Click2Annotate: Automated Insight Externalization with Rich Semantics

Yang Chen* Scott Barlowe† Jing Yang‡

Department of Computer Science
UNC Charlotte

ABSTRACT

Insight Externalization (IE) refers to the process of capturing and recording the semantics of insights in decision making and problem solving. To reduce human effort, Automated Insight Externalization (AIE) is desired. Most existing IE approaches achieve automation by capturing events (e.g., clicks and key presses) or actions (e.g., panning and zooming). In this paper, we propose a novel AIE approach named **Click2Annotate**. It allows semi-automatic insight annotation that captures low-level analytics task results (e.g., clusters and outliers), which have higher semantic richness and abstraction levels than actions and events. Click2Annotate has two significant benefits. First, it reduces human effort required in IE and generates annotations easy to understand. Second, the rich semantic information encoded in the annotations enables various insight management activities, such as insight browsing and insight retrieval. We present a formal user study that proved this first benefit. We also illustrate the second benefit by presenting the novel insight management activities we developed based on Click2Annotate, namely scented insight browsing and faceted insight search.

Keywords: Visual Analytics, Decision Making, Annotation, Insight Management, Multidimensional Visualization.

Index Terms: H.5.0 [Information Interfaces and Presentation]: General;

1 INTRODUCTION

Multidimensional data exist in a wide variety of applications, such as financial analytics, genomic analysis, and health analytics. In these applications, seeking insights from data and using them as evidence for hypothesis generation and evaluation are important steps in Decision Making and Problem Solving (DMPS). Since a DMPS process may involve a large number of insights, insight externalization, namely the process of capturing and recording the semantics of insights [12], is important for insight revisiting, association, comparison, and exchange. In this paper, we address the challenge of insight externalization for analytic insights. Analytic insights, a category of insights among other types, are the most traditional sense of insights supported in visualization systems [11]. They come from exploratory analysis, extrapolation, and consist in the large or small eureka moments where a body of data comes into focus for a user [11]. Wehrend and Lewis identified 11 low-level tasks which can result in an analytic insight for a user [15].

Insight Externalization (IE) in most existing information visualization systems, such as Many-Eyes [14] and Name Voyagers [7], requires users to type notes, draw marks, or connect associated insights manually. When the number of insights grows larger, these manual approaches become tedious, inefficient, inaccurate, and time consuming [6]. To address these problems, multiple efforts have been conducted toward Automated Insight Externalization (AIE) in recent years.

Existing AIE approaches can be classified according to the four-tier visual analytic activity model proposed by Gotz and Zhou [6]. In this model, visual analytic activities are abstracted into four levels namely tasks, sub-tasks, actions, and events. They range in semantic richness and abstraction levels from high to low. *Tasks* correspond to a user's highest-level analytic goals. *Sub-tasks* correspond to more objective, concrete analytic goals, such as finding clusters, outliers, or correlations. They are also called low level analytic tasks in other literatures [3]. *Actions* refer to atomic analytic steps such as zooming and panning. *Events* correspond to the lowest-level of interaction events, such as mouse clicks and button presses. The automation in most existing IE approaches are conducted at the action or event level. To the best of our knowledge, there exists no general IE approach for multidimensional datasets that conducts the automation at the sub-task level.

We argue that conducting AIE at the sub-task level is a promising research direction. The reasons are:

- Sub-tasks are less application-dependent than tasks. According to Amar and Stasko [3], there exists a set of low-level analytic tasks (sub-tasks) that are common to most multidimensional datasets. Therefore, it is possible to develop AIE techniques independent from particular domains and applications at the sub-task level.

- Information captured from the sub-task level, such as clusters and outliers, can have higher semantic richness and abstraction levels than that from the action and event levels, such as zooming and mouse clicks. The former will be easier to understand, recall, retrieve, and use in the DMPS process than the latter.

- Annotations with information from the sub-task level can be decoupled from the low level user exploration behaviors. For example, we can annotate an insight as a cluster without recording how this cluster is found. As a consequence, the annotations are independent from the visualization platforms on which the insights are captured. Thus the exchange of insights among different visualization systems can be enabled. In addition, the implementation of the annotation approach can be made simpler by not capturing the exploration process. The storage of the annotations can also be more efficient without the exploration process captured.

In this paper, we propose a novel AIE approach for analytic insights, insights for short in the rest of this paper. It conducts automation at the sub-task level and is named Click2Annotate. The approach is based on a three-component insight model [4]. In this model, an insight consists of a **fact** extracted from data under analysis, a **domain/application specific knowledge base** against which the fact is evaluated, and **objective and subjective evaluations** of the fact against the knowledge base. A fact, such as a cluster or an outlier, is a direct consequence of a sub-task. For example, a cluster is the consequence of finding a cluster sub-task. Facts are do-

*e-mail: ychen61@uncc.edu
†e-mail: sabarlow@uncc.edu
‡email: jyang13@uncc.edu

main and application independent while knowledge bases and evaluations are domain and application related [4]. As a general AIE approach, Click2Annotate enhances automation in **fact** annotation, and allows users to annotate knowledge bases and evaluations with light human effort.

The automation of Click2Annotate in fact annotation is based on the following observations reported in [4]. First, most facts extracted from multidimensional data fall into multiple categories independent from the domains/applications and visualization tools. Second, users can effectively and efficiently classify facts into these categories. Third, the same set of context and content information is often used to annotate facts falling into the same category. According to the above observations, the core components of Click2Annotate are annotation templates each of which guides the semi-automatic annotation of a certain type of facts. They are either pre-defined for popular fact types or interactively created by users. During the annotation process, the users only need to highlight data composing a fact, decide the type of the fact, and select the corresponding annotation template. The system will automatically follow the template to fetch information, encode it, and generate a narrative annotation for the insight. Click2Annotate is introduced in detail in Section 3. Our formal user study, which is reported in Section 4, proved that Click2Annotate enhanced annotation efficiency and the annotations generated were easy to understand.

Besides enhancing annotation efficiency, the semantic-rich information automatically captured at the sub-task level by our approach also enables a set of novel insight management activities, such as scented insight browsing and effective insight retrieval through faceted search. We have implemented a fully working prototype named **ManyInsights** that integrates these insight management activities and Click2Annotate. We present ManyInsights and its insight management functions in Section 5 to prove the benefits of our AIE approach in supporting insight management activities.

2 Related Work

2.1 Insight Externalization and Management

Most existing IE approaches for multidimensional datasets rely on human beings to input the task or sub-task information for insights captured. For example, Many-Eyes [14] provides a discussion forum where users can post threads to share their insights or free thoughts on visualizations. Systems such as Sandbox [16] and Name Voyagers [7] allow users to jot down their observations and opinions into visualization views.

Multiple efforts have been conducted on AIE approaches for multidimensional datasets. Most of them conduct the automation at the action or event level. They automatically capture the actions or events that lead to the discovery of the insight to be externalized. They allow users to infer the high-level logical constructs of the analysis and recall the insight itself by reviewing the full history of analytic activities [6]. For example, Aruvi [12] provides a historical record of users' insight exploration process by automatically capturing their navigation steps. Gotz and Zhou [6] propose an approach that automatically captures user exploration activities at the action level. Since information automatically captured from the action or event level has limited semantic meanings to users, human effort is usually required in these approaches to input the semantics of the insights. Click2Annotate conducts automation in the sub-task level to enhance IE effectiveness and efficiency. We need to point out that our approach does not support users in reviewing the visual exploration process. However, there is no conflict between our approach and other AIE approaches that capture visual exploration histories. They can be used together if desired.

Insight retrieval, the process of searching and processing information in insights, is supported by many existing multidimensional visualization systems. For example, Many-Eyes [14] and SparTag.us system [9] allow users to search insights with annotations containing keywords of interest. The effectiveness and efficiency of insight retrieval in visualization systems is limited by their IE approaches, since automatically captured actions and events are difficult to use as search criteria while manually generated annotations are not formalized and the information they contain is often incomplete and inaccurate. The semantic-rich annotations generated from Click2Annotate provide new potentials for effective and efficient insight retrieval as well as other insight management activities.

Many IE approaches have been proposed in the field of document visualization and management. For example, Zheng et al [18] propose a structured annotation approach that uses a unified annotation model to record and organize co-authors' insights in document revision tasks. This model is similar to the concept of templates in Click2Annotate. The tagging system developed by Hong et al uses two techniques, namely Click2Tag [9] and Fingerprint [8], to help users generate and browse annotations for online documents. Click2Tag allows users to annotate documents using simple mouse clicks while the Fingerprint technique achieves the goal of "annotate once, appear anywhere". Our approach achieves similar goals in the different environment of multidimensional visualization. Our work is also inspired by Document Cards [13], which uses a mixture of images and key words to capture the semantics of a document in a small space [13]. The annotations generated by Click2Annotate can also be examined in a card representation with mixed visualization and semantic information.

2.2 User Annotation Activities

In our previous research [4], we conducted a set of experiments, user studies, and interviews to investigate users' annotation activities. Since the proposed AIE approach is inspired by the conclusions of these studies, we briefly introduce the studies and their conclusions here.

In our experiments, we first conducted a literature survey on multidimensional visualization papers. It led to a rough fact categorization for facts from multidimensional data. We then manually classified a large number of annotations posted on Many-Eyes [14] into the initial fact categories and refined the categorization according to the results. After that, we conducted a user study in which users manually classified Many-Eyes annotations into the categories we provided. The task completion time and consistency of the classification results among the users were analyzed. Finally, we interviewed 16 experts from a variety of domains to learn what kinds of facts from multidimensional data they were interested in and how they annotated these facts. Three major conclusions were drawn from the above research.

First, the experiments and interviews showed that most facts discovered from multidimensional data fall into multiple categories. It suggested that we can predict the types of facts that are most frequently sought. Second, the user study showed that users could effectively and efficiently classify facts into the categories we identified. It suggested that users can help determine the types of facts. Third, the interviews showed that the same set of information is often used to annotate facts falling into the same category, despite the fact that the experts we interviewed were from a wide variety of domains. It suggested that information to be annotated for popular fact types is predictable.

3 Click2Annotate

3.1 Approach Overview

Click2Annotate is implemented in ManyInsights, a novel multidimensional visualization system we propose. It is developed using Flex [1]. ManyInsights allows users to seek insights from scatterplots and parallel coordinates [10] and annotate, browse, and examine them.

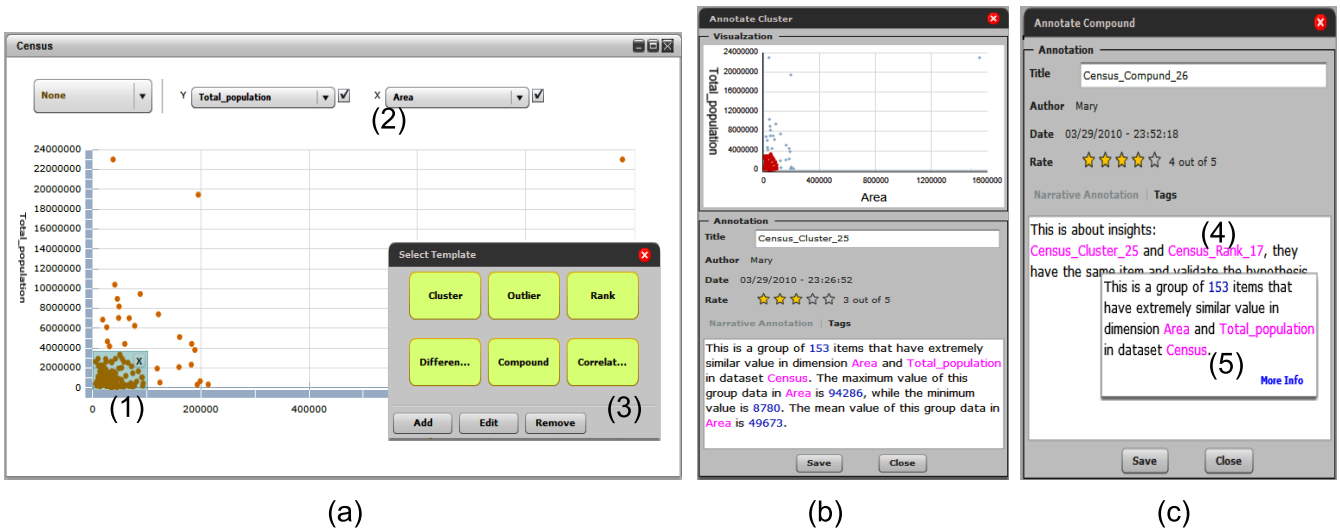

Figure 1: Semi-automatic annotation generation using Click2Annotate. (a) A scatterplot with a cluster in it and the annotation process. (b) The automatically generated annotation for the cluster. (c) The annotation generated for a compound fact.

We briefly describe how users use Click2Annotate to annotate an insight. When a user discovers a fact of interest during the visual exploration, such as the cluster shown in Figure 1(a), she brushes the relevant data (see Figure 1(a-1)), specifies the dimensions (see Figure 1(a-2)), judges the type of the fact, and selects the template for this type by a mouse click (see Figure 1(a-3)). The system will then automatically create an annotation based on the template and present the annotation to the user (see Figure 1(b)). The user reviews the annotation and interactively improves it, such as typing domain-related information and her evaluations (this step can be customized for individual applications to increase the level of automation). Since mouse clicks rather than intensive typing effort are required from the user to accomplish the majority of the annotation process, our approach is named Click2Annotate.

In the following sections, we introduce annotation templates, semi-automatic annotation generation, and interactive annotation review and modification in detail.

3.2 Annotation Templates

Annotation templates are the key components of Click2Annotate. Each annotation template is associated with a fact type. It tells the system what information needs to be retrieved from the data and how to generate a semantic-rich annotation for this type of fact. A template can be either pre-defined or user-defined.

3.2.1 Pre-Defined Templates

Click2Annotate provides pre-defined templates for popular fact types detected from our previous studies [4]. The templates are generated with the following steps.

Determining Popular Fact Types: Six fact types, namely *cluster*, *outlier*, *rank*, *difference*, *correlation*, and *compound fact*, are determined to be popular fact types. Facts of these types were frequently posted on Many-Eyes [2] as revealed by our experiment [4]. Their definitions are self-explained by the type names. These fact types are further classified into three categories, namely dimension-oriented facts, data item-oriented facts, and compound facts. Fact types within the same category share common features. Dimension-oriented facts, such as *correlation*, describe relationships of dimensions. Data item-oriented facts, such as *cluster*, *outlier*, *rank*, and *difference*, describe clusters, anomalies, patterns, and relationships of data items. Compound facts describe relationships among multiple facts, such as that fact A is related to fact B. The fact type hierarchy guides the generation of the templates by extracting the common features among the fact types in the same category. It also guides the use of annotations in insight management activities (see Section 5.1 for an example).

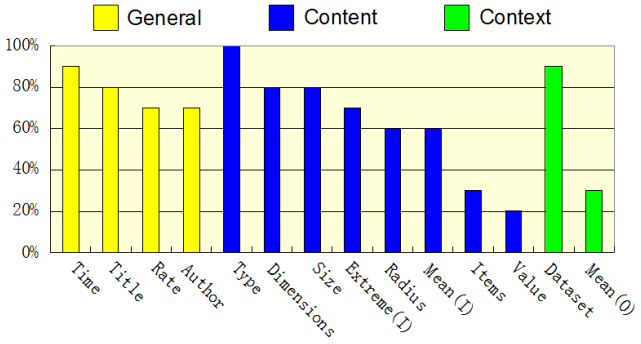

Figure 2: Frequencies of attributes used in cluster annotations

Predicting Information in Annotations: The templates tell the system what information needs to be retrieved from the data for generating an annotation. We predict such information for each pre-defined template based on the results of our domain expert interviews [4]. In these interviews, the experts reported what information they used to annotate the facts. We summarized the results and got an attribute list for each popular fact type. The percentage of the experts that used an attribute to annotate each type of facts was calculated. For example, Figure 2 shows how often the attributes listed were used to annotate a cluster by the experts.

As shown in Figure 2, an annotation of a cluster often consists of the following attributes, in descending order of their frequency: Type (the fact type, such as "cluster" in this example); Time (when the cluster was discovered); Dataset (the dataset where the cluster was discovered); Title (the title of the annotation); Dimensions (the dimension names of the subspace where the cluster existed); Size (the number of items in the cluster); Rate (users' subjective evaluation); Author (who discovered the cluster); Extreme(I) (the extreme of data inside the cluster); Radius (the radius of the cluster); Mean(I) (the mean of data inside the cluster); Items (the data item names of the cluster); Value (the data values of the cluster);

Table 1: Content attributes, context attributes, and narrative sentences for popular fact types. The attributes with "I" are about the data inside the fact. The attributes with "O" are about the data outside the fact.

Type	Content Attributes	Context Attributes	Narrative Sentences
Cluster	Type, Dimensions, Size, Extreme(I), Radius, Mean(I)	Dataset	This is a group of xx items that have (extremely/very/slightly) similar values in dimensions xx in dataset xx.
Outlier	Type, Dimensions, Size, Items, Mean(I)	Dataset, Mean(O), Distance	This is a group of xx items that are (extremely/very/slightly) different from the others in dimensions xx.
Rank	Type, Dimensions, Items, Value, Rank	Dataset	Item xx ranks xx in dimension xx in dataset xx from highest to lowest.
Difference	Type, Dimensions Items, Difference, Distance	Dataset	There is an/a (extremely large/large/slightly) difference between item xx and item xx in dimension xx. The value of item xx is higher by xx.
Correlation	Type, Dimensions Coefficient	Dataset	For xx percent of data items in dataset xx, the higher their value in dimension xx, the (higher/lower) their values in dimension xx.
Compound	Pointers to the related facts	N/A	This is about insights xx.

and Mean(O) (the mean of data outside the cluster).

According to the statistics, we identify three categories of attributes for each popular fact type (shown in yellow, blue, and green in Figure 2). They include: general attributes, such as Author, Time, Title, and Rate, which are important information for all types of insights and they are not directly related to the data; context attributes, such as Size, Items, and Mean(I), which are frequently used to describe the content of a certain type of facts; and context attributes, such as Mean(O), which are frequently used to capture the context of a certain type of facts.

Table 1 summarizes the frequently used content and context attributes (with percentage $\geq 50\%$) of the popular fact types. They are semantic-rich information widely used by the experts to describe facts. We include them into the templates of the types together with all the general attributes. Users can customize a template (refer to Section 3.2.2) if the information they desire is not included in the template.

Shaping the Templates: From the previous step, a fairly large amount of attributes are determined to be included into the templates. How should they be presented to users so that the users can enjoy reading the annotations and grasp their content effectively and efficiently? To address this problem, we conducted a user study.

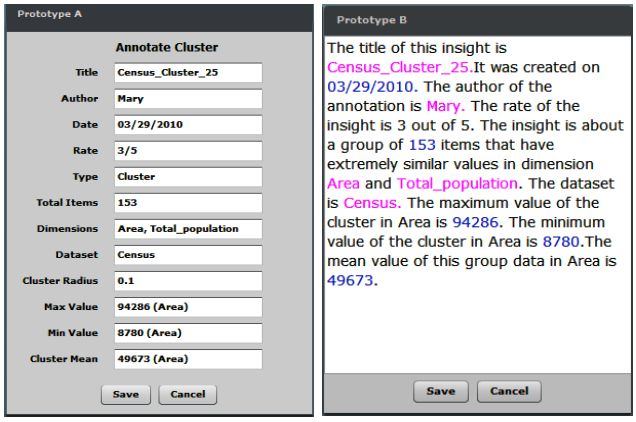

Figure 3: Examples of prototypes A and B.

First, we designed three template interface prototypes with the following goals: (1) Completeness: all attributes should be represented; (2) Clearness: the information should be easy to read and understand; and (3) Briefness: key information of the attributes should be easily accessed. In Prototype A, each attribute was represented as a form entry, such as "cluster radius: 0.1" for the radius of a cluster. Prototype B employed a narrative annotation that represents information textually [5]. All the attributes were presented in sentences that described the attributes using natural language. For example, the entry "cluster radius: 0.1" in the previous example was expressed as "The items in this group have very similar values". Prototype C used a mixed design. The general attributes were represented as form entries while the content and context attributes were represented textually. Two annotations were generated for each of the fact types, including cluster, outlier, and correlation following each prototype. A total of 18 annotations were generated. Figure 3 shows examples of the generated cluster annotations for prototype A and B. The annotation of the same content for prototype C is shown in Figure 1(b).

Twenty users who had good experiences with reading annotations in visualizations participated in the study one by one. The subject was asked to grade prototypes A, B, and C according to the following three criteria: (1) the annotations are pleasant to read; (2) the values of the attributes in the annotations can be quickly perceived; and (3) it is easy to compare facts of the same type and facts of different types. A 7-point scale was used for the rating (0=strongly disagree, 6=strongly agree). User feedbacks were also collected.

The results showed that there was a stronger preference to Prototype C. In particular, the average scores of Prototype A, B, and C in the first criterion were 2.8, 3.4, and 4.8 respectively; the average scores in the second criterion were 3.8, 3.2, and 4.4 respectively; and the average scores in the third criterion were 4.4, 3.0, and 4.2 respectively.

According to user feedback, Prototype C had the following advantages: First, it represented the general attributes as form entries and thus reduced the number of sentences in the narrative annotation. Users had no difficulty in understanding general attributes, such as author and dataset name, in the form entries. Second, it represented context and content attributes using natural language, which makes them easy to understand by users who were not familiar with terms such as cluster radius. Therefore, Prototype C is used in Click2Annotate for shaping the templates.

Encoding Attributes Using Natural Language: The context and content attributes are encoded into human-readable sentences in the templates to compose narrative annotations. Our encoding process is similar to the one described in [5] but improved from three aspects: First, multiple context and content attributes can be encoded in one sentence. This produces a less wordy annotation. Second, the numerical attribute values (e.g., the radius value) are explained in an easy to understand manner (e.g., "very similar"). Third, the key information in the sentences, such as dimension names, is automatically highlighted and hyperlinked so that users can easily access related insights sharing the common content. For example, the narrative of a cluster may start by a sentence that de-

scribes the size, the quality (indicated by radius), the dimension labels, and the dataset of the cluster: **this is a group of 8 data items that have extremely similar values in dimensions A and B in dataset NFL.** The information in pink is automatically extracted by the system after a user selected the data and the template. We summarize examples of narrative sentences for the popular fact types in Table 1.

3.2.2 User-Defined Templates

Although a set of templates is pre-defined for most popular fact types, it is impossible to predict all useful fact types as well as all possible attributes for each fact type. Therefore, Click2Annotate allows users to interactively modify pre-defined templates or create new templates from scratch.

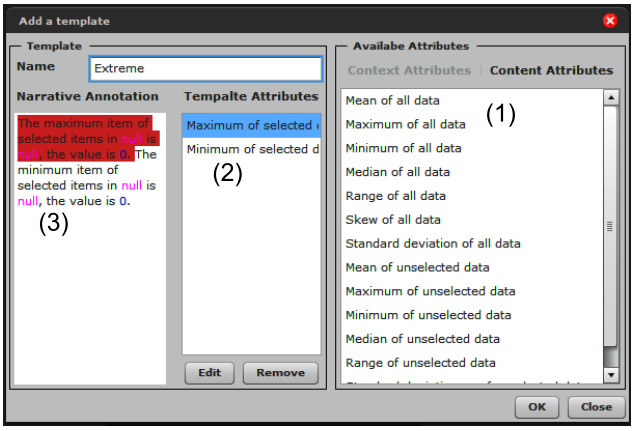

Figure 4: Interactive generation of a user-defined template.

Figure 4 shows an example of how to create a new template for a user-defined fact type named *extreme*. In this window, there a lists of available attributes (see Figure 4(1)), including all possible context and content attributes reported by the domain experts in the interviews [4]. They can be added to the template attributes list (see Figure 4(2)). In this example, the general attributes are automatically included and the maximum and minimum of the relevant dimension are manually added into the template. The narrative sentences of these attributes are represented in the annotation area, providing a preview for the annotations generated by this template (see Figure 4(3)). Users can interactively modify these sentences or change their order. The modification of an existing template can be accomplished in the same interface.

3.3 Semi-Automatic Annotation Generation

Click2Annotate semi-automatically generates annotations based on pre-defined or user-defined templates. To generate an annotation, users brush the relevant data items and dimensions and select a template according to the type of the fact. To allow quick access to the templates, a list of buttons are provided in a separated panel (see Figure 1(a-3)), which is shared by all created views. Each button corresponds to a template. Users can add or remove buttons from the panel so that it only contains the buttons for templates they need. The users click on a button to select a template. After the template is selected, the system will automatically fetch information from the data and encode it to fill the incomplete information in the template. Thus, an annotation is automatically generated.

The above process does not apply to compound facts because they contain pointers to other facts. To annotate a compound fact, an interactive approach is employed. In particular, users first open a compound fact annotation dialog (see Figure 1(c)) by clicking on a "compound" button and then use drag-and-drop interactions to add the flags of desired insights (Figure 8(a-1)) to the dialog. After an insight is added, its title will be displayed in the dialog (see Figure 1(c-4)). It is hyperlinked to the related annotation so that users can click on it to examine the annotation in detail (see Figure 1(c-5)). Users can add insights into the dialog and type their notes to complete the annotation.

3.4 Annotation Review and Modification

After an annotation is generated by the system, it will be presented to users in an annotation window (see Figure 5(a)) within which the annotation can be reviewed and improved by the users.

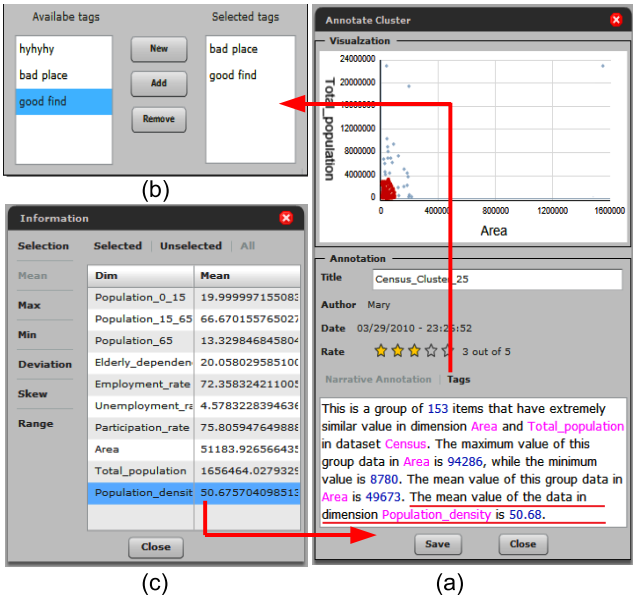

Figure 5: The review, tagging, and modification of generated annotations. (a) A modified annotation. (b) The tagging interface. (c) The statistics window.

The annotation window directly mimics the design of Prototype C with a thumbnail added. The thumbnail is a screenshot that captures the visualization at the moment when the fact was discovered to help users recall this insight. The general attributes are represented below the thumbnail, followed by a set of sentences that textually represent the context and content attributes.

If users are not satisfied with the automatically generated annotation, they can interactively improve it. In particular, the users can open a statistics window (see Figure 5(c)) which presents a list of all available statistics about the fact and the whole dataset, and a list of the information that has already been included in the current annotation. Users can use drag-and-drop interactions to add or remove statistics into or from the annotation and adjust the order of their presentations in the annotation. The statistics in the annotation is represented textually according to pre-defined templates. Users can manually customize the text representations if they are not satisfied with the pre-defined ones. For example, in Figure 5(c), a user drags and drops the mean value of the dimension *population density* to the annotation. A new sentence that conveys this mean value is then automatically added to the annotation, as shown in the sentence with the red underline in Figure 5(a).

The automatically generated annotation only captures the fact of an insight. To allow users to record the knowledge base and subjective evaluations of the insight, an interactive tagging function is supported. In particular, a user can click on a button in the annotation window to trigger a tagging interface (see Figure 5(b)). Through the interface, the user can create tags or select existing tags

to annotate the insight. A tag is generated once and reused later on. Thus users can type frequently used information once, save it as a tag, and reuse the tag in the future with light human effort.

4 User Study

A formal user study has been conducted to evaluate how Click2Annotate helped users generate annotations and if the generated annotations were understandable. The study was a 2×2 (system types×datasets) between-subjects design. We compared two systems: ManyInsights, which supported Click2Annotate, and a simple system, which provided users a text editor similar to those commonly found in many visualization systems to manually type notes for annotating insights. Our hypotheses were: (1) ManyInsights will reduce the time spent on annotating; and (2) annotations generated by ManyInsights will reduce the time-cost and errors for understanding the annotations.

4.1 Datasets and Insights

Two datasets were used in the user study: a small dataset (51 items, 4 dimensions) on state health measures and a large dataset (279 items, 10 dimensions) on the US census data. Before the user study, we manually extracted six insights, including a cluster, an outlier, a rank, a difference, a correlation, and a compound fact, from each dataset. The compound fact was about the difference between two clusters. All extracted insights were used in the user tasks described in the next section. The numbers of data items and dimensions involved in the insights were controlled according to the size of dataset. For example, a cluster in the large dataset had more data items and dimensions involved. This made annotating and comprehending insights in the large dataset more difficult.

4.2 Tasks

The experiment included two sessions.

Annotation session: Each participant was asked to annotate the six insights for each dataset on a computer. The insight was annotated one by one. For each insight, the fact type was explicated and the relevant data was highlighted in a parallel coordinates or scatterplot view of the dataset according to the number of dimensions involved in the insight. The participant was asked to record all possible information that could help them comprehend the insight. The task completion time was recorded.

Comprehension session: Each participant was asked to understand the annotations generated by other participants. There were six tasks to complete for each dataset, each of which for an insight used in the annotation session. In each task, an annotation (randomly picked from annotations generated by other participants and text only) was provided along with four images on paper. One image was the screenshot of the view with the insight described by the annotation highlighted, namely the original view provided to the participants when the insight was annotated. The other three images presented different views, such as the same display with other data highlighted or a different display with a similar pattern. The participant was asked to find the view with the insight described by the highlighted annotation. The task completion time was recorded.

4.3 Analysis Condition and Procedure

A total of 8 subjects (5 male and 3 female) participated in the study. All of them were graduate students and had strong English writing and reading abilities. Before the study, the subjects were evenly divided into two groups. One group of subjects used ManyInsights and the other group used the simple system. The same datasets and tasks were used in both groups. The subjects took the experiment one by one on the same computer following the same process.

In the annotation session, the same views were used in both groups. When making annotations, participants were allowed to read dimension names, data names, and data values on the visualizations. In the simple system, participants used the text editor to type notes for annotating insights. In ManyInsights, participants were allowed to edit existing templates and interactively modify automatically generated annotations.

At the beginning of the study, a tutorial was provided by an instructor to explain the definition of each fact type in the insights and show examples of how to annotate an insight. The annotation session was conducted right after the tutorial. The comprehension session was conducted three months after the annotation session. In each session, there were first practical tasks, second experimental tasks (the small dataset followed by the large dataset), and then survey questions specific to that session.

4.4 Results

We present two types of results from the study, namely quantitative data (completion time and correctness) captured through the system and the subjective preferences reported from survey questions, in the following sections respectively.

4.4.1 Task Completion Time and Correctness

The comparisons of the average completion time for annotating insights are shown in Figure 6(a) (for the small dataset) and 6(b) (for the large dataset). Figure 6(b) reveals the difficulty the subjects encountered in making annotations in the large dataset using the simple system, especially when annotating the cluster and rank. We observed that subjects had difficulty in manually summarizing information from complex data, such as estimating the size of a big cluster and the rank of a data item in a large dataset. Besides, in the simple system, the time the participants spent on determining the information to be recorded was often more than the time they spent on typing the note. Click2Annotate showed its strength in pre-defining the most essential information for insights and automatically capturing this information. Therefore, our first hypothesis was validated.

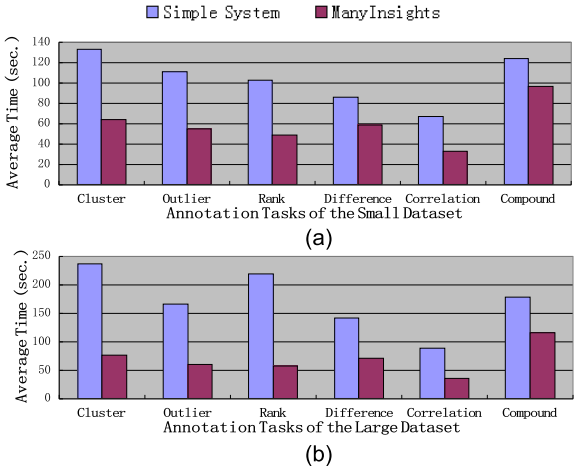

Figure 6: Results of Annotation sessions

The comparisons of the average completion time for the comprehension tasks are shown in Figure 7(a) (for the small dataset) and 7(b) (for the large dataset). The figures show that the participants were faster in selecting the views when reading the annotations generated by ManyInsights, especially for the large dataset. The average correct answer rate for all tasks was 89.6% for ManyInsights (with standard deviation 7%) and only 75.0% for the simple system (with standard deviation 11%). The result suggested that the subjects understood the ManyInsights annotations faster and better

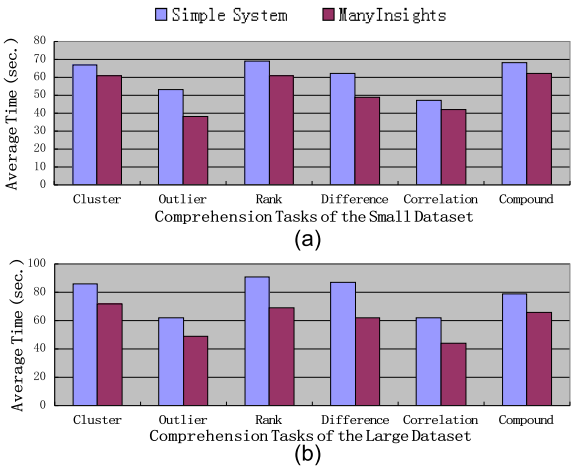

Figure 7: Results of Comprehension sessions

Table 2: Average ratings for four survey questions that have significant differences (difference≥2.5).

Questions	ManyInsights	Simple system
Q1. This tool helped me make annotations in the large dataset.	4.8	1.3
Q2. I enjoy using this tool to make annotations.	5.0	2.3
Q3. The content of annotations is accurate.	5.3	2.3
Q4. The annotations were helpful to me in understanding insights.	5.0	2.5

than the manually generated annotations. Thus our second hypothesis was validated.

4.4.2 Subjective Preferences

At the end of each session, the participant was asked to answer a set of survey questions each of which was answered in a 7-point Likert scale (0=strongly disagree, 6=strongly agree). A total of 10 questions were provided. The average score for ManyInsights was 4.8 and only 2.7 for the simple system. Table 2 summarizes the pair-wise comparisons of the questions where significant differences were detected.

The significant differences indicate that ManyInsights was judged to be more helpful than the simple system in annotating insights by the participants. Annotations generated by ManyInsights were judged to be more helpful in understanding insights.

5 Use Case: Insight Management

The semantic-rich information automatically captured by Click2Annotate provides rich potentials for enhancing various insight management activities, such as insight retrieval, browsing, and association. As a proof of the concept, we present scented insight browsing and faceted insight retrieval as concrete examples in this paper. These functions have been implemented in ManyInsights. In the following sections, we briefly introduce them.

5.1 Scented Insight Browsing

After insights are annotated, effective browsing approaches should be provided to help users review and reuse them. Towards this goal, we propose a scented insight browsing approach (see Figure 8(a)) based on Click2Annotate. If a user turns on the scented browsing mode, insight flags are attached to the visualizations, not only the views where the insights were captured, but also other views where the relevant data items/dimensions of the insights can be observed. Users can retrieve an insight from any view where its flag is displayed by clicking on the flag. Compared to existing approaches that require users to manually mark insights on the visualizations [7], our approach has several benefits.

First, based on the pre-defined essential information for different fact types, our approach automatically marks different types of insights in different ways to avoid cluttering the display. For example, Figure 8(a) shows a scatterplot with multiple annotated insights flagged. In this figure, the flags of data item-oriented facts are attached to their data items (see Figure 8(a-1)) while the flags of dimension-oriented facts are attached to their dimensions (see Figure 8(a-2)). In systems with manually generated annotations, users have to draw marks carefully to achieve similar effects.

Second, insights can be flagged in any display where the relevant data items/dimensions of them can be observed, not only the visualization where the insights were discovered. Thus it is an "annotate once, appear anywhere" approach [8]. For example, an insight of dimension correlation can be marked in any of the visualizations where any of the dimensions involved is displayed. This feature allows users to access relevant insights anywhere during their visual exploration process, without going back to the previous views. Thus the visual exploration becomes more convenient and flexible. Such an "annotate once, appear anywhere" approach is not easily supported by manual annotation approaches because of the lack of accuracy and formalization of manually generated annotations.

In addition, the scented insight browsing approach can work together with the faceted insight retrieval approach described below. In particular, users can interactively select the insights they want to flag using criteria such as the fact types and the dimensions involved (see Figure 8(a-3)). In this way the users can display only flags of insights of interest in the display to reduce clutter. Again, this benefit is brought by the accuracy and formalization of the automatically generated semantic-rich annotations.

5.2 Faceted Insight Retrieval

As the number of annotated insights grows larger, effective insight retrieval becomes essential. Faceted search [17], a popular searching approach used in mass online markets, has shown its efficiency and flexibility in finding items that can be aggregated based on multiple attributes. Since the annotations generated by Click2Annotate can also be aggregated based on multiple attributes, faceted search can be applied to help users retrieve insights according to their specific interest. In particular, a set of common attributes shared by multiple templates, including author, time, rate, title, fact type, dataset, dimensions, and tags, are used as faceted filters for searching insights in ManyInsights. Users can search insights in any order using these filters through the faceted search interface provided by ManyInsights (see Figure 8(b)).

For example, Figure 8(b) shows how a user retrieves the cluster annotated in Figure 1(b) using faceted search. First, she uses the fact type "cluster" to filter out insights that are not clusters. Second, she narrows down the results using the dataset name "census" (see Figure 8(b-4)). The search results dynamically roll over the screen from left to right. Inspired by [13], each insight is represented as an annotation card which summarizes the insight using a visualization thumbnail and a short sentence that captures the essential information of the insight. The user can sort the search results by different criteria, such as rate and title. When the user clicks on an annotation card, the annotation will be presented in full detail in an annotation pop-up window. Once the user finds interesting insights, she can export them to XML files for reporting or sharing.

Note that it is difficult to conduct faceted search on manually generated annotations since they are usually unformalized. It is also difficult to conduct faceted search based on AIE at the action

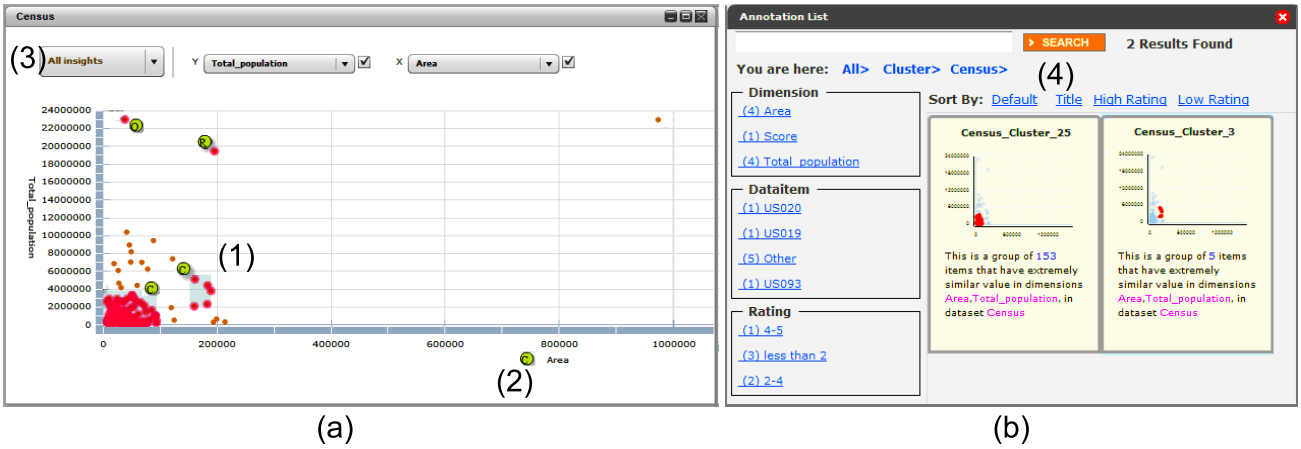

Figure 8: Insight management activities. (a) Scented insight browsing. (b) Faceted insight retrieval.

level or the event level since the lack of semantic richness. The annotations generated by Click2Annotate, on the other hand, are based on formalized templates and provide a set of attributes with rich semantics that can be used to classify insights in dimensions that are meaningful to users.

6 Conclusion

Insight Externalization is a critical requirement for an effective DMPS process. In this paper, we propose a novel approach that allows users to conduct semi-automatic insight externalization at the sub-task level. We also present a fully working prototype of this approach named Click2Annotate. Our formal user study showed that Click2Annotate reduced the human effort involved in IE and produced annotations that were easy to understand. Besides, Click2Annotate semi-automatically captures semantic-rich information that can be used in a variety of insight management activities, as illustrated in our use cases.

Click2Annotate is our first step toward a whole insight management solution, which includes insight browsing and retrieval, insight network, insight sharing/exporting in collaborative visualization, and insight recommendation and notification. We will work on these functions based on Click2Annotate. To support a wider range of data types, we will also extend Click2Annotate to trees, graphs, text, and geospatial data. In addition, more user studies and experiments will be conducted. For example, we will compare the performance of users who are familiar / unfamiliar with the datasets visualized. We will also investigate the effectiveness and efficiency of Click2Annotate with different design options in real analytical reasoning processes. The effectiveness of our current insight management approaches on handling large amounts of insights will be also investigated.

Acknowledgements

This work was performed with partial support from the DHS Visual Analytics for Command, Control, and Interoperability (VACCINE) Center of Excellence, under the auspices of the SouthEast Regional Visual Analytics Center.

References

[1] Flex. http://www.adobe.com/products/flex/.

[2] Many-eyes. http://www.many-eyes.com.

[3] R. Amar, J. Eagan, and J. Stasko. Low-level components of analytic activity in information visualization. *Proc. IEEE Symposium on Information Visualization*, pages 111–147, 2005.

[4] Y. Chen, J. Yang, and W. Ribarsky. Toward effective insight management in visual analytics systems. *Proc. IEEE Symposium on Pacific Visualization*, pages 49–56, 2009.

[5] D. Goncalves and J. Jorge. In search of personal information: Narrative-based interfaces. *Proc. ACM Conference on Intelligent User Interfaces*, pages 179–188, 2008.

[6] D. Gotz and M. Zhou. Characterizing users visual analytic activity for insight provenance. *Proc. IEEE Symposium on Visual Analytics Science and Technology*, pages 123–130, 2008.

[7] J. Heer, F. Viegas, and M. Wattenberg. Voyagers and voyeurs: Supporting asynchronous collaborative information visualization. *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1029–1038, 2007.

[8] L. Hong and E. Chi. Annotate once, appear anywhere: Collective foraging for snippets of interest using paragraph fingerprinting. *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1791–1794, 2009.

[9] L. Hong, E. Chi, R. Budiu, P. Pirolli, and L. Nelson. Spartag.us: A low cost tagging system for foraging of web content. *Proc. Working Conference on Advanced Visual Interfaces*, pages 65–72, 2008.

[10] A. Inselberg. The plane with parallel coordinates. *Special Issue on Computational Geometry, The Visual Computer*, 1:69–97, 1985.

[11] Z. Pousman and J. Stasko. Casual information visualization: Depictions of data in everyday life. *Proc. IEEE Symposium on Information Visualization*, pages 1145–1152, 2007.

[12] Y. Shrinivasan and J. van Wijk. Supporting the analytical reasoning process in information visualization. *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1237–1246, 2008.

[13] H. Strobelt, D. Oelke, C. Rohrdantz, A. Stoffel, D. Keim, and O. Deussen. Document cards: A top trumps visualization for documents. *Proc. IEEE Symposium on Information Visualization*, pages 1145–1152, 2009.

[14] F. Viegas, M. Wattenberg, F. van Ham, J. Kriss, and M. McKeon. Manyeyes: a site for visualization at internet scale. *Proc. IEEE Symposium on Information Visualization*, pages 1121–1128, 2007.

[15] S. Wehrend and C. Lewis. A problem-oriented classification of visualization technique. *Proc. 1st Conference on Visualization*, pages 139–143, 1990.

[16] W. Wright, D. Schroh, P. Proulx, A. Skaburskis, and B. Cort. The sandbox for analysis: Concepts and methods. *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 801–810, 2006.

[17] K. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 401–408, 2003.

[18] Q. Zheng, K. Booth, and J. McGrenere. Co-authoring with structured annotations. *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 131–140, 2006.