Visual Knowledge Discovery in Dynamic Enterprise Text Repositories

Vedran Sabol, Wolfgang Kienreich, Markus Muhr, Werner Klieber, Michael Granitzer
Know-Center
{vsabol|mmuhr|wklieber|wkien|mgrani@know-center.at}

Abstract

Knowledge discovery involves data driven processes where data is transformed and processed by various algorithms to identify new knowledge. KnowMiner is a service oriented framework providing a rich set of knowledge discovery functionalities with focus on text data sets. Complementing results of automatic machine analysis with the immense processing power of human visual apparatus has the potential of significantly improving the process of acquiring new knowledge. VisTools is a lightweight visual analytics framework based on multiple coordinated views (MCV) paradigm designed for deployment atop the KnowMiner's service architecture. In this paper we briefly present both frameworks and, driven by real-world customer requirements, describe how visual techniques can be synergistically combined with machine processing for effective analysis of dynamically changing, metadata-rich text documents sets.

1. Introduction

In today's knowledge driven, competitive society enterprises have to deal with large, heterogeneous document data bases. Depending on the industry sector these include documents such as scientific publications, patents, project documentation, technical reports, news, books etc. These repositories have a pronounced dynamic behaviour with documents being added, removed and modified at a high rate. In contrast to the Web, which is typically comprised of shorter, simpler documents, enterprise repositories mostly contain large, complex documents including rich metadata. Various document management systems are capable of handling storage and delivery; however, they do not meet the requirements of competitive intelligence, which include the abilities to explore, analyse and understand patterns, and to discover and communicate knowledge hidden in the underlying documents.

To improve the effectiveness of knowledge discovery it is crucial to include both the humans and the machines in the analytic process. If massive amounts of information processed by machines are presented in a convenient visual representation, human visual capabilities allow users to quickly discover, understand and explore complex patterns and relationships, and immediately apply their knowledge and creativity. Besides applying human perceptual abilities on large data sets the involvement of humans in the knowledge discovery process has several other advantages: experts can deal with noisy, ambiguous, conflicting or incomplete data, can apply explorative examination even when the data is poorly understood or the goals are vaguely defined, and are capable of adjusting their strategies and objectives on-the-fly based on their experience and intuition.

In this paper we apply visual knowledge discovery to realise workflows providing the enterprise knowledge worker with means for analysis of topical, temporal and metadata-related information as well correlations and interdependencies between these. Examples based on common customer requirements, which were realised with KnowMiner and VisTools frameworks, are presented. Evaluation results of critical visualisation aspects as well as user feedback were integrated in the system. These are discussed at the end of the paper and provide valuable hints for further refinement.

1.1. Related Work

Combining machine processing power with human visual capabilities is not a new idea [16] . In fact, visual analytics, an emerging interdisciplinary field focusing on reasoning facilitated by interactive visual interfaces [17] has also been defined as a combination of automated discovery and interactive visualization [3] .

Information landscapes have been traditionally used to visualize complex relationships in large text repositories, for example in systems such as IN-SPIRE [7] . While conceptually close to IN-SPIRE, we focus on the analysis of change in document sets as well as on correlation between metadata and topical and temporal aspects of the data. The concept of information landscape has been extended to hierarchically organized repositories in InfoSky [1] . For today's rapidly changing, dynamic repositories different approaches for visualizing temporal information play a growingly important role [10] . Visual representations such as ThemeRiver [2] have been applied on text data sets to visualize temporal developments of topical clusters. Information landscapes with dynamic topology have been applied in WebRat [12] for small search result sets.

Their application on dynamically changing repositories has been proposed in [12] and [15] , where adjustments and shifts in the topology should reflect changes of the data set. A discussion on techniques for visualization of topical and temporal aspects of dynamic text collections can be found in [14] .

2. KnowMiner Framework

KnowMiner [4] is a knowledge discovery framework based on a service-oriented architecture implemented in Java. Its main role is to automatically extract knowledge from large, heterogeneous text document repositories. The definition of the services was driven by the goal of maximum flexibility on one side and on the other side constrained by the intent of minimizing the complexity of orchestration for typical workflows and making services independent of the underlying algorithmic details.

2.1. Service-oriented Architecture

KnowMiner's data model revolves around a basic object called information entity. It typically corresponds to a document and carries content as well as additional information including:

- **Metadata**, which is additional data attached to an information entity, such as creation date or size.
- **Annotations,** which are automatically identified by information extraction methods to add further information to a part of the content.
- **Features vectors**, which contain statistical information on annotations and metadata, such as the frequency of nouns. Vector representation allows for comparison (i.e. similarity or distance computation) of any pair of information entities.
- **Relations** between information entities embedding them into a graph structure. For this purpose a Resource Description Framework (RDF) is used.

A powerful detail of KnowMiner's data model is the ability to maintain feature vectors in different vector spaces (such as person, geo-spatial or topical vector spaces) for every information entity. A flexible metrics definition allows algorithms to operate on any of these spaces, effectively analysing orthogonal aspects of the data set. Furthermore, several aspects of the data set can be considered at once depending in their importance. This is achieved by the metrics combining and weighting the influence of different vector spaces.

Typically, initial data provided by documents and carried by information entities is the content and some basic metadata. During the execution of services, algorithms manipulate information entities and their properties (e.g. features and annotations), create new information entities (such as clusters) and identify relationships between them (for example parent-child relationships between clusters and documents). Currently KnowMiner includes following services:

Import service gathers data from external data sources and transforms various formats into an internal meta-format under preservation of document content, structure and metadata.

Information extraction service annotates the text content making implicit information from unstructured content explicit. It performs text transformations such as case folding, stopword filtering and stemming, decomposes text into sentences, phrases and tokens, performs part-of-speech tagging and extracts named entities such as persons, places or organizations.

Feature extraction service identifies meaningful features (such as terms and named entities) from extracted annotations and metadata, and computes feature vectors for documents.

Summarization service computes a descriptive summary for a document in form of keywords from one or more high-dimensional vector spaces.

Indexing and search services: Indexing service creates a full-text and metadata search index. Search service executes search queries to quickly select relevant document sets. It provides comprehensive searching capabilities including: full text search, metadata search, range search, wildcard search, fuzzy search, Boolean search, search by example, relevance feedback, etc.

Associative indexing and associative search services: Indexing service associates terms between search index fields depending on their distributions and co-occurrences in the corpus (for example persons can be associated with locations). Search service executes search queries to identify and return related terms. These terms can be, for example, used for query expansion.

Classification and classification training services: Classification service classifies new, unseen documents to the known set of concepts. Training service learns specified concepts using documents as examples and stores the learned classification hypothesis.

Clustering service identifies groups of related documents depending on the similarity of their vectors.

Projection service performs a dimensionality reduction of the high-dimensional document vector space. The result is a low-dimensional layout where high-dimensional relationships (i.e. distances or similarities) are preserved as well as possible.

Persistence service is a high-performance storage solution optimized for knowledge discovery tasks.

2.2. Selected Algorithm Details

Every KnowMiner service encapsulates a variety of algorithms. Describing them into details is beyond the scope of this paper (see [6] for further references). For the purpose of illustrating visually supported analysis of topical, temporal and metadata-related aspects of text data sets clustering and projection algorithms are of particular interest:

Clustering algorithms: Clustering service encapsulates a variety of clustering algorithms such as k-means, ISODATA, hierarchical agglomerative clustering, affinity propagation, BIRCH etc. (see [18] for more information). The choice of the algorithm is performed automatically depending on the size of the

data set as well as depending of the specification of the clustering tasks to be performed, such as constraints on the minimum and maximum number of clusters, whether a single level or a hierarchy of clusters should be computed, performance vs. quality considerations etc. For browsing large data sets the clustering service can generate what we call a "virtual table of contents" by computing a hierarchy of clusters through recursive application of the k-means algorithm. Bisecting K-means combined with cosine similarity is known to perform well for text data [19] . For usability reasons minimum and maximum number of children at each hierarchy level is limited, typically to 3 and 10 respectively. A strategy for splitting and merging of clusters is applied attempting to determine the optimal amount of children at each hierarchy level.

Projection algorithms: For small data sets a simple projection algorithm based on a force-directed placement (FDP) algorithm is an adequate solution. As this algorithm does not scale we make use of the hierarchy produced by the clusterer to reduce the time complexity [1] . The recursive, hierarchical projection algorithm begins with top level clusters and applies the force-directed placement (FDP) to position cluster centroids in 2D. In the following step a Voronoi area subdivision is performed on the projected centroids to assign polygonal areas to the clusters. Sub-clusters of every top level cluster are also positioned with FDP and inscribed within the areas of their parent clusters. The algorithm proceeds recursively in this way until, at the bottom of the hierarchy, document vectors are projected.

Fast clustering and projection algorithms make it possible to process data sets on-the-fly: 10000 abstracts can be processed in under 30 seconds on a 2.5 GHz Core 2 processor using 64bit Java VM (1.6.0_12). Algorithms scale with the time and space complexity of O(n*log(n)), n being the number of clustered documents. 300000 abstracts can be clustered and projected in about ten minutes using less than 6GB memory.

Large, real world data sets often have a pronounced dynamic behaviour – data elements (documents) are added, removed and modified. In order to deal with dynamic data sets clustering and projection algorithms are incremental meaning that once a cluster hierarchy and a corresponding landscape have been computed, changes of the document set can be incorporated into the previously computed results without recomputing everything from scratch. This not only saves significant computing time, but also plays a crucial role for recognition of known, unchanged parts of a cluster hierarchy or a 2D projection. If for a change in the data set the computation was performed from the beginning it is likely that the new result would not look very much alike the previous one. The reason is that clustering and projection algorithms (basically optimisation problems searching for a local minimum) are sensitive to initial conditions. A small change in the initial configuration of data might lead to a different local minimum and therefore to a significantly different hierarchy and layout configuration. Incremental algorithms are capable of modifying a previously computed configuration and incorporate changes into it so that amount of incorporated changes approximately corresponds to the amount of changes in the data set. Parts of the hierarchy and of the layout not affected by the changes of the data set will most likely be left untouched. This is of critical importance for users browsing recognisable cluster hierarchy and projection of a known, dynamically changing data set. Note that algorithm details and evaluation results will be published in a separate paper.

3. VisTools Framework

VisTools is an information visualisation framework based on the coordinated multiple view (CMV) paradigm. It is designed to provide visual analytics capabilities atop KnowMiner's knowledge discovery functionality. VisTools is implemented in Java and, where performance is critical, makes use of 3D acceleration through the JOGL library. Note that other toolkits bearing the same generic name are not related to VisTools described in this paper.

Enterprise document sets are characterized by the diversity of information types they contain, such as topical, temporal and geospatial information, variety of different metadata types etc. To address various aspects of the data VisTools provides several specialised visual components and also provides means for combining and integrating them into consistent, explorative user interfaces. Currently available visual components focus on the following aspects of the data set:

- **Relatedness** between visualised items (i.e. documents, clusters) as well as their **quantitative distribution and cohesion** are conveyed by an information landscape component (Landscape3D).
- **Changes and trends** in the data set are handled by the StreamView time visualization component, which displays temporal changes of document clusters.
- For **metadata and high-dimensional data** a Scatterplot representation is suitable. Mapping of metadata onto visual properties of items visualised in an information landscape is also possible. A cooordinated table widget is available too.
- **Hierarchical structures** are be presented by a coordinated tree widget, as well as nested areas in the information landscape.

Each visual representation employs a specialized visual metaphor restricted to revealing relationships and patterns only for one, or a small amount of data aspects. In cases when the analysis of a data set necessitates considering more than just a single aspect of the data, visual tools capable of simultaneously handling many different aspects of the data are required. To achieve this VisTools components are built along the lines of the multiple coordinated view paradigm [9] . Through view coordination tight coupling of several visual components can be achieved effectively "fusing" them into a single unified, coherent user interface. Interactions performed in one component are immediately reflected in all components within the GUI.

3.1 Architecture

Data which shall be visualised is provided by an external component, such as the KnowMiner framework, and fed into VisTools components. Each visual component maintains a specialised, private data model and also relies on a shared data model for coordination. Shared data model is maintained by a view coordination framework. Coordination is implemented along the lines of the Model–View–Controller architectural pattern: When user interacts with a view, that view notifies the coordination controller communicating the IDs of the concerned items and the description the interaction type (such as a item colour change or visibility change due to zooming). Subsequently the coordination framework writes the changes into the shared data model and notifies all registered visual components so that these can adjust and repaint themselves accordingly.

Following user interactions can be subject to coordination: modifications of selection, colour, icon and size of visualised items, setting the current position in the visualisation, and ensuring visibility of items (e.g. by scrolling, zooming, expanding nodes etc.). A generic broadcasting capability is available for implementing application specific functionality. The coordination framework also offers means for colour, icon and metadata management which simplifies development and ensures that visualised items have the same appearance in all views used within a coordinated GUI.

3.2. Information Landscape

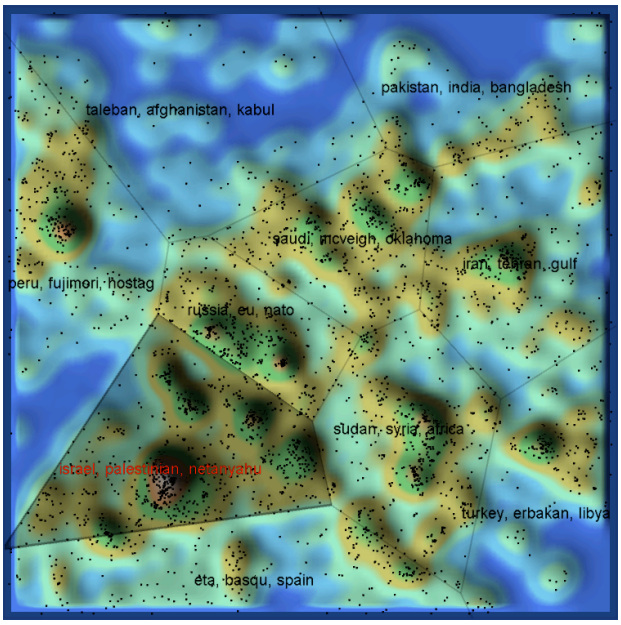

Figure 1: Information landscape showing 3600 Reuters documents, query was "terrorism".

Information landscapes (see Figure 1) are used for analysis of complex relationships in large data sets by conveying relatedness in the data through spatial proximity in the visualisation. We apply an information landscape component to visualise projection results provided by KnowMiner where relatedness between objects is defined as the similarity of corresponding feature vectors. Typically topical relatedness is visualised, but other feature vectors, such as person or geographical vectors (or any combination thereof) can be used to visualise other types of relatedness. Cluster hierarchy is represented through nested polygonal areas produced by the Voronoi area subdivision in the projection algorithm. Regions are labelled by highest weight terms of cluster centroids. At the bottom of the hierarchy single documents are visualised as tiny icons. The landscape also conveys the size and cohesion of clusters. Hills represent groups of related documents and emerge where the document count (density) is large. Hills (islands) are separated by sparsely populated areas represented as see. The compactness of the cluster area is an indicator of its cohesion.

Landscaspe3D is an interactive component allowing zooming, panning, rotating and tilting as well as manipulations of visual document properties. Zooming in on a cluster will reveal the areas and labels of underlying clusters as shown in Figure 2. In this way adaptive level of detail, adjusted to current zoom level, is provided in the area currently explored by the user. Cluster labels are also suitable for navigation: clicking on the label will smoothly fly to and zoom in on the corresponding cluster. In this way the information landscape adheres to the principles of the well-known InfoVis mantra ("overview first, zoom and filter, details-on-demand") providing an overview of the whole data set, and on demand offering insight into relationships at finer levels of detail. At the finest level of detail the user can navigate to and manipulate single documents. Searching for and/or filtering of documents is also available, with full KnowMiner search functionality being available.

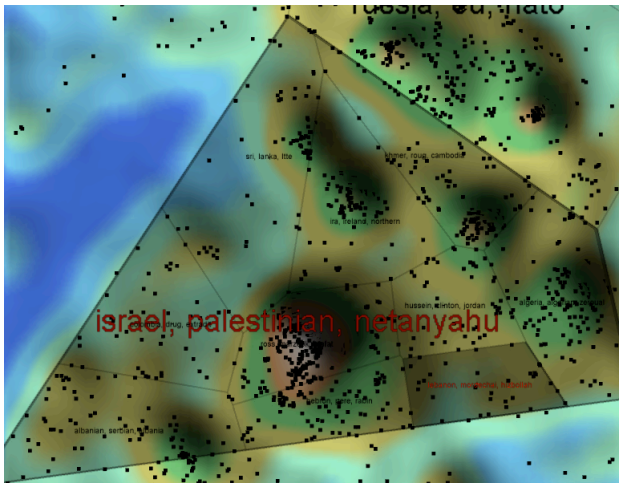

Figure 2: Zoom in revealing sub-clusters.

Information landscape component is configurable (colours, icons, fonts, interactivity etc.) and can be easily adapted to meet the requirements of different usage scenarios. It makes use 3D acceleration to visualise more than a million items in real time on a standard PC with integrated graphics and 1GB of main memory.

3.2.1 Collaboration: Information landscape can be employed as a collaborative platform empowering users to exchange information and communicate what they have discovered. Distinctive visual elements (such as "the elongated cluster up left" in Figure 1) can be easily identified by users. Using bookmarks to mark areas of interest, changing visual properties of items (for example red is irrelevant, green is relevant) and subsequently saving the landscape, allows other users to examine and modify the same data set at a later time point.

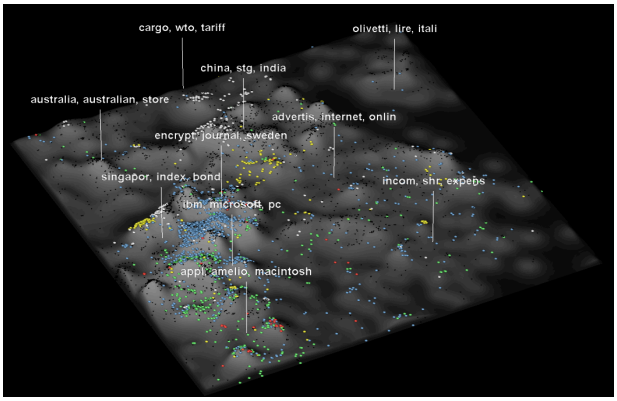

Figure 3: Mapping of document features onto visual properties (in this case colour)

3.2.2 Visual Feature Analysis: Mapping document features and metadata onto visual properties of the visualised visual items, such as colour, icon or size is a powerful feature for analysing metadata distribution over topical clusters. This is illustrated in Figure 3. Visualised are topical clusters of about 6000 documents from 1996-97 on "computer industry". Geographic entities extracted from document content were mapped to colours (note a monochromatic landscape texture for better recognition if item colours). Colour assignments are as follows: New York – blue, California – green, Tokyo – yellow, Boston – red, London – white. Relationships between topical clusters and extracted locations can be recognised immediately: it is obvious that "ibm, microsoft, pc" is connected with New York while "apple, amelio, macintosh" has more to do with California. Cluster distribution of other metadata, such as persons or organisations, can also be analysed in this way.

3.2.3 Dynamic Topology Landscape: Incremental clustering and projection algorithms provide the base for information landscapes with dynamic topology. When a visualized data set is modified corresponding visualised items are not only removed from and inserted into an existing, static information landscape. Rather than that the topology of the landscape is smoothly altered. Old island and hills may disappear or change their shape and position. New islands may arise from the seabed and eventually remain as a permanent addition to the landscape. Other modifications of the topology, such as drifting of hills towards each other (correspond to merging of previously separate clusters) or splitting of an island (corresponds to cluster breakup) may also occur.

Transitions of the landscape topology from an old to a new temporal configuration are incremental and adaptive so that only necessary changes are introduced in the topology: configuration of the parts of the topology, which are little or not at all affected by the modification of the data set, remain stable with respect to their previous position and shape. In this way the user can understand the modified landscape immediately through the recognition and orientation provided by the already known, preserved (or scarcely modified) elements of the topology. These incremental transitions can be smoothly animated by morphing between several incrementally computed landscapes so that the user can follow and understand the changes occurring in the data set.

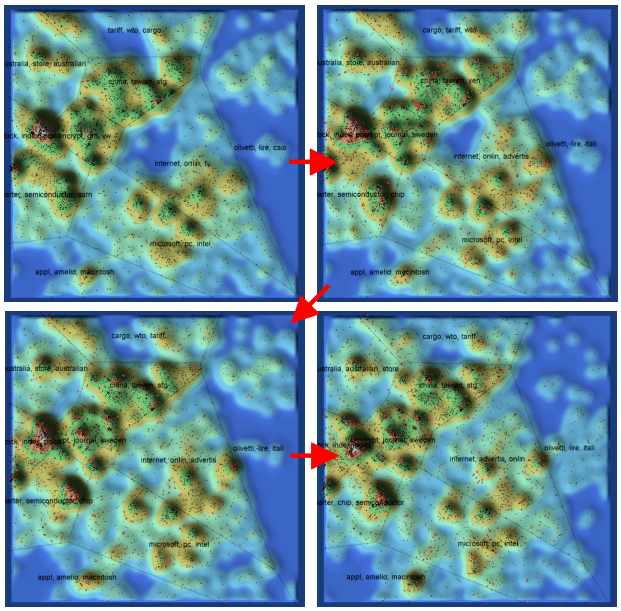

Figure 4: A sequence of growing, incrementally computed information landscapes.

An example of a dynamically changing topology can be seen in Figure 4. On the-top left is the first landscape containing 4382 documents on "computer industry" from 20.8.96 to 19.5.97. It was augmented with documents on the same topic for the following three months producing three additional, incrementally computed landscapes each containing approximately 500 new documents (shown in red). The global configuration and most areas remain recognisable, but several areas experienced changes: for example the sea area in the centre gradually disappears while the rightmost cluster experiences a major internal change.

3.3. StreamView Temporal Visualisation

StreamView component provides visualisation of change for clusters. Typically topical clusters are displayed, however it is possible to visualise documents grouped by any criteria, such as persons or organisations. In Figure 5 temporal development of ten topical clusters is visualised. Each cluster is represented as flow of a coloured stream. Colour assignment is shown in a legend

at the bottom of the component. Above the legend there is a timeline defining the time axis. Above the timeline the central part of the visualisation can bee seen showing clusters as streams stacked over each other. The amount of documents belonging to a cluster is represented by the thickness of the cluster's stream at the corresponding position along the time axis. Scaling can be adjusted by the user with linear, exponential and logarithmic (default) scaling being available. Interactivity includes cluster selection and temporal selection of documents using an interval selection bar.

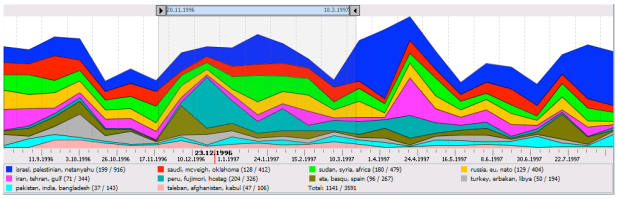

Figure 5: StreamView temporal visualization of topical clusters

The example in Figure 5 shows topical clusters in a data set from 1996-97 on "terrorism". For example it can be clearly seen that clusters "israel, palestinian…" and "eta, baque, spain" show significant continuous activity, with the first cluster being far more intensive. On the other side "taleban, afghanistan…" cluster shows the smallest overall activity. Cluster "peru, fujimori…" has almost no activity until a single large peak occurres, which then fades out into insignificance indicating that that this was a one-time event.

The visualisation can be computed on-the-fly for data sets of significant size: million documents in 10 clusters are processed in less than a quarter of a second on a 2.5GHz Core 2 PC. The component is implemented using Java2D. It is configurable including colours, edge-strokes, alignment, scaling, time segment granularity etc.

3.4 Example of a Coordinated View GUI

For the purpose of demonstrating and testing VisTools visual components and Know-Miner algorithms a demo-application is available. In the example shown in Figure 6 following components are integrated into a single, coordinated visual interface:

- Information landscape showing topical relatedness.
- StreamView showing temporal cluster changes.
- Two TreeViews: one showing topical clusters, the other one showing extracted named entities grouped by their class (i.e. persons, locations etc.).
- Two TableViews (hidden by the Tabbed-Pane): one displaying children of the current cluster ("russia, eu, nato"), the other showing selected documents.

All these views are integrated by a coordination framework to provide simultaneous, "fused" topical, temporal and metadata analysis of the data set. The user can navigate or modify visual properties in any view. Coordination framework works behind the scenes to ensure that the state of visualised items (selection, color, icon and size), as well as the user navigation (item visibility, zoom factor etc.) are consistent over all views. In the example in Figure 6 cluster "russia, eu, nato" was chosen by the user as the current position. As a consequence this cluster is highlighted in the topical TreeView, information landscape and the StreamView temporal visualisation. Also the landscape zoomed in and focused on the cluster, while the tree expanded and scrolled to the corresponding node. Coordination of visual features such as colour and selection of documents can also be seen in the landscape and in both TreeViews.

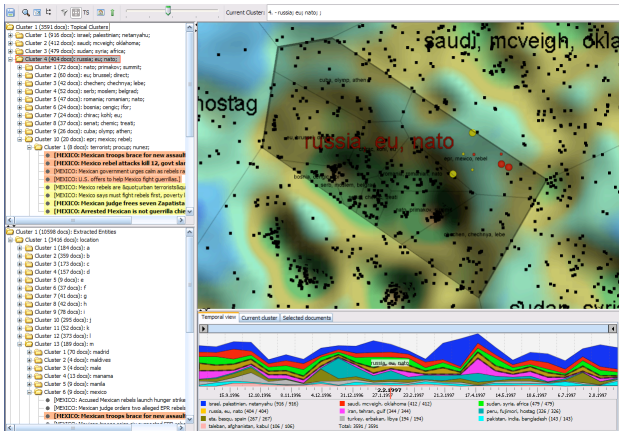

Figure 6: GUI consisting of coordinated views.

3.5. Evaluation

To improve the usability of visual components and applications, we performed following steps: heuristic evaluation of new functionality during the design phase, formal experiments and thinking aloud tests at later development phases, and collecting of user feedback during pilot installations. Our previous experiments [1] suggest that combining an information landscape with tree and table widgets provides tangible advantages for explorative use cases where goals are gaining an overview of the data set and understanding topical clusters and relationships. Thus, we decided to keep that configuration of components and to focus on improving interactive and visual aspects of the combined platform.

In recently performed experiments we focused on two different aspects of the information landscape [8] explorative browsing and perceptibility of visual item properties. The goal of explorative browsing testing was to discover whether automatic, animated fly-in to a selected cluster area provides advantages compared to manual zooming and panning. Users performed a set of three tasks, such as navigating to related clusters and documents, on two different data sets, the first with automatic fly-in and the second with manual navigation.

The goal of the perceptibility test was to discover whether combinations of different colours and shapes were easier to recognize than a single shape in different colours. For example one town was mapped onto a red plus while another town was mapped onto a yellow cross (see Figure 7). Documents mentioning both towns were represented by an overlay of both icons. Alternatively red and yellow spots were used. Users performed two

simple tasks on two different data sets, such as estimating the amounts of items with specific properties within different clusters. Once they used a combination of colours and shapes and once spots in different colours.

Figure 7: Icons used for encoding metadata.

Experiments were performed on a group of 10 users with technical background. Time required to perform each task was measured. After completing the tests user filled out a questionary and told us their impressions. The first experiment revealed that automatic fly-in was slightly better than manual zooming and panning. However, the results were inconclusive because some users would get completely confused in certain situations, for example when the automatic navigation would take them to a completely different part of the landscape. As a consequence we decided to add a switch to the interface for disabling the automatic navigation. The result of the second experiment clearly indicated that, contrary to expectations, using a single shape (spot) in different colours was superior to combinations of shapes and colours. The main reason for that was lower perceptibility of shape/colour combinations (caused in part by 3D filtering).

Performed experiments also revealed a series of usability issues most of which were subsequently solved. After that a pilot installation was performed visual components were tested with selected users in real-world scenarios producing additional feedback, which was (or will be) considered in newer versions of the software.

4. Analytic Applications

Building knowledge discovery applications around KnowMiner and VisTools framework usually includes three major, application specific tasks: connecting to external data repositories, configuring the frameworks, and implementation of application logic. These steps are typically performed by system integrators. Server–client architecture has been the preferred model, but plain desktop tools were also implemented. As of 2009 several applications were developed and deployed in production, addressing knowledge workers dealing with large amounts of documents in fields such as publishing industry, research, patent departments, government etc.

To realise application logic and workflow control third party frameworks have been successfully applied [11] . In Figure 8 two typical workflows can be seen. The workflow shown on the left is the most common one: it is a sequence of services which import, index, extract information from and persist the documents so that these can be retrieved and analysed later on. The workflow shown on the right is an example using combined machine and visual analysis, where the main goal is to identify a set of documents defining a topic (concept) of interest. As this set can contain over 10000 documents it is clear that scanning each document is not an option. Beginning with searching for documents of interest (keyword search, search by example) the user analyses the document set by means of clustering and visualization. Explorative analysis is performed using various criteria offered by the automated analysis and visualisation such as topical groups, relatedness information, temporal information, extracted named entities, key terms, and so on. These allow the user to separate relevant and irrelevant documents (which can be removed) in a group-wise manner. Using collaborative features the analysis can be performed by a group of people. Once the document set is defined it is used to train the classifier and learn a new concept. As later on new documents are imported they are automatically classified and dispatched to users who subscribed that topic. If the user is satisfied with the new documents these can be added to the training set to further improve the classification hypothesis. However, if a user detects too many non-relevant documents, or believes that relevant documents are not identified, the training set can be refined by the previously described analytic procedure. In this way discovered knowledge is fed back into the system to improve classifier performance and adapt it to topical and vocabulary drift.

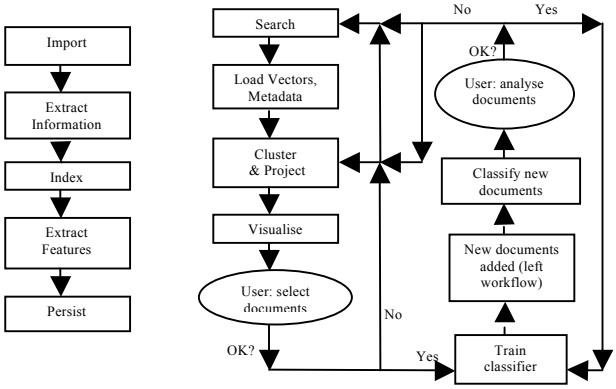

Figure 8: Workflow examples.

5. Future Work

User tests indicate that the current information landscape has occlusion problems for large data sets. Also memory issues arise on the client as the object count grows large. For more than 1000000 visualised objects a different approach is envisioned to address occlusion and to reduce memory and performance requirements of the client. Instead of displaying single items a “cloud” or an aggregation icon shall be displayed in zoomed out state. As the zoom factor changes additional information shall be dynamically retrieved from server and faded in, whereby group-wise item manipulation should remain possible at any zoom level.

Computing the cluster hierarchy in high-dimensional space is superior to clustering in low-dimensional space because the projection inevitably introduces an information loss. As we use Voronoi areas to delineate clusters, hills containing documents from neighbouring clusters may appear along the edges. If these clusters are not very closely related such hills might lack topical

coherence to a certain degree. This issue will be addressed in two ways: to improve the projection force-directed placement algorithm should also consider the neighbouring cluster positions; to eliminate hills containing documents from dissimilar clusters, inscribing of points into Voronoi areas will leave empty space along the borders proportional to the dissimilarity of the neighbouring clusters.

Dynamic topology information landscape is a new feature which has not yet been tested with users and deployed in productive environments. These steps are planned for 2009 with a selected customer. Evaluation results will be published in a future paper. Additional visual components, such as a geo-visualisation and a graph view are also under development.

Conclusions

In this paper we introduced KnowMiner and VisTools, two commercial frameworks developed at Know-Center [5] , and described how these can be applied to offer combined machine and visual analysis of dynamic, metadata-rich enterprise document sets. We presented real-world examples to illustrate how the combination of machine and visual analysis empowers users to quickly gain insight into large data sets, generate hypothesis, verify it by automatic techniques, and finally draw conclusions. To close the analytic cycle, user feedback can be considered for improving the quality of automated analysis.

Incremental clustering and projection methods capturing both the size changes of topical clusters and changes of topical relationships between them, as well as the capability to convey these changes through smoothly animated topology modifications in the information landscape, represent two outstanding features of this work.

Acknowledgements

The Know-Center is funded within the Austrian COMET Program - Competence Centers for Excellent Technologies - under the auspices of the Austrian Ministry of Transport, Innovation and Technology, the Austrian Ministry of Economics and Labor and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency (FFG).

StreamView component and the incremental k-means algorithm with cluster splitting and merging strategies were developed within the RAVEN project, which is financed by the Austrian Research Promotion Agency within the strategic objective FIT-IT.

References

[1] Andrews, K., Kienreich, W., Sabol, V., Becker, J., Droschl, G., Kappe, F., Granitzer, M., Auer, P., Tochtermann, K., The InfoSky Visual Explorer: Exploiting Hierarchical Structure and Document Similarities. *Information Visualization*, 1(3/4):166–181, December 2002.

[2] Havre, S., Hetzler, E., Whitney, P. and Nowell, L., ThemeRiver: Visualizing Thematic Changes in Large Document Collections, *IEEE Transactions on Visualization and Computer Graphics 2002*, 8(1): 9-20.

[3] Keim, D. A., Mansmann, F., Oelke, D., Ziegler, H., Visual Analytics: Combining Automated Discovery with Interactive Visualizations. *Discovery Science*, 2008, pp. 2-14.

[4] Klieber, W. Sabol, V. Muhr, M. Kern, R. Granitzer, M., Knowledge Discovery using the Knowminer framework. In *proceedings of the IADIS International Conference on Information Systems 2009*. Barcelona.

[5] Know-Center (2009): www.know-center.at

[6] Know-Center's Knowledge Relationship Discovery Division publication list (2009): http://en.know-center.at/forschung/knowledge_relationship_discovery/schriftliche_veroeffentlichungen

[7] Krishnan, M., Bohn, S., Cowley, W., Crow, V., Nieplocha, J., Scalable Visual Analytics of Massive Textual Datasets, *21st IEEE International Parallel & Distributed Processing Symposium*. Long Beach, USA: IEEE Computer Society. 2007

[8] Krnjic, V., Usability Evaluation of a Multiple Coordinated Views Application (in German), *Bachelor's Thesis at the Graz Univerity of Technology*. 2008

[9] Müller, F., Granularity based multiple coordinated views to improve the information seeking process, *PhD Thesis*, Konstanz, Germany. 2005.

[10] Müller, W. and Schumann, H. Visualization Methods for Time-dependent Data - An Overview, *Winter Simulation Conference 2003*. New Orleans, IEEE Press. 737- 745.

[11] m2n Intelligence Management Framework (2009): http://www.m2n.at/hm_c/index.htm

[12] Sabol V., Kienreich W., Granitzer M., Becker J., Tochtermann K., Andrews K., Applications of a Lightweight, Web-Based Retrieval, Clustering and Visualisation Framework. *Proceedings of the 4th International Conference on Practical Aspects of Knowledge Management*. Vienna, Austria, 2002.

[13] Sabol, V., Granitzer, M. and Kienreich, W. Fused Exploration of Temporal Developments and Topical Relationships in Heterogeneous Data Sets, *11th International Conference Information Visualisation*. London, UK, IEEE. 2007.

[14] Sabol, V., Andrews, K., Kienreich, W., Granitzer, M., Text mapping: Visualising Unstructured, Structured, and Time-Based Text Collections, *Intelligent Decision Technologies*, Vol 2, No. 2, pages 117-, IOS Press. 2008.

[15] Sabol V., Scharl A. Visualizing Temporal-Semantic Relations in Dynamic Information Landscapes. *GeoVisualization of Dynamics, Movement and Change Workshop at the AGILE 2008 Conference*, Girona, Spain.

[16] Shneiderman, B. Inventing discovery tools: Combining information visualization with data mining. *Information Visualization*, 1(1), pp. 5-12. 2002.

[17] Wong, P. C., Thomas, J., "Visual Analytics". In *IEEE Computer Graphics and Applications*, Volume 24, Issue 5, Sept.-Oct. 2004 Page(s): 20 - 21.

[18] Xu, R., Wunsch, D., Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, Volume 16, Issue 3, May 2005 Page(s):645 – 678

[19] Zhao, Y. Karypis, G., Evaluation of hierarchical clustering algorithms for document datasets, *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, 2002, pp. 515-524