

バージョン管理を考慮した Web に基づく リアルタイム協調型論文作成支援システムの実現

片山 真也[†] 合田 拓史[‡] 白松 俊[†] 大園 忠親[†] 新谷 虎松[†]
名古屋工業大学大学院情報工学専攻[†] 名古屋工業大学情報工学科[‡]

1. はじめに

本研究では、リアルタイム協調型の論文作成支援システムの実現を目的としている。この目的を達成するために、本研究では論文の添削の支援を対象としたシステムを開発している[1]。

論文の添削の課題として、1つの論文に対して何度も添削をしている場合、(1)添削者が以前に添削した論文からの変更箇所の見落としをしよう、(2)変更されていない箇所を読んでしまおう、(3)添削の履歴の活用がされない、といった点が挙げられる。

本研究では、(1)、(2)の課題をバージョン管理機構により解決した。これらの課題を解決することにより、必要十分な箇所に対しての添削、添削の時間の短縮といったことが可能となる。

ソースコードを主対象としたバージョン管理システムを論文のバージョン管理に適用するためには2つの課題を克服する必要がある。プログラムのソースコードに対するバージョン管理システムは Apache Subversion や Git が存在する。既存のバージョン管理システムを論文に適用する際の問題点は次の2点である。1つ目は、論文の一部の変更が変更された箇所以降の行にも影響する点である。2つ目は、論文の見込みの設定やコメント等、論文の添削とは関係の無い箇所のバージョンを管理してしまう点である。これらの点は添削者が論文の変更箇所を適切に把握することの妨げとなる。本稿では、論文を構造化し、整形することにより、これらの課題を解決した。

本稿では、既存のバージョン管理機構をリアルタイム協調型の論文添削に適用するための手法について説明する。

2. 論文作成のためのバージョン管理機構

本稿では、添削対象の論文を構造化した上で、既存のバージョン管理システムを利用することにより、論文作成に適したバージョン管理機構を実現した。このバージョン管理機構のシステム構成図を図1に示す。一般的に、論文の執筆には Microsoft Word や TeX を用いるが、本稿では TeX を対象としたバージョン管理について述べる。

論文解析モジュールは被添削者がアップロードした論文を構造化する。論文解析モジュールは入力された論文を解析し、構造化する。具体的には、1つの文や `\section` 等のコマンドで囲まれている箇所を1つのグループとする。また、コメントについては、異なるコメント間にコメント以外のグループが存在しない場合は1つのグループとして扱う。これにより、論文の添削とは関係の無い箇所のバージョンを管理してしまふことを防ぐことができる。

論文解析モジュールによって出力された構造化の情報は論文整形モジュールに入力される。論文整形モジュールは入力された情報を用いて論文を整形する。具体的には、1行につき、1つのグループの内容のみを論文に書くようにする。これにより、論文の一部の変更が、変更された箇所以降にも影響する問題を防ぐことができる。

整形した論文を既存のバージョン管理システムに入力することにより、論文のバージョン管理を行う。バージョン管理システムは論文の変更箇所を抽出するために用いる。ここではバージョン管理システムとして Git を用いた。

可視化モジュールは、バージョン管理システムから得られた論文の変更箇所の情報と、論文整形モジュールから得られた整形済みの論文を用いて、変更箇所を論文上で可視化する。具体的には、論文の変更箇所を、見込みを変更するコマンドで囲む。例えば、`\textcolor` コマンドで文字色を変更したり、`\it` コマンド等で書体を変更したりする。すべての変更箇所について見込みを変更する処理が完了したら、処理済みの論

Implementing a Web-based Realtime Collaborative Paper Editing System Considering Version Control.

[†]Shinya Katayama, Shun Shiramatsu, Tadachika Ozono, Toramatsu Shintani, Dept. of Computer Science and Engineering, Graduate School of Engineering, Nagoya Institute of Technology.

[‡]Takushi Goda, Dept. of Computer Science, Nagoya Institute of Technology.

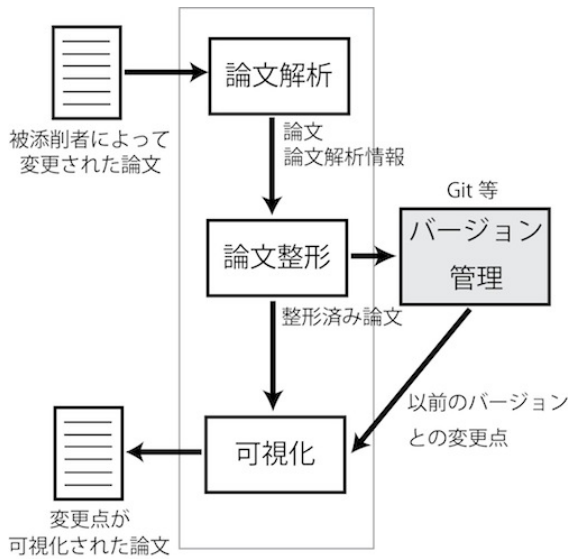


図1 バージョン管理機構のシステム構成図

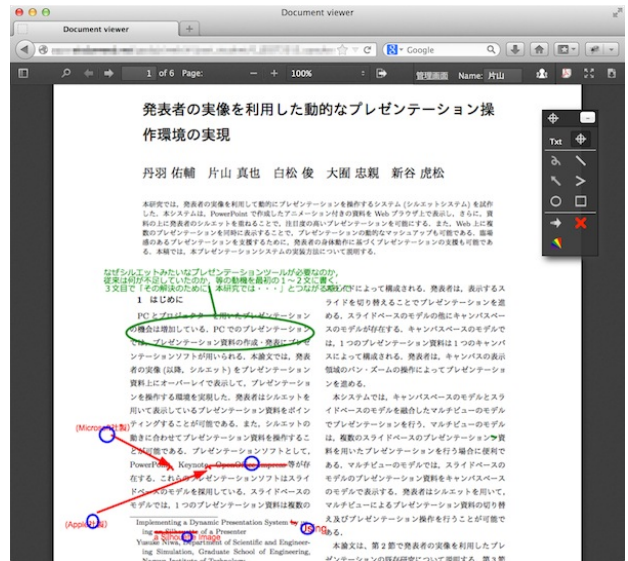


図2 GODA.js のスクリーンショット

文を出力する。

以上の処理で得られる変更点が可視化された論文は TeX ファイルである。よって、PDF ファイルに変換することにより、添削者が以前のバージョンからの変更点を確認することができる。

3. リアルタイム協調型論文作成支援システムの実装

2 節で示したバージョン管理機構を用い、リアルタイム協調型論文作成支援システム GODA.js[1]を実装した。図 2 に GODA.js のスクリーンショットを示す。

3.1 システムの概要

添削者は GODA.js を用いて、論文に文字列や簡単な図形を書き込むことにより、論文の添削をすることができる。添削者が書き込むことのできる図形は線分、矢印、楕円や四角等である。例えば、線分を論文中の特定の文の上に書き込むことで、その文を削除するように示すことができる。また、四角で論文の特定の箇所を囲み、囲った部分と、そことは別の箇所を矢印で結ぶことにより、四角で囲った箇所の移動を示すことができる。

3.2 論文のバージョン管理

被添削者は作成した論文とその関連ファイル(画像等)を ZIP ファイルとしてまとめて GODA.js にアップロードする。アップロードされた論文は GOF ファイルに変換される。GOF ファイルは GODA.js 独自のファイル形式であり、整形済みの TeX ファイルと、その関連ファイル、論文の構造化情報が格納されたファイル、GODA.js で

閲覧するための PDF ファイル、Git が生成するファイルによって構成される。

アップロードされた論文は 2 節で示したバージョン管理機構で以前のバージョンからの変更箇所が抽出、可視化される。変更箇所が可視化された論文は GODA.js で閲覧することができる。添削者は GODA.js 上で以前のバージョンからの変更箇所を確認し、添削を行うことができる。

4. おわりに

本研究では、論文のためのバージョン管理を考慮したリアルタイム協調型論文作成支援システムを実現した。論文の添削における課題として、以下の 3 点が挙げられる。(1)添削者が以前添削した論文から変更箇所の見落としをしてしまう。(2)論文の変更されていない箇所を読んでしまう。(3)添削の履歴の活用がされない。本稿では、(1), (2)の課題を克服するために論文を既存のバージョン管理機構に適した形状に構造化し、整形する手法を提案した。

今後の課題として、添削の履歴の活用に取り組む。本稿で示した論文の構造化の情報と、添削によって付加された文字列や図形の情報を結びつけることにより、論文の作成時に有効な付加価値が生まれると考えている。

参考文献

[1] S. Katayama *et al.*, “A Fast Synchronization Mechanism for Collaborative Web Applications based on HTML5,” In the Proc. of the SNPD2013, pp. 663-668, 2013.