

FORMATTING DATA BY EXAMPLE

BACKGROUND

[0001] The same type of data is often entered and stored in many different formats. For example, some dates are in the form CCYYMMDD (19990101), other dates in the format of MM/DD/CCYY (01/01/1999), yet other dates in the format of M/D/YY (1/1/99). To perform analysis on the data, it is converted to the same format. For example, some analysis may specify that phone numbers are to be formatted following the form (206) 555-1212, whereas other analysis may specify that formatting be removed from the phone numbers (i.e. 2065551212). Different methods may be used to transform the data. For example, different transformation functions may be used and/or code may be developed to transform the data.

SUMMARY

[0002] This Summary is provided to introduce a selection of concepts in a simplified form that are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used as an aid in determining the scope of the claimed subject matter.

[0003] Data formatting rules to convert data items from one form to another form are automatically determined based on an example set of outputs, e.g. a user's edits. A machine learning heuristic is applied to source data as well as example outputs, (e.g. a user's edits) to determine a data formatting rule that may be applied to additional data items. For example, a user may make edits that add/remove characters from data, concatenate data, extract data, rename data, and the like. By examining the original values along with the edited values, a rule can be derived that encapsulates this type of transform, and then that rule can be run on additional original values to automatically generate the desired edited values or outputs. The machine learning heuristic may be automatically triggered in response to an event (e.g. after a predetermined number of edits are made to a same type of data) or manually triggered (e.g. selecting a user interface option). The data formatting rule may be applied to other data and the results of the formatting reviewable by the user. Based on further edits/reviews, the data formatting rule may be updated. The data formatting rules may be stored for later use and/or modification. A confidence level may also be presented to assist a user in determining if an item(s) has been reformatted correctly.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] FIGURE 1 illustrates an exemplary computing environment;

[0005] FIGURE 2 shows a system for formatting data based on edits made to a document;

5 [0006] FIGURE 3 illustrates determining a data formatting rule based on a user's edits to a column and applying the data formatting rule to other cells within the column;

[0007] FIGURE 4 shows an example of a user making edits to the items in the social security number column;

10 [0008] FIGURE 5 illustrates an example of a user making edits to change the formatting of dates;

[0009] FIGURE 6 shows user interface elements that may be used to interact with the formatting of items;

[0010] FIGURE 7 shows a user interface for enabling/disabling fill by example; and

[0011] FIGURE 8 shows an illustrative process for formatting data by example.

DETAILED DESCRIPTION

15 [0012] Referring now to the drawings, in which like numerals represent like elements, various embodiment will be described. In particular, FIGURE 1 and the corresponding discussion are intended to provide a brief, general description of a suitable computing environment in which embodiments may be implemented.

20 [0013] Generally, program modules include routines, programs, components, data structures, and other types of structures that perform particular tasks or implement particular abstract data types. Other computer system configurations may also be used, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, minicomputers, mainframe computers, and the like.

25 Distributed computing environments may also be used where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

30 [0014] Referring now to FIGURE 1, an illustrative computer environment for a computer 100 utilized in the various embodiments will be described. The computer environment shown in FIGURE 1 includes computing devices that each may be configured as a server, a desktop or mobile computer, or some other type of computing device and includes a central processing unit 5 ("CPU"), a system memory 7, including a

random access memory 9 ("RAM") and a read-only memory ("ROM") 10, and a system bus 12 that couples the memory to the central processing unit ("CPU") 5.

[0015] A basic input/output system containing the basic routines that help to transfer information between elements within the computer, such as during startup, is stored in the ROM 10. The computer 100 further includes a mass storage device 14 for storing an operating system 16, spreadsheet 11, spreadsheet application 24, other program modules 25, and formatting manager 26 which will be described in greater detail below.

[0016] The mass storage device 14 is connected to the CPU 5 through a mass storage controller (not shown) connected to the bus 12. The mass storage device 14 and its associated computer-readable media provide non-volatile storage for the computer 100. Although the description of computer-readable media contained herein refers to a mass storage device, such as a hard disk or CD-ROM drive, the computer-readable media can be any available media that can be accessed by the computer 100.

[0017] By way of example, and not limitation, computer-readable media may comprise computer storage media and communication media. Computer storage media includes volatile and non-volatile, removable and non-removable media implemented in any method or technology for storage of information such as computer-readable instructions, data structures, program modules or other data. Computer storage media includes, but is not limited to, RAM, ROM, Erasable Programmable Read Only Memory ("EPROM"), Electrically Erasable Programmable Read Only Memory ("EEPROM"), flash memory or other solid state memory technology, CD-ROM, digital versatile disks ("DVD"), or other optical storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store the desired information and which can be accessed by the computer 100.

[0018] Computer 100 operates in a networked environment using logical connections to remote computers through a network 18, such as the Internet. The computer 100 may connect to the network 18 through a network interface unit 20 connected to the bus 12. The network connection may be wireless and/or wired. The network interface unit 20 may also be utilized to connect to other types of networks and remote computer systems, such as network service(s) 27. The computer 100 may also include an input/output controller 22 for receiving and processing input from a number of other devices, including a keyboard, mouse, or electronic stylus (not shown in FIGURE 1). Similarly, an input/output controller 22 may provide input/output to an IP phone, a display screen 23, a printer, or other type of output device.

[0019] As mentioned briefly above, a number of program modules and data files may be stored in the mass storage device 14 and RAM 9 of the computer 100, including an operating system 16 suitable for controlling the operation of a computer, such as the WINDOWS 7® operating system from MICROSOFT CORPORATION of Redmond, Washington. The mass storage device 14 and RAM 9 may also store one or more program modules. In particular, the mass storage device 14 and the RAM 9 may store one or more application programs, including a spreadsheet application 24 and program modules 25. According to an embodiment, the spreadsheet application 24 is the MICROSOFT EXCEL spreadsheet application. Other spreadsheet applications may also be used. A user interface, such as UI 28, allows a user to interact with an application, such as spreadsheet application 24.

[0020] Formatting manager 26 may be located externally from spreadsheet application 24 as shown or may be a part of spreadsheet application 24. Further, all/some of the functionality provided by formatting manager 26 may be located internally/externally from spreadsheet application 24.

[0021] Formatting manager 26 is configured to generate one or more data formatting rules to convert data from one form to another form based on original data and example outputs, for example a user's edits. According to an embodiment, formatting manager 26 applies a machine learning heuristic to the original data as well as example outputs a user's edits to determine the data formatting rule(s) that may be applied to data. For example, a user may make edits that add/remove characters from data, concatenate data, extract data, rename data, and the like. In response to the edits, a data formatting rule(s) is generated that is applied to other data within the document (e.g. a spreadsheet). The formatting that is applied to the data may be reviewable by the user such that the user may accept/reject changes. The formatting that is applied to the data may also comprise metadata formatting. According to an embodiment, a confidence level determined from the formatting rule is associated with the formatting that is applied to the data such that a user may more easily discern when the data is properly reformatted. For example, a high confidence level indicates that it is likely that the data is properly formatted, whereas a lower confidence level may indicate a user may wish to review the results. The machine learning heuristic may be automatically triggered in response to an event (e.g. after a predetermined number of edits are made to a same type of data) or manually triggered (e.g. selecting a user interface option). Based on further edits/reviews, the data formatting rule may be updated. The data formatting rules may also be stored for later use and/or

modification. For example, a user could modify the rule (e.g. a script) such that application of the data formatting rule follows the modified rule.

[0022] FIGURE 2 shows a system for formatting data based on edits made to a document. As illustrated, system 200 includes formatting manager 26, application
5 program 210, callback code 212, and display 215. The computing device(s) used may be any type of computing device that is configured to perform the operations relating to automatically formatting data based on a user's edits to a document. For example, some of the computing devices may be: mobile computing devices (e.g. cellular phones, tablets, smart phones, laptops, and the like); desktop computing devices and servers.

10 **[0023]** In order to facilitate communication with formatting manager 26, one or more callback routines, illustrated in FIGURE 2 as callback code 212 may be implemented. According to one embodiment, application program 210 is a spreadsheet application.

[0024] Display 215 is configured to display a document, such as spreadsheet document 220, and user interface elements used to interact with a document. As illustrated,
15 spreadsheet 220 shows three columns including a last name column (A), a first name column (B) and an edited column (C). In the current example, a user has made edits to the edited column. In cell C2, the user has entered for that row, row 2, the last name (that is also contained in cell A2), followed by a comma, that is followed by the first initial (that is also contained in cell B2). In cell C3, the user has entered for that row, row 3, the last
20 name (that is also contained in cell A3), followed by a comma, that is followed by the first initial (that is also contained in cell B3).

[0025] Generally, formatting manager 26 detects when the user is editing/modifying data that fits a pattern that can be filled down and applied to additional data in the spreadsheet, and automatically fills down the column with the results that are obtained
25 from applying the data formatting rule. In response to the edits, formatting manager 26 uses information that is associated with the edits to obtain a data formatting rule that is applied to other data within the spreadsheet. According to an embodiment, the information includes output examples that result from the edits to the text that is displayed within the edited cells (e.g. cells C2 and C3) and input examples that are associated with
30 the edits. In this case, column A and column B include input examples that are related to the edited column (e.g. cells A2 and B2 is an input example for the output example C2 and cells A3 and B3 is an input example for the output example C3). These input/output examples are determined by formatting manager 26 and are supplied to a process that generates a data formatting rule for other similarly formatted cells (e.g. cells C4:C7 (222)).

The machine learning heuristic obtains the set of input/output examples, determines a pattern, generates a data formatting rule, and then formatting manager 26 applies the data formatting rules to an output range to generate newly formatted values. According to an embodiment, an exemplary machine learning heuristic is described in “Automating String Processing in Spreadsheets Using Input-Output Examples,” by Sumit Gulwani, PoPL’11, January 26-28, 2011, Austin, Texas. Other machine learning heuristics may be utilized. Generally, any heuristic that takes original data as well as data edits as input and produces a data formatting rule that may be applied to other data to result in similarly formatted data may be used. According to an embodiment, the functionality of the machine learning heuristic is included within formatting manager 26. The functionality may also be located in other locations.

[0026] Formatting manager 26 automatically applies the data formatting rule to other cells within the spreadsheet that are similarly formatted. According to an embodiment, the data formatting rule is automatically applied to an output range of cells that fill down the column of the edited column. In the current example, the output range includes cells C4:C7. Box 222 shows that application of the data formatting rules to cells C4:C7 resulted in values being placed within cells C5 and C6. According to an embodiment, the data formatting rule that is applied to the output range is dynamic. In other words, when a value is edited within the output range, the data formatting rule is updated using the additional input/output example(s).

[0027] The data formatting rule may generate zero or more values for each of the cells. For example, a value is not returned for cells C4 and C7 since there is not a first name in the corresponding cell of the B column. More than one potential result may be generated by the data formatting rule when the data formatting rule is not sure of the result.

According to an embodiment, before automatically reformatting data, the data formatting rule is applied to a predetermined number of cells to determine whether application of the formatting rule is generating results that meet or exceed a predetermined confidence level threshold. For example, if application of the formatting rule to the predetermined number of cells results in a low confidence level, the data formatting rule is not automatically applied. [According to an embodiment, the data formatting rule is applied to the cells in the output range and a percentage of cells that have one answer is determined. According to an embodiment, the percentage of cells that have zero answers are excluded from the calculation. When the percentage is above a predetermined threshold (e.g. 70%) the cells in the output range are automatically filled down using the results provided by the data

formatting rule. When the threshold is not met, the results may not be applied to the cell and more edits are obtained before creating a new data formatting rule or the results may be applied and the cell and an indicator (e.g. highlighting, formatting) may be applied to the cell that indicates the confidence level is below the threshold. A unique result

5 generated by application of the data formatting rule to the cell is a good indicator that the data formatting rule is generating accurate results. Other thresholds and/or rules may be used to determine whether the data formatting rule is generating accurate results.

[0028] Many types of data formatting rules may be created based on a user's edits. For example, a concatenation of two columns, extracting information from a column (e.g.

10 extracting a top level domain name from an address, extracting an email address) and the like. Generally, a data formatting rule may be calculated based on any editing activity. In some cases, more than two input/output examples may be used to generate accurate results. For example, the machine learning heuristic may only be 50% accurate with two examples and be 95% accurate using three examples.

15 **[0029]** A data formatting rule may also be obtained based on a selection within a user interface (e.g. icon 224) or some other menu option may be selected. The example edits may be manually selected by a user (e.g. the user selects example cells) and/or the examples may be automatically determined by formatting manager 26. For example, formatting manager 26 may look at data and determine input/output examples from the

20 data (e.g. . a column with the least number of values may be considered as the output column, and the remaining columns may be considered as input columns).

[0030] FIGURES 3-6 show examples of formatting cells based on a user's edits.

[0031] FIGURE 3 illustrates determining a data formatting rule based on a user's edits to a column and applying the data formatting rule to other cells within the column. As

25 illustrated, a user is making edits to the Full Name column (C) of spreadsheet 310. In the current example, the user has typed a first initial followed by a period and a space that is followed by the last name. The last name is contained in column A of spreadsheet 310 and the first initial is contained in column B of the spreadsheet. In response to the user making the edits to cells C2 and C3, a data formatting rule is generated by a machine

30 learning heuristic that may be applied to other cells within the document.

[0032] In the current example, the input/output examples include the text in the C column and the text in the A and B columns for each row that was edited. The input data may be determined by scanning the document to locate data that may be used in application of the data formatting rule to create the desired result. In this case, the data

formatting rule creates a rule that obtains the first initial from column B and the last name from column A, as well as inserts a period character and space character after the first initial. The output range 312 indicates the cells to which the data formatting rule is to be applied.

5 [0033] Referring to spreadsheet 320 it can be seen that the automatic application of the data formatting rule has resulted in cells C3:C6 being filled in with a name that includes a first initial that is followed by a period and a space that is followed by a last name. Cell C7 was not filled in since application of the data formatting rule did not result in an accurate result since the first name column is empty.

10 [0034] Spreadsheet 320 also shows a reviewing user interface element 322 that may be used to accept/reject a change made by the application of the data formatting rule. An error user interface element 324 is also placed near the location of where the data formatting rule was not applied (in this case missing data from the First column) or where application of the data formatting rule may not be determined to be accurate (See FIGURE
15 6 and related discussion for more discussion regarding the reviewing user interface element and the error correction user interface element).

[0035] FIGURE 4 shows an example of a user making edits to the items in the social security number column. The user has changed the formatting of the social security number from the format “XXXXXXXXXX” to “XXX-XX-XXXX” (where X is any
20 numeral, 0-9). In other examples, the characters may non-numerical characters. According to an embodiment, after a user has made two or more edits the input/output examples are used by the formatting manager to generate the data formatting rule that is applied to the other data in the column. In the current example, the input examples are the original text that was contained in cells A2 and A3 and the output examples are the edited
25 text shown in cells A2 and A3. More or fewer edits may be collected before submitting the input/output examples. For example, in some cases (such as this one) one input/output example may be sufficient to generate an accurate data formatting rule. In more complex editing scenarios, more input/output examples may be used. Further, any additional edits made by the user may be used by the formatting manager to update the data formatting
30 rule. Application of the data formatting rule to cells A4:A7 result in the spreadsheet as illustrated by display 420. According to an embodiment, cells that already contain data are not changed automatically. Instead, a user may be requested to affirmatively accept the proposed changes before they are made to the cells containing data. The cells may be

also be changed automatically and the user provided with an opportunity to undo the changes.

5 [0036] FIGURE 5 illustrates an example of a user making edits to change the formatting of dates. The user has changed the formatting of two dates in spreadsheet 510 from the format “MM/DD/CCYY” to “MM/DD/YY.”

[0037] In the current example, the user has changed the formatting of the dates in cells A4 and A3. The input examples includes the original text in cells A3 and A4 and the output examples includes the edited text as illustrated in cells A3 and A4 of display 520. Application of the generated data formatting rule results in display 520. As shown, the
10 edits may be made anywhere within similarly formatted data and application of the data formatting rule may not only fill down as illustrated in FIGURES 2-4 but also be applied to other cells (e.g. cell A2).

[0038] FIGURE 6 shows user interface elements that may be used to interact with the formatting of items. As illustrated, spreadsheet 610 shows reviewing user interface
15 elements 612 and 618 and error correction user interface elements 614 and 616.

[0039] A cell may be marked with an error correction user interface element when the cell is flagged as having a value that is inconsistent and/or not determined to be accurate. According to an embodiment, a cell with inconsistent data means that the cell’s value either does not match what the data formatting rule would have generated or the value
20 within the cell was generated by the data formatting rule, but there is more than one possible result. As soon as the data formatting rule has been applied to the determined output range, any results that are inconsistent are flagged. According to an embodiment, a result is considered inconsistent when the number of possible results was more or less than one result (e.g. no results or 2 or more results provided by the data formatting rule) or the
25 pre-existing value is inconsistent with the result provided by the data formatting rule. Other heuristics may also be used. For example, a result may be considered inconsistent when the number of results exceeds a predetermined number of results and/or some other condition.

[0040] The error correction user interface element may be selected. When selected, the
30 error correction user interface element displays various selectable options (630). According to an embodiment, the selections include a next option, a previous option, a help option, an ignore option, an accept option, an edit in cell option, an error checking option and a possible values option. More or less options may be included within menu 630. The next option moves to the next cell that is marked as an error. The previous

option moves to the previous error. The help option provides a help display. The ignore option ignores the current error and removes the error correction user interface element from the display. The accept option removes the error condition and adds the associated input/output example for the cell to generate a new data formatting rule. The edit in cell
5 option places the user into an edit mode on the cell. When the user edits one or more of those error cells then the edit is treated as an input/output example, and new/updated data formatting rule is computed. According to an embodiment, the updated data formatting rule is applied to the remaining error cells that are related to the data formatting rule. The error checking option provides the user with various options relating to error checking.
10 The possible values option when selected displays a list of other possible values for the cell when reformatted. For example, each result that is generated by the data formatting rule may be displayed.

[0041] The reviewing user interface element 612 presents various options to interact with the cells that have been formatted using the data formatting rule. According to an
15 embodiment, the reviewing menu 620 comprises an undo option, a redo option, a stop option, a review option, an ignore all option, a save option, and an other option. More or fewer options may be included in menu 620. The undo operation reverts the document (e.g. the column of the document to which the data formatting rule was applied) to the state it was immediately before applying the data formatting rule to the cells. The redo
20 option restores the data in the cells that was previously undone by the user. The stop option disables the automatic behavior of the automatic fill down of applying the data formatting rule. The review option sets the active cell to be the first cell in the current conversion range (e.g. the output range) with an error tag. The ignore all option removes the error tags and any related error formatting from the cells in the current fill down range.
25 The save option allows a user to save the current data formatting rule. The save option saves information relating to the rule, such as column(s) that may be input as well as any input/output examples. The other option provides other options.

[0042] FIGURE 7 shows a user interface for enabling/disabling fill by example. Display 700 includes option 702 that allows a user to turn on/off the automatic filling of
30 data by example. Other options may also be included within a user interface, such as desired number of edits/selections before obtaining a data formatting rule, whether to overwrite existing data with/without confirmation, and the like.

[0043] Referring now to FIGURE 8, an illustrative process for formatting data by example will be described. When reading the discussion of the routines presented herein,

it should be appreciated that the logical operations of various embodiments are implemented (1) as a sequence of computer implemented acts or program modules running on a computing system and/or (2) as interconnected machine logic circuits or circuit modules within the computing system. The implementation is a matter of choice
5 dependent on the performance requirements of the computing system implementing the invention. Accordingly, the logical operations illustrated and making up the embodiments described herein are referred to variously as operations, structural devices, acts or modules. These operations, structural devices, acts and modules may be implemented in software, in firmware, in special purpose digital logic, and any combination thereof.

10 **[0044]** After a start block, process 800 moves to operation 810, where edits that are made to data within a document are detected. The edits may be any edits to the document. According to an embodiment, the edits are to data that is contained within cells of a document (i.e. spreadsheet, table, list) that are a same type of data and are similarly formatted. Generally, each cell within a column may contain the same type of data (i.e.
15 dates, addresses, names, numbers, and the like). The edits that are applied to each of the items fits a pattern that may be applied to other cells having the same type of item.

[0045] Moving to decision operation 820, a determination is made as to whether the number of edits has exceeded a predetermined number of edits and has triggered the process to obtain the data formatting rule that is to be applied to other similarly formatted
20 cells. According to an embodiment, the number of edits to trigger obtaining a data formatting rule is two. The trigger point may be set to other values manually/automatically. For example, the trigger point may be based on a predicted accuracy of applying the data formatting rule to other similar data items within the document. In some cases the trigger point may be one, whereas others may be three or
25 more.

[0046] When the trigger point has not been reached, the process returns to operation 810 to detect when further edits are made.

[0047] When the trigger point has been reached, the process flows to operation 830, where input/output examples are obtained and provided to a machine heuristic to obtain a
30 data formatting rule. The input/output examples provide examples of data in a before state and an after state relating to the edits of data. For example, when the edits are to existing data, then the input examples are the data before editing and the output examples are the data after editing. When the edits are to a new cell, the output examples are the edited

data in the cell and the input are the data related to creation of the output (e.g. one or more other columns of data).

[0048] Transitioning to operation 840, the data formatting rule is obtained. According to an embodiment, the data formatting rule is a function that receives textual input (e.g. from one or more cells) and produces zero or more results. The data formatting rule is directed at formatting other similar items within the document (e.g. the other cells within a column) to match the edits made by the user.

[0049] Moving to operation 850, the output range is determined. The output range identifies the items to which the data formatting rule is to be applied. For example, the other items may be all or a portion of the cells in a column in which items have been edited by a user and are the basis for the data formatting rule. In some examples, the output range are the cells within the column that are of the same item type (e.g. date, number, address, and the like). In other examples the output range are all the cells with values that are adjacent to each other, and that are adjacent to the edited cells.

[0050] Flowing to operation 860, the data formatting rule is applied to each of the items in the determined output range. Any results produced by applying the data formatting rule may be temporarily stored before making any changes to the document.

[0051] Transitioning to decision operation 870, a determination is made as to whether application of the data formatting rule resulted in accurate results. According to an embodiment, the accuracy is estimated by a number of results returned by the data formatting rule when applied to an item. When the number of results for an item is zero, the data formatting rule did not have enough data to generate a result. When the number of results is greater than one, the accuracy of the results may be questionable. When the number of results is one, then the result is likely accurate. The number/percentage of cells estimated to have an accurate result may be used to determine when a confidence threshold has been exceeded (e.g. > 70%, 80%, 90%). When the confidence level is not exceeded, the process returns to operation 810 to detect more edits. Generally, the more examples obtained, the more accurate the results. When the confidence level is exceeded, the process flows to operation 880.

[0052] At operation 880, the document is updated with the results created by applying the data formatting rule to each of the items. For example, the cells having a single result are updated with the result. The cells having a different number of results may be marked with an error indicator as discussed above. A reviewing user interface element may also

be displayed that allows a user to perform various operations relating to the application of the data formatting rule.

[0053] The process then flows to an end block and returns to processing other actions.

[0054] The above specification, examples and data provide a complete description of the manufacture and use of the composition of the invention. Since many embodiments of the invention can be made without departing from the spirit and scope of the invention, the invention resides in the claims hereinafter appended.

WHAT IS CLAIMED IS:

1. A method for formatting data based on edits, comprising:
determining when edits are made to different items within a document,
wherein each of the different items are related;

5 obtaining a data formatting rule based on the edits;
automatically applying the data formatting rule to items within the
document that are the same type of data; wherein the data formatting rule attempts to
format the items to a format as defined by the edits made to the different items; and
displaying the items reflecting the application of the data formatting rule.

10 2. The method of Claim 1, wherein obtaining the data formatting rule based
on the edits comprises submitting information relating to each of the edits to a machine
learning heuristic that creates the data formatting rule.

3. The method of Claim 1, wherein the document is a spreadsheet document
and wherein the edits are made to different cells within a same column of the spreadsheet.

15 4. The method of Claim 1, further comprising displaying a graphical user
interface next to at least one of the items formatted by the data formatting rule that when
selected provides options for performing operations relating to the formatted item,
including options for undoing the formatting, redoing the formatting, stopping the
formatting, reviewing potential errors and ignoring errors.

20 5. The method of Claim 1, further comprising displaying an indicator with the
formatted item when a confidence level is below a predetermined threshold.

6. The method of Claim 1, wherein applying the data formatting rule to the
data items comprises applying the data formatting rule to data items within at least one of
a same column and a same row.

25 7. The method of Claim 1, wherein determining when the edits are made to
items of the same type of data comprises determining when edits are made to a first
column that includes data that is also included in a second column and a third column.

8. A computer-readable storage medium storing computer-executable
instructions for formatting data based on examples, comprising:

30 determining examples from different items within a same column of a spreadsheet
document;

obtaining a data formatting rule based on the examples, wherein obtaining
the data formatting rule based on the examples comprises submitting information relating

to each of the examples to a machine learning heuristic that creates the data formatting rule based on the examples

automatically applying the data formatting rule to items within the same column of the spreadsheet document; wherein the data formatting rule attempts to format the items to a format as defined by the examples ; and

displaying the items reflecting the application of the data formatting rule.

9. A system for formatting data based on edits, comprising:

a network connection that is configured to connect to a network;

a processor, memory, and a computer-readable storage medium;

an operating environment stored on the computer-readable storage medium and executing on the processor;

a display;

a spreadsheet application;

a spreadsheet; wherein the spreadsheet comprises items that are arranged in rows

and columns; and

a formatting manager operating in conjunction with the spreadsheet application that is configured to perform actions comprising:

determine when edits are made to different items within a same column of the spreadsheet;

obtaining a data formatting rule based on the edits;

automatically applying the data formatting rule to items within the same column of the spreadsheet document; wherein the data formatting rule attempts to format the items to a format as defined by the edits made to the different items within the same column of the spreadsheet;

displaying the items on the display reflecting the application of the data formatting rule; and

displaying a graphical user interface next to at least one of the items formatted by the data formatting rule that when selected provides options for reviewing formatting changes.

10. The system of Claim 9, further comprising displaying a user interface element in the same column when a confidence level is below a predetermined threshold.