propose an approach that aims at employing some of the key mechanisms of *computational thinking* but on one activity that is most likely to engage students more oriented to humanities than to technical and scientific fields, namely closed reading.

Closed reading [8] is an approach to text analysis regularly used in scholarly studies [24] and literacy education [3, 27]. It consists in examining a text thoroughly, methodically engaging in active reading to understand its layers of meanings and eventually reach deep comprehension. Computational approaches to humanities studies have often been employed to analyze large corpora of texts, the so-called *distant reading* [13], and, although there are interesting examples of digital tools to support closed reading too (as briefly summarized in the related work below), in our work, we take a different perspective: we indeed want to provide a useful tool to support close reading, but we primarily aim to leverage computational thinking in order to encourage students to foster these skills in a natural and engaging activity which might be perceived as different from (the usual way of doing) coding.

In our approach, we rely on *end-user development*: an approach to computation that aims at empowering users with respect to digital technologies by positing that users should be put in a position of control of their own devices by acquiring some programming skills [7]. Although it is different from other instances of end-user development, we believe that our approach can still be classified as such. It entails formulating a problem in a structured way, and it uses decomposition and abstraction (which are among the basic skills of computational thinking) to identify and visualize specific elements in a given text. This approach might be suitable for our purpose since from one side, *end-user development* requires the acquisition of *computational thinking skills*, but, at the same time, it foster the development of such skills without (or before) engaging in standard computer science classes.

2 RELATED WORKS

The use of computational resources in humanities is not a novel idea, and there is a whole field of study called *digital humanities* [23]. Although most of the research is on supporting scholars, there have been several proposals for using computational resources in teaching as well (for example, [2, 22]).

Most digital humanities research focuses on dealing in semi-automatic ways with large corpora, the so-called distant reading initially suggested by Moretti [13]. Nevertheless, several attempts

of supporting close reading have been proposed, although often based on automatic statistical summaries and visualizations [9]. An interesting approach of using an NLP pipeline to *close in* into a text is presented by Eve [6] from a scholarly (rather than an educational) perspective. Digital humanities' approaches usually require relative high competencies in programming when applied to scholarly research while it is based on searching and visualizing large corpora when proposed in educational settings.

In our project, we are aiming to provide a reading tool that allows contextual querying for analyzing the morpho-syntactical structure of a text but with an emphasis in *closing in* into the text by means of the possibility of highlighting or changing the font of fragments of the text rather than fostering a *distant reading* of it by means of statistics and external visualizations (although, of course, the two approaches might eventually be combined).

End-user development is proposed as a set of methods and tools to allow end users to create, modify or extend a software artifact without requiring coding skills [10]. As mentioned before, it is usually include a recognition of the need to empowering stance for end-users [7].

There are different approaches to realize *end-user development* tools, in particular, the most recent research is focusing on the so-called *action-triggering rules*, which are well suited for applications in the field of *internet-of-things* when digital tools should be programmed to react to external events [1, 17]. Of course, that approach might not be well suited to analyze a text but there are other paradigms that might be employed in *end-user development* beyond the event-driving paradigm: among others, contextual reasoning [12] and tabular methods [16]; also the formulas in spreadsheets might be regarded as end-user development since they allow complex computations without high coding skills [15].

In our project, we are experimenting with an approach closer to the tabular methods which consist in defining and embedding structural queries that implement a simpler version of regular expressions.

3 THE DESIGN CONCEPT AND THE PROBE

Our proposed tool is based on a standard NLP pipeline written in Python and uses a mix of language models, regular expression, and lexical resources. It uses the Stanford Stanza core [18], and it employs other specific lexical resources, specifically Morph-it [11] and Paisa [28]. The procedure follows the steps: (1) tokenization (sentence and word segmentation); (2) part of speech tagging (tokens are assigned their syntactical category); (3) multi-word expressions recognition and (4) lemmatization (e.g words like *went*, *geese* or *runs* are mapped to their lemmas *go*, *goose*, and *run* respectively).

The NLP pipeline is used to pre-annotate literary texts. The user will not see the linguistically annotated version, but the tool will use it to instantiate the queries.

The graphical user interface appears in the form of a standard e-book reader while offering an augmented functionality for text annotation (that is, highlighting parts of the text as well as changing color or size of fonts for specific segments) by specifying morpho-syntactical constraints.

The design concept consists in offering a set of modular *filters* to extract parts of a text and a set of *lenses* that can be used to visualize the filters' results directly into the text.

In our approach, a *filter* is a structural query that extracts portions of a text. Similar to the table of contents, which extract headings, sections and paragraphs, but based on linguistic properties rather than structural properties of the text. In our tool, basic filters are defined on the output of the NLP pipeline. They allow the detection of words with specific morpho-syntactic features (for example, plural adjectives), words in the text that are instances of a given lemma, sentences with a specific structure (for example longer than a set number of words or containing relative clauses) and so on. Filters can be combined to build more complex filters. For example, a combined filter may extract those sentences that have a relative clause with at least one adjective. Every element in the text is identified with a unique id, therefore the result of the application of a filter to a text is a list of identifiers.

A *lens* is a set of visual properties such as font type, font background or foreground color. Lens are applied to filters to visualize the results of applying a filter on a text. For example, all the sentences with a relative clause can be highlighted in yellow, and the adjectives in those sentences can be written in larger red font.

Examples of activities that the users can do are the following ones: highlighting in yellow all the occurrences of a given lemma, irrespectively of the specific forms in which it appears in the text (for example, if "lion" is selected as a lemma, the tool will highlights also "lions" and "lioness"); write in a red color all the sentences with a subjective verb; highlights in green all the nouns preceded by a pronoun; and so on. Furthermore, the filters can be combined or just applied together to be able to make apparent specific patterns in the text. It is worth noting that in our tool, the linguistic data is not extracted, counted, or visualized outside the text but the filters, through the lenses, are always located in the text, therefore realizing a *close in* into the text or what Even [6] calls "close-textual digital microscopy".

While *filters* may resemble the standard language-based queries of several *distant reading* tools (for which the results are tabulated or graphically represented), their combination with lenses allows to *come back* to the text and this represents a support for close reading. The iterative application of filters and lens on a given text foster recursive and heuristic reasoning further supporting *computational thinking practices*.

In summary, our design concept consists of encouraging the readers to define semi-structured annotations directly in a text by building visual queries in a language that does not require explicit programming (along the lines of end-user development) while encoding some of the principles of *computational thinking*. The purpose is twofold: (i) to propose a new tool to support the practice of *close reading* and (ii) to foster *computational thinking* to young students of humanities, notoriously less prone to scientific and technical challenges.

These design ideas have been preliminary implemented in a simple prototype meant to be used as a probe to foster the discussion among teachers of literature and humanities students (as prospective humanities scholars or teachers).

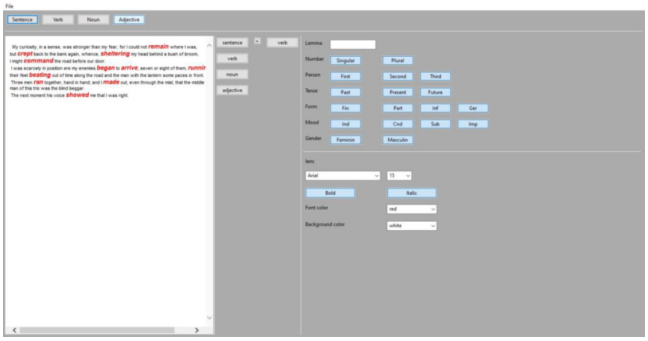

Figure 1: A screenshot of the probe prototype with the annotated text (on the left) and a language query (on the right).

4 AN INITIAL USERS RESEARCH

To have an initial assessment of the potentialities of the design concept, we decided to involve experienced educators in different fields of humanities as well as literature and language university students as future educators and scholars.

Ten (5 females) teachers or assistant university professors of literature or foreign language from Iran, Italy, the UK, and the US have been recruited for individual interviews. Furthermore, 15 (13 females) Italian university students of literature and other fields in humanities have been involved in 4 focus groups. The participants have been recruited with snowball sampling starting from personal contacts. The language used for interviews and focus groups was English, which all spoke fluently.

Both interviews and focus groups started with a general introduction of the participant(s), followed by one researcher demonstrating the probe as an educational tool without further specifications. The participants were then asked to compare it with other similar tools and discuss its perceived strengths or weaknesses. For the focus group, the researcher also encouraged interaction among the participants to increase the discussion and the confrontation.

The verbal reports have been analyzed following the approach of *reflexive thematic analysis* [4, 5]. An inductive orientation of coding has been used. That is, the analysis started the analytic process from the data without defining a-priori categories. The process of analysis followed the six steps of thematic analysis: (1) familiarization with the data, (2) analytical coding, (3) generation of initial themes, (4) review of themes, (5) definition and naming of themes and (6) summarizing the results and linking to the extant literature. Consistently with the tenets of Reflexive Thematic Analysis, we did not seek inter-annotator agreement or saturation but rather to share an understanding of the themes' definitions and their structure.

Four interrelated themes emerged from the analysis; 1) potential strengths of the tool, describing what they find interesting and intriguing in the tool; 2) teaching problems, regarding which problematic aspects of teaching the tool can potentially address; 3) user practice, describing the tasks, goals, and circumstances that users are now involved with while studying or teaching literature that might be affected by the tool; 4) tool problems, anticipating potential issues that might cause the tool not to be used. In the following, these themes are discussed using representative quotes from the participants.

Theme 1 potential strengths. For what concerns the potential strengths of the tool. All participants recognize that the tool might quite useful to support and assist during classroom activities.

The majority of students seem to focus mainly on the teaching of grammar aspects: "*So this tool can help us to show students how grammar is used in the language and learn the correct order of using words in the sentence by seeing many examples.*"; *"For example, I remember in high school, I studied Latin, German, Russian and English and we were very focused on recognizing which part of the phrase, was the verb, which was the subject, which was… and this tool can be very useful for people that starting to learn"*.

For several professors, grammar analysis was the most prominent aspect too: *"This tool could be beneficial in a class where we read a book with the students and then use the tool to analyze it grammatically."*; *"...or you have to pick the gender of the nouns or you have to study particular that tense of the verbs that sort of noun or that kind of stuff that are particularly related to grammar or you have to really analyze the text deeply or doing research about it.."*;

Because of the strong emphasis on grammar, the students mainly suggested usage in teaching foreign languages "*[…]something useful for learning a new language, or even for us, maybe for study Latin or Greek"* and "*it would be very useful actually especially if you're studying a language that maybe you don't know and you have to identify the different parts may be as you said, the different verbs, nouns,..."* .

As basically a tool for grammar analysis, it was regarded more as a basic tool *"High school students are certainly going to profit from it more than university students now that I think about it. Because the grammar analysis is something that university students should already be able to perform. They learned it during their secondary school period."*.

Some professors proposed the possibility to have students analyze their own or classmates' writings: *"[…] some students would use it just to kind of analyze their own writing to get an idea... of like when they construct a sentence or a paragraph, [...] I could see them analyzing their own writing with this. […] I could see some of them analyzing their classmates' writing with it, so they would possibly like when critiquing each other's writing…"* .

Nevertheless, several professors did also relate it to textual and critical analysis (which are similar to close reading): "*[…] I think I would use it in courses dedicated to novels."* and *"I would use this tool in my translation classes."*. Some professors argue the possibility of using for discussing stylistic variations in texts, including composition classes and poets.

In one case, close reading was explicitly mentioned: *"I could see them using it for, um, you know, close reading like you know they take a passage from a novel or a reading and put it in there and use it to try to understand what the author is up to..."* .

Often, it is conceived as a tool for the teacher *"[…] I could see using this in my lectures to pull stuff out in front of the class to actually like on the board like call out and say, okay, let's look at this sentence and look at this paragraph."*; *"I could totally see using something like this in a classroom situation... um, to really demonstrate the complexity of what the authors are up to, but also to demystify the writing um, I think a lot of times, we hold these writers like way up*

here somewhere, you know, kind of we put them on pedestals when in reality they just write differently for their time.".

But, in several cases, it is also proposed as a tool to foster students to to engage with the textual material autonomously: *"I think something like this could actually assist my students in a way that um, in, especially in my distance learning classes, web base classes where I'm not there for them a lot of times, and they just kind of have to work on their own; this is something that, if integrated, could be, could be something they could use on their own, which could be very helpful."*; *"...maybe to do their homework or to analyze texts. I mean, we do a lot of, in second year, we do a lot of analysis of newspaper articles, for instance.".*

As often happen for computational tools, the efficiency aspect is the most prominent one: "It is useful for searching quickly and in a structured way for the elements of the text." (P).

In a case, a professor suggested that the tool may be the base for a new study method: *"it would be an interesting new tool to use to maybe to add technology to our way of study or to have a new study method instead of like classic rules and classic text"* .

Theme 2 teaching problems: When asked about the problems in teaching literacy content that might be alleviated using a tool like the one proposed. Most of the participants mentioned keeping the class interactive, motivating students to engage in the discussions: *"[the one usually employed in teaching humanities] it is a very old, classic, monotonous method where the teacher writes something on the board and students repeat it and so on".*

Theme 3 teaching practices: In both the interviews and the focus groups, the participants were prompted to discuss how their teaching or study practice can be related to the tool's functionalities or that can affect its use. In many cases, students stated that they prefer to read and study on physical books over digital ones *"[...] people that are very fond of literature prefer the paper, I'm like that. I can't study on the computer in general."* and *"I usually read books on paper because I don't want to read them on a screen mostly because I get tired after a while but also because I get distracted a lot when I have access to the internet… "*

The use of digital technologies, for the students, is mainly related to foreign language computer-based exercises: *"I just use an application to study language"*; *"I use websites that are more for training for specific exams for example the C1 the Cambridge […] but it was more exercise rather than analyzing a text".*

Some professors were intrigued by the possibility of highlighting (the effect of applying *lenses*) parts of the text: *"I do know some of my colleagues do use PowerPoint, for instance, or they use tools like that. And then they highlight words that they want the students to know where to find in the text.".* One of the professor recognized the similarity with corpora-based distant reading tools. However, she appreciated the possibility of *closing in* the text of our tool through text highlighting: *"some corpora more or less do this already... But then you don't have the chance to make it graphically evident […]what these corpora don't do is to show it in such a graphically evident way. So you may inquire about verbs and get the verbs in the text, but it is not put... highlighted… emphasized… put to the font […]".*

Theme 4:tool problems: Nevertheless, some professors find the focus on morpho-syntactic aspects quite limiting: *"[...] Well, I don't categorize words in this way. So the only thing I could do with this would be to highlight a word that I want to think about with the students [...] But it would not be in terms of these categories.[…] So the categories for me are a problem because I don't categorize in this way. I want to get the students away from categories not to use categories myself, or to ask them to use categories [...]".*

Nevertheless, again, the greater problem seems to be the general attitude toward technology: *"...many teachers use… also speaking from my high school, they don't use technology as much as they should and so they use physical books … "* and in the words of a professor: *"[...] I do feel the need, but I'm not so technological. I'm a [humanities] teacher and therefore, I had no idea that something like this could be useful [...]".*

5 DISCUSSION AND CONCLUSION

In our design concept, closed reading is approached as a problem-solving process that entails formulating a problem and applying procedural thinking, decomposition, and abstraction (which are among the basic skills of computational thinking) to identify and visualize specific elements in a given text. Contrary to other approaches to applying digital humanities in educational contexts, in our tool the linguistic data is not extracted, counted, or visualized outside the text but instead a sophisticated form of text annotation is offered to encourage *close in* into the text.

In our initial formative evaluation, the proposed design ideas seem to be regarded as interesting by participants as actual and prospective teachers of literature. The main obstacles are the scarce attitude toward technology by the teachers and by students too. While the former might be addressed by specific training and compensated by a recognized need for more engaging ways of conducting lectures, the latter, particularly the dislike of digital books, should be carefully considered. The possibility of actively engaging with the text is generally appreciated, and in several cases, it has been positively recognized the difference between better-known approaches based on corpora. The narrow focus on morpho-syntactical properties might be a limitation for implementing a full support to close reading, and it indeed created the expectation of a tool based on grammar and focus on basic language skills more suited to learn a foreign language. Those aspects should be carefully considered in the further development of the tool. For example, by adding functionalities at the semantic level (leveraging dictionaries and thesaurus) and possibly at the pragmatic level (perhaps by analyzing rhetorical relations in the text).

The ultimate goal of fostering computational thinking was not recognized by the participants, and it would have been pretentious to actually expect that. The main purpose of our project is to provide an effective tool for teachers and possibly effective training for them to modify their teaching practices. Only after that, we could assess whether this approach can also affect the computational thinking skills or the attitude of students (and maybe teachers as well).

Although the results are preliminary and the study has several limitations (a small sample, an incomplete tool only used as a probe and the lack of involvement of actual high school students), we believe that, because of the lack of extant literature, it may provide evidence to support our twofold innovative approach: digital support to close reading and the need to foster computational thinking skills and attitude outside scientific and technical (STEM) educational classes.

REFERENCES

[1] Carmelo Ardito, Paolo Buono, Giuseppe Desolda, Rosa Lanzilotti, and Maristella Matera. 2019. Enabling Non-Technical Users to Shape their own Automation Experience. In *In Proceedings of the workshop "Everyday Automation Experience'19" in conjunction with CHI'19*.

[2] Claire Battershill and Shawna Ross. 2017. *Using digital humanities in the classroom: a practical introduction for teachers, lecturers, and students*. Bloomsbury Academic, London Oxford New York New Delhi Sydney.

[3] Nancy Boyles. 2012. We can't wait until middle school to teach students to read closely. Three practices bring close reading to the lower grades. *Educational Leadership* December 2012/January 2013 (2012), 36–41.

[4] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology* 3, 2 (2006), 77–101. https://doi.org/10.1191/1478088706qp063oa

[5] Virginia Braun and Victoria Clarke. 2019. Reflecting on reflexive thematic analysis. *Qualitative Research in Sport, Exercise and Health* 11, 4 (2019), 589–597. https://doi.org/10.1080/2159676X.2019.1628806

[6] Paul Martin Eve. 2019. *Close Reading with Computers: Textual Scholarship, Computational Formalism, and David Mitchell's Cloud Atlas*. Stanford University Press. https://doi.org/10.21627/9781503609372

[7] Gerhard Fischer, Elisa Giaccardi, Yunwen Ye, Alistair G Sutcliffe, and Nikolay Mehandjiev. 2004. Meta-design: a manifesto for end-user development. *Commun. ACM* 47, 9 (2004), 33–37.

[8] J Hawthorn. 2000. *A glossary of contemporary literary*. Oxford University Press.

[9] S Jänicke, G Franzini, M F Cheema, and G Scheuermann. 2015. On Close and Distant Reading in Digital Humanities: A Survey and Future Challenges. In *In Proceedings of Eurographics Conference on Visualization (EuroVis)*.

[10] Henry Lieberman, Fabio Paternò, Markus Klann, and Volker Wulf. 2006. End-User Development: An Emerging Paradigm. In *End User Development*, Henry Lieberman, Fabio Paternò, and Volker Wulf (Eds.). Vol. 9. Springer Netherlands, Dordrecht, 1–8. https://doi.org/10.1007/1-4020-5386-X_1 Series Title: Human-Computer Interaction Series.

[11] Verena Lyding, Egon Stemle, Claudia Borghetti, Marco Brunello, Sara Castagnoli, Felice Dell'Orletta, Henrik Dittmann, Alessandro Lenci, and Vito Pirrelli. 2015. The paisa'corpus of italian web texts. In *In Proceedings of 9th Web as Corpus Workshop (WaC-9) held in conjunction with EACL 2014*.

[12] Georgios Metaxas and Panos Markopoulos. 2017. Natural contextual reasoning for end users. *ACM Transactions on Computer-Human Interaction (TOCHI)* 24, 2 (2017), 1–36.

[13] Franco Moretti. 2013. *Distant reading*. Verso Books.

[14] Enrico Nardelli. 2019. Do we really need computational thinking? *Commun. ACM* 62, 2 (Jan. 2019), 32–35. https://doi.org/10.1145/3231587 Number: 2.

[15] Bonnie A Nardi. 1993. *A small matter of programming: perspectives on end user computing*. MIT press.

[16] John F Pane and Brad A Myers. 2000. Tabular and textual methods for selecting objects from a group. In *Proceeding 2000 IEEE International Symposium on Visual Languages*. IEEE, 157–164.

[17] Fabio Paternò and Carmen Santoro. 2019. End-user development for personalizing applications, things, and robots. *International Journal of Human-Computer Studies* 131 (Nov. 2019), 120–130. https://doi.org/10.1016/j.ijhcs.2019.06.002

[18] Peng Qi, Yuhao Zhang, Yuhui Zhang, Bolton Jason, and Christopher D. Manning. 2020. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In *In Proceedings of Association for Computational Linguistics (ACL) System Demonstrations*.

[19] Mohammed Saqr, Kwok Ng, Solomon sunday Oyelere, and Matti Tedre. 2021. People, Ideas, Milestones: A Scientometric Study of Computational Thinking. *ACM Transactions on Computing Education* 21, 3 (March 2021), 1–17. https://doi.org/10.1145/3445984

[20] Valerie J. Shute, Chen Sun, and Jodi Asbell-Clarke. 2017. Demystifying computational thinking. *Educational Research Review* 22 (Nov. 2017), 142–158. https://doi.org/10.1016/j.edurev.2017.09.003

[21] Xiaodan Tang, Yue Yin, Qiao Lin, Roxana Hadad, and Xiaoming Zhai. 2020. Assessing computational thinking: A systematic review of empirical studies. *Computers & Education* 148 (April 2020), 103798. https://doi.org/10.1016/j.compedu.2019.103798

[22] Anna Wing-bo Tso (Ed.). 2019. *Digital Humanities and New Ways of Teaching*. Digital Culture and Humanities, Vol. 1. Springer Singapore, Singapore. https://doi.org/10.1007/978-981-13-1277-9

[23] Edward Vanhoutte, Julianne Nyhan, and Melissa M Terras (Eds.). 2013. *Defining Digital Humanities*. Ashgate Publishing Limited, Farnham, UK.

[24] Matthew Wilkens. 2012. *Debates in the Digital Humanities*. University of Minnesota Press, Minneapolis, Chapter Canons, Close Reading, and the Evolution of Method, 249–258.

[25] Jeannette M. Wing. 2006. Computational thinking. *Commun. ACM* 49, 3 (March 2006), 33. https://doi.org/10.1145/1118178.1118215 Number: 3.

[26] J. M Wing. 2008. Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 366, 1881 (Oct. 2008), 3717–3725. https://doi.org/10.1098/rsta.2008.0118 Number: 1881.

[27] Maryanne Wolf and Mirit Barzillai. 2009. The Importance of Deep Reading. *Educational Leadership* 66, 6 (2009), 32–37.

[28] Eros Zanchetta and Marco Baroni. 2004. Morph-it! A free corpus-based morphological resource for the Italian language. In *In Proceedings of Corpus Linguistics*.