Comparison of Tools for Digitally Tracking Changes in Text

Eve Washington¹, Bernat Ivancsics¹, Emily Sidnam-Mauch², Ayana Monroe³, Kelly Caine², Susan McGregor¹
Columbia University¹, Clemson University², The University of North Carolina at Chapel Hill³

Introduction
Over the past four decades, the need to trace and attribute changes to a digital text has been concentrated in two primary areas of digital technology use: word processing and software development. In this work, we survey the landscape of tools for digitally tracking changes in text and identify similarities and differences across tools.

Background
The widespread use of digital word processing by the late 1970s and early 1980s concentrated in two primary areas of digital technology use: word processing and software development. In this work, we survey the landscape of tools for digitally tracking changes in text and identify similarities and differences across tools.

Methodology
Our goal is to identify the user interface presentation mechanisms for the following elements in text tracking tools:

1. Authorship attribution;
2. Date/time of change;
3. Type of change (e.g. addition, deletion)
4. Substance of change (new text/old text)
5. Location of change within the document.

We reviewed four instances of each type of software tool (VCSs and word processing programs).

- Software: Git, Github, Visual Studio Code, IntelliJ IDEA

Acknowledgments
This material is based upon work supported by the National Science Foundation under Grant No. 1940670, Grant No. 1940679, Grant No. 1940713. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Discussion
All tools make use of text color and formatting to visually indicate text additions, deletions, and replacements.

While both types of tool can track precise text changes, the unit of emphasis differs, with code editors emphasizing changes at the line level (ex. Fig. 1) while word processing programs typically show character-level changes (ex. Fig. 2). Some of that distinction may be due to different cultures of use.

Usability Research⁴ indicates that users prefer that similar tools use similar interface conventions, so the shared visual representations of text changes within tool types is not surprising.

Wikipedia (Fig. 3) represents a type of crossover, following code editor conventions at the paragraph level. Wikipedia remains the only substantive example of an effort to display change tracking on a large corpus of published, human-readable text, suggesting that there is opportunity for significant experimentation in this area.

Customization options, where available in five out of eight of the tools explored. Fig. 4 and 5 show one option for customization in Word. They offer a glimpse into the breadth options for visually representing changes in digital text, many of which are underexplored.

References