J of W Analytics

Advances in Code

ThesisWorkshopper: An Automated Thesis Statement Evaluator

Raymond Oenbring, *University of the Bahamas*

Structured Abstract

- **Identification of Innovation:** While scholars in rhetoric and composition are showing increasing interest in using large corpora of student writing to analyze student essays and instructor feedback on them, they have largely avoided developing natural language processing technologies designed to provide rich feedback to students on their writing. At the same time, those involved with the development of automated essay evaluation (AEE) systems—who are, almost always, not compositionists—are increasingly developing systems that provide rich feedback on a variety of essay features, rather than simple scores. The current study aims to show an example of how a web-based natural language processing tool can provide students with rich feedback on their papers under a process pedagogy framework.
- **Exposition of Innovation:** The current study presents ThesisWorkshopper (www.thesisworkshopper.com), a still-in-alpha-development web-based tool designed to provide feedback on students' draft thesis statements early in their essay writing process. The ThesisWorkshopper algorithm is coded mostly in Python and relies upon a number of off-the-shelf open-source Python libraries (e.g., a Python part-of-speech tagger).
- **Application of Innovation:** The study provides examples of the application's execution using different draft student thesis statements. The study also discusses how the researcher has used ThesisWorkshopper in their classes and the caveats that they have expressed to their students, including asking students to think critically about the limitations of algorithms and artificial intelligence.

- **Directions for Further Research:** While the algorithm is currently informed by a growing corpus of submitted student thesis statements, the researcher hopes in the future to expand the features of the algorithm by training it with corpora of example student thesis statements. What's more, the researcher hopes to build out the website's accounts system in the near future, allowing instructors to customize the algorithm for their students. The researcher invites other interested scholars to participate in the project or related projects.

Keywords: writing analytics, thesis statements, automated essay evaluation, natural language processing

1.0 Identification of Innovation

Decades ago, scholars in the field of artificial intelligence recognized that computer systems are more likely to have success in modelling human intelligence when the issues being focused on are specifically formulated questions within a narrow domain (see, for example, Leondes [2001] for an overview of the history of *expert systems*). Similarly, contemporary scholars within the field of automated essay evaluation (AEE)—an interdisciplinary field encompassing research from computer science, artificial intelligence (AI), natural language processing (NLP), psychology, education, and English composition/writing studies—are increasingly recognizing that AEE systems are more successful and useful when those systems have narrower foci and more modest aims. Early work in AEE construed the field as *automated essay scoring* (AES; cf. Shermis & Burstein, 2013)—that is, through its very name making stronger claims that the technology could replace trained human graders/instructors. However, the more recent title for the field, *automated essay evaluation*, seemingly recognizes that the products of AEE technologies should not be understood as a *be all, end all* but should rather be valued as tools capable of assisting students, instructors, and writing programs by providing automated analysis of and/or feedback on certain aspects of student writing (cf. Shermis & Burstein, 2013; Whithaus, 2013, p. vii; Zupanc & Bosnić, 2015, p. 384). What's more, a number of AES/AEE's traditional exponents, who are, almost always, not trained rhetoric and composition scholars, are increasingly recognizing the importance of not *fully* automating the process of essay assessment; that is, AEE promoters are increasingly recognizing the importance of "keeping humans in the loop" (see, for example, Foltz, 2020, p. 110).

Recent research in AEE increasingly aims to provide metrics for student papers across a number of linguistic dimensions and essay features, rather than simple holistic scores. For example, the algorithm used by Janda et al. (2019) uses 23 different syntactic, semantic, and sentiment features to analyze student papers. Making a case for an AEE technology specifically focused on the feature of essay organization (rather than holistic scoring), Persing et al. (2010) note that

> A major weakness of many existing essay scoring engines such as IntelliMetric (Elliot, 2001) and Intelligent Essay Assessor (Landauer et al., 2003) is that they adopt a holistic scoring scheme, which summarizes the quality of an essay with a single score and thus provides very limited feedback to the writer. In particular, it is not clear which dimension of an essay (e.g., coherence, relevance) a score should be attributed to. (p. 229)

AEE and NLP scholars' concern about early AES systems' lack of student feedback dovetails with some of the ideological problems that composition/writing studies scholars have with AEE systems that have undoubtably significantly slowed compositionists' adoption of these systems: that is, AES systems with holistic grading focus exclusively on the *product* of student writing assignments, rather than (encouraging students to reflect upon) the *process* that they go through in their writing and producing multiple drafts of their essays (see Shermis et al., 2013 for an overview of these concerns). Put differently, compositionists might be more willing to adopt NLP technologies if those technologies focused on *formative evaluation* (so the student and the instructor can know where students should focus their attention in order to improve their writing) rather than *summative evaluation* (used for grading and/or placing students in classes). Indeed, as Shermis (2003) notes, AEE/NLP technologies "can be used in a formative fashion to provide feedback at times when students like to write (any one of the 24 hours in a day). By submitting their essays to a web site … , they can get nonjudgmental feedback from a program that never gets tired. Based on the feedback, students may elect to revise and improve their work or leave it unchanged" (p. 5).

While compositionists have been correct to resist the complete automation of summative evaluation (that is, exclusive "robograding" of essays [e.g., Perelman, 2013, 2018]), this study aims to provide an example of how NLP/AEE technologies can potentially have a place in an English composition classroom informed by process pedagogies. Indeed, the recent establishment of *The Journal of Writing Analytics* suggests increasing interest among composition/writing studies scholars regarding the use of NLP tools and large corpora of essays to provide insight on patterns in student writing development[1] and instructors' feedback on student writing (e.g., Lang, 2018). However, despite this growing interest among compositionists and associated scholars in using corpus/computational linguistic technologies to analyze patterns in student writing, rhetoric and composition scholars have largely avoided developing NLP software tools to help students improve and/or analyze their writing (a notable exception is David Kaufer of Carnegie Mellon University).While composition scholars are right to be suspicious of wholehearted uncritical adoption of AI and NLP technologies, this study attempts to stake out a middle ground: that is, to show how we can *join the dark side on our own terms.* Indeed, if we as compositionists want to have a say in the pedagogical philosophies undergirding

[1] See, for example, Aull (2015, 2017) and Lang and Baehr (2012).

the tools that we are using in our increasingly digital classrooms, we should, wholeheartedly, participate in the development of those applications.

2.0 Exposition of Innovation

In this study, I describe a still-in-alpha-development web-based NLP technology entitled ThesisWorkshopper (www.thesisworkshopper.com). I contend that the student thesis statement is a particularly productive dimension of student essays for AEE/NLP technologies that provide automated feedback, for a variety of reasons. First of all, the student thesis statement is (or at least should be) the lynchpin of a student's line of reasoning in most English-composition-type essays, and thus, a well-formed student thesis statement is particularly important for a successful essay. Secondly, the student's development of the thesis statement is an early *choke point* in many instructors' essay assignment sequences (that is, many English instructors have a *thesis workshop* where they ask the student to submit/bring in their draft thesis statement for feedback/approval before the student drafts the complete essay). Finally, I contend that many students' draft thesis statements have a number of issues that are particularly amenable to detection by NLP systems (e.g., insufficient length or syntactic complexity in the thesis statement, insufficiently argumentative language, and poor syntactic parallelism). The goal of ThesisWorkshopper is not to replace instructor feedback, but instead to use computer technology to knock out the *low hanging fruit* in student thesis statements so that instructors can focus on higher-level issues when they provide feedback on student writing. While this is not the first project in the AEE/NLP tradition to focus specifically on student thesis statements rather than student English composition essays as a whole (see, for example, Persing & Ng, 2013), this project is distinct from previous research in that it focuses specifically on providing feedback on draft thesis statements.

Although I believe that the ThesisWorkshopper algorithm can be beneficial to all first-year composition students (and instructors), the application is, as might be expected, particularly suited to composition courses at the basic writing and/or developmental level (as a significant number of my first-year composition students are). Indeed, a number of the algorithm's functionalities are geared toward encouraging students to move toward a more complex, focused academic register. While it is an uncontroversial fact that NLP digital technologies have played a more significant role in other more developmental academic writing pedagogies[2] in comparison to "mainstream" rhetoric and composition pedagogy, the use of NLP technologies in the composition classroom remains controversial. Indeed, an obvious—and probably not unwarranted—critique of the use of NLP tools for assessing students' writing is that the tools are often limited to surface-level grammatical features (see, for example, Aull, 2017, p. 9; Dean, 2013).

[2] See, for example, the role that *computer-assisted language learning* has played in English for Academic Purposes (EAP) pedagogy in the past few decades (e.g., Cotos et al., 2015; Dodigovic, 2003; Levy & Stockwell, 2006).

2.1 Design Rationale

While it might have been possible to implement the ThesisWorkshopper algorithm in a smartphone app, I decided that a web-based implementation would work best for the needs of students and instructors using the tool; students generally write their essays on laptop or desktop computers rather than smartphones. For example, major AEE/NLP technologies such as Turnitin.com are largely web-based. ThesisWorskhopper has been implemented on a virtual private server (VPS) since I discovered that the performance of the algorithm was very slow in its execution when implemented using shared hosting. On the server side, the language analysis code is written, like much NLP code, entirely in Python, relying upon a number of off-the-shelf open-source NLP Python libraries (including a part-of-speech tagger, a parser, and a spell check library). SQL is used to interact with the databases, and PHP is used for other server-side scripts. On the browser side, ThesisWorskhopper uses a variety of languages and libraries, including JavaScript (for highlighting text on cursor rollover), jQuery, AJAX, and Bootstrap.

3.0 Application of Innovation

At present, the ThesisWorkshopper algorithm is capable of recognizing (and in some cases suggesting strategies for revising) a number of common issues in student draft thesis statements of varying technical complexity. On the simpler end of the scale, the ThesisWorkshopper algorithm can identify common issues noted by most contemporary word processing programs (and even some web browsers) such as misspelled words and common grammatical errors (e.g., sentence fragments, run-on sentences, and comma splices). Also rather simple in its technical implementation, the ThesisWorkshopper algorithm can make a number of suggestions triggered by the presence of certain words. For example, the presence of the word *discuss* (as in *this essay is going to discuss*) will cause the algorithm to advise the student to select a stronger verb (such as *argue*) to make a stronger assertion. The algorithm is also triggered by the occurrence of certain pairs of words (e.g., the presence of the words *positive* and *negative* [as in *there are many positives and negatives to issue X*] will cause the algorithm to suggest that the student choose less noncommittal language and argue one rather than both sides of the issue in their thesis statement). The algorithm is also capable of identifying when thesis statements lack sufficient syntactic complexity and length (presumably indicating that the student's thesis statement is simplistic).

Moving up in technical complexity, the algorithm is capable of recognizing the following: stylistic issues such as passive voice and nominalizations), poor parallelism (e.g., *To improve the education system, the school board should hire more teachers,* ***installing*** *better technology in classrooms, and improve school facilities)*, and basic logical fallacies (e.g., tautologies, sweeping generalizations, and oversimplifications). The ThesisWorkshopper algorithm can also recognize an overly broad list of topics in the thesis statement (i.e., when a thesis statement is construed as a list [*I argue X, Y, and* Z], but there is insufficient semantic correspondence between X, Y, and Z). The algorithm is trained on this feature using a section of a Wikipedia corpus.

Table 1

Example Algorithm Features

Category	Examples
Traditional grammar errors	Sentence fragments, comma splices, subject-verb agreement issues
Lexical choice issues	Weak verbs (e.g., *discuss*), personal judgmental language (e.g., *feel*), noncommittal language (e.g., *advantages* and *disadvantages*)
Stylistic issues	Passive voice constructions, excessive nominalizations, weak subject and verb (e.g., *there is*).
Syntactic issues	Lack of subordination/syntactic complexity, poor parallel structure
Advanced features	Basic logical fallacies, overly broad list of topics

The following figures show examples of the execution of the ThesisWorkshopper algorithm and the feedback provided by the website when students input draft thesis statements with common issues. Figure 1 shows the initial screen students see when they access the ThesisWorkshopper website.

Figure 1

ThesisWorkshopper Initial Screen

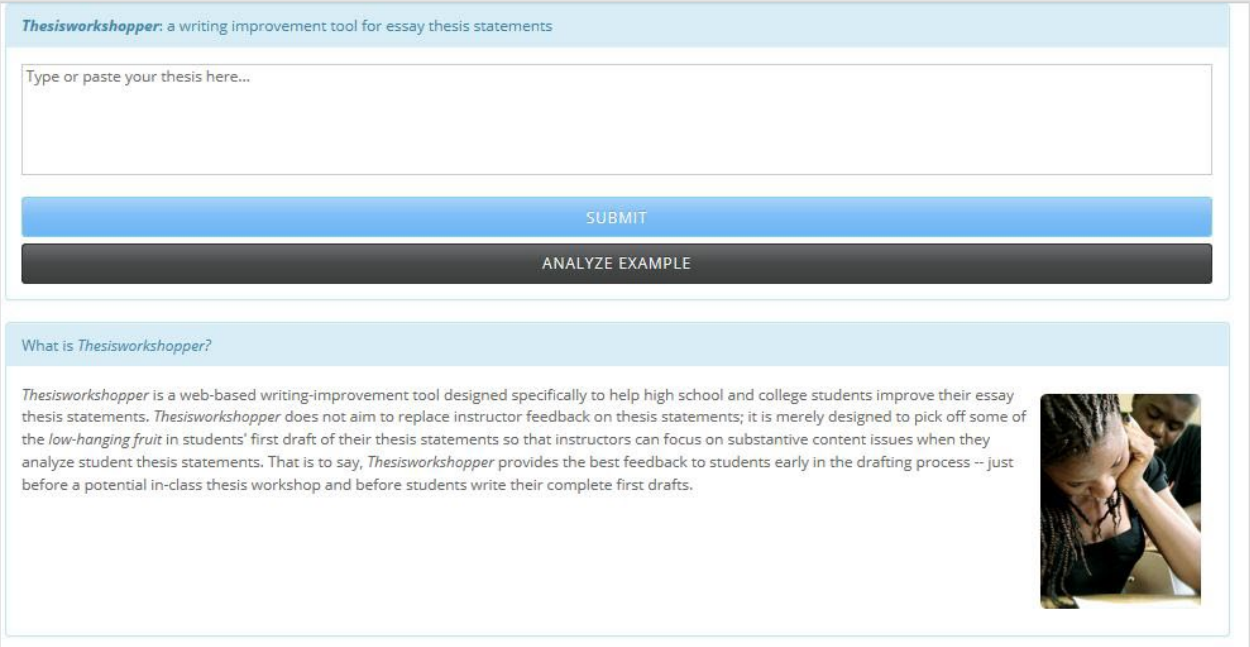


Figure 2 shows the output when a student submits the draft thesis statement *There are many advantages and disadvantages to legalizing marijuana.* The algorithm detects that the thesis statement

1. is too short, thus, presumably, lacking the complexity of longer multi-clause assertions;
2. has a weak main subject and main verb (with the word *subject* providing a link to a different part of the website that provides further explanation of the grammatical concept); and
3. has the noncommittal language *advantages* and *disadvantages* (that is, the best thesis statements choose one side of an issue rather than attempting to *play both sides*).

Figure 2

Output for Draft Thesis Statement: There are many advantages and disadvantages to legalizing marijuana.

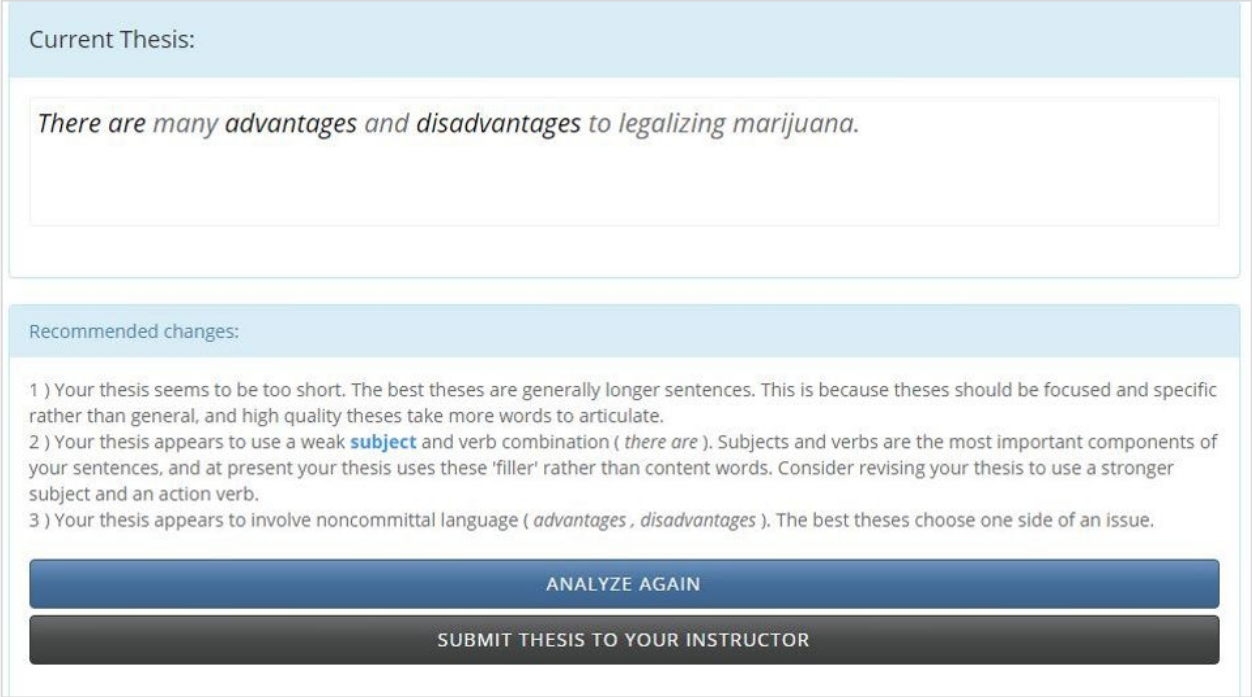


Figure 3 shows the output when a student submits the draft thesis statement *All Canadians are nice.* As with the previous example student thesis, the algorithm notices that the draft thesis statement is too short. Here, it also detects a lack of syntactic complexity and suggests that the student add a dependent clause to the thesis statement. Furthermore, the algorithm recognizes that the draft thesis statement is a sweeping generalization.

Figure 3

Output for Draft Thesis Statement: All Canadians are nice.

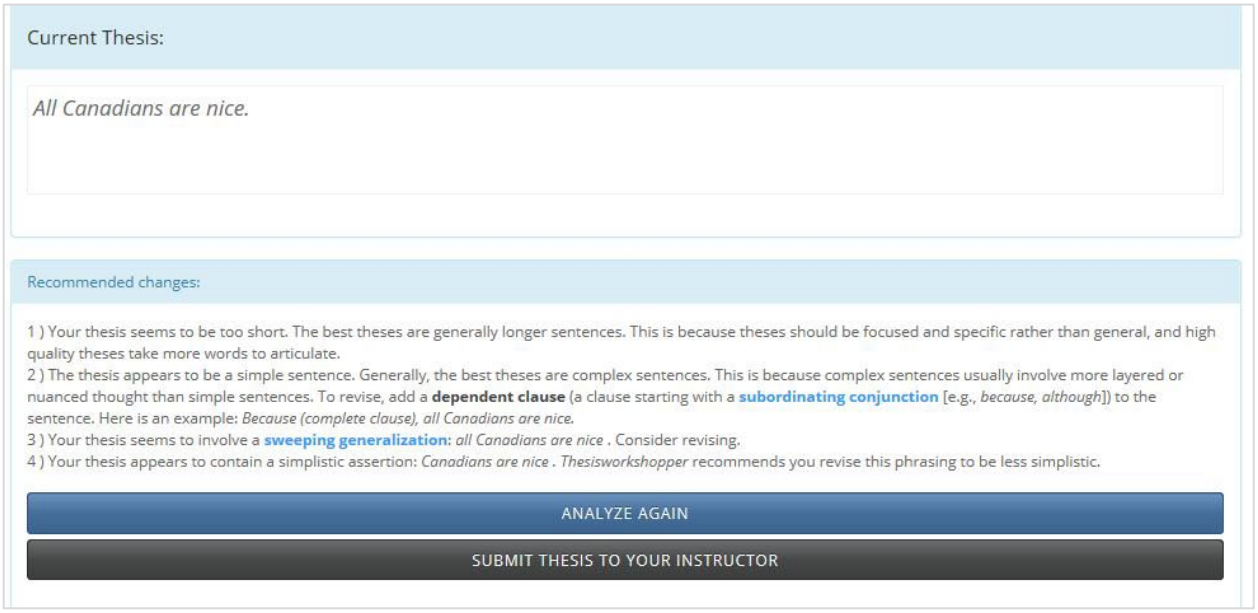
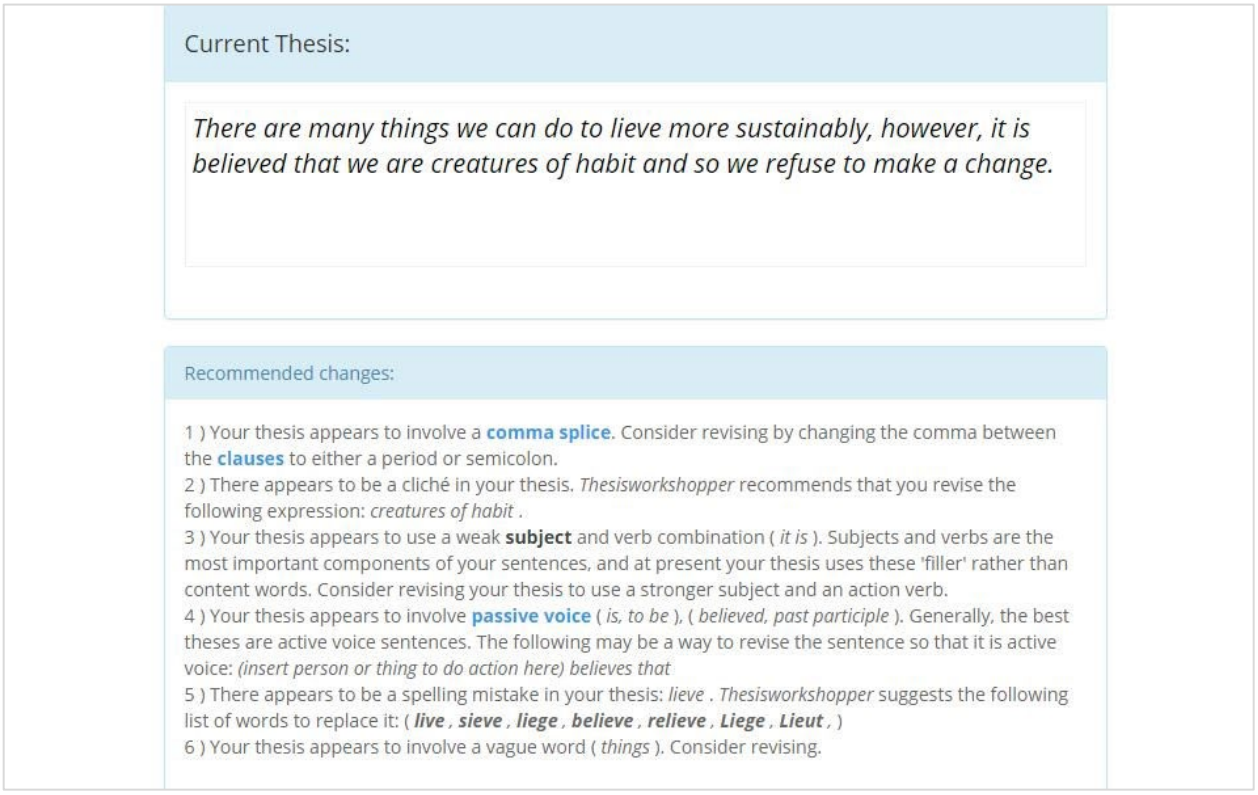


Figure 4 shows the output when a student submits the draft thesis statement *There are many things we can do to lieve more sustainably, however, it is believed that we are creatures of habit and so we refuse to make a change*. First of all, the ThesisWorkshopper algorithm recognizes the comma splice. Secondly, the algorithm notices the cliché *creatures of habit*. Thirdly, the algorithm, like in the first example, queries the weak subject and verb combination in the second independent clause (*it is*). Next, the algorithm recognizes the presence of a passive voice construction, and suggests a strategy to revise the syntax of the sentence. Next, the algorithm recognizes the spelling error *lieve*, and uses a Python spell check library to provide suggestions for revision (note that browser-side JavaScript functionality will replace the misspelled word with the selected word when the user clicks on it). Finally, the algorithm recognizes the presence of a vague word (*things*) and encourages the user to revise it.

Figure 4

Output for Draft Thesis Statement: There are many things we can do to lieve more sustainably. . .



3.1 Using ThesisWorkshopper in the (Physical or Virtual) Classroom

As I have suggested, ThesisWorkshopper is designed to be used early in the drafting process, before the in-person thesis workshop. That is, the goal of the application is not to replace instructor feedback on student papers, or even thesis statements. Rather, the goal of the application is to catch certain recurring *low hanging fruit* in student draft thesis statements so that the instructor can focus on higher-level issues when responding to student papers. While the application is obviously designed to provide rich feedback to improve students' (and instructors') experiences, I do not encourage instructors using the application with their students to do so completely blindly. In fact, I urge instructors using ThesisWorkshopper to encourage their students to think critically about the feedback the application provides. For example, if the student's draft thesis statement has none of the features that ThesisWorkshopper searches for, then the algorithm will state "ThesisWorkshopper makes no suggestions to improve your thesis. This indicates you may have a strong thesis. Congratulations!" However, I make clear to my students that just because they receive that feedback, that doesn't mean they have a strong thesis; it merely means that the algorithm did not find any of the features that it searches for in draft student thesis statements. Indeed, I believe that using ThesisWorkshopper in the composition

classroom offers students and instructors an opportunity to think critically about the limitations of algorithms and other artificial intelligence technologies.

4.0 Directions for Future Research

As the traditional focus of AEE technologies has been on writing assessment rather than the writing process, mainstream rhetoric and composition scholars' suspicious attitude toward the wholesale adoption of AEE technologies that take the human out of the equation—part of what I have cheekily referred to as *the dark side*—has been, I believe, reasonable. However, even before the COVID-19 pandemic, it was becoming clear that our writing classes are becoming increasingly—and probably permanently—digital. Furthermore, given recent leaps in AI NLP technologies, as demonstrated by the OpenAI's GPT-3 algorithm's ability to craft (almost) coherent original prose on general topics,[3] it seems clear that compositionists can no longer ignore the dark side of these technologies. That is, we as composition scholars can no longer remain on the sidelines as technology proceeds at its own pace and with its own agenda. Indeed, as I have argued in this piece, if we as compositionists wish to have a say in the pedagogical philosophies undergirding our increasingly digital classrooms, we should be actively involved in developing the applications that we use in our classrooms; that is the only way to assure that the technologies we are using are informed by our values. Accordingly, I welcome the participation of other scholars and/or entrepreneurs in the ThesisWorkshopper project, to expand and enhance the technologies outlined in this study.

While the ThesisWorkshopper algorithm currently provides mostly suggestions that have been hard coded by the researcher, I hope to soon extend the algorithm's capabilities by training it using corpora of student theses (as was done in Persing & Ng, 2013). Eventually, I hope to further incorporate advanced machine learning techniques as well. Finally, in the future, I would also like to develop ThesisWorkshopper's account system, which could allow developments such as letting instructors customize the algorithm for their students and logging the feedback the algorithm provides, tracking the progress of the student-writer's thesis through various revisions.

Author Biography

Raymond Oenbring is Professor of English at the University of the Bahamas, where he serves as Writing Program Coordinator. He is a coeditor of *Creole Composition: Academic Writing and Rhetoric in the Anglophone Caribbean,* which received the MLA's Mina P. Shaughnessy Prize and the CCCC's Outstanding Book Award.

[3] The stunningly (nearly) cogent prose that can be produced by OpenAI's GPT-3 algorithm (see, for example, GPT-3, 2020) suggests that we may be only a few years away from students being able to pay a small fee for an AI-produced high quality original essay. Clearly, such a possibility represents an existential threat to writing instruction as we know it.

References

Aull, L. L. (2015). *First-year university writing: A corpus-based study with implications for pedagogy*. Palgrave Macmillan.

Aull, L. L. (2017). Corpus analysis of argumentative versus explanatory discourse in writing task genres. *The Journal of Writing Analytics*, *1*, 1-47. https://wac.colostate.edu/docs/jwa/vol1/aull.pdf

Cotos, E., Huffman, S., & Link, S. (2015). Furthering and applying move/step constructs: Technology-driven marshalling of Swalesian genre theory for EAP pedagogy. *Journal of English for Academic Purposes*, *19*, 52-72. https://doi.org/10.1016/j.jeap.2015.05.004

Deane, P. (2013). On the relation between automated essay scoring and modern views of the writing construct. *Assessing Writing*, *18*(1), 7-24.

Dodigovic, M. (2003). Natural Language Processing (NLP) as an instrument of raising the language awareness of learners of English as a Second Language. *Language Awareness*, *12*(3-4), 187-203. https://doi.org/10.1080/09658410308667076

Elliot, S. (2001). *IntelliMetric: From here to validity* [Paper presentation]. Annual Meeting of the American Educational Research Association, Seattle, WA, United States.

Foltz, P. W. (2020). Practical considerations for using AI models in automated scoring of writing. In H. Jiao & R. Lissitz (Eds.), *Application of artificial intelligence to assessment* (pp. 101-113). Information Age.

GPT-3. (2020, September 8). A robot wrote this entire article. Are you scared yet, human? *The Guardian*. https://www.theguardian.com/commentisfree/2020/sep/08/robot-wrote-this-article-gpt-3

Janda, H. K., Pawar, A., Du, S., & Mago, V. (2019). Syntactic, semantic and sentiment analysis: The joint effect on automated essay evaluation. *IEEE Access*, *7*, 108486-108503.

Landauer, T. K., Laham, D., & Foltz, P. W. (2003). Automated scoring and annotation of essays with the Intelligent Essay Assessor. In M. Shermis & J. Burstein (Eds.), *Automated essay scoring: A cross-disciplinary perspective* (pp. 87-112). Lawrence Erlbaum Associates.

Lang, S. (2018). Evolution of instructor response? Analysis of five years of feedback to students. *The Journal of Writing Analytics*, *2*, 1-33. https://wac.colostate.edu/docs/jwa/vol2/lang.pdf

Lang, S., & Baehr, C. (2012). Data mining: A hybrid methodology for complex and dynamic research. *College Composition and Communication*, *64*(1), 172-194.

Leondes, C. T. (2001). *Expert systems: The technology of knowledge management and decision making for the 21st century*. Academic Press.

Levy, M., & Stockwell, G. (2006). *CALL dimensions: Options and issues in computer-assisted language learning*. Lawrence Erlbaum.

Perelman, L. (2013). Critique of Mark D. Shermis & Ben Hamner, "Contrasting state-of-the-art automated scoring of essays: Analysis." *Journal of Writing Assessment*, *6*(1). http://www.journalofwritingassessment.org/article.php?article=69

Perelman, L. (2018). *Why and how automated essay scoring doesn't work (most of the time) & the perils and promise of automated essay evaluation*. http://lesperelman.com/writing-assessment-robo-grading/presentation-on-automated-essay-scoring/

Persing, I., Davis, A., & Ng, V. (2010). Modeling organization in student essays. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing* (pp. 229-239). Association for Computational Linguistics.

Persing, I., & Ng, V. (2013). Modeling thesis clarity in student essays. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (pp. 260-269). Association for Computational Linguistics.

Shermis, M. (2003). Facing off on automated scoring. *Assessment Update*, *15*(2), 4-5.

Shermis, M., & Burstein, J. (2013). *Handbook of automated essay evaluation*. Routledge.

Shermis, M., Burstein, J., & Apel Bursky, S. (2013). Introduction to automated essay evaluation. In M. Shermis & J. Burstein (Eds.), *Handbook of automated essay evaluation* (pp. 1- 15). Routledge.

Whithaus, C. (2013). Foreword. In M. Shermis & J. Burstein (Eds.), *Handbook of automated essay evaluation* (pp. vii-ix). Routledge.

Zupanc, K., & Bosnić, Z. (2015). Advances in the field of automated essay evaluation. *Informatica*, *39*, 383-395.