Analysis and Visualization of Temporal Changes in Bloggers' Activities and Interests

Masahiko Itoh* Naoki Yoshinaga† Masashi Toyoda‡ Masaru Kitsuregawa§

Institute of Industrial Science, University of Tokyo

ABSTRACT

Social media such as blogs and microblogs enable users to easily and rapidly publish information on their personal activities and interests. They are considered to provide valuable information from the viewpoints of sociology, linguistics, and marketing. This paper proposes a novel system for analyzing temporal changes in the activities and interests of bloggers through a 3D visualization of phrase dependency structures in sentences. We first extract events that represent bloggers' activities and interests through analyzing the phrase dependencies of sentences in a blog archive. We roughly categorize the retrieved events according to the thematic roles (such as the experiencer, agent, and location) of the noun within the events, and then store them in a dependency database so that we can retrieve events that involve a given topic. Second, we present a 3D visualization framework for exploring temporal changes in events related to a topic. Our framework enables users to find events about a topic that appear within a specific timing, and drill down details of the events. It also enables users to compare events with different timings and/or on multiple topics. Moreover, it allows them to observe an overview of temporal changes in sets of events, and long-term changes in the frequency of events to assist users in finding trends. We implement the proposed system on our own five-year blog archive that focused on Japanese, and we report the usefulness of our system by using various examples.

Index Terms: D.2.2 [Software engineering]: Design Tools and Techniques—User interfaces; I.2.7 [Artificial Intelligence]: Natural Language Processing—Text analysis

1 INTRODUCTION

Social media such as blogs and microblogs enable users to easily and rapidly publish information on their personal activities and interests. They have been considered to provide important diachronic data on marketing, linguistics, and sociology [17]. Such data are, however, too diverse and enormous to manually analyze for individual purposes, which implies a growing demand for a system that can analyze long-term changes in activities and interests of people from multiple viewpoints, and that will enable us to overview and explore the analyzed results.

Existing systems for analyzing the blogosphere [2, 3] focus on topic detection based on the emergence of topical keywords or phrases. After they have found a hot topic, users often want to obtain more detailed information that explains how and in which context the given topic has become popular. Although some systems provide a supplementary set of words related to the given topic (e.g., 'Prime Minister Hatoyama' often occurs with 'Obama', 'Washington D.C.', and 'Loopy') [3], these words co-occur with the topic in various contexts (e.g., 'Hatoyama meets Obama in Washington D.C.' or 'Hatoyama is called Loopy') and it is quite onerous for users to recover the individual contexts from the mixture of co-occurring words. The contexts between a topic and a particular related word, especially in diachronic data, can change over time (e.g., 'Hatoyama welcomes Obama in Tokyo'), which further increases the complexity of recognizing contexts. Users are therefore forced to read individual posts on the topic to understand the contexts between the topic and related words, while summarizing surrounding but valuable information that may be buried in co-occurring words.

One of the most effective ways of understanding these contexts is by analyzing the dependency structures of sentences on a topic. When the topic is given by a word, we can semantically connect the topic with related words via verbs through dependency analysis. The context can then be represented by the structural relationships between these words.

The challenges here are how to summarize the dependency structures of all sentences on the topic during the long term, and how to enable users to investigate the evolution of the topic. Four visual and interactive supports are required to investigate and explore temporal dependency structures. These are:

- finding the dependency structures for a given topic to understand contexts (such as "Hatoyama meets Obama") and the frequency for each relation (word pair) in the structures within a specific timing, and tracking their temporal changes.
- exploring dependency in detail to discover surrounding valuable information (such as "Hatoyama meets Obama *in Washington, D.C*").
- comparing dependency structures with different timings and different topics.
- finding changes in the frequencies of each dependency relation to observe timing of bursting or periodicity of appearances.

This paper proposes a 3D system for exploring temporal changes in bloggers' activities and interests using phrase dependency structures. We utilize a third dimension in a 3D environment as a timeline and integrate each function using interactive components in the 3D environment. Each function is mapped to an intuitive operation for the 3D components. The proposed 3D visualization framework supports the previously mentioned requirements by providing four fundamental functionalities for:

- visualizing dependency structures as a unified tree representation in a slidable 2D plane on the timeline in the 3D environment. Sliding operation for the 2D plane along the timeline indicates changes in the structure and frequencies of dependency relations.
- interactively navigating to the detailed dependency by expanding operations in the tree representation.
- providing multiple 2D planes to compare multiple structures side by side for different timings and topics.
- visualizing changes in the frequencies of each dependency relation as a line of bubbles.

*e-mail:imash@tkl.iis.u-tokyo.ac.jp
†e-mail:ynaga@tkl.iis.u-tokyo.ac.jp
‡e-mail:toyoda@tkl.iis.u-tokyo.ac.jp
§e-mail:kitsure@tkl.iis.u-tokyo.ac.jp

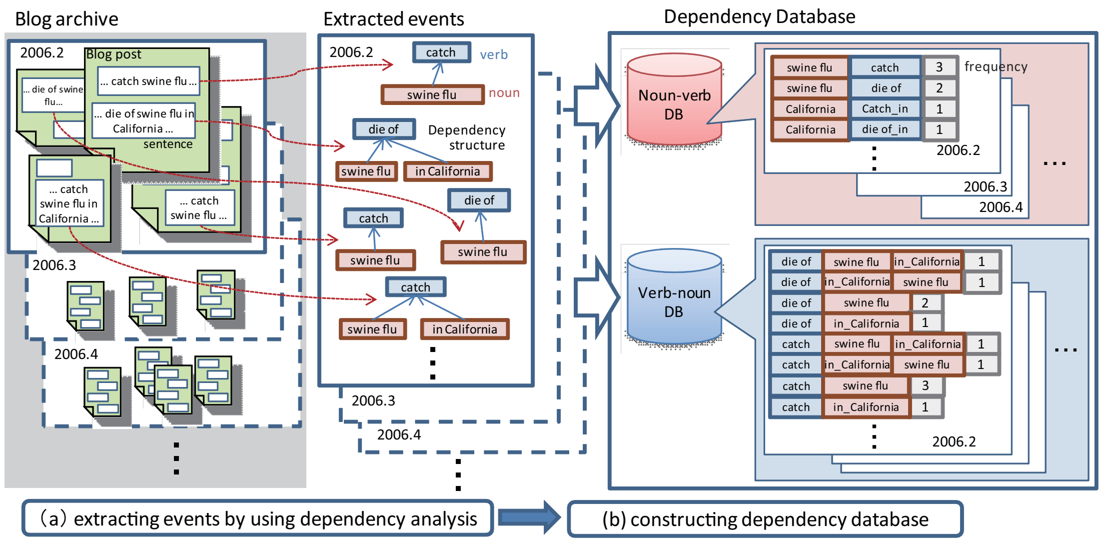


Figure 1: Overview for constructing event databases from blog posts in blog archive by using dependency analysis

The design of the integrated functions using interactive 3D components and 3D environments naturally supports overview+detail exploration mechanisms that enable users to seamlessly change overviews and detailed views while retaining their mental models. It allows users not only to overview one or multiple snapshots of structures for different aspects on the timeline and long-term changes in frequencies, but also to seamlessly change time stamps for snapshots and go to the details of the dependency relations.

The two major contributions of this research are that it is a novel approach for:

- summarizing the contexts of a given topic by aggregating dependency structures including the topics that are extracted from texts
- visualizing the aggregated set of dependency structures and exploring them in detail with temporal changes through interactive and seamless overview+detail operations

We implement the proposed system on our own five-year blog archive that focused on Japanese, and here we present some case studies with exploratory scenarios.

In what follows, we first introduce a dependency structure for a sentence and the definition of events we use in this study, and we then introduce a dependency database in Section 2. We next describe our 3D visualization framework in Section 3. We then present some case studies in Section 4. We give an outline of related work in Section 5 and the paper is concluded with an explanation on future work in Section 6.

2 Dependency Analysis and Database

We construct a dependency database that consists of *events*, each of which is defined as a set of dependency relations on a particular verb and is regarded as an essential context for a topic, from the blog archive. After this, we regard a set of dependency relations between a verb and its dependents (nouns) as a target event. The verb represents an action or a state of being (in which bloggers may have an interest), and the nouns modifying the verb further characterize the action/state. We can therefore understand details of a particular event anchored by a verb, by collecting a set of dependency relations between the verb and its dependents. Words/phrases are in a certain syntactic relation, and noisy words in retrieved results can be naturally reduced.

Figure 1 overviews the construction of the dependency database. We have collected Japanese blog feeds since February 2006. We first analyze the sentences in blog posts every month in the blog archive by using a dependency parser to collect dependency relations that represent events. We utilize the most efficient dependency parser currently available for Japanese[1] to parse them. It took around two days with a single server with two 3.2-GHz Quad-Core Intel Xeon CPUs to analyze over two million blogs with a total of over 350 million posts. We next build a dependency database for each month. The dependency database allows us to search events that involve a given topic (noun).

Our approach does not focus on real-time analysis of blogs, but long-term analysis of changes in events described in the large blog archive. We therefore construct the dependency database to immediately access analytical results for specified time stamps.

2.1 Dependency Structure

We first introduce a dependency structure for a sentence and the definition of events we use in this study. A *dependency structure* of a sentence represents the relation between a head (word) and its dependents. Each word in a sentence has a single head in the definition, with the exception of the head for the entire sentence (usually a verb), and the dependency relations thereby form a tree structure called a dependency structure. Figures 2 and 3 outline the dependency structures for Japanese sentences corresponding to "A 13-year-old girl newly caught swine flu" and "A child died of swine flu in California". These sentences include four/five base phrases (boxed in Figures 2 and 3), and each edge in the dependency structure represents a single dependency relation between each base phrase (a child node (dependent) modifies a parent node (head)). After this, we will assume that a dependency relation is defined over base phrases, unless otherwise noted.

We regard a set of dependency relations between a verb (head) and its dependents (nouns) as a target event (the dependency relations with thick lines in Figures 2 and 3) in this paper.

We categorize each dependency relation in the events based on a *thematic role* (e.g., an agent, experiencer, theme, and location) of the noun for the verb to summarize events. In Japanese, we can exploit *case particles* attached to the nouns ('-ga' or '-ni' in Figure 2 and '-ga' or '-de' in Figure 3) to guess their thematic roles. Here, we briefly introduce five case particles, '-ga', '-wo', '-ni', '-to', and '-de', which occur frequently and are related to important thematic roles that characterize events. Nouns with case particle '-ga' represent agents or experiencers of the action signified by the verb (indicated in English by 'subj(ect)' in Figures 2 and 3). Nouns

[1] http://www.tkl.iis.u-tokyo.ac.jp/˜ynaga/jdepp/

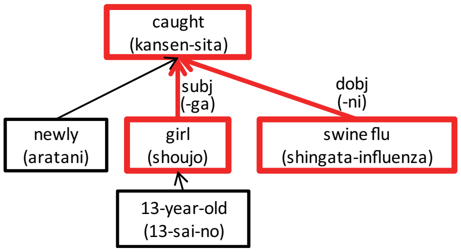


Figure 2: Dependency structure for Japanese sentence: "A 13-year-old girl newly caught swine flu (Aratani 13-sai-no shoujo-ga shingata-influenza-ni kansensita)". Dependency relations with a thick line form an event

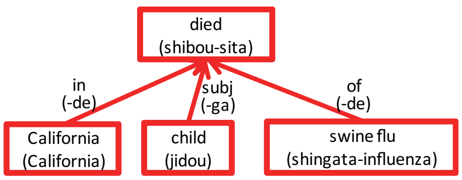


Figure 3: Dependency structure for Japanese sentence: "A child died of swine flu in California (California-de jidou-ga shingata-influenza-de shibou-sita)"

with case particles '-wo' and '-to' represent themes of the action, nouns with case particle '-ni' represent goals, directions, recipients, or the time the action occurred, nouns with case particle '-de' represent instruments or locations of the action (indicated by d(irect) obj or prepositions corresponding to case particles in Figures 2 and 3). Note that for languages without case particles (e.g., English), we can use a dependency parser that outputs typed dependency relations[2] to find the thematic roles of the nouns.

2.2 Dependency Database

We next built a dependency database having a dependency relation between a head (verb) and its dependants (nouns with case particles). We selected nouns/verbs appearing at least 100 times over the blog archive. The resulting number of nouns/verbs was 1,498,345/105,987. We then collected dependency relations whose constituents were composed only of these words. The total number of collected dependency relations was 1,187,544,311.

The following two types of dependency databases are built per month by using a trie. The input to the two databases expresses constituents of events and the output is the frequency of the events that include the given constituents.

Noun-verb DB: This database is constructed to support NOUN queries to search events (verbs). The NOUN query retrieves the verbs that take the given noun as their argument (dependant). For example, giving 'shingata-influenza (swine flu)' as a NOUN query, the database returns 'kansen-suru (catch)'[3] with the case particle '-ni' (Figure 2) and 'shibou-suru (die)' with the case particle '-de' (Figure 3). This retrieval can be efficiently carried out by storing keys that concatenate a noun, a case particle, and a verb in this order (e.g., 'shingata-influenza -ni kansen-suru') in the trie and by traversing the sub-tree rooted by the given noun.

Verb-noun DB: This database is constructed to support VERB and EVENT queries to collect events. These two types of queries retrieve the nouns that modify a given verb (head) or the other nouns that modify the verb in a given event. For example, given 'kansen-suru (catch)' as a VERB query, the database returns their dependents 'shoujo-ga (-ga, girl)' and 'shingata-influenza-ni (-ni, swine flu)' in Figure 2, while given 'jidou-ga shibou-suru (a child dies)' as an EVENT query, it returns further contexts for this (partial) event 'California-de (in California)' in Figure 3. This retrieval can be efficiently undertaken by storing keys that concatenate a verb and a sequence of dependants ((case particle and noun) pairs)[4] in this order to the trie and by traversing the trie as in NOUN query.

[2] For example, multilingual maltparser (http://maltparser.org/) returns Stanford typed dependencies:
http://nlp.stanford.edu/software/dependencies_manual.pdf

[3] All the verbs in the events are lemmatized in the databases: 'kansen-sita (caught)' $\mapsto$ 'kansen-suru' (catch).

3 3D Visualization

We provide a 3D visualization framework to explore changes in events, which are retrieved from the dependency database described in Section 2, related to a particular topic every month. Our framework provides the following functions for: (i) visualizing snapshots of events per month, (ii) interactively drilling down details about specified events, (iii) comparing multiple timings and topics to find differences and/or similarities, and (iv) visualizing long-term changes in the frequency of particular events to find the timing of bursting or periodicity of events that appeared. To accomplish the framework, we utilize two 3D components, i.e., TimeSlice [15] and TimeFlux, and extend their functions.

A TimeSlice enables us to visualize temporal changes in the structure of relations by using a tree visualization in 3D space, and to drill down the surrounding contexts of the event by expanding nodes in the tree. It is a 2D plane for visualizing a snapshot of an event tree that represents events with a specified timing. The TimeSlice can be dragged along time, which is represented by the third dimension. We can seamlessly change the positions of the TimeSlice along the timeline. Such manipulation generates an animated temporal change in the tree structure. A user can incrementally add new TimeSlices on the timeline side by side to compare different months (Figure 4). Moreover, he/she can add TimeSlices to represent different topics at the top and bottom (Figure 4). TimeFlux enables us to visualize temporal changes in the frequencies of particular events every month (Figure 4).

Our framework helps users at the three levels of tasks described by Chi et al. [12] that are 1) local: finding specific content, 2) comparison: comparing information in two places, and 3) global: discovering trends. A TimeSlice allows users to find the appearance and/or disappearance of specific events through a visualizing set of events and their structure, and through using animation to show changes in them. Multiple TimeSlices enable users to compare multiple situations. Moreover, multiple TimeSlices provide a global view of the structure of events belonging to different timings and aspects by using 3D space. TimeFluxes visualize an overview of changes in the frequencies of events, and assist users in finding trends.

The remainder of this section first explains how events about a particular topic on a TimeSlice are represented, introduces an interface to explore more specific events, and provides details on how different topics and timings are visualized by using multiple TimeSlices. We finally provide details on visualizing temporal changes in the frequency of events by using TimeFlux.

3.1 Visualizing Events on a TimeSlice

A TimeSlice visualizes a retrieved set of events that are represented as a unified tree structure. It only displays events belonging to a selected month and can be dragged along a timeline to change visualized events. When a user inputs a topic noun, the system retrieves events from the noun-verb DB. Each event is represented as a triplet (a noun, a case particle, and a verb) and a path between constituents of the triplet on a TimeSlice. A set of events is visualized as a tree

[4] Most verbs have at most four dependants in the blog feeds and we discard verbs with more than five dependants (nouns with case particles).

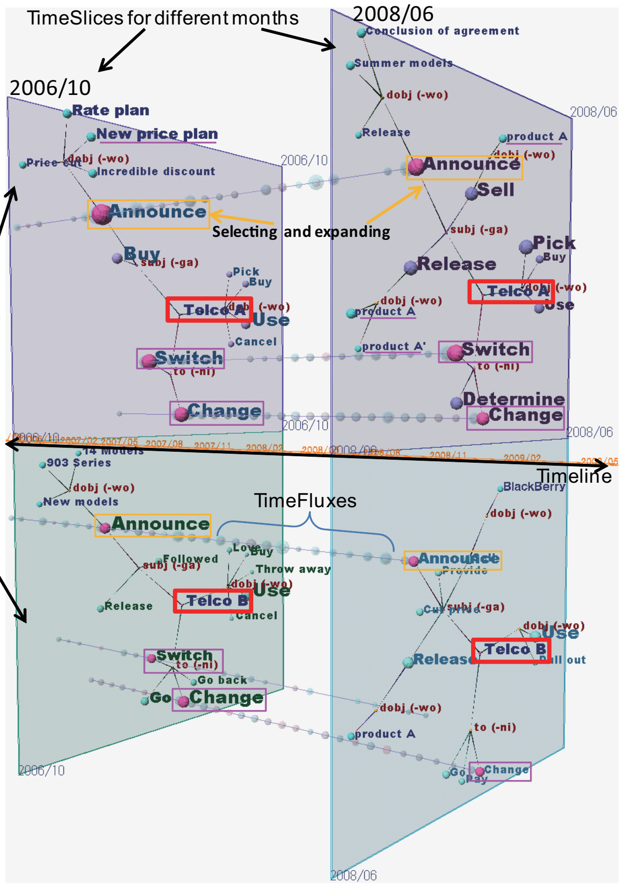


Figure 4: Multiple TimeSlices for comparing events in different months and for comparing events on different topics

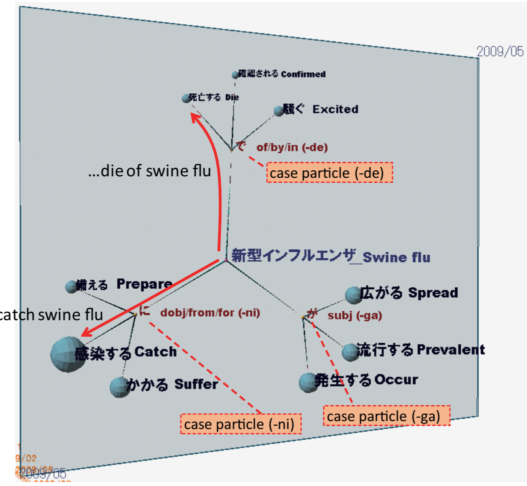
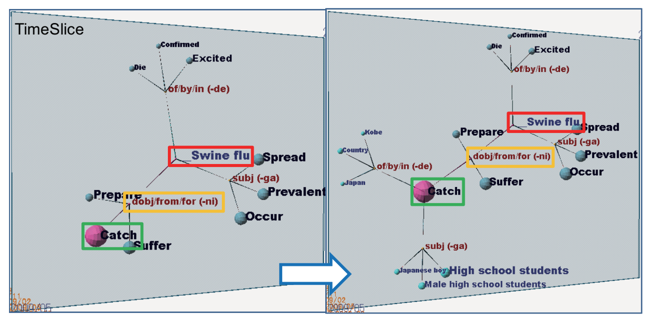


Figure 5: Visualizing set of events by using tree on TimeSlice, which indicates both original in Japanese and translated words in English



(a) Visualizing events related to "swine flu" (b) Expanding nodes to show detail of the event "catch swine flu"

Figure 6: Visualization of detailed information on selected event

as depicted in Figure 5. First, a TimeSlice places a query noun as a topic at the center of the tree, and then displays case particles '-wo', '-to', '-ni', '-ga', and '-de' as child nodes of the noun. The system then arranges verbs that take the noun with a case particle as subtrees of that case particle. A TimeSlice only displays top-n frequent events related to each case particle. The system in Figure 5 visualizes events about 'swine flu' such as "die of swine flu" and "catch swine flu". In this case, the topical noun 'swine flu' is centered, and case particles '-de' and '-ni' are around the noun 'swine flu'. Verbs such as 'die' and 'catch' are arranged around the topic noun with case particles, i.e., "swine flu -de" or "swine flu -ni". We can filter out case particles that the user does not want to see. We filter out '-wo' and '-to' in Figure 5.

We roughly translate Japanese nouns/verbs into English words for illustrative purposes in the following figures. We translate each case particle in Japanese with the corresponding typed dependencies shown in Figures 2, 3, and 5. The 'ga' and 'wo' are mapped to subj (subject) and dobj (direct object), and the other case particles are mapped to one or more prepositional typed dependencies such as 'in', 'of', and 'from'. Due to ambiguities in translation between case particles and typed dependencies, the figures specify all possible typed dependencies for each case particle (individual usages of case particles can be a unique typed dependency).

A TimeSlice changes the size of nodes representing verbs according to the frequency of events in the selected month. The size of nodes is calculated as $r\sqrt[3]{c_i(t)}$ (r: constant, $c_i(t)$: the frequency of event i in month t) to prevent each node from becoming huge. Nodes have been colored differently according to constituent types (nouns, case particles, or verbs) so that they can be easily distinguished.

3.2 Interface for Exploration of Events

We provide an interface for drilling down detailed information after events on a particular topic have been visualized on a TimeSlice. First, we select a node for a verb. This means that we select an event consisting of the topic noun (centered node), a case particle, and the selected verb. Second, the system creates an event query from the selected event, and retrieves events that subsume the (partial) event from the verb-noun DB. Next, the system creates nodes and edges, and expands them at a lower hierarchy of the selected node (verb).

We select the event 'catch swine flu' in Figure 6 by selecting the verb 'catch'. Then, the system shows more specific events such as "high school students catch swine flu" and "catch swine flu in Kobe", and then visualizes them as child nodes of the selected event.

If there are the same nodes in different TimeSlices that have different timings or are on different topics, this expansion operation is also propagated to the other nodes on the different TimeSlices to enable TimeSlices in the same situation to be easily compared. There are four nodes 'announce' in Figure 4. If we select one of these, e.g., 'announce' related to 'Telco A' on the TimeSlice in June 2008, and expand it, the node 'announce' 'Telco B' is automatically expanded. The nodes on TimeSlices in October 2006 are automatically expanded in the same way.

3.3 Comparison of Multiple Aspects

3.3.1 Comparison of Multiple Topics

We utilize the *split view* proposed by [15] to visualize two sets of events in different TimeSlices (the top and bottom of Figure 4). The positions of the different TimeSlices along the timeline are synchronized with one another to enable changes in events to be easily compared on them. Users can also add sets of TimeSlices having different time stamps to the timeline.

The system in the split view treats the central noun, the same case particles just under the center nodes or under the same events, and the same verbs (or nouns) under the same paths on different TimeSlices as the same nodes and arranges them in the same position on different TimeSlices. Moreover, when we select nodes on one TimeSlice, the system automatically selects and highlights the same nodes on different TimeSlices. It helps users to understand differences and/or similarities between events related to different topics.

3.3.2 Comparison of Multiple Timings

A TimeSlice only displays events in a specified month, and can be dragged along the timeline to change visualized events. In addition, our system allows us to interactively add new TimeSlices to different positions along the timeline, and to compare multiple TimeSlices in a 3D environment (Figure 9). This enables us to compare events in different months.

The same events belonging to different timings, which are visualized in different TimeSlices, have the same position to enable the similarities and differences between different months to be easily understood.

3.3.3 Layout of nodes in multiple TimeSlices

Multiple TimeSlices visualize events related to different topics and with different timings. They arrange the same nodes in different TimeSlices in the same position. For this purpose, we extend the method proposed by Diehl et al.[6]. The system constructs one aggregated tree from all visible events in the specified TimeSlices, and then calculates the positions of all visible nodes. Each TimeSlice only displays the necessary nodes for a selected month and a specified topic. TimeSlices independently calculate the positions of nodes in the different topics except for the same nodes mentioned in Section 3.3.1.

We adopt automatic and dynamic graph layout algorithms based on a force-directed model [8] to visualize the tree structures. It considers a graph (in this case a tree) to be a physical system. Attractive forces in the force-directed model are exerted on all pairs of connected nodes, and repulsive forces are exerted on all pairs of nodes.

We can select and drag nodes and edges to interactively check details on the relationship between nodes. Moreover, we can zoom and pan a canvas to interactively change a region being focused on. The positions of nodes on different TimeSlices can be completely synchronized with one another even if users drag a node. Interactions such as selecting and highlighting nodes, and expanding nodes to show detailed information are propagated to other TimeSlices. Panning and zooming operations are also propagated to other TimeSlices.

3.3.4 Parallel View

Utilizing a third dimension to represent a timeline in 3D space helps users to observe an overview of changes in the structure of events from a global aspect, and to understand the time intervals of changes [12]. Moreover, by using TimeFlux together in 3D space, users can observe an overview of changes in the frequencies of events.

However, placing TimeSlices side by side in 3D Space causes problems with occluded TimeSlices [5]. Distorted perspective causes poorer legibility of elements on each TimeSlice [12, 5]. Perspective also makes TimeSlices in the center occupy a smaller space than TimeSlices on the sides. Users occasionally see the front side of TimeSlices on the left of the screen, and the backside of TimeSlices on the right of the screen [12].

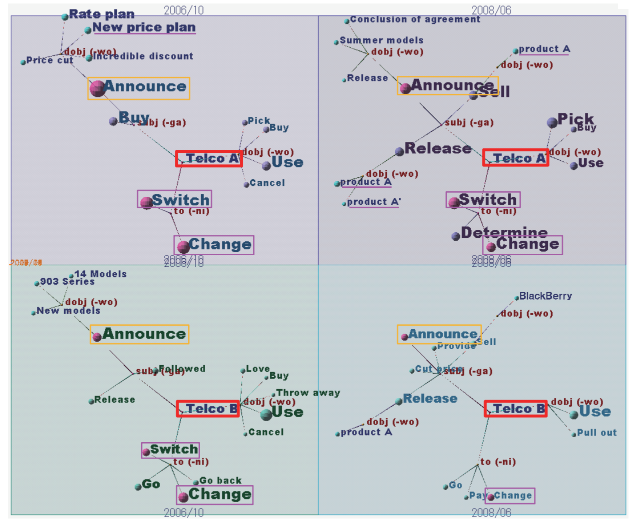


Figure 7: Parallel view

To compare events in different TimeSlices in detail and to solve these problems, our framework utilizes mechanisms for a parallel view in a 3D environment (Figure 7) [15]. It enables us to seamlessly change the normal 3D view to a parallel view, and to interactively explore information through different aspects using multiple views. We can see an overview by using the normal 3D view and find areas of interest, and then change the view mode to a parallel view to explore the area in detail. Figure 7 shows a parallel view that has been changed from the normal view given in Figure 4.

To achieve this, the system can automatically slide TimeSlices in the direction of depth, change the eye position to see TimeSlices from the front, and then change the projection mode. We can obtain a parallel view after that. We normally use perspective projection in 3D environments. The same nodes in different TimeSlices are displayed in different positions because of perspective in perspective projection. We organize an orthogonal projection mode to solve such problems, where the same nodes in different TimeSlices completely overlap one another.

3.4 Visualizing Temporal Changes in Frequency of Events

A TimeFlux is a line of spheres to visualize changes in the amount of information such as the number of events within a given period of time (Figure 4). They enable us to intuitively observe when a selected event attracted attention such as a bursting point, and the periodicity of its trends. We can show multiple TimeFluxes by selecting nodes. These allow us to observe differences in trends between other events.

For example, the TimeFluxes at the top of Figure 8 indicate that events related to ‘catch swine flu’ explosively increase after May 2009. The TimeFluxes at the bottom of Figure 8, on the other hand, indicate that events related to ‘catch seasonal flu’ increase and decrease annually.

Before we construct the dependency database, we identified the gender of bloggers by using a simple heuristic that determines their gender according to the number of clue expressions (in the blog) that indicate either gender; the clue expressions include first-person pronouns and sentence-ending particles specific to each gender [13]. The dependency database independently counts the number of events for each gender. TimeFluxes enable us to visualize and compare differences in tendencies between females and males by using two different colors. We normally assign red to females and blue to males. We can observe the phenomenon of change in the gender ratio in Figure 9.

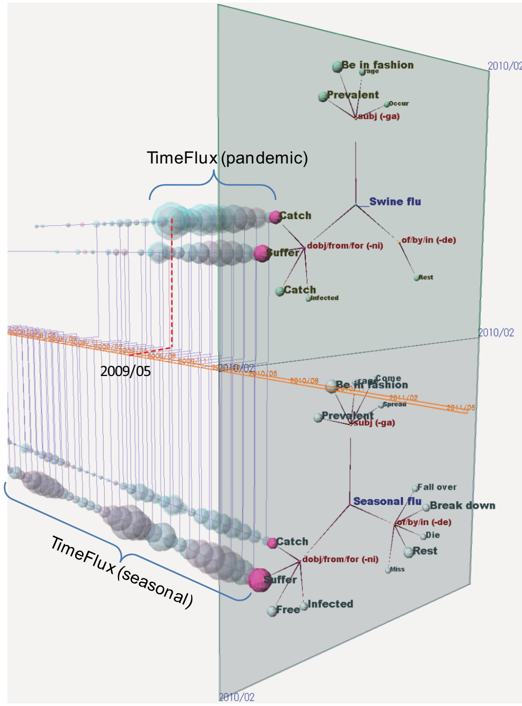


Figure 8: Comparison of temporal changes in frequency between events on different topics by using multiple TimeFluxes

4 CASE STUDIES

We have thus far described the functionality of our 3D visualization system through sociological analysis of the 2009 swine flu pandemic (Figures 5, 6, 8) as a case study. This case study demonstrated that our system could highlight when the pandemic on swine flu in Japan started, while providing the surrounding contexts (information on the first patients or places).

This section provides two further case studies that highlight the usefulness of our 3D system of visualization. The first presents the effects of marketing by two telephone companies in Japan that are described in Section 4.1. The second presents changes in trends in Japanese society related to chocolate around St. Valentine's Day that are described in Section 4.2.

These case studies clearly distinguish our 3D visualization system from the conventional systems for analyzing trends that only visualize co-occurring words. By using dependency relations instead of word co-occurrences, users can understand within what contexts (events specified by verbs) the target topic noun interacts with other topic nouns. Furthermore, since the dependency relation captures the thematic roles of topic nouns associated with the events (as described in Section 2), users can be aware of changes in the thematic roles of the target noun (Section 4.2). Observing these examples, readers will also notice that our system greatly reduces noisy words in retrieved results compared to inherently noisy word co-occurrences (mentioned previously in Section 2).

4.1 Comparing Marketing Effect of two Telcos

The mobile market in Japan is mature and is reaching saturation. Customer retention is currently a very critical issue for telephone companies. We compared the effects of marketing strategies by two rival telephone companies (Telcos for short) in Japan in this case study by focusing on a situation where customers switched from one Telco to another.

The upper TimeSlice in Figure 4 shows a topic for 'Telco A', while the lower one shows a topic for 'Telco B'[5]. We can recognize events related to 'change/switch to Telco A' are more popular than 'change/switch to Telco B' in most months by observing changes in the structure and frequencies for events. It is difficult for us to discover such situations from co-occurring words because they do not provide the "direction of changing/switching".

To start exploring possible reasons for such actions and plans, we first find times when events "change/switch to Telco A" are popular, and then observe events around them on the TimeSlice in detail. We then find TimeFluxes related to the events 'change/switch to Telco A' to observe times when such events become popular. We can see that there are some peaks in the events 'switch to Telco A' by observing the size of the spheres on the TimeFluxes.

We can simultaneously visualize other events related to the Telcos in detail on the TimeSlices to explore reasons why people want to switch. We place TimeSlices at the left on the position of timing in Figure 4 when the first peak is observed (October 2006), and TimeSlices at the right for the second peak (June 2008). We find that 'Telco A' announced or released something because the size of the 'announce' and 'release' nodes increased. It is onerous to manually identify these different events from words co-occurring with the topic, when we do not know which company is actually involved with what event.

We next expand these nodes to find details on announcements and products that were released, and we then find that they announced a 'new price plan' in the first peak, and released 'product A' in the second peak. However, although 'Telco B' also announced new products very frequently, there are few peaks related to events 'switch' for 'Telco B'.

These results mean that 'Telco A's' marketing activities had more impact in Japan than those by 'Telco B'.

4.2 Visualizing Trends on St. Valentine's Day

It is crucial for food companies to track customer trends in interests in buying products. Figure 9 visualizes the evolution of trends on the topic 'chocolate' in Japan.

First, we select the verb node 'give', and then find the TimeFlux for the event 'give chocolate'. We can see that there are peaks for the number of these events around St. Valentine's Day, which reflect this Japanese tradition[6]. The TimeSlice at the left in Figure 9 indicates events in February 2006, and the TimeSlice at the right indicates events in February 2009. We can see that the gender ratio for the number of events 'give chocolate' between two TimeSlices changes from '#female $>$ #male' to '#female $\leq$ #male'. Next, we check details on the events 'give chocolate' by expanding the node. There are only events "a woman/girl gives chocolates to a man/boy" in February 2006. However, we can find events "a boy gives chocolates to a woman/girl" in February 2009. These situations indicate that such a chocolate-giving tradition has recently undergone slight changes.

We should note that dependency analysis enables us to find changes in the thematic role of topic nouns (girls and boys) within the event ('give chocolate'), while the word collocation between boy, girl, and chocolate does not maintain their semantic relations.

5 RELATED WORK

5.1 Temporal analysis of word usage

Kehoe and Gee [16] used a heatmap technique to analyze collocational words around a given word. Collocational words are likely to include more noisy words when we increase the size of the collocation window.

There have been some studies on the storage and look-up interface of syntactic relations. Resnik and Elkiss [21] provided a phrase structure for a given sentence to help users build a structured query from it. Atterer and Schütze [1] broke down individual dependency

[5] Companies' names have been anonymized.

[6] Females normally give chocolates to males.

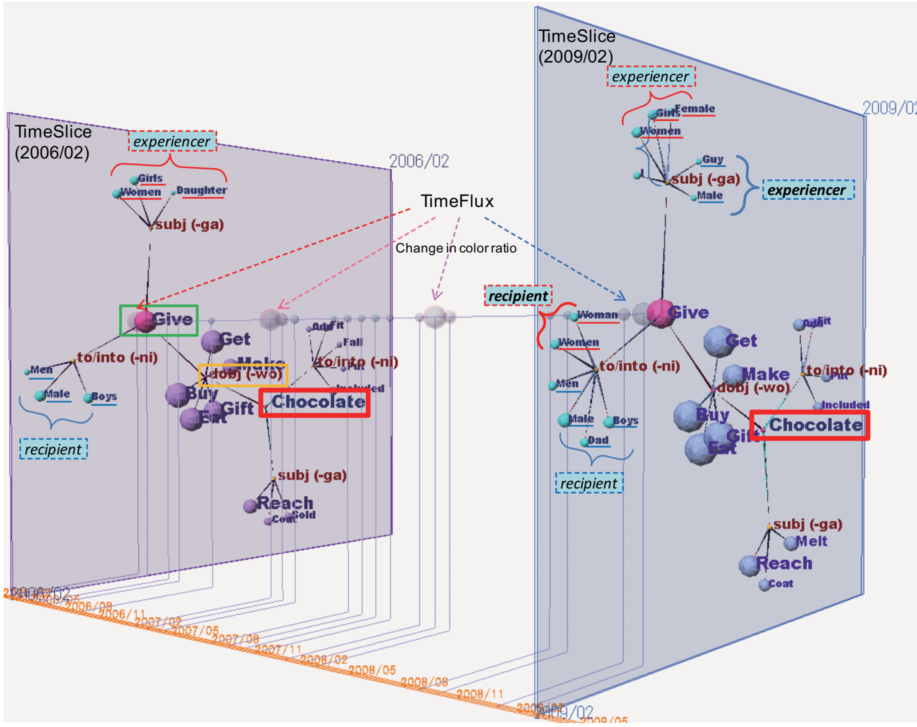


Figure 9: Visualizing changes in trends on St. Valentine's Day in Japan

relations to build inverted indices (from a relation to a sentence) on Lucene[7]. These studies only returned sentences that included all the dependency relations, while our database enabled a quick predictive search of collocated dependency relations by using a trie structure.

Inui et al. proposed a text-based experience mining system [14] that collected events related to experiences when people used a topic object (e.g., consumer products, public places, and social systems) and categorized the events according to tense, aspect, polarity, and modality. Our system provides temporal analysis of all events that involve a given topic (not limited to usable objects) to find trends in users' activities and interests on a given topic. Also, our visualization system provides novel functionality that drills down more specific events by recursively specifying the topics involved, which can greatly decrease the burden on users having to read individual sentences.

5.2 Visualization of temporal/structural evolution

Four types of methods have been proposed that have enabled the evolution of information structures to become visualized; however, these have their own pros and cons.

The first is using *animation* to dynamically display changes in structures [24, 19, 12]. Although this enables users to dynamically observe changes in structures, it reduces user recognition, because they lose the context in previous situations. Users occasionally miss where changes have occurred and when they have changed throughout the entire space.

The second is mapping a *timeline* to one of the axes in a 3D environment [12]. This enables users to observe global differences between multiple graphs. However, it is difficult to check local differences in detail. VisuaLinda [18] and Time-Tunnel [20] also use the combination of 3D space and a timeline. They effectively use 3D space to simultaneously represent two kinds of relations including a time relation.

The third is using multiple *tiled views* to display multiple Web graphs [24, 11]. Users can compare the differences between Web graphs in parallel, enabling them to comprehend global differences. However, it is difficult to intuitively understand time intervals between Web graphs. Users occasionally cannot determine how long it has taken for changes to have occurred.

The fourth is *overlaying* graphs for different time periods on one view [19, 4]. This is advantageous for comparing graphs in detail; however, it is difficult to display global changes in structures.

Although these studies have enabled users to dynamically observe changes in structures, they have not prepared the mechanisms for visualizing overviews of temporal changes in the number of appearances of elements at all timings.

ThemeRiver [10] provided methods of visualizing changes in the values of multiple attributes on a timeline. TIARA [25] simultaneously combined ThemeRiver with tag-clouds to visualize changes in elements consisting of topics. These techniques were useful for visualizing overviews of changes in values. However, it was difficult to visualize structural changes and to drill down elements to explore information in detail.

5.3 2.5D Representation

TimeSlices introduce multiple 2D planes into 3D space. Such methods are sometimes referred to as 2.5D representations. A 2.5D representation is mainly used for visualizing multiple situations in 3D environments such as visualizing different content [9, 7, 5, 23], visualizing time sequential changes [12, 4, 7], and visualizing different visual representations and/or models [23, 5, 9].

Erten et al. [7] proposed methods of visualizing series of related graphs, which shared all or parts of the same nodes, to compare them. Their method could treat time sequential graphs as simultaneous graphs. However, they did not provide an interface to support the exploration of temporal changes in graphs. It was also difficult to represent time sequential graphs for multiple topics.

VisLink [5] utilized multiple 2D planes to visualize various kinds of data sets with different representations. They provided inter-plane edges between 2D planes in 3D space to link the same data on different planes. They solved one of the limitations of coordi-

[7]http://lucene.apache.org/

nated multiple views (CMV) [22] that could not explicitly visualize the relationships between views. Each plane had hinge widgets to change the orientations of planes like those in books. It enabled users to avoid the problem with occluded planes, and the problems caused by distorted perspective. Although they did not describe methods of treating time sequential data sets, their book metaphor was useful for arranging multiple planes that were legible.

5.4 Coordinated Multiple Views

CMV [22] is one solution to visualize both temporal changes in event structures and the frequency of events.

We have already provided a table view as a Web application to visualize lists of events that are retrieved from the dependency database. Each column represents a month in this view. There are five rows that represent case particles, and each row includes verbs with top-n frequencies. We can move into another table that visualizes more detailed information on selected events by clicking on a verb in the table.

This view has various limitations and it is also difficult to simultaneously display both the current table and detailed tables on selected events to compare various contexts for events on the topic. It also cannot visualize temporal changes in frequency for events.

We may solve these limitations by providing linked multiple views that provide detailed tables of selected events, and bar charts that visualize temporal changes in frequency for selected events. However, we need too many windows to observe overviews and compare various situations. This causes problems with cluttered and occluded windows. We may also miss correspondence between windows because the relationships between them are not explicitly visualized.

6 Conclusion

We proposed a novel system to temporally analyze blog content based on dependency analysis and a 3D visualization framework to explore temporal changes in bloggers' activities and interests. The dependency database and 3D visualization system allowed us to search events that involved given topics, to visualize structural changes in a set of events every month to observe trends in these monthly events, and to visualize long-term changes in the frequency of events to find the timing for the bursting or periodicity of events that appeared. These also enabled us to interactively drill down each event in detail. We implemented the system on our own five-year blog archive, which consisted of over two million blogs in total for over 350 million posts. We focused on Japanese and demonstrated the effectiveness of our system with some case studies that represented novel results that could not be extracted by using conventional systems used for analyzing trends.

Our framework provided a generic framework for analyzing and visualizing temporal data, and the framework can therefore be applied to various kinds of temporal data to construct applications to explore such data. We have already generated an application for exploring changes in sentiment in the blogosphere by using a dependency database for adjectives instead of verbs. We have also already constructed an application using a data set collected from Twitter.

It is still difficult for casual social media users who only know vague keyword nouns to specify topic nouns. We plan to provide mechanisms to monitor overviews of huge information spaces such as those generated from various keywords and to automatically find areas of interest. This should help casual users of social media to explore interesting topics.

Acknowledgements

This work was supported by the *Multimedia Web Analysis Framework towards Development of Social Analysis Software* program of the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

[1] M. Atterer and H. Schütze. An inverted index for storing and retrieving grammatical dependencies. In *Proc. LREC 2008*, pages 62–65, 2008.
[2] BlogPulse. http://www.blogpulse.com/.
[3] BlogScope. http://www.blogscope.net/.
[4] U. Brandes and S. R. Corman. Visual Unrolling of Network Evolution and the Analysis of Dynamic Discourse. *Information Visualization*, 2(1):40–50, 2003.
[5] C. Collins and S. Carpendale. Vislink: revealing relationships amongst visualizations. *IEEE Trans Vis Comput Graph*, 2007:1192–1199.
[6] S. Diehl and C. Görg. Graphs, They Are Changing. In *Proc. GD 2002*, pages 23–30, 2002.
[7] C. Erten, S. G.Kobourov, V. Le, and A. Navabi. Simultaneous graph drawing: Layout algorithms and visualization schemes. In *Proc. GD 2003*, pages 437–449, 2003.
[8] T. M. J. Fruchterman and E. M. Reingold. Graph Drawing by Force-Directed Placement. *Software Practice and Experience*, 21(11):1129–1164, 1991.
[9] D. C. Y. Fung, S.-H. Hong, D. Koschutzki, F. Schreiber, and K. Xu. Visual analysis of overlapping biological networks. In *Proc. IV 2009*, pages 337–342, 2009.
[10] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. ThemeRiver: Visualizing Thematic Changes in Large Document Collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, 2002.
[11] E. H.Chi and S. K.Card. Sensemaking of Evolving Web Sites Using Visualization Spreadsheets. In *Proc. INFOVIS 1999*, pages 18–25, 1999.
[12] E. H.Chi, J. Pitkow, J. Mackinlay, P. Pirolli, R. Gossweiler, and S. K.Card. Visualizing the evolution of Web ecologies. In *Proc. CHI 1998*, pages 400–407, 1998.
[13] D. Ikeda, T. Nanno, and M. Okumura. Gender Classification of Blog Authors. In *Proc. the Annual Meeting for Natural Language Processing 2006*, 2006.
[14] K. Inui, S. Abe, K. Hara, H. Morita, C. Sao, M. Eguchi, A. Sumida, K. Murakami, and S. Matsuyoshi. Experience Mining: Building a Large-Scale Database of Personal Experiences and Opinions fromWeb Documents. In *Proc. WI-IAT 2008*, pages 314–321, 2008.
[15] M. Itoh, M. Toyoda, and M. Kitsuregawa. An Interactive Visualization Framework for Time-Series of Web Graphs in a 3D Environment. In *Proc. IV 2010*, pages 54–60, 2010.
[16] A. Kehoe and M. Gee. Weaving Web Data into a Diachronic Corpus Patchwork. *Language and Computers*, 69(1):255–279, 2009.
[17] M. Kitsuregawa, T. Tamura, M. Toyoda, and N. Kaji. Socio-Sense: A System for Analysing the Societal Behavior from Long Term Web Archive. In *Proc. APWeb 2008*, pages 1–8, 2008.
[18] H. Koike, T. Takada, and T. Masui. Visualinda: A framework for visualizing parallel linda programs. In *Proc. VL 1997*, pages 174–182, 1997.
[19] N. Nakazono, K. Misue, and J. Tanaka. NeL2: Network Drawing Tool for Handling Layered Structured Network Diagram. In *Proc. APVis 2006*, pages 109–115, 2006.
[20] H. Notsu, Y. Okada, M. Akaishi, and K. Niijima. Time-tunnel: Visual analysis tool for time-series numerical data and its extension toward parallel coordinates. In *Proc. CGIV 2005*, pages 167–172, 2005.
[21] P. Resnik and A. Elkiss. The Linguist's Search Engine: an Overview. In *Proc. ACLdemo 2005*, pages 33–36, 2005.
[22] J. C. Roberts. State of the Art: Coordinated & Multiple Views in Exploratory Visualization. In *Proc. CMV 07*, pages 61–71, 2007.
[23] Z. Shen, M. Ogawa, S. T. Teoh, and K.-L. Ma. Biblioviz: a system for visualizing bibliography information. In *Proc. APVis 2006*, pages 93–102, 2006.
[24] M. Toyoda and M. Kitsuregawa. A System for Visualizing and Analyzing the Evolution of the Web with a Time Series of Graphs. In *Proc. HYPERTEXT 2005*, pages 151–160, 2005.
[25] F. Wei, S. Liu, Y. Song, S. Pan, M. X. Zhou, W. Qian, L. Shi, L. Tan, and Q. Zhang. TIARA: A Visual Exploratory Text Analytic System. In *Proc. KDD 2010*, pages 153–162, 2010.