DiscoverText - Distinguishing features

This document should be read in conjunction with the 'Choosing a CAQDAS Package Working Paper' which provides a more general commentary of common CAQDAS functionality. This document does not provide an exhaustive account of all the features and functions provided by DiscoverText, but is designed to highlight some distinguishing elements. The Comment section at the end details our opinions on certain aspects of functionality and usability. Thanks to Dr. Stuart W. Shulman for checking this document for accuracy.

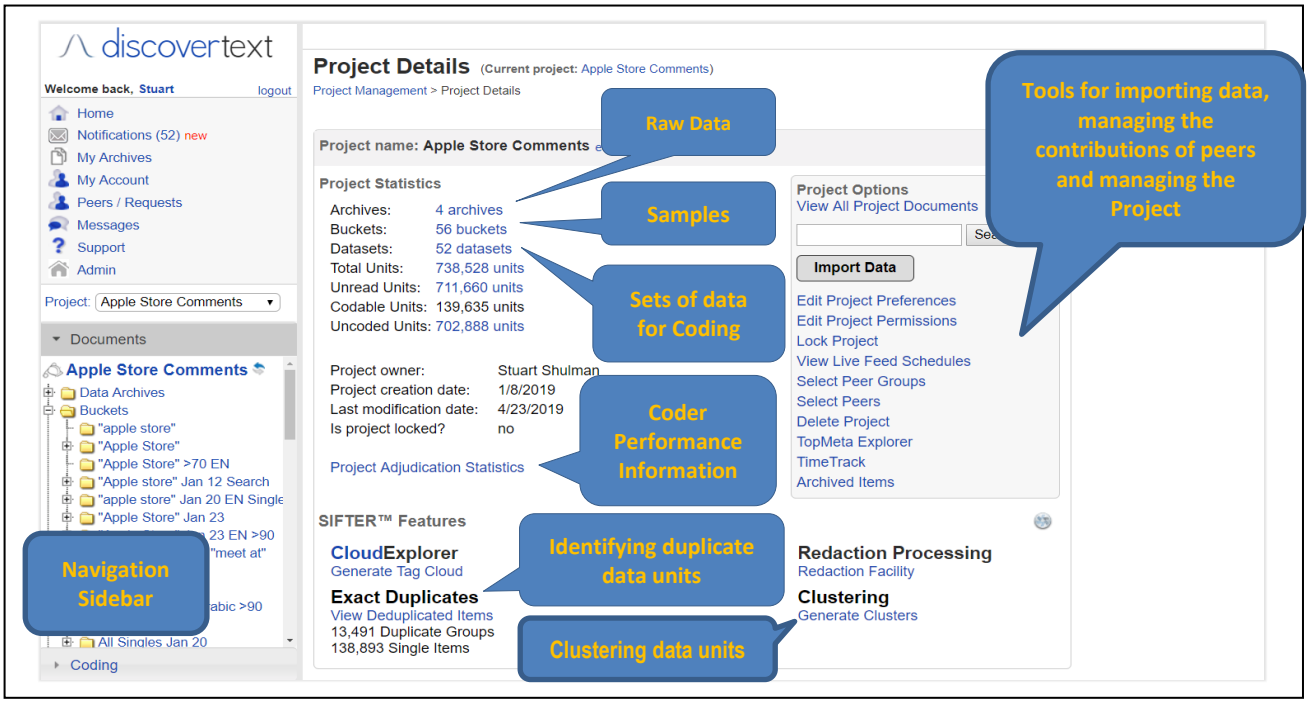
Background

https://DiscoverText.com DiscoverText is a web-based, collaborative text analytics software application developed by Dr. Stuart Shulman under the auspices of U.S. National Science Foundation-funded academic research into sorting public comments on proposed federal regulations ■ DiscoverText was released in 2009 and enables analysis of diverse medium- and large-scale text data and associated metadata. It combines human interpretation and machine-learning principles to perform text classification ■ Underlying DiscoverText is the principle that combining what humans and computers each do best leads to powerful and robust analysis of text. Consistent iteration between humans and computers ("Active Learning") increases the ability of both to adapt to particular classification challenges ■ The predecessor to DiscoverText was the Coding Analysis Toolkit (CAT), a free, open source, web-based text analysis service for coding and annotating text data in teams that Dr. Shulman designed to enable validation of coding performed using ATLAS.ti. https://cat.texifter.com/

Minimum system specifications (recommended by developer)

DiscoverText is an online application supported by common internet browsers (Internet Explorer, Firefox, Safari, Chrome, Opera) ■ The developers recommend optimum browser performance requires high-speed internet access, that cache be cleared every 10-14 days and that users close unnecessary other browser tabs when running DiscoverText.

Figure 1. Project details page in DiscoverText



Structure of work in DiscoverText

The DiscoverText management interface comprises the Project, Archive, Bucket, and Dataset "Details" pages and the Navigation Sidebar. To start working, users create and name new Projects ■ The Project Details page provides an overview of information relating to the currently loaded project. As particular areas are selected the information shown changes accordingly ■ The Navigation Sidebar provides access to global navigation links and expandable project-specific navigation. For example, the Data Archives link leads to a list of raw data currently loaded in the project ■ *Data Archives* are collections of raw data that are either manually uploaded to Projects, or pulled in using an Application Programming Interface (API) ■ Each Project can have an unlimited number of Data Archives from different sources, which can be organised into user-specified sub-folders. Data Archives are static when imported from external files (e.g. from spreadsheets, text files etc.), and dynamic when collected from live social media feeds (e.g. Twitter, Facebook, LinkedIn) ■ *Buckets* are refined archives representing specialised sub-sets (or "samples") of raw data

created from one or more Data Archives. Buckets can be created in several ways, including by searching, filtering, coding, detecting duplicates, clustering and machine classification ■ *Datasets* are collections of raw data prepared for human coding, validation and machine classification. They can contain raw data from whole Data Archives or parts of them (e.g. Buckets) ■ The Coding navigation provides access to tools for coding Datasets, creating and managing Codesets, validating the coding of Datasets and adjudicating validations ■ The Analytics / Export Navigation provides access to generating reports about and exporting different aspects of the project. There are three types of report – Project Reports, Dataset Reports, and Archive Reports ■ The Tools Navigation provides access to additional features relating to the account, previous searches, the Redaction Facility and Live Feed Scheduler ■ The Help Navigation provides access to support materials (video tutorials and written articles) for using DiscoverText and direct access to reporting issues with its use ■ Choosing an option from one of the Navigation areas changes the display on the Homepage and the features available.

Data types and formats in DiscoverText Data files are called Documents within DiscoverText and can be imported into Projects in several ways ■ Data Files are split into Units upon importing. Units are data segments that are analysable ■ Social media content can be automatically imported according to user-specified search terms via live feeds from Twitter, Facebook and LinkedIn. Links to search terms for accessing Twitter content are provided and up to 100 fetches of Twitter data can be scheduled (e.g. every 15 minutes, every hour, every day, every 7 days etc.) ■ Text data can be uploaded from MS Word, and from PDF files with associated metadata (via .zip format) ■ Survey-type data can be uploaded from spreadsheets (.csv format), or directly from Survey Monkey (via an authorised account) ■ Email collections can be imported from Outlook (via.pst format) ■ Importing from Twitter, Facebook and LinkedIn requires authorising the social media account through DiscoverText.

Handling multimedia data in DiscoverText Direct import of standalone multimedia data files (e.g. image, audio and video file formats) into DiscoverText is not currently possible, but images and videos posted within tweets are visible and thus can be analysed as part of each data unit (see Figure 2).

Data organisation in DiscoverText Data can be organised in DiscoverText on several levels ■ Data Archives are used to organise raw data. They are either created as part of the import process or empty Data Archives can be created up-front and data imported into them retrospectively. Data Archives need not be discrete in terms of data type – i.e. data can be imported into a Data Archive that already contains data of a different type ■ Buckets are another possible level of data organisation, used to create meaningful sub-sets of the raw data held in Data Archives ■ Datasets are created in order to human code and subsequently verify and machine classify Data Units. Random samples can be automatically created from datasets for processing ■ Metadata are factual information pertaining to data units. Metadata values can be of various types (numeric, categorical etc.). Metadata linked to Data Units can be used to create specialised Buckets in several ways – e.g. by filtering Data Units using simple searches for (combinations of) keywords or using advanced filters which allow the identification of Data Units based on combinations of metadata criteria (e.g. type, date, annotations, coding etc.). Units identified by searching and/or filtering can be assigned to new or existing Buckets.

Coding schema in DiscoverText Codes are organised into Code Sets which can be flat or hierarchical ■ Codes can be defined and be given a colour attribute. In addition, users can specify short-cut keystrokes for quick coding ■ It is possible to choose an existing Code Set created for a previously analysed Dataset, or to create a new Code Set to apply to a Dataset ■ Code Sets can be imported and exported in xml format

Human coding processes in DiscoverText Codes can be applied to Data Units by individual users or peers collaborating on a DiscoverText Project in several ways (see below for more information about peers in the Teamworking section) ■ Having multiple humans involved in coding underlies the principle of combining human and machine coding ■ Existing codes can be applied by humans using checkboxes from a List, or using short-cut keystrokes ■ New codes can be created as additional concepts are identified in Data Units ■ Data Units can be filtered using search terms and/or metadata criteria and codes applied to all or selected units en-masse (called Batch coding) ■ Codes are applied to whole data units ■ Any number of codes can be applied to individual Data Units ■ Datasets can be automatically sampled in order to create random sub-sets for human coding

Coding adjudication and machine learning in DiscoverText Human coding is validated via an adjudication process, which enables the accuracy of the coding of multiple peers involved in a Project to be verified ■ Interpretively inaccurate human coding choices are identified by humans, and are then excluded from the training data delivered to the Active Learning Machine ■ DiscoverText can subsequently statistically analyse how often human coders are accurate, and the application learns from the adjudication process, converting valid observations for machine learning and subsequent automatic classification.

Figure 2. Illustrations of some processes in DiscoverText

Machine classification in DiscoverText After human coding and adjudication on a sub-set of Data Units within a Dataset, Sifter - a clustering engine based on document similarity – can be used to automatically code the remaining Data Units ■ Custom machine classifiers (called 'sifters') can be created in order to identify and cluster relevant content ■ Machine classification in DiscoverText is based on the principle that it is more efficient and accurate to train a machine to identify relevant content, rather than to rely on searching content using Boolean operators that will produce some erroneous content.

Closeness to data and interactivity in DiscoverText The Navigation Sidebar is always on view making it easy to flick back and forth between different areas of the application, and there is good interactivity between the different aspects of work within DiscoverText ■ The basic analysable segments of qualitative data in DiscoverText are called Data Units (sometimes referred to as items). When viewing and coding Data Units the metadata pertaining to them is also displayed ■ When viewing Twitter data DiscoverText uses the Twitter interface and displayed tweets are "live" – meaning that, as tweets are liked and retweeted the information updates within the DiscoverText interface ■ There are various display options for listing data units within Data Archives, Buckets and Datasets - allowing for them to be viewed and worked with in many different ways ■ The Cloud Explorer displaying single terms, bigrams and trigrams (as well as a tag cloud visualisation) is fully interactive, allowing terms to be accessed within their data unit context (Key Word In Context (KWIC) functionality) as well as to delete and merge items, and build more complex linguistic models

Basic retrieval of coded data in DiscoverText Retrieval of coded Data Units based on the application of individual codes is enabled by running a Coding Report which displays information relating to coding in different charts ■ The Overview Chart displays frequency and percentage information for codes in charts (pie, funnel or pyramid). The Metadata Breakdown Chart displays coded units according to user-specified metadata values in charts (stacked, column, line, spline) ■ Charts are interactive to coded data units.

Writing tools in DiscoverText Writing in DiscoverText is enabled by the Annotation feature. Annotations are comments written about and linked to Data Units ■ Existing Annotations are indicated in Dataset Lists and when

viewing Data Units ■ Annotations can be shared amongst peers collaborating on a Project, or remain private to individual users ■ Annotations can be aggregated and exported in several formats

Linking devices in DiscoverText Linking tools in DiscoverText refer to being able to access the original source file from which a Data Unit derived – e.g. accessing a tweet live on the Twitter website, or accessing unit within its source file outside of DiscoverText (e.g. for PDF files).

Visual tools in DiscoverText There are several visual tools for exploration and representation of results within DiscoverText ■ The Cloud Explorer provides a visualisation of frequently occurring words across a Dataset and can be customised in several ways (e.g. to display as a list or cloud, to display single terms, bigrams or trigrams, to colour particular terms, or remove them from the display, and to display user-specified number of terms). The results are interactively linked to data units via the Search function ■ Interactive Charts can also be generated on the basis of coding (see above).

Searching and interrogating the database in DiscoverText Archives, Buckets and Datasets have details pages providing access to various searching and interrogation features ■ Filtering is the key way of identifying and grouping Data Units and can be based on combinations of various criteria, including keywords, metadata values, annotations, file types, and applied codes. Different ways of filtering can be used discretely or in combination ■ The Detect Duplicates feature provided by Sifter automatically identifies exact duplicate and near duplicate units within Archives, which can then be excluded from a Dataset for analysis ■ The Redaction Facility provided by Sifter allows sensitive information contained within documents to be identified and hidden for publication without destroying the information. Redacted words can be identified manually by creating user-defined keyword lists, or automatically using automated redaction sets. Three styles of redaction – highlight only, blackout, or remove text.

Output in DiscoverText Output is enabled by DiscoverText by generating Reports ■ There are three types of Report ■ Project Reports provide an overview of a whole Project in Summary form, based on the history of a Project, or generating an overview of Annotations ■ Dataset Reports provide an overview of analysed Data Units – in the form of a Dataset Coding report, a Dataset Annotation report, or a summary of Dataset Annotations ■ Archive Reports focus on information relating to Data Archives and include the Exact Duplicates report, Cluster Summary report, and Archive Cloud Explorer report ■ Via Reports raw data, human coded data and machine classified data can be exported ■ Aggregate reports and filtered reports are interactive – meaning visualisations can be drilled into to see the raw data behind

Team-working in DiscoverText DiscoverText is an analytic network designed for concurrent collaborative analysis ■ The Peer Network allows collaborative working in real-time or asynchronously. Peers are invited to collaborate on the coding and adjudication of Datasets. Peers can be organised into Groups when for example a small number of trusted analysts are required to work together on a project. Alternatively, crowd-based collaboration can take place via DiscoverText. Enabling concurrent team-working is thus at the heart of DiscoverText in that it was designed as a quality control system to enable the system to learn from range of human coders, all of whom have different expertise levels for a given dataset ■ Three options for different peer-based coding strategies: standard coding where every peer codes all the data units; assigning each peer a range of data units to code; and prompting peers to code as yet un-coded items ■ Advanced options for coding Datasets include choosing whether to turn on a verification step for coders (which forces coders to check each coding before proceeding), allow user-defined codes, allow multiple codes to be applied to each data unit, whether to use a hierarchical code set, and where applicable, whether to order the Dataset by unit date stamp

Comments on DiscoverText

DiscoverText is a unique platform *for enabling groups of human analysts to work together and combine the value of human interpretation with machine learning and automatic classification, and it has several features not found in other CAQDAS-packages. Firmly rooted in an inductive qualitative tradition, Coding Analysis Toolkit (CAT) and DiscoverText have uniquely attended to the issues of volume, teamwork, and organization in its tools, paying particular – and powerful - attention to the value of combining the benefits that humans and computers can bring to qualitative data analysis.*

As a web-based application users needn't install software*, although will only be able to access projects when connected to the internet. The pricing structure is different from other similar tools and may offer added flexibility to some users. Some users may be concerned about uploading sensitive data to the DiscoverText server, although the developers have gone to a great deal of effort to ensure data security. Information on this can be accessed* ***here***

Features for facilitating collaborative analysis amongst 'peers' are highly developed. *There are many flexible options to suit the requirements of differently organised teams and to enable team members with different backgrounds and levels of expertise to collaborate efficiently, effectively and quickly.*

The validation of human coding via the Adjudication feature is unique*. It is easy to use and operates as a second level of coding allowing users to learn from one another and generate the best possible training dataset for machine learning and subsequent automatic machine classification. This feature is the key getting the most out of DiscoverText and succeeds in balancing the value of human interpretation and machine classification. In addition, the interrogations and Reports for comparing human coding are flexible and refined.*

The ability to automatically identify and remove duplicate and near-duplicate data units is unique *and incredibly useful when working with large volumes of data. Together with the multitude of ways to filter data units and automatically create random samples DiscoverText provides a great deal of flexibility for analysing subsets of very large datasets for specific analyses.*

The minimum segment of text to which codes can be applied is the unit. *This is beneficial when working with large volumes of short texts that can meaningfully be analysed as discrete units – such as tweets and free-text responses to open ended survey questions – because it is easy to code large volumes of data very quickly. However, it also means that DiscoverText is less flexible than other CAQDAS packages for text data where units are larger because codes and annotations cannot be applied to parts of data units. For example, if working with interview data the whole transcript would be imported as a unit and could thus only be coded as a whole. The workaround would be to import a transcript as many small units but this would result in losing the sense of the whole conversation which would be problematic for some methodologies.*

There is good integration between qualitative text and quantitative metadata characteristics*. In particular, the ability to view metadata values associated with data units whilst coding and retrieving is unique and analytically very useful. In addition, the embedded use of the live Twitter display is useful.*

The combination of search and advanced filters is extremely powerful *for creating and analysing subsets of very large datasets. The ability to create systematic random samples of data units within datasets at any time is unique amongst the CAQDAS packages reviewed by us, and is incredibly powerful and analytically useful, especially when working with large volumes of data, such as social media content*

Writing and linking tools are less well-developed *(at time of writing) than comparable functions in other CAQDAS packages. As are the facilities for analysing multimedia data, other than images or videos embedded in tweets. The lack of a mapping tool to visually represent connections between and within data may be seen as a limitation by some users.*

There is a good range of visualisations *based on the words contained within texts (e.g. The Cloud Explorer) and based on how data units have been coded (e.g. Dataset Coding Summary Report). In particular, the Cloud Explorer provides immediate access to bigrams and trigrams in frequency-based list as well as single terms in a tag-cloud visualisation, which is*

The range of options for querying and visualising data units *based on how they have been coded, and in relation to metadata values (e.g. using Boolean and Proximity operators), are limited in comparison to some other CAQDAS packages.*

Further reading

- Chi-Jung Lu & Stuart W Shulman (2008) ***Rigor and flexibility in computer-based qualitative research: Introducing the Coding Analysis Toolkit***, International Journal of Multiple Research Approaches, 2:1, 105-117, DOI: 10.5172/mra.455.2.1.105
- Shulman S (2011) ***DiscoverText: software training to unlock the power of text***. Proceedings of the 12th Annual International Digital Government Conference: Digital Government Innovation in Challenging Times, pp373-373
- For a list of scholarly articles and other published mentions of DiscoverText, see the publications page https://discovertext.com/publications/ and scholarly citations of Coding Analysis Toolkit (CAT) can be found at https://discovertext.com/2018/03/31/scholarly-citations-of-the-coding-analysis-toolkit/