Analysis of Collaborative Writing Processes Using Revision Maps and Probabilistic Topic Models

Vilaythong Southavilay, Kalina Yacef*, Peter Reimann†, Rafael A. Calvo*
*School of Information Technologies, University of Sydney
kalina.yacef@sydney.edu.au
†Centre for Research on Computer-supported Learning and Cognition, University of Sydney
peter.reimann@sydney.edu.au

ABSTRACT
The use of cloud computing writing tools, such as Google Docs, by students to write collaboratively provides unprecedented data about the progress of writing. This data can be exploited to gain insights on how learners’ collaborative activities, ideas and concepts are developed during the process of writing. Ultimately, it can also be used to provide support to improve the quality of the written documents and the writing skills of learners involved. In this paper, we propose three visualisation approaches and their underlying techniques for analysing writing processes used in a document written by a group of authors: (1) the revision map, which summarises the text edits made at the paragraph level, over the time of writing. (2) the topic evolution chart, which uses probabilistic topic models, especially Latent Dirichlet Allocation (LDA) and its extension, DiffLDA, to extract topics and follow their evolution during the writing process. (3) the topic-based collaboration network, which allows a deeper analysis of topics in relation to author contribution and collaboration, using our novel algorithm DiffATM in conjunction with a DiffLDA-related technique. These models are evaluated to examine whether these automatically discovered topics accurately describe the evolution of writing processes. We illustrate how these visualisations are used with real documents written by groups of graduate students.

Categories and Subject Descriptors

General Terms

Keywords
Collaborative writing processes, visualisation, probabilistic topic models.

1. INTRODUCTION
Collaborative writing (CW) is an essential skill in academia and industry. It combines the cognitive and communication requirements of writing with the social requirements of collaboration. Cognitive studies show that CW challenging in all these aspects [6]. Yet, CW is not explicitly taught in the school or higher education systems. Providing support on the processes of CW can be useful to improve not only the quality of the documents but also, more importantly, the CW skills of those involved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Our project explores how to support collaborative writing skills by providing feedback to students and teachers about that process. We provide feedback in the form of mirroring visualisations [8, 21] improving the awareness of a group’s writing activities. The aim is that individual students can perform their collaborative writing tasks more efficiently and effectively and teachers may monitor groups more effectively and detect problems early.

There is disparate prolific research for improving the support of quality writing such as tools for automatic scoring of essays [15], visualization of documents [11, 22], automatic question generation [10] and document clustering [1]. However, these approaches, unlike ours, focus on the final product, not on the writing process itself which can be used to gain insight on how student write their documents.

In this paper, we propose three visualisation approaches and their underlying techniques for analysing writing processes. We first create revision maps showing a snapshot of text edits performed by students on their jointly authored documents. This visualization depicts the development of documents at the paragraph level over a period of time. Based on text edits made to the paragraphs, we then extract topics by using several types of probabilistic topic models. We use the topic evolution charts to gain insights on how topics are created and developed during writing processes. Finally, we develop topic-based collaboration networks to analyse student collaboration based on their written topics. The topic-based collaboration networks present network diagrams showing students who commonly write about the same topics during their writing tasks.

This paper is organised as follows. First, we review related work in Section 2. Then, we outline an overview of our approach in Section 3. In Section 4, 5 and 6, we present our revision maps, topic evolution charts and topic-based collaboration networks, respectively. We validate our techniques with simulated data in Section 7. We then illustrate the applicability of our techniques using real word data of documents written by graduate students in Section 8. Finally, discussion and conclusion are provided in Section 9.
2. RELATED WORK

The recent work by Perrin and Wildi [12] investigated the dynamics of cursor movement during the process of writing and presented the movement in progression graphs. Based on the graphs, they proposed a statistical method to discover writing stages that authors went through. Particularly, they considered the progression graphs as time series consisting of large ‘bursts of signals’ and used statistical signal extraction to decompose the series into sequences of interesting features. Writing stages are then extracted based on the changes of the features. However, this analysis focused only on the temporal dynamics of cursor movements, not on text edits in the content. Unlike this work, we are interested in how the text content changes over time for deriving the writing processes. Another work by Kim and Lebanon [9] proposed a novel representation of document versions based on the location of the content words. They built a matrix in which columns correspond to position of each word and rows represent versions (time). They used the space-time smoothing method to extract and visualise changing in words over time [9]. Based on these changes, they discovered revision patterns. Although this method can discover which parts (i.e. word contents) of documents change over time, it does not incorporate topic evolution and author information in the visualisation, which we propose.

Another writing process visualisation was proposed by Caporossi and Leblay [5]. The visualisation is based on the graph theory that captures the overview of the writing processes. The graph of a writing process consists of nodes and links. The size and colour of each node indicate the number of text edits it represents and their nature respectively. For instance, yellow nodes represent additions that have later been removed, whereas the red ones depict additions that remain until the final text, and the blue nodes mean deletions. These nodes are connected by links or edges representing a spatial or temporal relationship, indicated by the shape and color of the edges. From the graphs, writing processes can be analysed. This representation technique of writing process can handle moving text position. This technique is useful for analyzing writing processes of documents written by single authors, but does not apply to collaborative writing contexts.

3. APPROACH

In this paper, we extended the framework proposed in [16] in order to visualising and analysing writing processes based on text edits and topic evolutions. Figure 1 depicts the architecture of our approach. It consists of a writing environment, Google Docs in the front end with Google Documents List API (GD API) for retrieving revisions and their information; a text comparison utility; Topic Model, and Author-Topic Model components.

In Google Docs, each document created is uniquely assigned a document identification number (document ID). Google Docs also keeps track of all version numbers (revision ID) of each document. Every time an author makes changes and edits a particular document, Google Docs stores the edited text content of the document as well as a record consisting of the following information in the revision history of the document:

- The version number (revision ID).
- The identification of the author (author ID).
- The timestamp (date and time) of the committed changes.

Some technical adjustments need to be made: for instance, several authors can make changes to the same content at almost the same time., but, the GD API only provides one main author for each revision. Therefore, we have to manually reconcile the list of authors for each revision by using the revision history function on the web interface of Google Docs.

There are three kinds of visualisations generated. The first visualisation, which is called a revision map, depicts text edits performed on individual paragraphs during writing processes. In order to understand the semantic of these text edits, the second visualisation, called a topic evolution chart, depicts how written topics were created and developed during writing processes. The third visualisation, called a topic-based collaboration network, shows the network of authors’ collaboration based on topics. For instance, it displays whether any two authors have written the same topics during their writing tasks. These three visualisations will be detailed in the next three sections.

As depicted in Figure 1, after retrieving the text contents of all the revisions and the revision history for a particular document, we use the text comparison utility to identify the text edits in successive revisions and identify the list of added and deleted texts. These identified text edits, mapped against the list of author IDs and the timestamps of corresponding revisions, are used in the revision map. They are also used as input to both the topic and author-topic modelling algorithms. The topic modelling, especially DILDA [19], creates the topic evolution chart, while the author-topic modelling [13–14] outputs the topic-based collaboration network using author IDs provided in the revision history. The detail of how these three visualisations are created will be described in the next three sections.

4. REVISION MAP

In order to gain insights on how students develop their jointly authored document over a period of time, it is useful to present a chronological picture of what happened during the writing of the document. Revision maps summarise the text edits made by
students at the paragraph level over their period of writing. Figure 2 depicts the revision map of a real document written by a group of students. Each column refers to a revision of the document. Each small rectangle depicts a paragraph of the document. Each row shows the evolution of an individual paragraph over time, as it is created, altered, or deleted during the writing process. Rectangles are colour-coded to visualise the nature and the extent of the edits made to the paragraph: green means more words were added than deleted, red means more words were deleted than added, and white represents no change in the paragraph. The intensity of these colours approximately denotes the extent of those edits. If there are as many added words as deleted ones, the rectangle is coloured in yellow-green. Lastly, the horizontal bar under author IDs’ row shows the aggregated edits of individual revisions and the last vertical column represents the aggregated edits of individual paragraphs across all revisions during the writing process.

Each paragraph evolution is positioned relative to its position in the current (final) revision. This means that the paragraph evolution rows can move up and down over time, especially when a new paragraph is added. In addition, the paragraph evolutions were grouped into sections based on the structure of the document.

For instance, the revision map shown in Figure 2 represents the edits in a document written by a group of students over a period of 6 days (from 04 to 09/05/2011). The text edits of four paragraphs: P1, P2, P3 and P4 (as indicated in the revision map) can be described as following. The first paragraph of Section A (P1) was added by c1 on 04/05/2011 22:29 with lots of words. It was not edited until 06/05/2011 16:48 (by author c1), in which more words were deleted than added to it. Towards the end of the week (on 08/05/2011 21:46), P1 was modified again by c1 with more words added. The first paragraph of Section B (P2) was inserted on 05/05/2011 13:57 by author c2. It was not modified at all and was deleted altogether from the document on 09/05/2011 02:38 by c2. A new paragraph (P3) was inserted by c2 after P2 was removed. A paragraph can be split and merged. For instance, the paragraph, P4 was inserted on 05/05/2011 13:57 by c2. It has been changed once, with few words added on 06/05/2011 16:48 by c1. P4 was then split into two paragraphs on 09/05/2011 02:38 by c2.

Revision maps can help provide answers to the following five questions:

1. Which sections of the document were the most or the least worked on? (Location of text edits)
2. When (at what dates) did major edits (i.e. addition and deletion) occur during the writing process? (Time)
3. Did students work sequential or in parallel? (i.e. sequential – single paragraphs were written at different writing sessions or days, or parallel – many paragraphs were written almost in parallel at the same writing sessions or days).
4. Who made the most or the least edits to the document? (Authorship)
5. How many authors worked on each paragraph and each section? (Collaboration)

First, by examining the vertical bar showing the aggregated edits of individual paragraphs, we can easily answer Question 1. Particularly, we learn that Section A of the document has been edited more than Section B. The second interesting observation based on Question 2 above is that most of text edits, especially text additions happened at the beginning of the process when students started their writing tasks. There were some extensive text edits (only by author c3) in the middle of the process. In addition, there were many text deletions towards the end of the writing process. For Question 3, it is interesting to note that students wrote their document sequentially for Section A, especially the first three paragraphs by two students, c1 and c5. In contrast, paragraphs of Section B were created almost at the same time (by student c2). However, these paragraphs of Section B did not make the final revision. All of them were replaced at the last stage of the writing.

The revision map also provides an insight on how student collaborate over the period of writing, in which Question 4 and 5 can be used as a guideline. Among the five students, we observe that c4 involved the least in the development of the document. c2 worked mainly on Section B during the period of writing. In addition, c1 and c5 collaborated a lot to develop their documents, especially in Section A.

In Figure 2, although c4 perform very little of text edition comparing relatively to other four students, it is quite difficult to conclude that c4 contributed the least to the development of the document. Particularly, we would like to know if small text editions of c4 increase the assignment of a topic, and thus make the content text clearer and improve the coherent of the content.

This initial analysis provides some insight on how students created their jointly authored document at the paragraph level. We will now look at ways to investigate how they developed ideas during writing, especially how their topics evolved over time.
5. TOPIC EVOLUTION CHART

Knowing how topics have evolved during text edits can provide help in understanding how students developed their ideas and concepts during their writing tasks. This is addressed by our topic evolution chart, shown in Figure 3 with four topics (T1, T2, T3, and T4). A topic consists of a cluster of words that frequently occur together in a revision, and each document revision is represented by a set of topics. The topic evolution chart depicts changes in the membership of topics throughout the sequence of revisions. For instance, in Figure 3, T1-T3 appeared at the start of writing (Revision 1), whereas T4 emerged in the sixth revision (Revision 6) and disappeared later in the writing process (Revision 7). The ratio of importance of the other three topics (T1-T3) changes over time. In the beginning of the writing process, the document contains more text about topic T1 than about T2 or T3 (66% vs 17%). However, towards the end of the process, topic T2 is more dominant in the document than T1 and T3.

![Figure 3. A topic evolution chart of three topics: T1, T2, T3, and T4 over 17 revisions.](image)

Latent Dirichlet Allocation (LDA) [3] is a popular probabilistic topic modeling technique, which, to our knowledge, has not been used to extract the evolution of topics during the writing of the same document. The closest method to do so is DiffLDA [19], which has been used for extracting topic evolution in software repositories. In DiffLDA, the GNU diff utility was used to identify text edits at the line level before using LDA. We build on these techniques to extract topics and their evolution during the writing process.

In our work, we developed a text comparison utility to extract text edits at both paragraph and word levels. Unlike DiffLDA, the number of topics and hyper-parameters, α and β (of the two Dirichlet distributions: author’s topic distribution and topic-specific word distribution), are selected using a trade-off between the model fitting (i.e. perplexity) and the simplicity of model structure (i.e. the smallest number of topics). Particularly, the number of topics is selected independently for each document. In the following subsections, we will first give an overview of the probabilistic topic model, Latent Dirichlet Allocation (LDA) and DiffLDA before describing how we extend DiffLDA for mining topic evolution of writing processes.

5.1 Probabilistic Topic Models

**Topic modeling** automatically discovers topics within a corpus of text documents [2] in which topics are collections of words that co-occur frequently in the corpus. Due to the nature of language usage, the words that constitute a topic are often semantically related. Each document is represented as a probability distribution over some topics, while each topic is represented as a probability distribution over a number of words. LDA defines the following generative process for each document in the collection:

1. For each document, pick a topic from its distribution over topics.
2. Sample a word from the distribution over the words associated with the chosen topic.
3. Repeat the process for all the words in the document.

More formally, in the generative process, LDA infers, for each of $T$ topics, an $N$-dimensional word membership vector $z(d, n)$ that describes which words appear in topic $z$, and to what extent. In addition, for each document $d$ in the corpus, LDA infers a $T$-dimensional topic membership vector $d(\theta, d)$ that describes the extent to which each topic appear in $d$. Both $\theta$ and $\phi$ have Dirichlet prior with hyper-parameters $\alpha$ and $\beta$, respectively. LDA performs these inferences using Bayesian techniques such as collapsed Gibbs sampling, a Markov-chain Monte Carlo (MCMC) method, which is currently in widespread use as an inference tool among topic modelers [7].

Topic evolution models using LDA suffers the duplication effect as explained in [18]. These topic evolution models have an assumption that documents in the corpus are unique across time. This assumption holds for the collection of journals, blog posts, and newspaper articles, which are typically studied in the topic modeling. It is very unlikely that an article published in one year is only slightly updated and republished the next year in the same conference proceedings. Instead, each article (i.e. the specific combination of words within article) is unique across time. However, the nature of writing process is quite different. Jointly authored documents are usually updated incrementally from one revision to another revision as authors developed the documents. Although sometimes there can be lots of text edits occurring in one revision, there still exists some overlap of text contents between the revision and the previous one. This particularity has been addressed by DiffLDA, which is described next.

5.2 DiffLDA for Mining Writing Processes

In order to address the data duplication effect found in software repositories, Thomas et al. [18-19] proposed a simple technique used in the pre-processing step before applying LDA to their source codes. On top of the normal pre-processing steps, they included the **diff step** to identify text edits between every pair of two successive versions of each source code. In particular, for every pair of successive versions, DiffLDA uses the standard GNU diff utility to compute the edits (i.e. **add**, **delete** or **change**) at the line levels. According to DiffLDA [19], if an existing line is changed, it is considered to be deleted and then added again. Identifed edits (added and deleted lines) were then used as documents, called **delta documents** [19]. The corpus then consisted of all delta documents in the software repository. This diff step effectively removes all duplication and thus prevents the duplication effect when applying LDA for the corpus.

However, the pre-processing step used in DiffLDA can not be applied directly in our work. In writing processes, it is common that authors revised a paragraph, which is a line in pain text, several times by changing some words in the paragraph. Therefore, a number of words in the revised paragraph have not been altered at all. Using the pre-processing step of DiffLDA will generate many change edits for particular paragraphs or lines. Consequently, the resulting delta documents will have many duplicated words.

In our approach, we develop a text comparison utility (TCU) to compute the edits between successive revisions. TCU first identify text edits at paragraph levels. In other words, for each revision, it compares its individual paragraphs to the corresponding paragraphs of the previous revision, using the GNU diff utility. As a result, a paragraph can be classified as added, deleted or changed depending on whether the text edits that transform the previous to the current revision produces a new paragraph, remove the existing one, or change some words in the
existing one. For changes in the existing paragraphs, TCU then computes text edits at word levels. Particularly, it identifies words as added, deleted, or equal (no change) depending on whether they have been newly added, removed or not altered at all. The added and deleted words and paragraphs are then used as documents for LDA to extract topics and topic evolution. Therefore, our method can prevent the duplication effect.

The procedure used in our work can be described below. For each document, we first identify text edits (at paragraph and word levels) between two consecutive revisions \( R_i \) and \( R_{i+1} \) using the text comparison utility, as explained above. For each revision of document, we create two delta documents, \( \delta^A \) and \( \delta^D \) to capture both types of text edits: addition and deletion, similar to [19]. We place all added word and paragraph edits between \( R_i \) and \( R_{i+1} \) into \( \delta^A \) and all deleted paragraph and word edits into \( \delta^D \). For the first revision \( (j=1) \), we classify the entire revision as added paragraphs, and therefore we add the entire paragraphs of the revision to the delta document, \( \delta^A \). Using this method, each revision has at most two delta documents. Sometimes a revision can have one delta document of either added or deleted paragraphs. After that, we apply LDA to the entire set of delta documents. As a result, we obtain a set of extracted topics and membership values for each delta document.

LDA requires us to set parameters, \( \alpha \) and \( \beta \), of the two Dirichlet distributions: author’s topic distribution and topic-specific word distribution. We use the strategy to fix \( \alpha \) and \( \beta \) to depend on the number of topics, \( T \) and explore the consequence of very \( T \) as explained below. Therefore, based on the techniques proposed by Griffiths and Steyvers[7] and our empirical testing, we set the value of \( \alpha=50/(\#\text{topics}) \) and \( \beta=200/(\#\text{words}) \).

After defining the hyper-parameter values as mentioned above, we chose the number of topics \( T \) by using perplexity [7], which is a standard measure for estimating the performance of a probabilistic model based on its ability to predict words on new unseen documents. We selected \( T \) as small as possible while maintaining a good model fit. The number of topics is selected independently for each document. We used the technique proposed [4] to ensure that the chosen model is not too fit and can be generalized for modeling data.

The extracted topic and topic evolutions provide an overview of how topics are created and evolved. Knowing whether students collaborate and commonly write about the same topics can assist instructors and learners understand how the documents have been developed. To perform analysis about learner collaboration, we created topic-based collaboration networks which will be explained below.

6. TOPIC-BASED COLLABORATION NETWORK

Let us now turn towards visualising the presence of collaboration of students around topics. In particular, we are interested to know whether students develop their ideas and concepts independently or work together on the same topics. Figure 4 shows a topic-based collaboration network from a group of four students writing a document. Each node represents a student (an author). A square depicts a group coordinator. Circles represent group members. A connection (link) between two nodes shows that the corresponding students have written about the same topics during their writing tasks. From the figure, we can see that the group coordinator \( a1 \) has worked with all group members to draft, revise, and edit some topics written in the document. Similarly, \( a2 \) also worked on some topics with all other group members. However, \( a3 \) and \( a4 \) have not written about the same topics. In other words, independently \( a3 \) and \( a4 \) have worked with both \( a1 \) and \( a2 \) to develop some topics.

Our contribution in the creation of this visualisation resides in the creation of a Diff Author-Topic Model (DiffATM), which is an extension of Author-Topic Model (ATM), proposed by [14]. As DiffLDA can overcome the duplication effect in LDA, DiffATM is developed to deal with the duplication effect in ATM. For this work, similar to DiffLDA, DiffATM is applied to text edits identified at the paragraph and word levels in order to extract topics. However, DiffATM does not provide a cluster of topics per revision, but a cluster of topics per author. Based on a number of revisions, a particular author can be represented by a membership of topics written in those revisions. Similar to DiffLDA for writing processes, DiffATM is developed by selecting the number of topics and hyper-parameters based on the trade-off between the model fitting and the simplicity of model structure. In addition, we apply social networks proposed by [4] for collaborative writing tasks based on the membership of topics of individual authors.

In this section, the author-topic model will be described first. Then, how topic-based collaboration networks are constructed will be explained.

6.1 The Author-Topic Model

The Author-Topic Model (AT Model) is an extension of LDA, which was first proposed in [14] and further extended in [13]. Under this model, each word \( w \) in a document is associated with two variables: au author, \( x \) and a topic, \( z \). Similar to LDA, each author is associated with a multinomial distribution over \( T \) topics, denoted as \( \theta \). Each topic is associated with a multinomial distribution over words, denoted as \( \phi \). Differently to LDA, the observed variables for an individual document are the set of authors and the words in the document. The formal generation process of Author-Topic Model is as follows [13]:

For each document, given the vector of authors, \( a_t \):

- For each word in the document:
  1. Conditioning on \( a_t \), choose an author \( x_{dt} \sim \text{Uniform}(a_t) \).
  2. Conditioning on \( x_{dt} \), choose a topic \( z_{dt} \).
  3. Conditioning on \( z_{dt} \), choose a word \( w_{dt} \).

One important difference between the Author-Topic Model and LDA is that there is no multinomial distribution over \( T \) topics for an individual document. Therefore, if we want to model documents and authors simultaneously, further treatment is needed. A detailed description can be found in [13].
6.2 Diff Author-Topic Model for Writing Processes

The DiffATM model provides an analysis that is guided by the authorship data of the documents (provided by revision histories), in addition to the word co-occurrence data used by DiffLDA. Each author is modeled as a multinomial distribution over a fixed number of topics that is selected empirically as explained below. Each topic is, in turn, modeled as a multinomial distribution over words.

As described in Subsection 5.2, the Text Comparison Utility (TCU) outputs the delta documents (i.e., added and deleted words and paragraphs). As mentioned in Section 3, each revision is produced by one or more authors. The authors of each revision are mapped to delta documents of that revision. Next we apply the Author-Topic Model (ATM) to the entire set of delta documents.

As in DiffLDA, the hyper-parameters defining each Dirichlet prior \( \alpha \) and \( \beta \) of DiffATM are depended on the number of topics, which is selected independently for each document using the trade-off between the model fitting and the simplicity of the model structure as described in the previous section. The likelihood of two authors co-writing on the same topic will depend on the hyper-parameters chosen. In general, larger values of \( \alpha \) will lead to a larger topic overlap for any given corpus, motivating the use of a consistent hyper-parameter selection algorithm across all corpora analysed. All hyper-parameter settings used for the analyses presented here follow the guidelines derived empirically by Griffiths and Steyvers [7]. In particular, \( \alpha = 50/\# \text{topics} \), inducing topics that are mildly smoothed across authors, and \( \beta = 200/\# \text{words} \), inducing topics that are specific to small numbers of words.

Like DiffLDA, DiffATM is fit using a MCMC approach. Information about individual authors is included in the Bayesian inference mechanism, such that each word is assigned to a topic in proportion to the number of words by that author already in that topic, and in proportion to the number of times that specific word appears in that topic. Thus, if two authors use the same word in different senses, the DiffATM will account for this polysemy. Details of the MCMC algorithm derivation are given by Rosen-Zvi et al. in [14].

After the number of topics, \( T \), has been selected, a \( T \)-topic DiffATM is fit to all delta documents. 10 samples are taken from 20 randomly initialised Markov chains, such that there are 200 samples in total. The result of the final samples are used to construct the topic-based collaboration network, described next.

6.3 Construction of Topic-Based Collaboration Network

The aim of this network visualisation is to mirror whether students commonly use the same topics of discourse over the period of writing. We use the same method proposed by [4], in which we compute each pair of authors’ joint probability of writing about the same topic as:

\[
P(X_1 \cap X_2) = \sum_{Z} P(Z = z_1 | X_1)P(Z = z_2 | X_2)
\]

If the joint probability of two authors exceeded 1/\( T \) (e.g. 0.1 if \( T = 10 \)), we create a link between the two nodes. The reason for choosing this condition is explained in [4]. We construct a square author-author matrix with entries equal to one for each linked author pair, and entries equal to zero otherwise. For each document, we then repeat this procedure several times as suggested by [4] to average across whatever probabilistic noise might exist in the DiffATM fit. Authors who link across multiple DiffATM fits more often than would be expected due to chance were considered to be linked in the network for that document. After sampling DiffLDA for 200 times, we obtain the author-author matrix as a result. Based on the matrix, for each author pair if its entry is more than 125, we consider there is a link between that pair of authors.

7. TECHNICAL VALIDATION

We describe how we validate the accuracy of DiffLDA and DiffATM, which are used to create topic evolution and topic-based collaboration networks. Since there is no public dataset for evaluating the accuracy of topic evolution models, we created a synthetic dataset, as inspired by [19] where we simulated text edits on a document. In particular, we created two simple scenarios representing several types of text edits so that we were able to evaluate whether the evolutions discovered by the models were accurate. Especially, we would like to check if the text edit events detected by the models correspond to the actual changes that happened during writing (precision) and if the discovered evolutions contained all the text edits that were actually performed during writing (recall).

7.1 Data Generation

To evaluate the DiffLDA model for collaborative writing, we first created a document with 17 revisions (R1 – R17). The document consisted of three paragraphs, which are generated from three topic distributions with equal weight. Table 1 shows the dictionary and topic distribution of the data. After created (or first added to the document), each paragraph was changed three times. The changed paragraphs were also generated from the tree topic distributions as presented in Table 1. The text changes of these paragraphs were depicted in the event log file shown in Table 2. It is important to note that there were no text changes in some revisions. The 17 revisions formed our baseline scenario.

<table>
<thead>
<tr>
<th>Words</th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>River</td>
<td>0.37</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stream</td>
<td>0.31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bank</td>
<td>0.22</td>
<td>0.28</td>
<td></td>
</tr>
<tr>
<td>Money</td>
<td>0.3</td>
<td>0.07</td>
<td></td>
</tr>
<tr>
<td>Loan</td>
<td>0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Debt</td>
<td>0.12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factory</td>
<td>0.33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Product</td>
<td>0.25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Labor</td>
<td>0.25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>News</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>Reporter</td>
<td>0.05</td>
<td>0.05</td>
<td>0.05</td>
</tr>
</tbody>
</table>

We set up two simulated scenarios as follows. The first scenario modifies the baseline scenario by adding one paragraph in the revision R6 as shown in Table 2, and deleting it in the revision R7. The added paragraph was generated from a new topic (i.e. the four code names of Ubuntu operating system) totally unrelated to the three topics in the baseline scenario. This scenario simulated two types of text edits: 1) addition of a new paragraph and the deletion of an existing paragraph.

The second scenario was created based on the first scenario by adding two paragraphs: 1) A paragraph from a new topic
unrelated to the four topics mentioned above was added in the first revision R1, remained (unchanged) in the revision R2 and R3, and deleted in the revision R4. 2) A paragraph from another unrelated new topic was added in R14 and R15. The first half of the paragraph was added in the revision R14, while the second half of the paragraph was added in the final revision R16. These two scenarios were created for testing multiple text edits happening simultaneously in the same revision.

Table 2 shows the text edition events. The simulation was designed in the way that there are no more than four paragraphs in any revisions at any time. The baseline scenario consists of three paragraphs P1, P2, and P3. The first controlled scenario (C1) is to add and delete P4. The second one (C2) is to add and delete P5 and to add and change P6. There are four text edition events: no change, adding, changing, and deleting a corresponding paragraph, presented as ‘-’, a, c, and d, respectively. Each revision is produced by at most two authors. There are five authors: a1 – a5.

<table>
<thead>
<tr>
<th>Rev.</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>C1</th>
<th>C2</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>a</td>
<td></td>
<td>a</td>
<td></td>
<td>a1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>a1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R3</td>
<td>-</td>
<td>a</td>
<td></td>
<td></td>
<td></td>
<td>a2</td>
<td>a5</td>
<td></td>
</tr>
<tr>
<td>R4</td>
<td></td>
<td>-</td>
<td>a</td>
<td></td>
<td></td>
<td>a2</td>
<td>a5</td>
<td></td>
</tr>
<tr>
<td>R5</td>
<td>c</td>
<td></td>
<td></td>
<td>d</td>
<td>a3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R6</td>
<td></td>
<td></td>
<td>a</td>
<td></td>
<td></td>
<td>a4</td>
<td></td>
<td>a1</td>
</tr>
<tr>
<td>R7</td>
<td>-</td>
<td>d</td>
<td>a</td>
<td></td>
<td></td>
<td>a4</td>
<td></td>
<td>a1</td>
</tr>
<tr>
<td>R8</td>
<td>c</td>
<td></td>
<td></td>
<td></td>
<td>a5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R9</td>
<td>-</td>
<td>c</td>
<td></td>
<td></td>
<td>a5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R10</td>
<td>c</td>
<td></td>
<td>a</td>
<td></td>
<td>a5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R11</td>
<td>c</td>
<td>c</td>
<td></td>
<td></td>
<td>a5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R12</td>
<td>-</td>
<td>c</td>
<td></td>
<td></td>
<td>a3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R13</td>
<td>c</td>
<td></td>
<td></td>
<td></td>
<td>a3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R14</td>
<td>-</td>
<td>c</td>
<td></td>
<td>a</td>
<td>a2</td>
<td>a5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R15</td>
<td>-</td>
<td>-</td>
<td>a</td>
<td></td>
<td></td>
<td>a2</td>
<td>a5</td>
<td></td>
</tr>
<tr>
<td>R16</td>
<td>-</td>
<td>-</td>
<td>c</td>
<td></td>
<td>a2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R17</td>
<td>-</td>
<td>c</td>
<td></td>
<td></td>
<td>a3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7.2 Preprocessing and Study Setup

After identifying text editions and creating delta documents as described earlier, we preprocessed them. For the analysis reported in this paper, a word-document matrix and author-document matrix were constructed using doc2mat utility from the CLUTO package [17], which removes all stop-words and stems all words to their roots using the Porter stemming algorithm.

For Scenario 1, the preprocessing results in a total of 417 words (15 of which are unique) in 23 (delta) documents. There are (M=18.13, STD=0.81) words per revision. The Scenario 2 consists of 485 words (23 of which are unique) in 26 (delta) documents. There are (M=18.65, STD=4.25) words per revision in Scenario 2.

For the actual LDA and ATM computations, we used the Topic Modeling Toolbox [20] implemented in MATLAB. We ran 500 sampling iterations. Because our simulated data is quite small, we did not perform any parameter optimisation and thus set the burning period.

7.3 Results

Scenario 1 consists of a change in Topic 4 when a paragraph is added. Figure 5(a) shows that the model detects the topic because the evolution of T4 has a value of 0 at all revisions except at revision R6, where its distribution spikes to a bit more than 20%. We checked the corresponding revision, especially the added paragraph, and found that the paragraph has high membership in this topic and low membership in all other topics. In fact, except this paragraph, there are no paragraph having a non-zero membership in this topic.

Figure 5(b) shows the discovered topic evolutions for Scenario 2. The model indeed captured all three changes of the topic evolutions.

Based on the simulated authors: a1-a5, we also evaluate the technique used in constructing the topic-based collaboration networks. We correctly obtain a network diagram showing five nodes and two links: the first link is between a2 and a5 who work on P2 either alone or both at the same time, and the second link is between a1 and a4 who together add and delete P4.

From the evaluation above, we can conclude that DiffLDA can be used discovering topic evolutions for writing processes. In addition, the topic-based collaboration networks showing authors who write about the same topics can be constructed correctly as described above. In the next section, we will illustrate the applicability of our techniques using the real documents in the real learning environment.

8. Case Study

We conducted a case study in a semester-long graduate course called “Foundation of Learning Science” at the Faculty of Education and Social Work, University of Sydney in 2012. The study aimed at deploying our techniques within a course and exploring how the visualisations were used for understanding the writing processes of documents written by students.

8.1 Data

There were 22 students in the course. The course was structured in the following way. Every two weeks, students were divided into five different groups of four to five students each. During each fortnight, groups were required to write about a topic, which vary each fortnight, in a jointly authored document of approximately 3000 words. For this study, a writing duration of each fortnight is called a cycle. This writing component of this course lasts for 12 weeks. Thus, there are six cycles. Throughout the semester, therefore each student was asked to write collaboratively about six documents. At the end of the semester, there were 30 documents
In total to be analysed. All documents were assessed and graded as Pass (P), Credit (C), Distinction (D), and High Distinction (HD).

During the two weeks of writing about an assigned topic, individual students in each group were assigned reading materials. There were six readings per group. Students were encouraged to incorporate ideas and concepts learned in the class lectures and reading materials in their writing tasks. For every document, students had to make a plan for their writing tasks and discuss it among group members during the first week.

Each document comprised of two sections: In the first section, A, students were required to write about their assigned reading materials. They had to describe the main ideas of the articles they read and show evidence that they were grappling with the ideas, that they could articulate difficult concepts, or put into context. They also had to provide evidence of critical thinking in this section. In the second section, B, students were required to identify relationships between the reading materials of this cycle and the ones from the previous cycle. Students had to identify the “big ideas” that the readings can be seen as a contribution to.

8.2 Hypotheses

Based on the task description mentioned above, we explored two hypotheses in our analysis. First, individual students develop their own ideas and topics from their assigned readings by writing several paragraphs to explain their ideas and show evidence of their understanding in Section A. Therefore, we would expect that each of these individual paragraphs would be mainly developed by one student only. Second, student collaborate a lot to develop Section B in order to relate their idea developed from the readings. We would expect paragraphs in this section to be edited by several group members.

For each topic evolution chart (of each document), the creation and development of main topics are examined. In particular, topics in Section B will emerge and developed later than topics in Section A or vice versa. We expected the former scenario will be more likely because students may start their writing tasks and developed their idea in Section A based on their assign reading materials. They then work with others to develop idea in Section B.

In term of topic based author collaboration, it is obvious to expect that for each group (each topic-based collaboration network) there is at least one link connecting two nodes. This is because at least two students collaborated and wrote about the same topics in Section B as explained above. This link, if exists, may be a link connecting a group coordinator (node) to other team member (node) depending on the nature of text edits performed by individual group coordinators. If a group coordinator only edits (e.g. performs surface changes), there will not be any links connecting the coordinator to the other group members. However, if the coordinator revises (e.g. elaborates on topics developed by other group members), there will be a link connecting the coordinator to the others. This is not a strong requirement and is quite difficult to check since a group coordinator’s responsibility is to assign writing tasks to individual group members and make sure the assigned tasks are progressing according to the plan. Thus, the coordinator may not spend time collaborating and writing about the same topics with other group members.

In order to test our hypotheses mentioned above and gain insights on how students developed their documents, we use the revision maps, topic evolution charts, and topic-based collaboration networks, as described below.

8.3 Analysis

Among the six fortnightly cycles of writing, we randomly select the third cycle for our analysis. There were five groups of students in this cycle, thus five documents. After downloading all the revisions of these documents, we used the text comparison utility to identify text edits performed to produce these revisions. As a result, we obtained delta documents containing the added and deleted paragraphs.

Table 3 summarises the numbers of revisions, delta documents, vocabularies, delta topics, authors per revision, and final marks of 5 documents.

<table>
<thead>
<tr>
<th>Group</th>
<th>#revisions</th>
<th>#delta documents</th>
<th>#vocabularies</th>
<th>#total authors</th>
<th># inferred topics</th>
<th>Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>49</td>
<td>73</td>
<td>821</td>
<td>4</td>
<td>10</td>
<td>P</td>
</tr>
<tr>
<td>02</td>
<td>61</td>
<td>85</td>
<td>1040</td>
<td>5</td>
<td>11</td>
<td>P</td>
</tr>
<tr>
<td>03</td>
<td>144</td>
<td>229</td>
<td>1056</td>
<td>5</td>
<td>32</td>
<td>P</td>
</tr>
<tr>
<td>04</td>
<td>36</td>
<td>47</td>
<td>640</td>
<td>4</td>
<td>9</td>
<td>D</td>
</tr>
<tr>
<td>05</td>
<td>46</td>
<td>67</td>
<td>844</td>
<td>4</td>
<td>11</td>
<td>C</td>
</tr>
</tbody>
</table>

Table 3 summarises the numbers of revisions, delta documents, vocabularies (unique words), and authors of the five documents. The table also shows the final grades. We can see that the number of revisions varies from 144 for Group 03 (receiving Pass) to 36 for Group 04 (receiving Distinction). Group 4 which received the highest mark among the five groups actually produced less number of revisions.

8.3.1 Revision Maps

After identifying text edits made on the five documents as described in Section 5, we created the revision maps for each document. Using the revision maps of individual documents, we can observe how individual paragraphs of the two sections (A and B) have been created and have evolved during the process of writing. We use the five questions presented in Section 3 to guide our analysis.

Particularly, we perform the analysis on the five documents of the third cycle. From the five revision maps, we see that students performed approximately equal amount of text edits in the two sections. Secondly, as expected, in all five documents, Section A was created before Section B. In fact, we see that a lot of text edits happen in Section A at the beginning of the writing process and a lot of text edits produced in Section B towards the end of the writing, suggesting that most students spent their time writing in the beginning and rush their writing toward the end. Thirdly, in all five documents, more than 50% paragraphs were created and changed during the same writing sessions or days, revealing that most students prefer to do their writing in one session rather than drafting them sequentially over several days.

We then looked at the authorship of the edits made. In all five documents, most of the paragraphs in Section A were edited and revised by one student. For Section B, many paragraphs were edited by more than one student. The number of paragraphs in Section B written by several students is more than 10 for Group 11, 6 for Group 2, 4 for Group 3, 9 for Group 4, and 9 for Group 5. This suggests that most students collaboratively wrote Section B as we expected.
Table 4. After that, we applied the technique described in Section 5 to extract topics and create topic evolution maps.

Figure 6 shows the topic evolution chart of some topics of Document 2. There are 11 topics for this document. The topic evolution chart only depicts three topics: T3, T4, and T9. The top 10 words of the three topics are also shown below the figure. From the topic evolution chart, we gain insights on how topics have been developed during students’ writing. Particularly, T4 is about the instruction and explanation of the assignment that appears since the beginning of the document and decreases it assignment over time. Unlike T4, T3 is about a reading material related to the work of Hamilton about a “theory of personalized learning communities”. Students wrote about this topic to reflect on the topic. The topic spikes up at the third revision. On the other hand, T9 arrives later than the two topics because it is for Section B of the document. The topic is about “teacher’s recognition of their learners’ cognitive and motivational potential”. Although we can learn how topic evolve during writing, we also would like to know if students wrote about the same topics over time, in which we will perform the analysis base on the topic-based collaboration networks.

8.3.3 Topic-Based Collaboration Networks

Using the technique described in Section 6, we obtain the networks shown in Figure 7.

In all five groups, the coordinators have at least one link to group members. In other words, students who coordinated their groups worked with other group member on the same topics to develop their documents. In fact, the group coordinator worked on the same topics with all group members for all groups, except Group 2 and 4.

All of the networks except Group 4 are mostly connected. In some groups, especially Group 1, all students wrote on the same topics. Let us turn our attention on Group 4. Although there are four students in this group, the revision history of Group 4 shows that only three (i.e. a1, a2, and a3 shown in the figure) involved in developing the document. There were 36 revisions for this document. After checking the revision map of Group 4, we realize that a2 and a3 were only involved in 6 revisions each. Four revisions edited by a2 were also edited by a1, the group coordinator. After checking with the revision maps, we found that the four revisions were for paragraphs in Section B. In fact, most of the revisions were produced only by the coordinator a1. Nevertheless, the group managed to score a high grade.

9. DISCUSSION AND CONCLUSION

We have presented a case study with real documents written by graduate students and illustrated the use of our visualisations to analyse the students’ writing processes, since simple statistics and access to the final documents did not provide information about the writing process. The revision map allowed us to visualise text edits made by students at the paragraph level overtime, the topic evolution chart showed how topics evolve during students’ writing and the topic-based collaboration network showed which students wrote about the same topics during their writing.

This is a first step to gain some understanding about how students worked and created their collaborative document. Our aim is to support this collaborative writing process by providing visualisation as feedback to students about that process. This is to provide an awareness of the group’s writing activities to individual students so that they can perform their collaborative writing tasks more efficiently and effectively. The support can also be tools for teachers to monitor groups effectively and detect problems early.
10. ACKNOWLEDGMENTS
This project has been funded by Australian Research Council DP0986873 and a Google Research Award. We would like to thank Anindito Aditomo for setting up and administrating the course in the case study.

11. REFERENCES


