Linnæus University
School of Computer Science, Physics and Mathematics

Degree Project

Visualization of Sensory Perception Descriptions

Mimi Prangova
2010-10-20
Subject: Information Systems
Level: Master
Course code: 5DV00E

School of Computer Science, Physics and Mathematics

Reports from DFM

Visualization of Sensory Perception Descriptions

Student Name: Mimi Atanasova Prangova

Supervisor: Prof. Dr. Andreas Kerren

Linnaeus University

Abstract

Visualization of Sensory Perception Descriptors is a topic in the field of Information Visualization. It is concentrated on the research and development of methods for analyses of data related with human modalities description. One possibility for investigating sensory perception descriptors is analyzing a great number of wine tasting notes.

This thesis is concerned with the visualization of wine tasting notes in order to aid linguistic analyses. It strives to find proper visualizations that will give a better insight into the language used in wine tasting notes. Two main processes are described in the following report. First it sets out the process of researching of different methods of information visualization that led to the final approach for representing the data. A number of concepts for text analyses are discussed and the most useful of them are developed further. Several approaches for text visualization and statistical information are combined to build a system for tasting notes analyses. The second part of the report describes the process of developing a prototype that implements the represented approaches and gives an opportunity for real testing and conclusions.

Keywords

Information Visualization, Geographic Visualization, wine tasting note, word tag, sensory perception descriptor.

Acknowledgements

I wish to express my gratitude to my supervisor, Prof. Dr. Andreas Kerren, for his continuous support through the whole period of working on my master thesis. His encouragement, guidance and responsiveness from the initial to the final level of the project enabled me to develop an understanding of the subject. He introduced me to Information Visualization and gave me the basic knowledge in this field. This thesis would not have been possible without his help and directions.

I would like to thank to Carita Paradis for her collaboration and initiation in the application field. I am grateful for her opportune assistance and involvement in the process of my work and results.

Note: In the report, the word “we” means Andreas Kerren, Carita Paradis and I. The word “I” means myself.

Table of content

List of Figures

List of Tables

List of abbreviations

1. IDE – Integrated Development Environment
2. GUI – Graphical User Interface
3. GeoVis – Geographic Visualization
4. InfoVis – Information Visualization
5. SQL – Structured Query Language
6. DML – Data Manipulation Language
7. API – Application Programming Interface
8. DBMS – Database Management System

1. Introduction

This chapter describes the problem and purpose standing in the basis of this thesis. It also contains background information and related work description that the reader needs to be familiar with before getting into more details about the problem and solution of the thesis.

1.1. Background

The section describes concepts and purposes in the fields of Information Visualization and Geographic Visualization as they are in the bases of the project's solution and development.

1.1.1. Information Visualization

Nowadays there are plenty of data sources like internet, radio and TV, books and articles, news, etc. The information flow increases every year and it is getting harder and harder for people to perceive this great amount of intelligence and especially to search for and sift out the data of their interest [5]. Analyzing the structure of text or documents collections, for example, without the assistance of a specific for the purpose tool could be extremely expensive and could take years. This entails the necessity of more efficient way of representing data giving the possibility to people to examine the data and gain insight into it. The need of alternative methods for data overview and analyses arises. This is what Information Visualization deals with. One definition of visualization is "the use of computer-supported, interactive, visual representation of abstract data to amplify cognition" [7, 8]. The aim of visualization is to represent data in a way that will give people a better overview and understanding of it. The focus of Information Visualization is the representation of abstract data that usually does not correspond to subjects from the physical world [7]. Visualization needs not to be associated with visual sense perception as any other type of sensation could be used to derive information and therefore the data can be represented not only by graphics, but also by sounds for example. Figure 1.1 illustrates a simple diagram of the process of information visualization. The data is transformed into pictures and a person interprets the pictures to gain insight. The diagram emphasizes that there is a big difference between data and information. One of the main tasks of Information Visualization is to aid people in deriving information from great amounts of data [9].

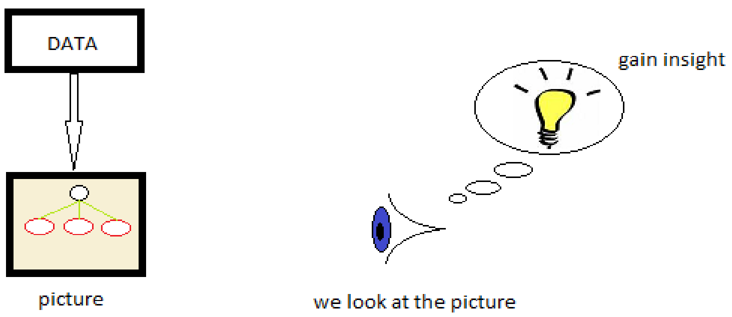

Figure 1.1 A simple diagram summarizing the process of Information Visualization [9].

The basic principle processes of information visualization are organized in the diagram of Figure 1.2. The process of representing the data consists of encoding the data values visually. There are many different methods that can be used for data representation. The choice of an approach to encode data values visually depends thoroughly on the following three aspects: the type of the represented data, complexity and the way the user interprets this data. The next step is presentation of the represented data. It has to be found a proper way for displaying the data that will give the user a good overview of it. Here a challenge is coping with the limited display area and different techniques for providing overview and details are used. Usually the corpus of explored data is too large to be presented in a single view giving a good and detailed understanding about it. The interaction between human and computer is one of the most useful techniques for giving users the opportunity to browse the data, to analyze it and to discover the information that is of their interest. A determinant factor in Information Visualization is the role of human visual perception system including activities like perception, interpretation and other "higher-ordered cognitive processes" [9]. More detailed description of the used representation, presentation and interaction techniques in the solution of this thesis' problem is given in Chapter 3.

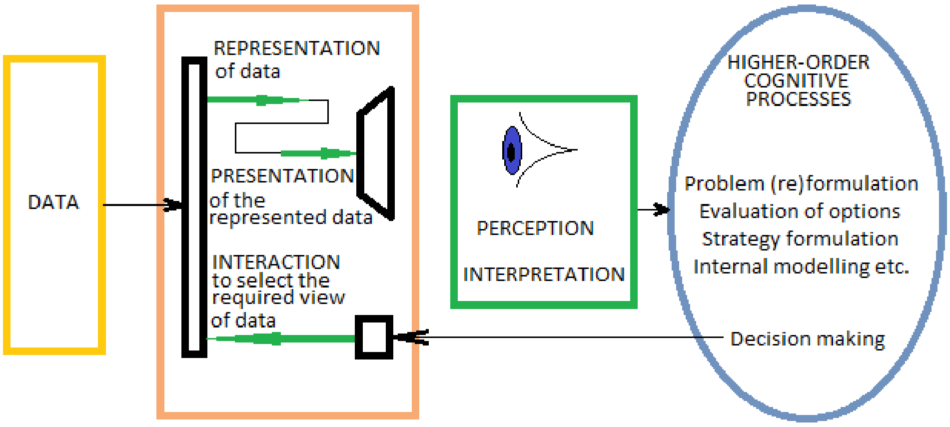

Figure 1.2 Identification of the interaction with data governed by high-order cognitive processes [9].

1.1.2. Geographic Visualization

Geographic Visualization, also referred as GeoVis, is concerned with a set of techniques and tools for interactive visualizations supporting geospatial data analyses. In contrast to Information Visualization the focus of GeoVis is the representation of real data that corresponds to subjects from the physical world. GeoVis is an area of research for visualization methods that ensue basically from cartography. The purpose in this field is to improve cartography and to provide more efficient exploration of various geographical objects or areas. GeoVis combines different methods of computing and interactive visualization approaches together with human cognitive processes to achieve these goals [20].

1.2. Motivation

The expansion of wine industry and the competition between the different producers causes the need of better means of wine description and advertising approaches. This along with the spreading interest in wine attracts the attention of the linguists to investigating the terminological language used in wine description. One of the methods for investigating the wine language is analyzing the language constructions and patterns used in wine tasting notes.

A wine tasting note is a comment given by a wine critic containing description and evaluation of a tasted wine. Wine tasting notes have a strict structure and follow definite rules for constructing. They usually consist of three parts. The first part gives information about the produce of the tasted wine. The middle of the tasting note is the essential part and it contains a description and evaluation of the wine given by the taster. The end of the comment provides some assessments and recommendations to the consumer [19].

From a linguistic point of view, analyses of wine tasting notes could be of great benefit for exploring the descriptors used for human sensory perceptions. The language of wine comments combines a number of terms and correlations to depict the wine's visual appearance, smelling, tasting and texture. Terms representing wine's clarity and color are used in describing its appearance. Olfactory perceptions are described by words for objects or taste. The reason for this is that there are no specific terms for depiction of odors and therefore words representing other modalities are used for this purpose. Exploring various correlations and patterns and the context of their use in wine description language can provide a lot of information about human's determination for descriptors of odor and other modalities [1].

In contrast with other senses smell does not provide information that can be reliable enough for making a conscious deduction about the object of attention. Vision provides information that can have a great influence on the perception of odors. It is considered to be a more reliable modality than smell for making a confident decision. The degree of confidence in human senses is affected by the differences existing between individuals [1]. The "Reliability hierarchy of Evidentiality" defines a hierarchy of the perception domains from lower to higher sensory modalities: from touch, taste, smell to sound and vision [19]. Analysis of wine tasting notes is an effective contrivance for investigating the interaction between the different sensory modalities [1].

1.3. Problem description

This thesis uses a created in advance database containing preliminarily loaded data about different types of wines along with additional descriptive information about them and their tasting notes. Analyzing a great amount of information like this can be significant for the linguists but also extremely laborious and time consuming. Therefore it needs to be represented in a way that will give a better insight into it. Several problems arise here in the field of Information Visualization.

There is a large amount of data that needs to be visualized. It has to be found a proper and efficient approach for representing this data that will provide a wide view over it as a starting point. Techniques for user interaction with the visualization are necessary as they will assure the opportunity for selecting a subset of tasting notes for getting detailed information about them and proceeding with further analyses.

Another challenge is selecting beneficial methods for interactive text visualizations. A research in the application field needs to be done to investigate the interests and goals of the linguists in the process of analyzing wine tasting notes. According to their necessities it has to be defined what could be useful for them and how the means of Information Visualization could aid them to get a better understanding of the texts content in a large collection of wine tasting notes. A number of compatible visualization approaches can be combined in order to give the possibility for efficient exploration of the language used in wine description.

Wines are described by a number of characteristics like variety, color, dryness, vintage, origin, etc. They need to be represented and integrated with the created visualizations in a suitable way corresponding to their data types and significance.

1.4. Purpose

The purpose of this master thesis is to be analyzed and developed a prototype of an efficient visualization tool for tasting notes analyses. It will combine a number of interactive visualization techniques supported by additional statistical information in order to give linguists the opportunity to make a study of the different correlations and patterns used in wine description language. The availability of a prototype will give us the possibility to test the selected approaches, to get feedback from potential users of the system and in this way to get a better idea of the benefits from a powerful future visualization tool.

1.5. Report structure

The next chapter contains information about related work in the application field and in the field of Information Visualization. It gives an initial direction of the problem research. Chapter 3 describes the process of investigating the thesis' problem and solution. It introduces different visualization approaches that are applied to give a better understanding of the represented data. Chapter 4 gives details about the analysis and design of the implemented tool. It contains description of the application and its functional and non-functional requirements. An explanation of the system components and classes design is also given in Chapter 4. Chapter 5 presents the user interface of the developed visualization tool and the possibilities given to the user to interact with the visualizations. Chapter 6 contains information about the programming language, software libraries and tools used in the process of implementing the application. The last chapter summarizes the results of our work and describes possible improvements of the tool and the visualization approaches.

2. Related work

This chapter contains information about previous work in the application field and in the field of Information Visualization. First it describes some previous work in wine tasting notes analyses. Then it gives information about preliminary work in the area of InfoVis that serves as a starting point for the research and implementation of the thesis solution.

2.1. Analysis of wine tasting notes using ALCESTE tool

In the article "The Color of Odors" published in 2001, Gil Morrot, Frederic Brochet and Denis Dubourdieu describe a study about the semantic of associations between colors and odors. They present the results of an experiment and do some analyses and conclusions of it. The experiment is carried out by the help of the ALCESTE tool. This tool uses a method of hierarchical cluster analysis over a corpus of text. The analysis is based on statistics about the words distribution in the text to determine groups of words co-occurring in the same context. Each group is supposed to contain words with similar semantic meaning used in the context of a specific lexical field. This gives the opportunity for statistical analyses of the co-occurrences of words and lexical analyses of the language used in a given piece of text. The ALCESTE tool also produces a dendrogram of the hierarchical division of the lexical fields as a result of the hierarchical cluster analysis. The input data for the experiment is a file containing the texts of a great number of wine tasting notes. The output of the ALCESTE tool consists of lists of words for each lexical field and dendrograms obtained from the input text. The experiment is carried out for four corpuses of wine tasting notes, three of them are in French and one of them is in English. In Table 2.1 is given a sample of the lists of the most often used olfactory terms for the lexical fields "white wine" and "red wine" obtained as a result of the previously described experiment. The ALCESTE tool generates a hierarchy clustering dendrogram of the lexical fields "white wine" and "red wine" for each of the four input corpuses of text. A sample of the generated dendrograms is given in Figure 2.1.

White wine				Red wine			
Descriptors (French)	Descriptors (English)	Occ. corpus	% group	Descriptors (French)	Descriptors (English)	Occ. corpus	% group
Miel	Honey	151	100	Chicore´e	Chicorey	6	100
Citron	Lemon	103	100	Charbon	Coal	5	100
Pamplemousse	Grapefruit	43	100	Povoine	Peony	27	88.89
Paille	Straw	30	100	Pruneau	Prune	48	87.5
Banane	Banana	27	100	Myrtille	Bilberry	46	86.96
Litchi	Lychee	23	100	Framboise	Raspberry	173	86.71
Pe´trole	Petroleum	21	100	Girofle	Clove	19	84.21
Acacia	Acacia	18	100	Cerise	Cherry	309	83.5
Aube´pine	May blosson	17	100	Fraise	Mills	101	83.17
Soufre	Sulfur	17	100	Ce`dre	Cedar	41	82.93
Buis	Boxtree	14	100	Musc	Musk	25	80
Arachide	Peanut	14	100	Havane	Havana	15	80
Mirabelle	Mirabelle plum	14	100	Chocolat	Chocolate	61	78.69
Mangue	Mango	13	100	Violette	Violet	154	77.27
Melon	Melon	12	100	Cacao	Cocoa	35	77.14

Tilleul	Lime tree	9	100	Cassis	Blackcurrant	245	75.92
Beurre	Butter	97	98.97	Tabac	Tobacco	61	75.41

Table 2.1 A sample of the lists of the most often used olfactory terms for the lexical fields "white wine" and "red wine" obtained from the experiment [1].

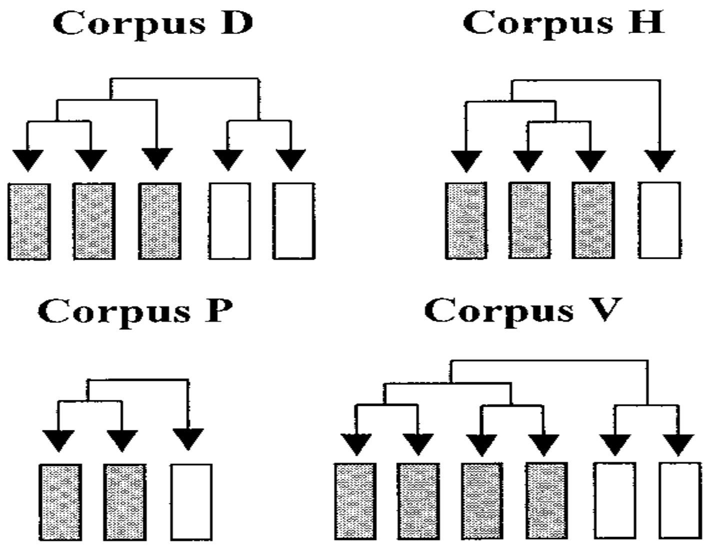

Figure 2.1 A sample of the generated dendrograms of the lexical fields "white wine" and "red wine" generated by the ALCESTE tool [1].

Using these statistical results the authors come to a number of conclusions about the wine perception of the wine tasters, the descriptors they use to characterize white and red wines and the impact of colors on the humans' odor perception. More details about the experiment and the results obtained from it can be found in resource [1].

2.2. Related work in the field of Information Visualization

2.2.1. FilmFinder

The FilmFinder is a tool for exploring a film database combining different representation techniques (Figure 2.2). It is the first tool integrating the concept of two-dimensional scatter-plot with color coding, filtering and details provided on demand. It is a very significant example of the use of different encoding and interaction techniques for representing hypervariate data [10, 12].

The main display presents the distribution of the films according to the attributes plotted on the scatter-plot axes. The film attributes plotted on the axes in Figure 2.2 are "Popularity" and "Year of Production". The range sliders integrated to the scatter-plot axes can be used to confine the view to a specific range of values and to have a closer look at the displayed data. Each film is represented by a rectangle in a specific color. Each color encodes a different film subject, for example "Drama", "Mystery", "Comedy", etc. The right part of the display contains different sliders that can be used by the user to specify the values of other film attributes and to filter out the visualized data according to the selected values. The FilmFinder tool demonstrates a good and convenient possible solution of the presentation of great amount of data in a single display.

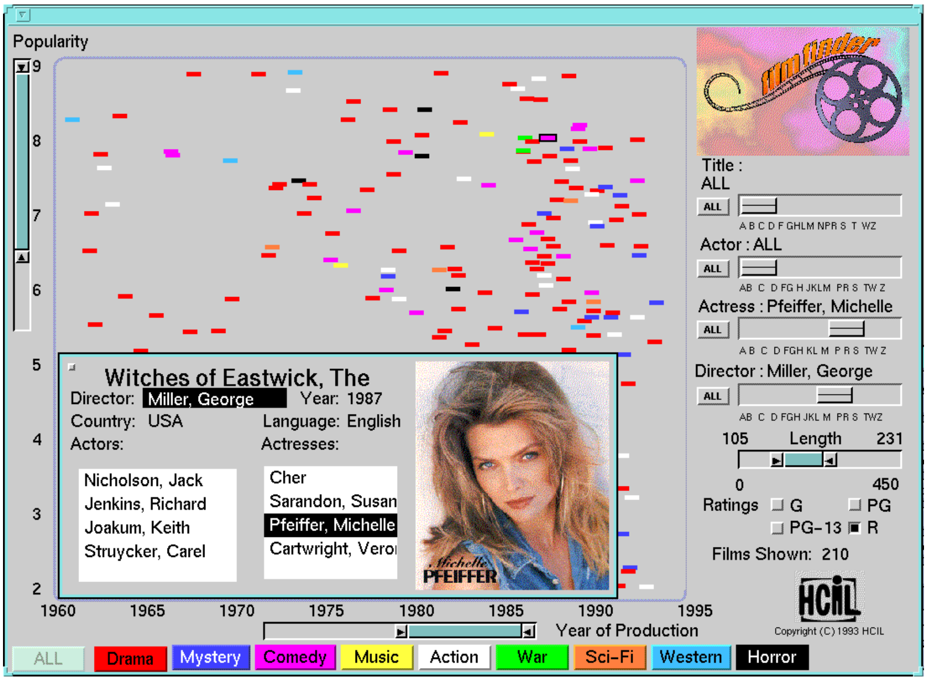

Figure 2.2 A snapshot of the FilmFinder tool [10].

2.2.2.Word tree of Many Eyes

Many Eyes is an IBM research project in the area of Information Visualization that provides alternative methods for data analyses using innovative visualization techniques. One of their approaches for supporting text analysis is the representation of a given text as a word tree. The purpose of this visualization method is to afford an insight into the different contexts in which a word is met in an unstructured text. They also provide an implemented visualization of this approach.

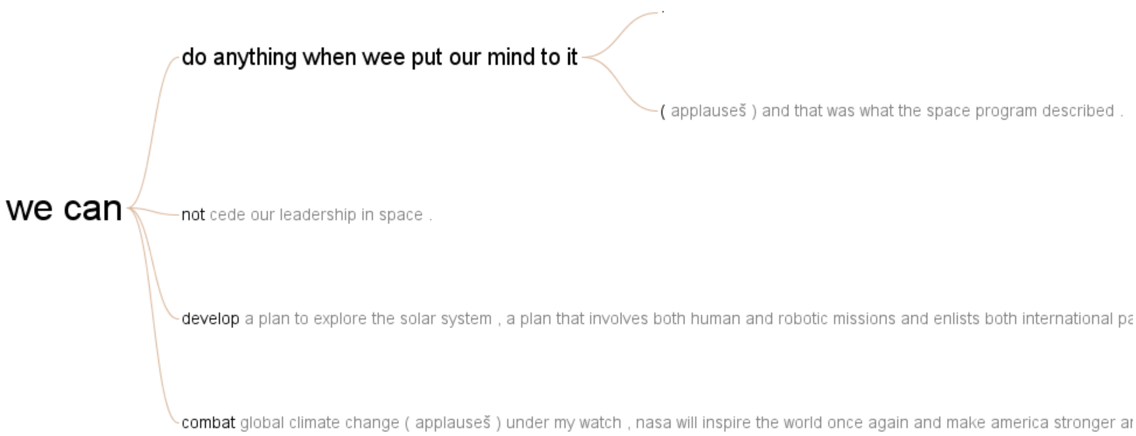

Figure 2.3 An example generated by the Many Eyes' Word Tree Visualization [3].

The word tree is constructed on the base of a previously input text and a word or phrase typed by the user and used as a search term. As a result the method finds all the occurrences of this word or phrase in the text and all the phrases that appear after it and structures this data into a tree. Figure 2.3 gives an example generated by the Many Eyes' Word Tree Visualization that demonstrates a word tree as a result of a text about the president Obama's speech in 2008 with the phrase "we can" used as a searching term. Different possibilities are realized for interaction between the user and the visualization. It is possible for the uninteresting branches to be filtered out; a new word can be chosen as a new search term by clicking on it, etc.

This method is in some way related with our work as it gives a good overview about the frequency of a word occurrence in a text and the contexts of its usage. One disadvantage of this approach is that when a term occurs very often the word tree becomes too big to be displayed in the screen. In these cases the users need to scroll the visualization and therefore they are not able to get an overall view of the data structure [3].

3. Problem investigation

This chapter describes the process of problem investigation. First it presents the database and its content. After that different concepts and visualization approaches are described as a result of the research of possible problem solutions.

3.1. Database understanding

We were preliminary provided with two databases of tasting notes about different kinds of wine. In each database the tasting notes are represented in a different way. The first database is the original one and contains some descriptive information about the wines, their origin, vintages, wine ratings, information sources and the whole original text for each of their tasting notes. Figure 3.1 presents the entity relationship diagram of the first database.

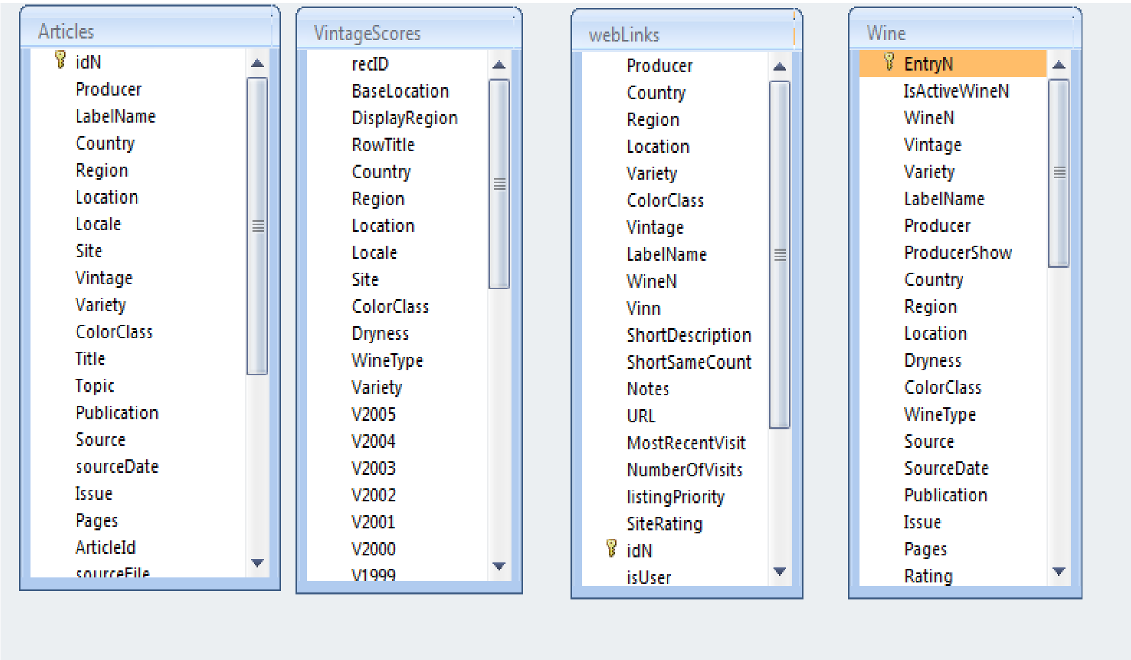

Figure 3.1 Entity relationship diagram of the original database.

The main focus of interest from a linguistic point of view is the Wine table, where each row contains the text of one tasting note. We are now concentrated on the linguists' needs for doing there analyses and investigation in the wine description language and therefore most on visualization of the tasting notes content. On another hand for a general user, who is interested in the quality of the wine itself, it could be very beneficial to get insight into the data about its origin, vintages and scores contained in the VintageScores table. Visualizing the density of wines produced in different regions through various periods of time is a challenging task for the field of Information Visualization and it is an interesting idea for further work and research.

The second database is derived from the original one by using the WineConverter tool, developed by the PhD students Susanne Ekeklint and Jens Nilsson at Växjö University. This tool takes as an input the original database and creates as a result a new database that contains the same tasting notes as in the original one, but segmented into words and given word classes (also called word tags). Only the abbreviations of the word tags are stored in the table. Not all of the information from the original database is contained in the derived one as not all of it is important for the analyses of the tasting notes. Figure 3.2 shows the entity relationship diagram of the second database. The table "Notes" contains the words that are used in the tasting notes. Each row consists of the following data: number of the tasting note, number of a sentence in the tasting note, number of a word in this sentence, the word itself and the word tag given to this word, i.e., each row contains one word from a tasting note and additional information that accurately specifies the position of this word in the respective tasting note.

The WineConverter tool offers an option to divide the data in the output database into a number of smaller databases. This number can be specified by the user. I used this option to divide the derived database into 10 pieces as I had difficulties with processing the whole data at ones.

The data stored in the "Wine" table of the initial database is closely related to the data stored in the manipulated database. The execution of queries in two different databases causes some difficulties in retrieving and processing the necessary data. Copying the "Wine" table from the initial database to the derived one is an easy way to avoid such complicated operations and this action should be taken before proceeding to the next steps.

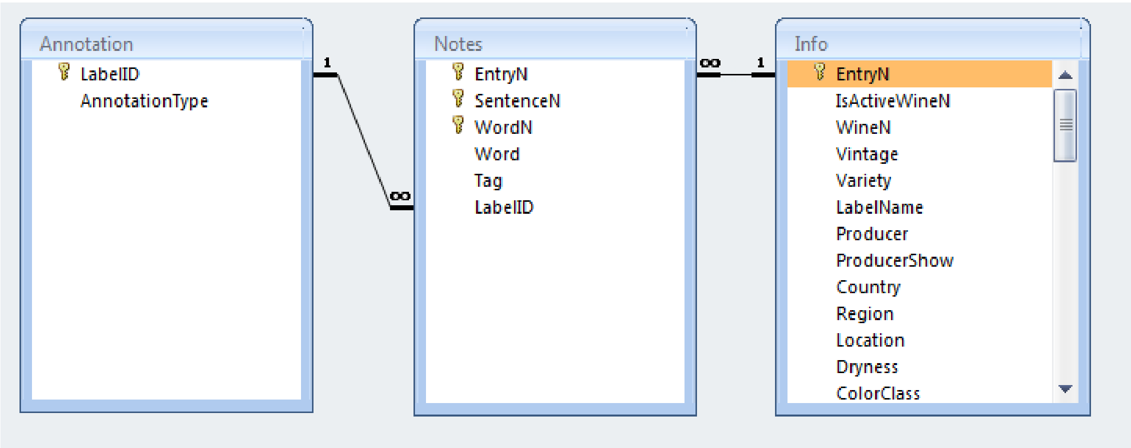

Figure 3.2 Entity relationship diagram of the derived database after manipulation on the original database.

As the linguists' purpose is analysis of the wine tasting notes texts and studying the language used in wine comments, our focus of interest is more on the wine tasting notes and words than on the wines themselves.

3.2. Word tags corpus

The "Notes" table in the derived database described in Section 3.1 contains data about the word tags of each word. These tags are presented by their abbreviations and the database does not contain information about their meanings. More details about the word tags corpus used in this project is presented in Appendix A and reference [14] giving a list of the word tags abbreviations and there long names.

3.3. Wine attributes

Each wine tasting note is described by a number of characteristics that give more details about it and the evaluated wine. From now on such characteristics will be referred as wine attributes in this report. Some of the wine attributes that can be obtained from the database described in Section 3.1 are: wine color class; wine vintage; country of wine produce; wine rating; wine variety; length of the wine tasting note; etc.

3.4. Problem solution concepts

This section describes several initial concepts of information retrieval and presentation that could give the linguists a good insight into the previously described database. These ideas were discussed with a domain expert and the most useful of them were combined and represented by the proper visualization methods to build a tool for tasting notes analyses.

3.4.1. Common words from specified sequence of word tags used in a group of tasting notes

One of the activities in wine description language analyses is concentrated on investigating the descriptors of perceptions activated in wine tasting. The succession of human sensory activation plays a significant role in the choice and sequence of wine descriptors. According to the "Conceptual Preference Principle" the preferred direction of mappings in metaphorical combination of words is from lower modalities like touch and taste to higher modalities like smell and vision [19]. A structured overview of the combination of words used in the wine tasting notes can facilitate the linguists in their investigation of these issues. Figure 3.3 is an abstract diagram for possible organization of the words contained by two tasting notes that can give information about the succession and frequency of their usage. In the given example there are two tasting notes. The focus of interest is in the combination of words corresponding to the pattern "adjective, adjective, noun". The combination of words that correspond to this pattern are extracted from the tasting notes' texts and distributed in three basic groups: words that belong to the first tasting note, word that belong to the second one and words that belong to both of them. The example is for two tasting notes but it can be applied to three or more. The purpose of this concept is to give overview about the order of the words and the frequency of their use corresponding to a given pattern of tags combination.

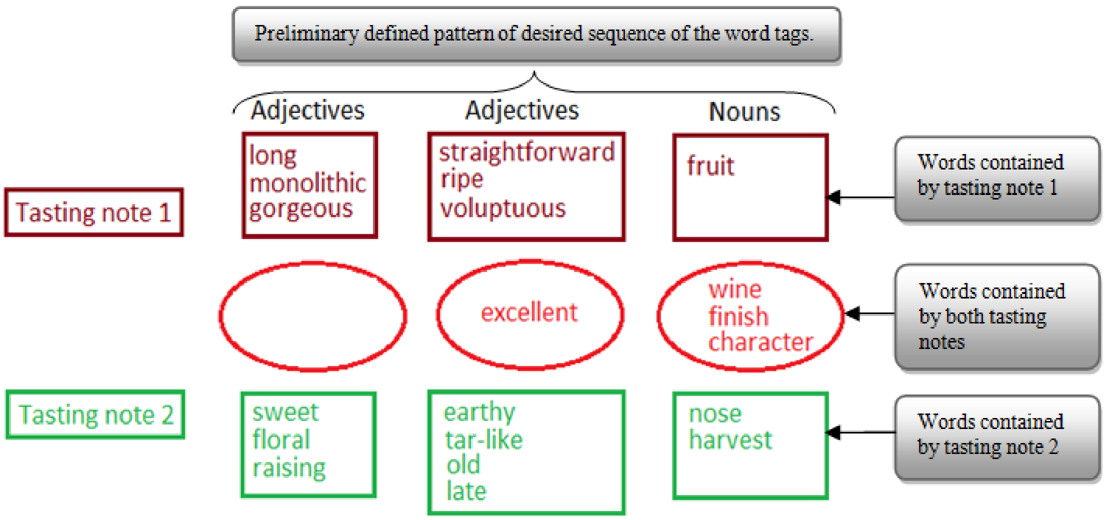

Figure 3.3 A structured overview of the combination of words used in the texts of two tasting notes.

3.4.2. Construe of tasting notes sentences

This concept has for a purpose to give an overview of the words and the frequency of their use in a list of tasting notes. The focus is on words corresponding to some preliminarily specified tags that are interesting for the wine description language analysts. Figure 3.4 gives an abstract example of the idea. In the diagram there is a group of wine tasting notes corresponding to wines with the following characteristics: red, dry and table. The text of each tasting note is written and partly construed. In the given example the adjectives and nouns are marked respectively by oval and rectangular shapes. The same words are emphasized by red color to give an impression about the frequency of words occurrences with a specific tag.

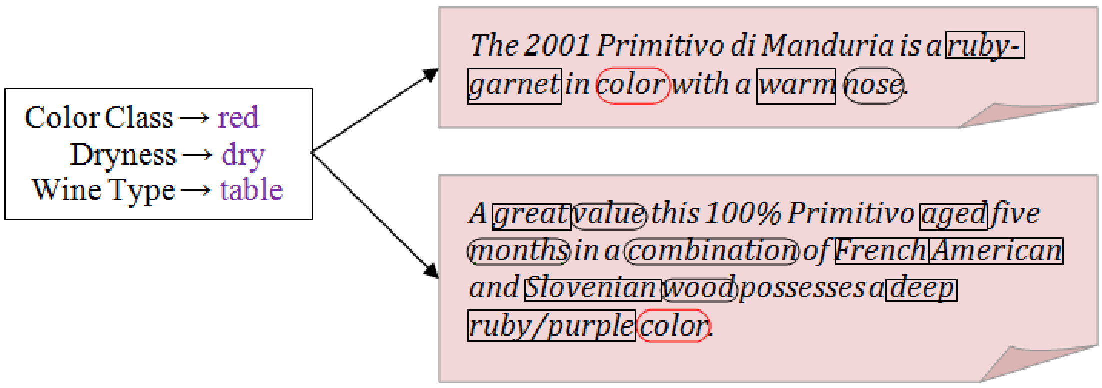

Figure 3.4 An example of tasting notes sentences construe concerning the adjectives and nouns marked respectively by oval and rectangular shapes.

3.4.3. Word combinations construction

The purpose of this concept is to structure the text in a way that gives information about all combination of words found in a group of tasting notes. Figure 3.5 contains an abstract diagram of the idea. The initial words that serve as a starting point for the combinations depend on the choice of the analysts. The words are grouped together according to their tags to aid human's perception of the information. The provided information gives the opportunity for text exploration and a better understanding of a word or phrase usage.

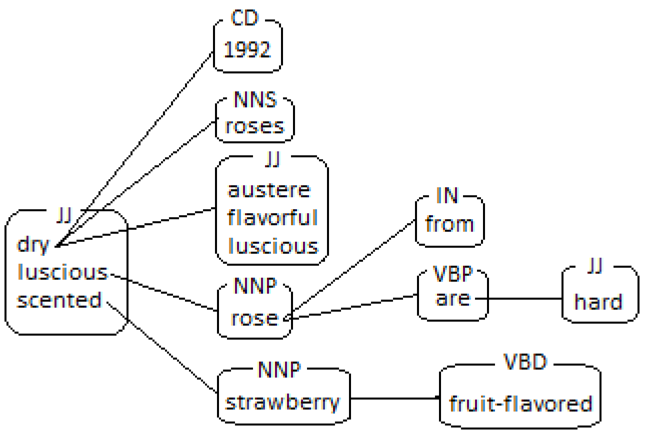

Figure 3.5 An example of the word combinations in a group of tasting notes starting with preliminary specified adjectives.

3.4.4. Words distribution according to their tags

Information about the number of words with a specific tag and the frequency of their occurrence in the wine tasting notes texts can give a good overview of the characteristic descriptors used in the wine description language. The purpose of this concept is to provide statistical information about the words and tags used in a group of tasting notes. The pie chart diagram in Figure 3.6 illustrates an abstract example of the described idea. Each pie part of the circle designates a different word tag and has a different color. The practical example in Figure 3.6 gives information about the following five word tags: verbs, nouns, adjectives, prepositions and singular determiners. The size of the pies signifies the ratio of the numbers of words occurring in the tasting notes texts grouped by their tags. The pie parts are divided into smaller pieces each one of which presents a word. The closer a word is to the center of the circle, the more often it occurs in the texts. The words located in the same level occur in the texts the same number of times. It can be noticed in the given example that the most often used nouns are “Wine”, ”nose”, ”cassis” and ”drink”. Another conclusion is that they occur more often than the nouns “lack” and elegance.

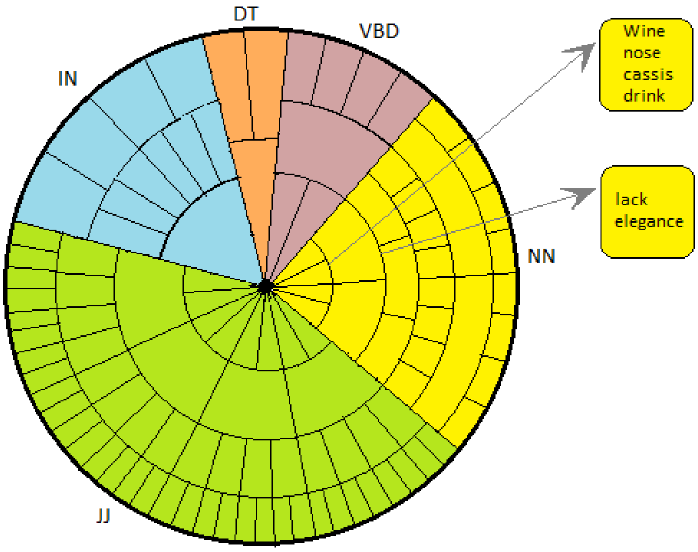

Figure 3.6 An example of the words distribution according to their tags in a group of tasting notes.

3.4.5. Sentences structure analysis

The purpose of this concept is to assist analysis of the sentences structure in a corpus of text. Figure 3.7 gives a diagram providing statistical information about several tags distribution: CC, CD, DT, EX, FW, IN and JJ. The left part of the diagram is a table of line charts representing the word tags occurrences in different positions of the tasting notes sentences. The first column contains the sequence numbers of words in the sentences structure. In the concrete example each sentence in the analyzed text consists of no more than 7 words. The right part of the diagram represents a zoomed in view of the first row of the table. The x-axis of the line chart diagram represents word tags and the y-axis represents the number of occurrences of the respective tags in the first position of the tasting notes sentences. It can be noticed that the tag “CC” happens to be the tag of the first word in the analyzed sentences for 300 times.

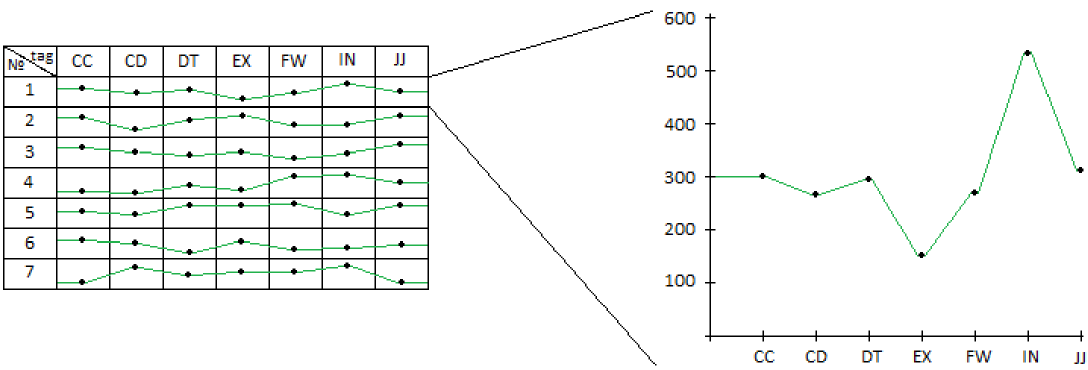

Figure 3.7 An example of a diagram providing statistical information about the tags depending on their occurring in different positions in the tasting notes sentences.

This concept can be useful not only for wine description language analyses but also for language analyses in general as it gives an insight into the language structure and sentences construction.

3.5. Essential numbers

Before proceeding to the investigation of different visualization approaches we should take into account the size of the analyzed data and data attributes. I derived some statistical numbers about the different data values from the database. The results are given in Table 3.1.

Number of tasting notes	84 864
Total number of words used in the tasting notes	8 332 666
Number of different words used in the tasting notes	46 000
Maximum length of a tasting note	496
Number of word classes	43
Number of vintages	104
Wine rating values range	From 1 to 100

Table 3.1 Essential numbers derived from the wine database.

The conclusion after estimating these numbers is that we have to find a proper visualization method for the representation of a great amount of data. As the main purpose of our visualization is to give insight into the wine tasting notes we come to the idea that it has to provide an overall view of all the tasting notes as a starting point. This will give the user an initial overview of the explored data and the possibility to proceed with investigation of a smaller subset of it.

3.6. Visualization approaches

Several visualization approaches are combined in order to achieve our goals: a scatter-plot; bar charts; tag clouds; a word tree and a world map. The starting point of the application strives for giving an overview of all the information providing the possibility for applying filters and selection of a smaller subset of the analyzed tasting notes. Statistical diagrams and representations like bar charts and tag clouds are added to aid this process. The word tree visualization gives insight into the structure of the filtered out tasting notes' texts. This section describes in details the different methods and approaches that we combine in building the solution of this thesis.

3.6.1. Starting point

The idea of the starting visualization is to give a general view of the distribution of the tasting notes according to their different wine attributes.

Scatter-plot

We can see in Table 3.1 that the total number of tasting notes is 84 864. As this is a large number of elements we came to the idea to use a pixel for representing each one of the tasting notes. One of the possibilities for such representation is a scatter-plot. The scatter-plot consists of two orthogonal axes in two-dimensional space each of which is associated with one of the elements' attributes. This representation gives an idea about the distribution of a number of elements depending on the values of two of their attributes. Figure 3.8 presents an example of a scatter-plot that shows the distribution of several tasting notes according to their length and wine rating. The representation in the given example affords a conclusion that the longer the wine comment is, the higher the rating of the wine is.

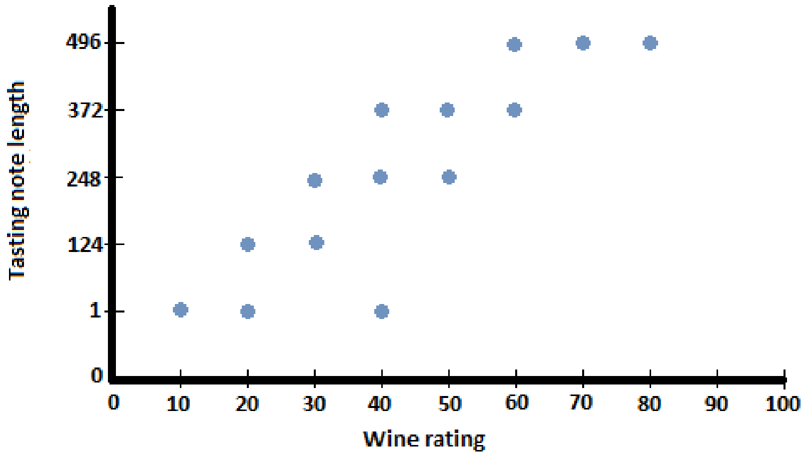

Figure 3.8 An example of a scatter-plot that shows the distribution of several tasting notes according to their length and wine rating.

Although the scatter-plot gives a good overall view of the tasting notes distribution, it is a conventional approach for representation of bivariate data [9]. Here comes the question which two of the wine attributes to be plotted on the scatter-plot axes and how the user can get an impression about the tasting notes distribution according to other wine attributes. A natural solution is to combine this representation with some interaction techniques and to provide the opportunity of selecting different possibilities.

3.6.2. Dynamic queries

After getting an overall view of the data users need to be provided the opportunity to interact with the visualization and to select a subset of elements that are of their interest. There is a set of elements with different values for their attributes and the purpose of the interaction is to find the best elements or subset of elements to be further analyzed. Dynamic queries are widely used concept in Information Visualization for aiding the solution of this task [13]. Dynamic queries give the opportunity for selecting a value or range of values for the elements' attributes and getting a fast visualized answer for the elements corresponding to the selected values. This approach offers simultaneous display of query and result and it is useful for dynamic exploration of the data [13]. Following this concept different types of filters were integrated for some of the attribute values to be used for finer delimitation of the visualized elements on the scatter-plot.

3.6.3. Text visualization

The next step is to find a proper representation of the text content of the selected for further analyses tasting notes. The visualization approaches that we decided to apply to the analyzed texts are a word tree and tag clouds.

Word tree

The word tree describes the sequence of words and phrases used in a group of tasting notes. The word tree visualization facilitates rapid querying and exploration of bodies of text [15]. There are three prerequisites for proceeding with the word tree visualization. The users need to select a group of tasting notes for further analyses, a specific tasting note for deriving the initial data and word classes of the words in this tasting note that they would like to analyze. The first three levels of the word tree contain data about the selected tasting note (from now on referred as "root tasting note" of the tree as the initial data is obtained from this tasting note). The root node of the tree is artificially added and it contains the static text "Tags" which suggests that the following level consists of word tags. The second level contains the selected in advance word tags that correspond to words in the root tasting note. The third level consists of all the words from the root tasting note that have the selected in advance word tags. The structure of the word tree is organized in two main groups of nodes: word tags and words. Figure 3.9 gives an example of the word tree and the organization of its nodes. The levels of the tree alternate with each other to represent either word tags or words that correspond to the tags of the previous level. The children of each node representing a word class are the words from the analyzed tasting notes that belong to this word class. There exists another division of the nodes in the word tree as well: nodes that contain data from the root tasting note and nodes that contain data only from the other tasting notes. This division is visualized by color coding of the nodes' text. Red color of the text means that the represented data is contained in the root tasting note and black color of the text means the opposite, i.e. the data is not contained in the root tasting note. It should be noticed that red color of the node's text does not mean that it is not contained by other tasting notes, though it indicates that it is contained by the root tasting note.

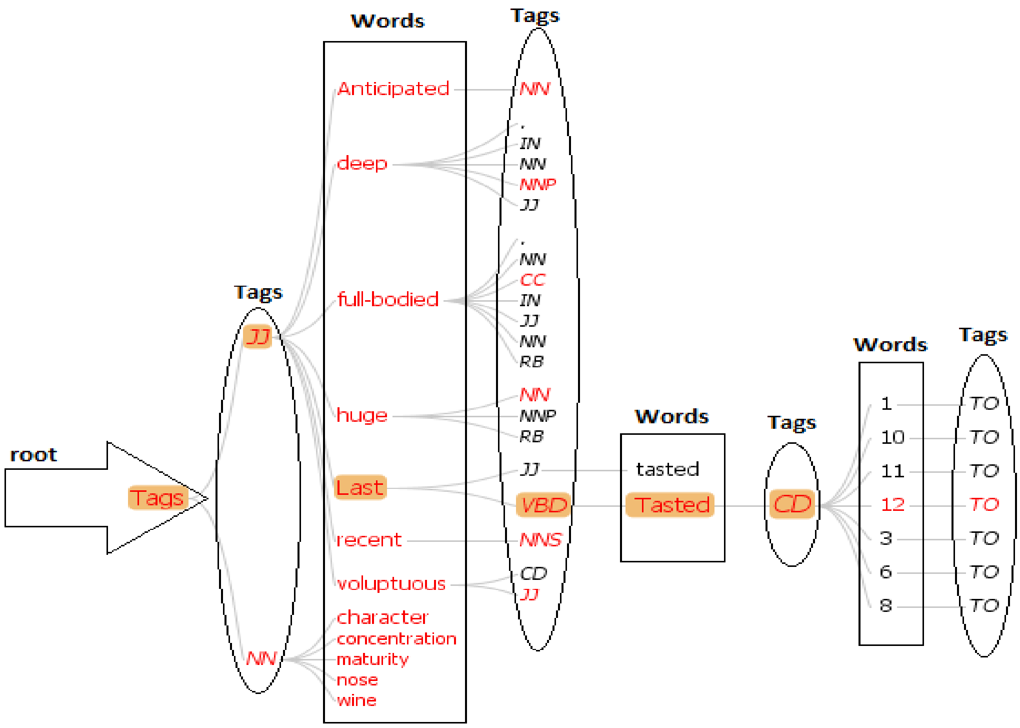

Figure 3.9 An example of a word tree that shows the node organization into two main groups: word tags and words.

Degree-of-interest trees

The word tree visualization approach brings up several issues that have to be taken into consideration. It represents a large data set of words and word tags and its visualization is restricted by the size of the display and people's perceptive capabilities [17]. The Degree-of-interest trees are interactive trees that give a solution of these problems. They combine focus + context visualization techniques and degree-of-interest calculations to find a proper layout that fits within the bounds of the display [17, 18]. Some of these techniques are: logical filtering of the displayed nodes of the tree based on a preliminary estimated degree-of-interest value; geometric distortion using different size for the nodes depending on their degree-of-interest value; semantic zooming of the nodes determined by their focus; etc. [17, 18] In the base of the tree visualization approach is the use of a "Degree of interest" function which assigns a number value to each node indicating how interested the user is in this node. This value is then used as a criterion for determining which nodes and how to be visualized on the display. More details about the Degree-of-interest trees approach can be found in [16], [17] and [18].

Tag clouds

The tag clouds visualization provides information about the frequency of words used in a corpus of text. This approach uses different font size for each word in the text to indicate how often this word is used by comparison with the others. There are two prerequisites for applying the tag clouds visualization to a tasting note's text. A group of tasting notes for further analyses needs to be defined and then one of them has to be selected for its text visualization.

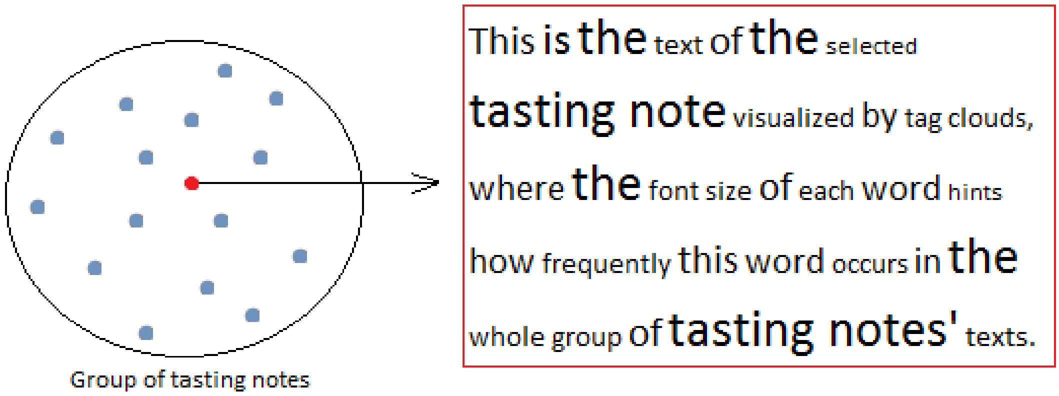

Figure 3.10 An example of a tag clouds visualization of a tasting note's text in a group of tasting notes.

In Figure 3.10 there is an example of a tag clouds visualization of a tasting note's text. A group of tasting notes is defined in the black circle. Each tasting note is represented by a dot in blue color except the selected one which is colored in red. The text of the selected tasting note is visualized by tag clouds in the right side of the diagram. Each word has a different font size depending on the frequency of usage of this word in the texts of all tasting notes in the group. In the given example we can see that the most often used words contained by the selected note and the other tasting notes in the given group are "the", "tasting", "note" and "notes".

3.6.4. World map visualization

One of the attributes describing the different types of evaluated wines is there origin. The country where the wine is produced is an important characteristic and it is useful to be visualized in a way that will give a wide overview of this information. A natural visualization approach of presenting the wine origin is an interactive world map indicating the density of wine production in different countries depending on the data contained in the provided database. Figure 3.11 gives a world map representing information about the tasted wines produced in different parts of the world. The darkness of the color is proportional to the density of wines produced in the respective country. It can be deduced from the given example that most of the tasted wines come from France as it has the darkest color of all other places in the world map. The countries that are in white color do not produce any of the tasted wines.

This approach can be combined with the dynamic queries approach to give users the opportunity to select a country and get an immediate result of the tasting notes for wines produced in the respective place.

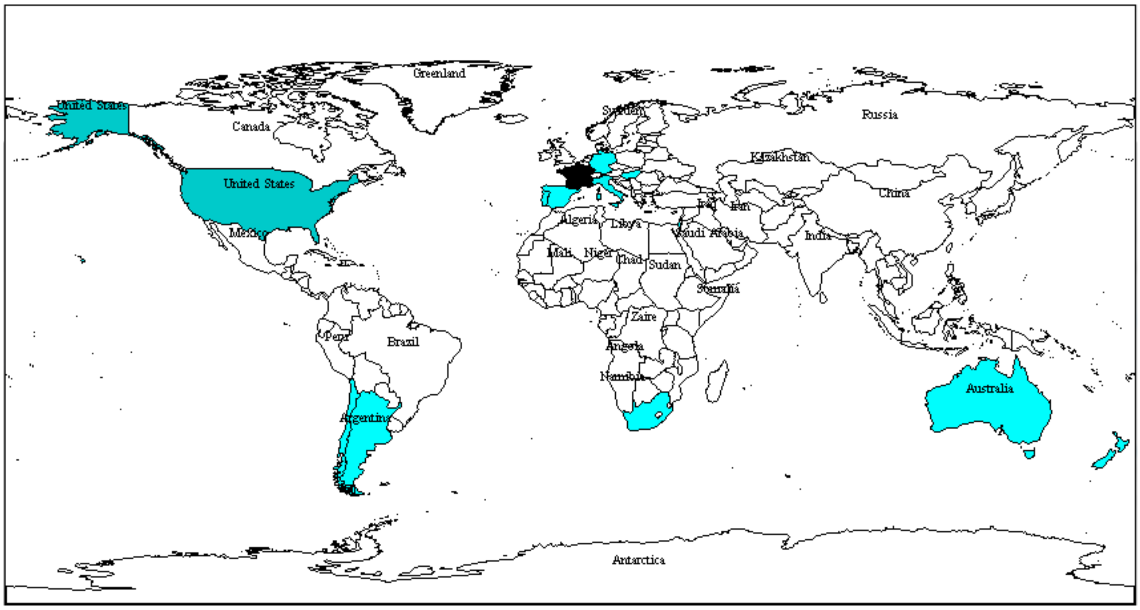

Figure 3.11 World map providing information about the tasted wines produced in different countries.

3.6.5. Bar chart diagrams

Getting statistical information can aid analysts to understand more about the currently visualized data and to find their way to the desired set of tasting notes that are interesting for them. Bar chart diagrams are a general approach for statistical data visualization. In this project they are supplied for showing the number of wines corresponding to different wine attribute values. Figure 3.12 gives an example of a bar chart diagram presenting the number of wines produced in each of the given wine vintages.

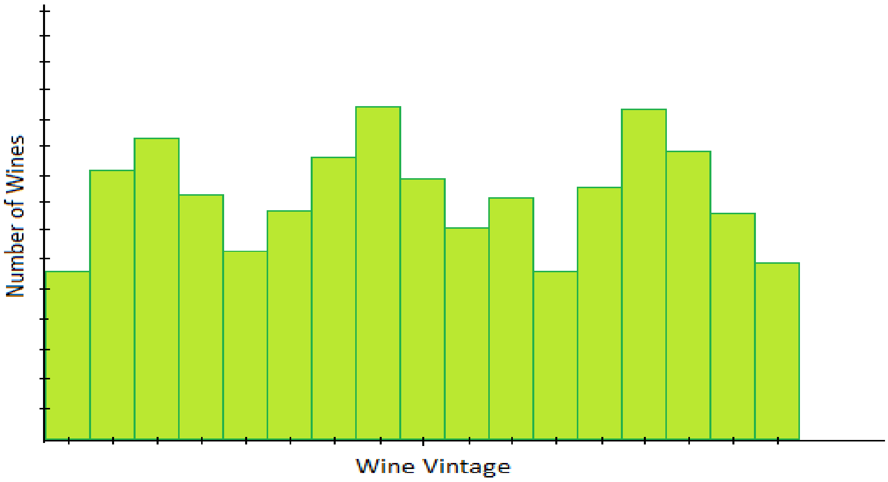

Figure 3.12 A bar chart diagram showing the number of wines that correspond to different wine vintages.

4. Analyses and Design

This chapter describes the process of software analyses and design of the developed prototype. It gives a description of the visualization tool and a list of the functional and non-functional requirements.

4.1. Visualization tool description

The purpose of the visualization tool is to give users a good insight into the information retrieved from the database and to give them useful instruments for analyses over it. We are concentrated basically on the text of the tasting notes and on the standard wine attributes. A very important feature of the tool is to realize the opportunity for interaction with the user. The starting point of the visualization tool is visualization of all the tasting notes on a scatter-plot. Each tasting note is represented by a pixel as this is one of the possibilities to visualize the great amount of tasting notes that we have in the database. The axes of the scatter plot represent the values distribution of two wine attributes. The user is given the possibility to change the wine attributes plotted on the scatter plot axes. As it is a big number of tasting notes it is essential that users have the opportunity to filter them out and to view only those of them that are of their interest. This feature is realized by adding controls (also called filters) that allow selecting the values of different wine attributes and getting visualized on the scatter plot only the tasting notes that satisfy the selected values. Another useful opportunity is the integration of range sliders to the scatter plot axes that allow the user to narrow down or up the range of the visualized region. As the tasting notes are represented by pixels, no other details about their texts can be observed on the scatter plot. More detailed information about a tasting note can be obtained on demand by selecting it and requesting for narrow visualizations. The text of the selected tasting note is then visualized by tag clouds displaying the frequency of words usage and their respective word tags. There are bar chart diagrams integrated with the scatter-plot and providing additional statistical information about the number of visualized tasting notes that corresponds to different wine attribute values. In the initial state of the scatter-plot all the tasting notes are visible on the screen. Users are given the opportunity to filter out some of the tasting notes using several types of filters like range sliders, combo boxes, radio groups, checkboxes and a world map. The world map filter is intended to perform two purposes. First it represents the density of wines produce in different countries on the map and second it gives users the opportunity to select a particular country and to get visualized the tasting notes of wines produced there. The functional requirements of the system are listed in the next section.

4.2. Functional requirements

The Wine Tasting Notes Visualization Tool is intended to provide many functional opportunities for data analyses that will reveal different comprehensive visualizations of the wine attributes and tasting notes. Table 4.1 presents the functional requirements that are implemented throughout this project working time and are meant to serve as a starting point for future extensions:

FR-1	View of all the tasting notes in a scatter plot.
Importance	Essential
Description	When starting the application the user should get an initial overview of all the tasting notes distributed in a scatter plot. Each scatter plot's axis

	represents the values of one wine attribute.
FR-2	Change the attribute to be plotted on the X axis of the scatter plot.
Importance	Desirable
Description	The user should be given the possibility to choose the wine attribute plotted on the X axis of the scatter plot.
FR-3	Change the attribute to be plotted on the Y axis of the scatter plot.
Importance	Desirable
Description	The user should be given the possibility to choose the wine attribute plotted on the Y axis of the scatter plot.
FR-4	Change the axes' values scope of the scatter plot
Importance	Essential
Description	The user should be given the possibility to change the range of attributes values in the scatter plot's axes and to have a closer look at the corresponding tasting notes.
FR-5	Filter the tasting notes of interest.
Importance	Essential
Description	Users should be able to filter the tasting notes of their interest by selecting different wine attribute values.
FR-6	View a bar chart diagram about the number of tasting notes corresponding to each value of a wine attribute.
Importance	Desirable
Description	A bar chart diagram should provide statistical information about the number of tasting notes that correspond to the different values of a wine attribute.
FR-7	Change the wine attribute for the bar chart diagram.
Importance	Desirable
Description	Users should be able to select the wine attribute used by the bar chart diagram.
FR-8	Select a tasting note from the scatter plot
Importance	Essential
Description	User should have the possibility to select a tasting note from the scatter plot.
FR-9	View the text of a selected tasting note represented by tag clouds.
Importance	Essential
Description	Users should be provided the possibility to view the text of the selected tasting note represented by tag clouds.
FR-10	Choose word tags.
Importance	Desirable

Description	Users should be able to select the word tags that are interesting for them.
FR-11	Generate a word tree.
Importance	Essential
Description	A word tree should be generated to give the possibility for tasting notes' text exploration.

Table 4.1 List of the functional requirements of the Wine Tasting Notes Visualization Tool.

Figure 4.1 illustrates the use case diagram created on the base of the functional requirements from Table 4.1.

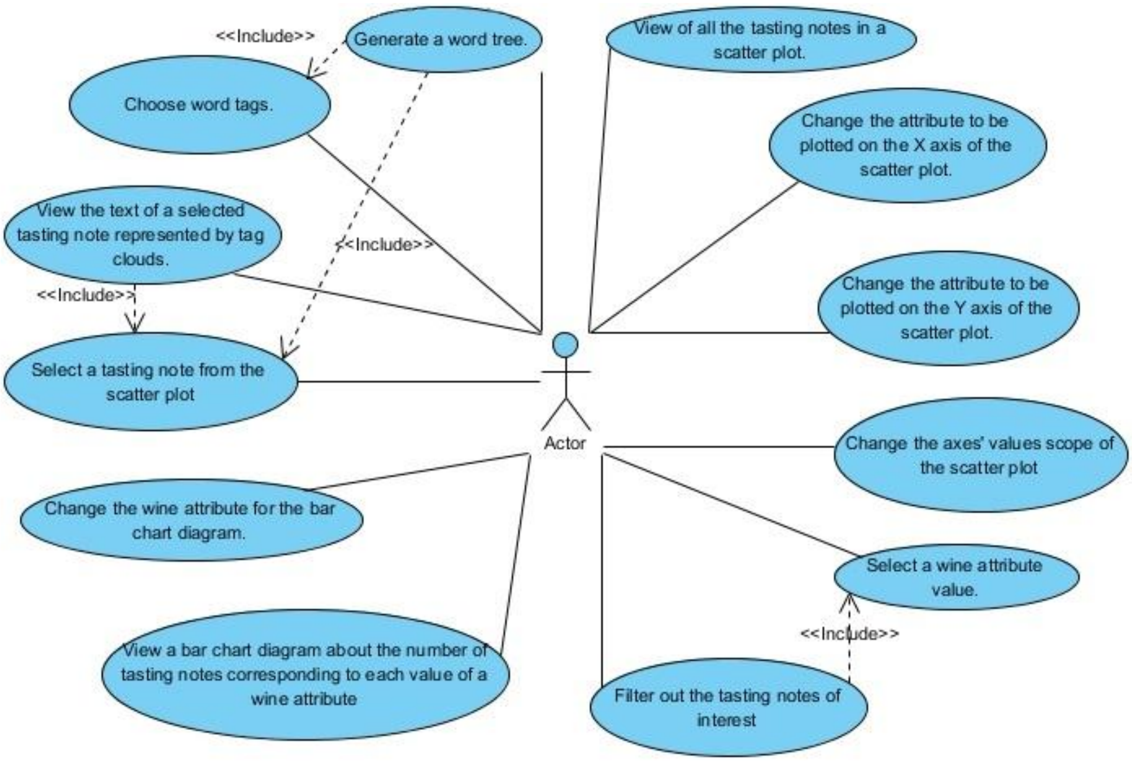

Figure 4.1 Use case diagram that describes the functionality of the Wine Tasting Notes Visualization prototype.

4.3. Nonfunctional requirements

The system should satisfy a number of quality goals. Table 4.2 gives a list of the nonfunctional requirements of the Wine Tasting Notes Visualization Tool.

NFR-1	Performance
Importance	Essential
Description	The application should have acceptable response time and processing speed.
NFR-2	Stability
Importance	Essential
Description	The application should be stable against possible crashes.
NFR-3	User friendly interface
Importance	Essential
Description	The user interface should be designed to be user friendly and intuitive.

FR-4	Maintainability
Importance	Desirable
Description	The application should be well documented and maintained to meet new requirements.
FR-5	Extensibility
Importance	Desirable
Description	The application should be designed to be extensible so that adding new functionalities would require minimal efforts and time.

Table 4.2 List of the nonfunctional requirements of the Wine Tasting Notes Visualization Tool.

4.4. System components

The system is build by several independent components communicating with each other through particular interfaces. This structure provides the opportunity a component to be easily changed by another one without more supervening changes in the rest of the system. Figure 4.2 represents the component diagram of the Wine Tasting Notes Visualization prototype. The system consists of three basic components. “Wine Database” represents the database of wines descriptions and tasting notes that provides all the data necessary for the application. The “Prefuse Visualization Toolkit” is an extensible software framework for creating interactive information visualizations and it is widely used in building the prototype. The third basic component is the “Wine tasting notes visualization tool” which consists of several subcomponents. The “Database access” subcomponent is responsible for retrieving and processing the requisite data from the database. JDBC driver is used for interaction with the database. The “Scatter Plot” subcomponent is set with the task to visualize the scatter-plot with the tasting notes. The “Filters Settings” subcomponent gives filters for determining ranges of wine attributes values and applies them to the visualized tasting notes. “Filters Settings” uses the “IScatterplot” interface provided by the “Scatter Plot” to restrict the visualized data depending on the selected values of interest. The “Statistical Charts” subcomponent represents statistical data about the visualized tasting notes. It uses bar charts for each one of the wine attributes to provide information about the number of visualized tasting notes corresponding to its values. The text of a selected tasting note from the scatter-plot is visualized by the “Tag Clouds Visualization” subcomponent. The “ITextTagClouds” interface is provided for communication with the “Scatter Plot”. The “Word Tree Visualization” subcomponent is responsible for the word tree construction and representation. Each one of the previously described subcomponents uses the “Prefuse Visualization Toolkit” to create efficient and interactive visualizations.

4.5. Classes design

Each of the system’s components described above is represented by an individual package of classes that realize the respective functionality. The class diagram of the Wine Tasting Notes Visualization prototype is given in Figure 4.3. It consists of the basic classes that realize the functionality of the system.

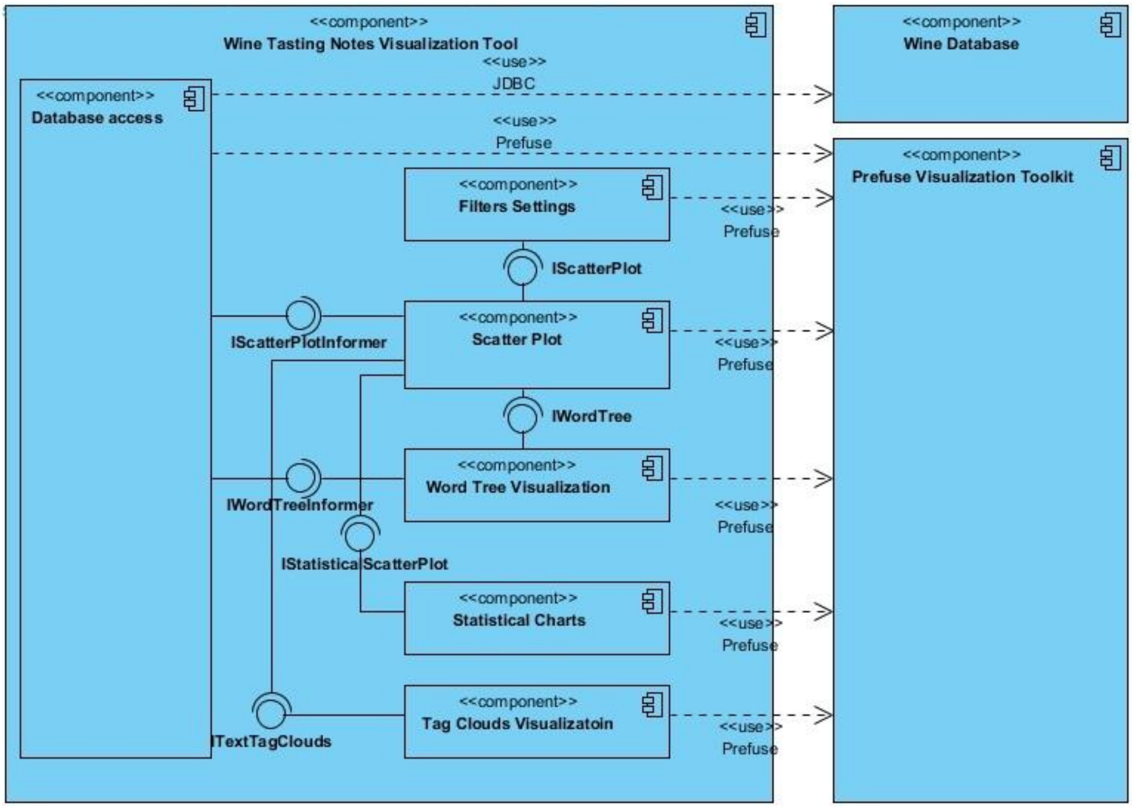

Figure 4.2 Component diagram of the Wine Tasting Notes Visualization prototype.

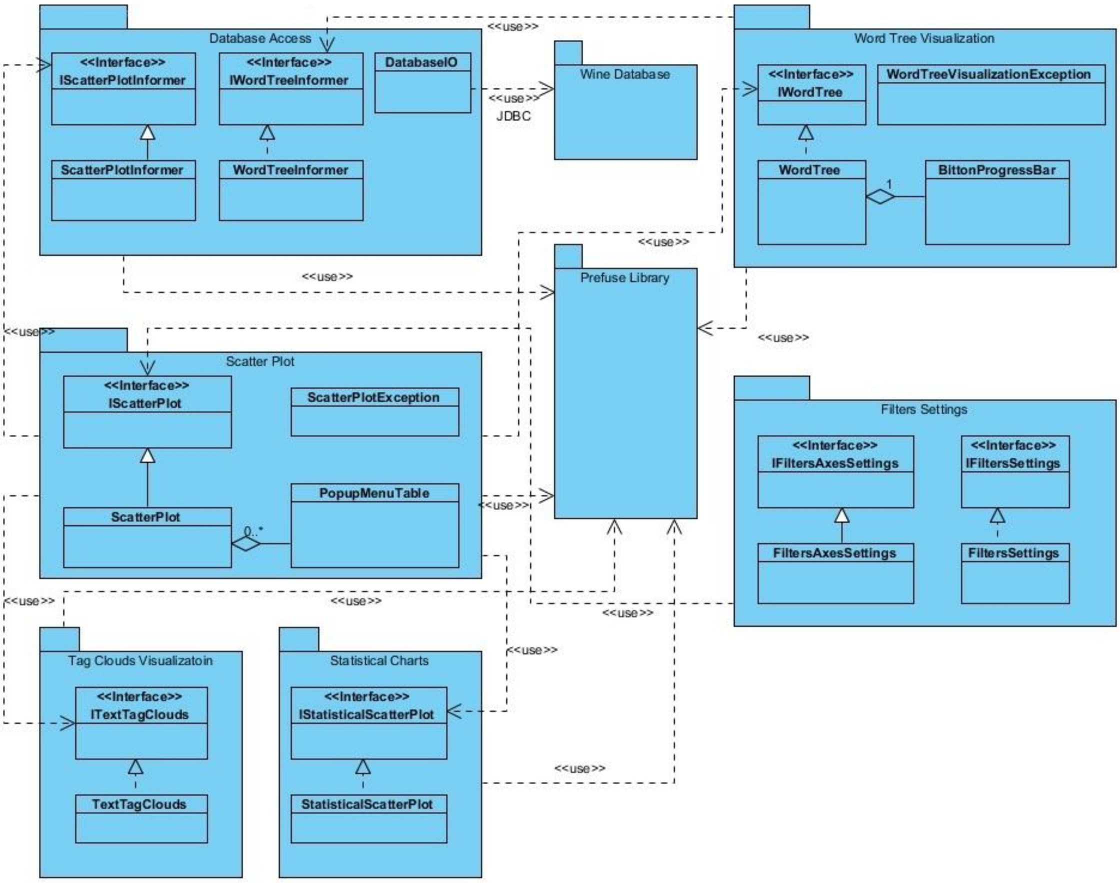

Figure 4.3 Class diagram of the Wine Tasting Notes Visualization prototype.

5. Interactive visualizations and user interface

This chapter contains details about the implementation of the visualizations described in Section 3.6 and the techniques integrated for user interaction with them. It also gives a notion of the user interface approaches and the overall layout of the application.

5.1. Starting point

The initial screen of the application is considered to combine many viewpoints of the overall data so that the user can rapidly become acquainted with the presented information. Figure 5.1 gives a snapshot of the screen appearing after starting the application. There are five particular parts intended to co-operate in building an efficient starting visualization: 1 – a scatter-plot; 2 – filters; 3 – bar chart diagrams; 4 – tag clouds text visualization; 5 – tags checkboxes panel. The functionality of each of these parts is described in the next sections.

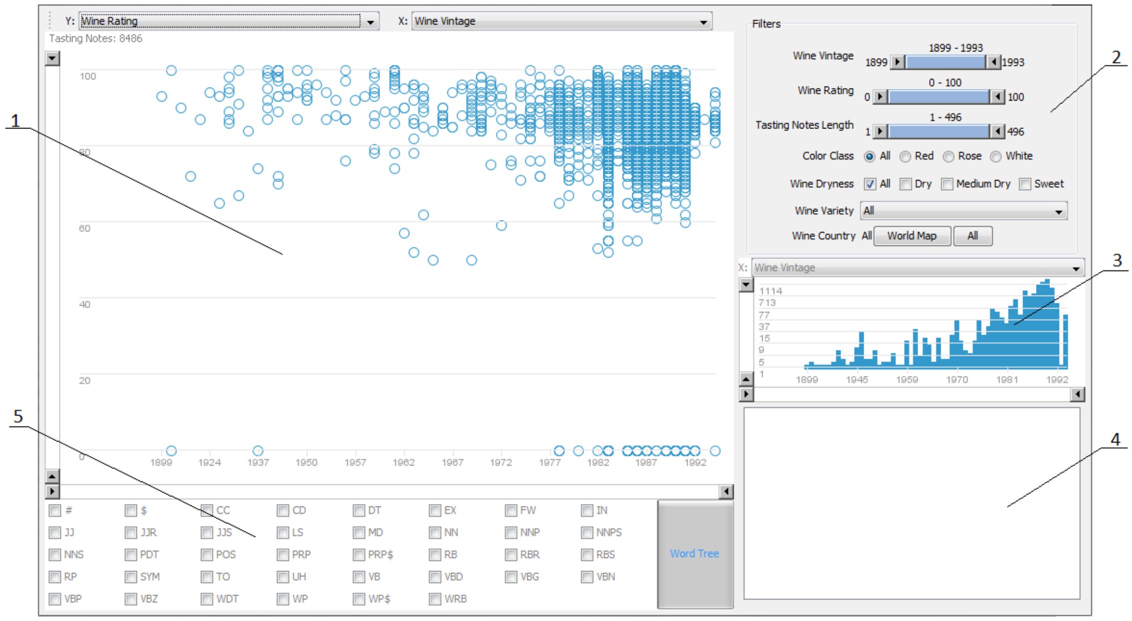

Figure 5.1 A snapshot of the starting screen of the application.

5.2. Scatter-plot visualization

The scatter-plot visualization approach and its purposes and advantages are described in Section 3.6.1. The scatter-plot provides information about the distribution of the tasting notes according to two of their wine attributes' values. The tasting notes are represented by circles. A snapshot of the implemented scatter plot of the "Wine tasting notes visualization tool" is given in Figure 5.2 a). The scatter-plot axes are laid in the left and bottom sides of the display and each one of them corresponds to a wine attribute. Range sliders are added to the axes so that users can change the range of wine attributes' values and therefore the scope of tasting notes visualized on the scatter-plot. In the given example the range of wine vintages is narrowed to be between the years 1964 and 1974 and the rating is between 80 and 100. Only the tasting notes corresponding to wines produced between 1964 and 1974 and having rating between 80 and 100 are visualized on the display as distinct from Figure 5.1 where all elements are visible. The number of tasting notes on the display can be observed in the upper left corner of the scatter-plot. In Figure 5.2 a) the number of currently visible elements is 79. Another possibility given to the user is to change the wine attributes plotted on the X-axis and Y-axis. In Figure 5.2 a) the selected wine attributes for the X and Y axes are respectively

"Wine Vintage" and "Wine Rating" and can be replaced by selecting other values from the combo boxes on top of the display.

Figure 5.2 a) Scatter-plot axes range sliders and wine attribute selection; b) Overlapping tasting notes; c) Tasting note selection.

There is a special case appearing as a consequence of the scatter-plot concept and the data stored in the database. It happens to be found more than one tasting note with the same values for both wine attributes plotted on the axes. Such tasting notes overlap each other as they are visualized at the same spot on the scatter-plot. This makes the selection of an element from the display a little more complicated. Figure 5.2 b) and c) demonstrate the means that we used to cope with this situation. A tooltip was added to each element to give a hint to the user about the number of overlapped tasting notes at the specific position (Figure 5.2 b). An individual element can be selected from a pop up list of the overlapping tasting notes that appears on click with the right mouse button (Figure 5.2 c). The selected tasting note is distinct from the others as it is colored in blue in the popup list and on the scatter-plot.

5.3. Filters

The range sliders integrated with the scatter-plot's axes give users the possibility to filter out some of the tasting notes on the display. More opportunities for these purposes are provided by the filters component. Figure 5.3 gives a snapshot of the filters panel implemented in the application. There are five types of filters intended to be used for restriction of the visualized tasting notes: radio groups, checkboxes, combo boxes, range slider and a world map. The variety of the filters depends on the wine attributes' data types and signification. Range sliders are used for selection of a range of values for a wine attribute instead of a specified single value.

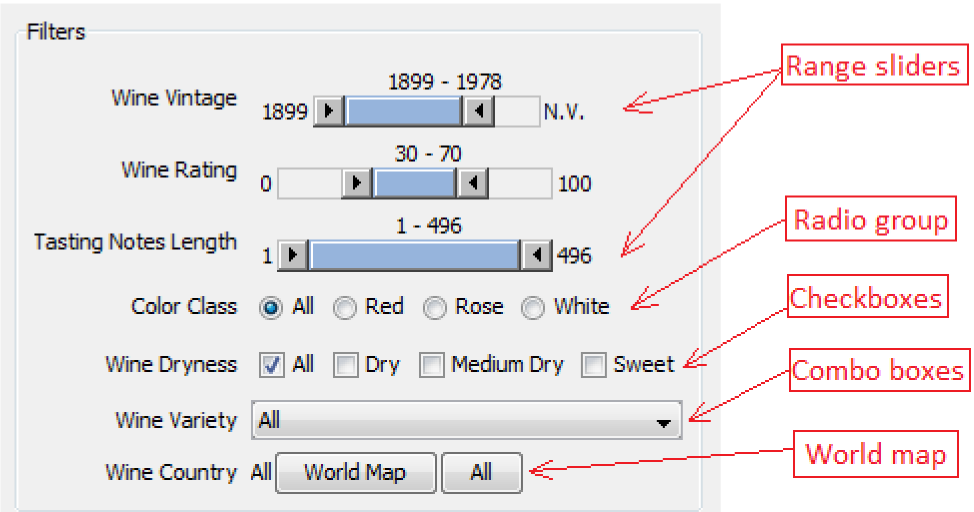

Figure 5.3 Types of filters implemented in the application.

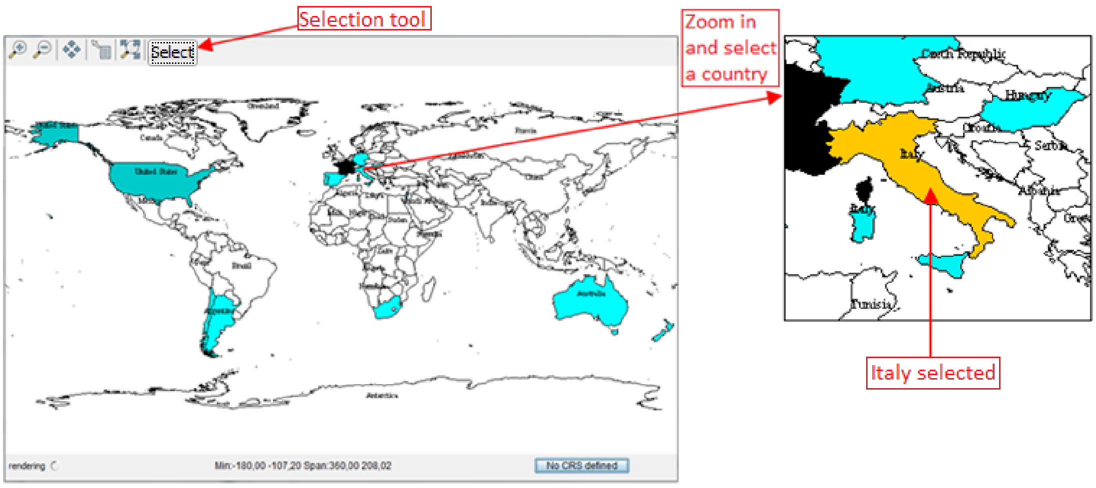

Figure 5.4 World map filter.

The world map filter uses a realization of the geographic visualization approach described in Section 3.6.4 to give users the possibility to filter out tasting notes in the scatter-plot depending on their origin. Figure 5.4 gives a snapshot of the world map filter. Different functionalities are provided to facilitate working with the map like zooming in, zooming out and panning to a specific region of interest. Users are given the possibility to select a country on the map. Only tasting notes of wines produced in the specified region will be visualized on the scatter-plot.

There is a property file used in the application that contains a list of wine attributes and their required filter types. Filters are dynamically created on the basis of this information and therefore can be easily added or removed. Exception is the world map filter as it is generated only for the countries wine attribute.

5.4.Bar chart diagrams

The purpose and usage of bar charts in this project are described in Section 3.6.5. There is a property file used in the application that contains a list of wine attributes that need to be represented by a bar chart diagram. An individual bar chart diagram is created for each of the listed attributes showing the number of visible tasting notes corresponding to each of their values. By reason of the restrictions imposed by the screen size, some improvements were applied to the classical bar chart diagram. Figure 5.5 presents snapshots of the bar chart diagrams implemented in the application. Range slider were added to the X and Y axes to assist users to change the range of visualized attribute values and to have a closer look at a specific section of the diagram (Figure 5.5 a) and b)). There is one diagram visualized at a time in order to save space. The attribute participating in the bar chart diagram in Figure 5.5 is “Wine Vintage” and it can be changed from a combo box containing the other possibilities (Figure 5.5 c).

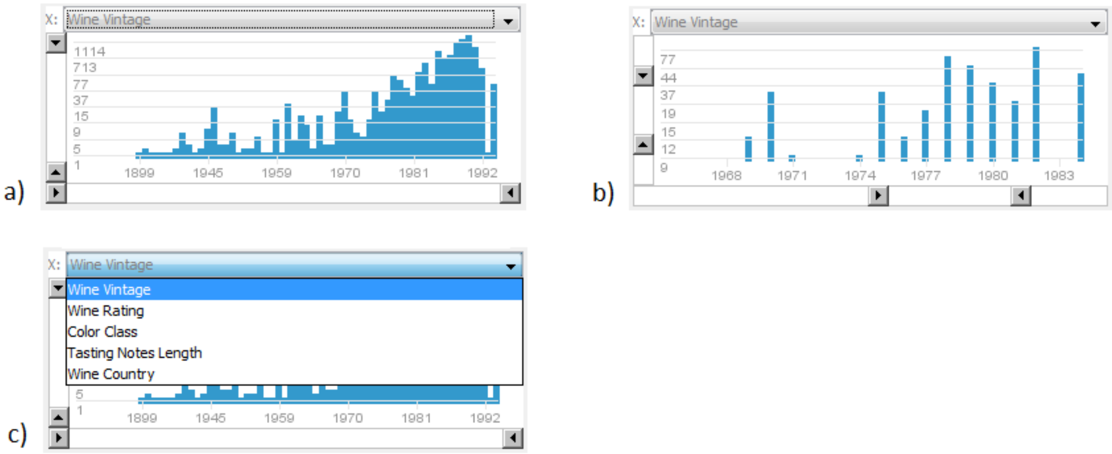

Figure 5.5 Snapshots of the bar chart diagrams implemented in the application.

5.5. Tag clouds text visualization and tags checkboxes panel

The concept and use of the tag clouds visualization approach was described in Section 3.6.3. It uses font size for the different words in a corpus of text to give a hint about the frequency of their usage. Figure 5.6 a) represents a snapshot of the tag clouds visualization implemented in the application. After a tasting note is selected from the scatter-plot its text is visualized using the tag clouds approach. The font size of each word is estimated according to the frequency of its occurrence in all currently visible elements including the selected one. A tooltip was added to the words to give accurate information about the number of times they occur in the analyzed tasting notes and the tag they belong to. It can be observed in the given example that the word “in” occurs 5924 times and it is a preposition.

The tags checkboxes panel contains word tags. Users need to check the tags they are interested in. A bilateral interaction between the tag clouds visualization and the checkboxes panel was implementing in order to aid users in making their choices. First, after the user selects one word from the tag clouds visualization all the words with the same tag are highlighted in the text together with the tag itself in the checkboxes panel. Figure 5.6 b) demonstrates this interaction after selecting the word “Last” in the tasting note. Second, after a tag is checked it is highlighted in the checkboxes panel together with all the words corresponding to this tag in the text.

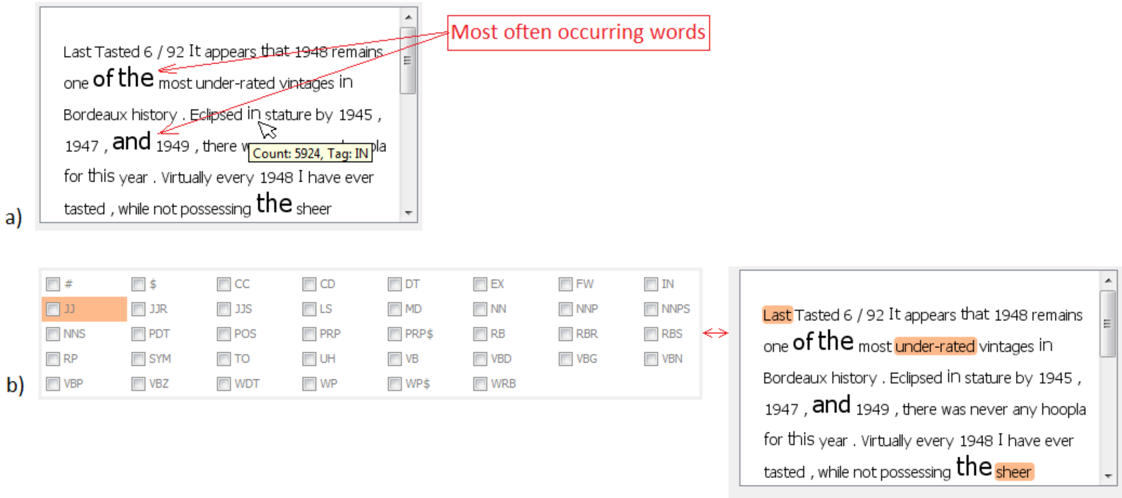

Figure 5.6 Snapshots of a) the tag clouds visualization implemented in the application and b) its interaction with the tags checkboxes panel.

5.6. Instantaneous response of the system

Information exploration is an interactive process between the user and the visualizations and it should be well supported by the systems potentialities. This process consists of series of questions and answers while each succeeding question depends on the prior answer [9]. To provide an adequate environment for data analyses the system needs to perform an instantaneous response to the users’ actions. The means that the wine tasting notes visualization tool supplies for asking questions are filters. Changing the value of a filter has an instantaneous effect on the displayed visualizations. The user does not need to perform an action to request for refresh of the displays. For example if the wines’ color class is changed from “All” to “red” the following three displays will be immediately refreshed: the scatter-plot will visualize just the tasting notes of red wines; the bar chart diagram will count the number of red wines; and the tag clouds visualization will estimate the words’ font size taking into account only the tasting notes of red wines. This feature is very beneficial for performing efficient data analyses.

5.7. Word tree visualization

The concept and structure of the word tree visualization was described in Section 3.6.3. Figure 5.7 gives a snapshot of a word tree generated by the system. The visualization consists of three basic components: (1) a display containing the word tree, (2) a text area presenting the text of the root tasting note, and (3) a text area presenting the currently constructed sequence of words. The texts of the nodes in the word tree corresponding to words that are contained by the root tasting note are colored in red. The nodes constructing the path from the root node to the currently selected node are on focus. Selecting a node from the tree changes the focus to the nodes contained by the path from the root to this node. A smooth animation is used to change the state of the tree to the newly selected focus. Some of the sub trees expand with their children flowing out from them and others become hidden. Each node has a degree of interest value that is estimated each time the focus changes. On the base of this value it is determined whether a node is to be visible or hidden. The nodes on focus are distinguished as highlighted and magnified. In the given example the currently selected node is “raspberries” and therefore all the nodes from the root to the node “raspberries” are on focus. These nodes construct a sequence of words that is contained in one or more tasting notes from the scatter-plot. This sequence of words is displayed in the bottom of the word tree and it is also highlighted in the root tasting note if contained there. In Figure 5.7 the currently constructed sequence of words is “glass, offering aromas of ripe raspberries.” The node’s texts are in red color as they are contained by the root tasting note and therefore they are highlighted in its content. Often the tree depth and width exceed the display bounds, and it is not possible all the nodes to be visualized in the available space. In order to surmount such problems different techniques are integrated into the visualization like zooming in, zooming out and panning controls. More detailed description of the methods employed to implement the visualization of the word tree can be found in [17].

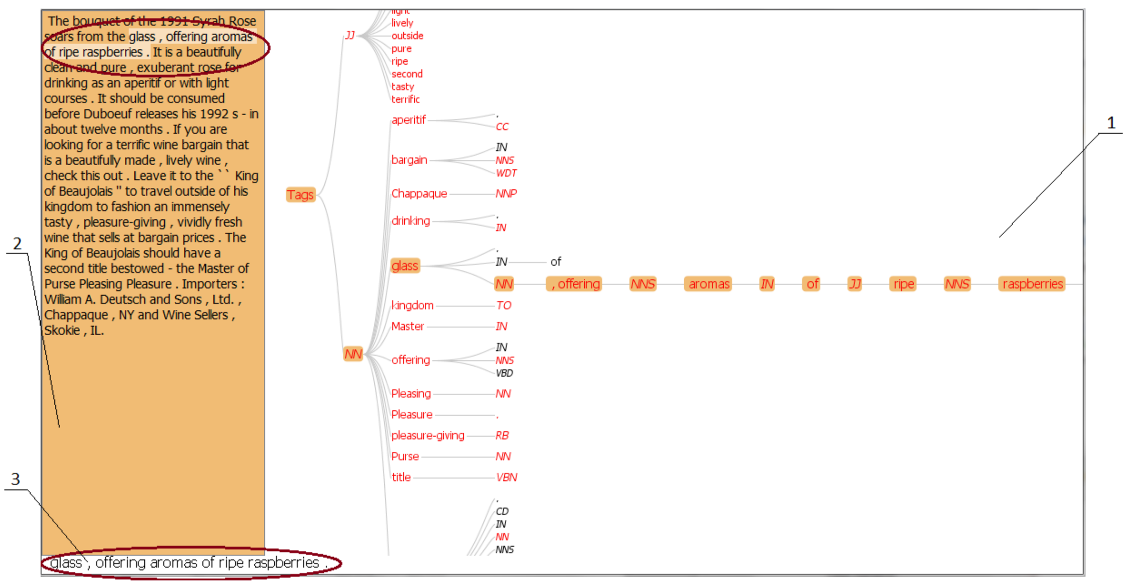

Figure 5.7 Word tree visualization consisting of three basic components.

There exists a close bond between the word tree visualization and the scatter-plot. The word tree is constructed according to all combinations of words beginning with words from the root tasting note and followed by words from all the tasting notes visualized in the scatter-plot. This means that each sequence of words constructed through the word tree exploration is contained by one or more tasting notes on the scatter-plot. This relation is indicated by highlighting the tasting notes on the scatter-plot containing the currently constructed sequence of words by the word tree. Figure 5.8 a) presents a snapshot of a highlighted tasting note on the scatter-plot. As described in Section 5.2 it happens to appear a special case of overlapping elements on the display. The same case arises here and it is not clear which of the overlapping tasting notes are highlighted. To avoid the possibility a not highlighted tasting note to overlap and hide a highlighted element the order of drawing the element on the scatter-plot is taken into account. Tasting notes that are not highlighted are drawn first and highlighted elements are drawn after that. In this way highlighted element are always visible at the front. To distinguish the highlighted elements, their texts are colored in blue in the pop up list of overlapping tasting notes. In the given example there is one tasting note that contains the sequence of words "glass, offering aromas of ripe raspberries" and its text is colored in blue in the pop up list (Figure 5.8. b). It can be noticed that it is the same tasting note that is selected for a root of the word tree as it is also highlighted in blue color.

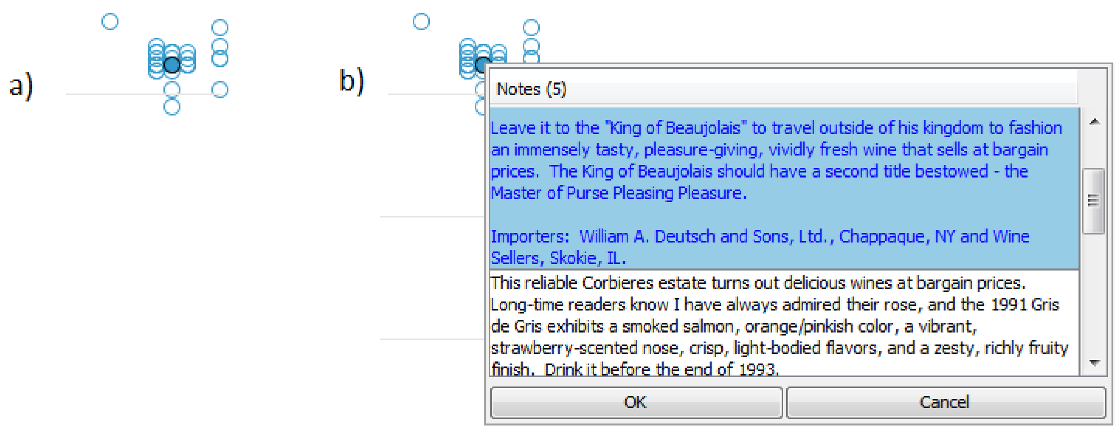

Figure 5.8 Highlighting the tasting notes on the scatter-plot that contain the currently constructed sequence of words by the word tree.

5.8. User guide

A user guide was added to the application to give assistance to the users in using the tool. Figure 5.9 presents a snapshot of the user guide window.

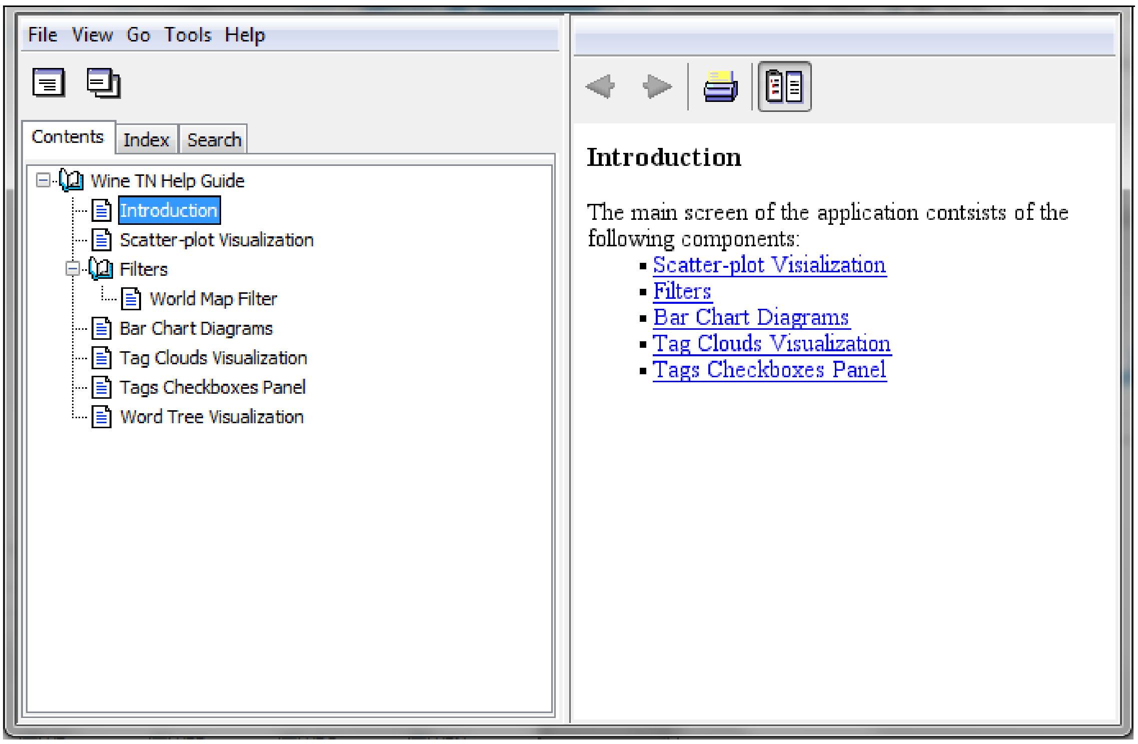

Figure 5.9 User guide describing the instructions for operating with the system.

Instructions for operating with each of the system components and visualizations were added to the user guide topics. There exists a possibility for searching in the topics for characteristic words and phrases. The user guide's content is presented in Appendix C.

6. Implementation

This chapter describes the process of implementing a prototype of the visualization tool. It specifies the programming language, programming environment and the software libraries used for developing the prototype. A description of the classes realizing the main functionality for each package is also included.

6.1. Programming language, software libraries and tools

In this section I will specify the means that we decided to use for implementing the prototype of the tool.

Operating system	Windows 7
IDE	Eclipse
Description	Eclipse is a multi-language software development environment comprising an integrated development environment and an extensible plug-in system. It can be used for developing applications in Java [21].
Advantages	It is free and provides a powerful debugging system.
Programming language	JAVA
Description	An object-oriented programming language
Advantages	There are many open source libraries written in JAVA.
Database Management System	Microsoft Access 2007
Description	Database management system
Advantages	The preliminary given database was created and managed by Microsoft Access.
Software library	Java Swing Toolkit
Description	The Java Swing toolkit is a powerful instrument for building GUI with good possibilities for users interaction with java applications. Swing is built on the Java 2D. It contains a rich set of components as table controls, list controls, tree controls, buttons, check boxes, labels, etc. If none of the provided components are suitable for the developers' purpose they are given the possibility to reuse the available functionality to create custom components [4].
Software library	Prefuse Toolkit
Description	Prefuse is a visualization toolkit that provides a rich set of software tools for creating interactive information visualizations. The original Prefuse toolkit provides a visualization framework for the Java programming language. The Prefuse toolkit features gives the opportunity for data modeling, visualization and interaction. It supports dynamic queries, integrated search, database connectivity and provides optimal representation of data structures as tables, graphs, trees, etc. Prefuse is easily integrated into Java Swing applications as it is implemented in Java using the Java 2D graphics library. Finally Prefuse is extensible in accordance to the developers needs as it is open source

	and can be also freely used for both commercial and non-commercial purposes [2].
Advantages	Prefuse is open source, maintained and well documented.
Software library	Geotools
Description	Geotools is an open source library for creating interactive geographic visualizations.
Advantages	Geotools is an open source library which is maintained and well documented.
Software library	JDBC
Description	The JDBC API is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases. The JDBC API provides a call-level API for SQL-based database access [23].
Advantages	The JDBC API can be used for connection and communication with a Microsoft Access database.
Software library	Oracle Help for Java
Description	Oracle Help for Java is a set of Java components, a Java API, and a file formats specification for developing and displaying HTML-based help content in a Java environment. OHJ is designed primarily for displaying help for Java applications, although it can also be implemented as a stand-alone document viewer for use in a Java environment [22].
Advantages	OHJ includes a set of default Java user interface components that together comprise a complete help system, with a table of contents, index, search, and topic windows [22].

Table 6.1 A list of software libraries and tools used in the system implementation.

6.2. Classes implementation

This section contains description of the classes implemented through the system developing process. Figure 4.3 (Section 4.5) presents a class diagram of the basic classes responsible for the required functionality of the prototype. The classes are distributed into several packages according to their purpose and functions.

6.2.1. Scatter-plot package

The classes in the scatter-plot package realize the functionality of the scatter-plot visualization. The class realizing the basic functionality of this package is ScatterplotVis and it is described in Table 6.2. The table contains listed only the methods responsible for the main functions of the class.

Class	ScatterplotVis
Description	The ScatterplotVis class is responsible for the scatter-plot visualization and organizes the communication between the other classes in the package. It implements the IScatterplotVis interface which is used by the classes in the Filters Settings package for communication. It sets the initial state of the application. It realizes functions like visualizing tasting notes; selecting a tasting note; changing the range of the axes values; estimation of the visualized elements and providing information

	to other components of the system that depend on the current state of the scatter-plot.
Methods	public int getSelectedItemPK(): gets the primary key of the selected item on the scatter-plot. public void setSelectedItem(int selectedNotePK): sets the selected item on the scatter-plot using the given as an argument primary key. public ArrayList<String> getCheckedTags(): gets a list of the selected tags that are used by the words tree generation. public void updateVisualization(): refreshes the display containing the scatter-plot visualization. public ArrayList<Integer> getSelectedElements(): returns a list of the tasting notes on the scatter-plot that correspond to the currently selected filters values. public Table getTagCloudsTable(): estimates the data necessary for the tag clouds visualization. public void setNotesTreeView(String newNotesTreeView, String sentence): sets and visualizes the tasting notes containing the currently constructed sentence by the word tree visualization. public void setDataFilters(AndPredicate newDataFilters): applies the currently selected filters values to the visualized tasting notes. public void refreshXScatterplotAxis (String newDataField): refreshes the wine attribute plotted on the X-axis of the scatter-plot. public void refreshYScatterplotAxis (String newDataField): refreshes the wine attribute plotted on the Y-axis of the scatter-plot.

Table 6.2 Description of the ScatterplotVis class.

The CheckBoxesPanel class realizes the selection of tags used in the word tree generation and tag clouds visualization. Table 6.3 contains description of this class and its basic methods. The CheckBoxesPanel class listens for ITagCheckBoxItemListener listeners and in this way communicates with the tag clouds visualization package (Table 6.4).

Class	CheckBoxesPanel
Description	The CheckBoxesPanel class implements the selection of tags that are a prerequisite for the word tree generation. Another functionality that is realized by this class is the highlighting of the tags corresponding to words from the tag clouds visualization.
Methods	public ArrayList<String> getCheckedValues(): gives a list of the selected tags. public void highlightTag (String tag): highlights a specified tag.

	public void clearHighlights (): clears highlights.

Table 6.3 Description of the CheckBoxesPanel class.

Class	ITagCheckBoxItemListener
Description	Listener interface for receiving events when a tag is selected in the checkboxes panel.
Methods	public void tagSelected(String tag): handles the event when a tag is selected in the checkboxes panel.

Table 6.4 Description of the ItagCheckBoxItemListener listener interface.

The Scatter-plot package throws a ScatterplotException exception to warn that a problem has occurred in this package.

6.2.2. Filters settings package

The filters functionality in the application is realized by the classes grouped by the Filters settings package. The basic classes in this package are FiltersSettings and WorldMap. The FiltersSettings class is responsible for the filters creation depending on their types. The wine attributes used for filtering and their respective types are described in the application's property file "Properies.properties". Filters are dynamically created using this information. The methods of the FiltersSettings class are listed in Table 6.5.

Class	FiltersSettings
Description	The FiltersSettings class is the class that creates and organizes different types of filters.
Methods	void createRadioGroupFilter(String filterDataField): creates a radio group filter using the values of the wine attribute given as an argument. void createCheckBoxGroupFilter(String filterDataField): creates a checkbox group filter using the values of the wine attribute given as an argument. void createComboBoxFilter(String filterDataField): creates a combo box filter using the values of the wine attribute given as an argument. void createWorldMapFilter(String filterDataField): creates a world map filter using the values of the wine attribute given as an argument. This method makes sense to be used only for the wine country attribute. void createRangeFilter(String filterDataField): creates a range slider filter using the range of values of the wine attribute given as an argument.

Table 6.5 Description of the FiltersSettings class.

The world map filter is implemented by the WorldMap class. It is described in Table 6.6.

Class	WorldMap
Description	The WorldMap class realizes the world map filter. It implements the functionality to visualize the world map and to represent the countries

	with different colors depending on their wine produce density. The most significant possibility realized by this class is the selection of a specific country.
Methods	void setMapFrameVisible(boolean b): displays the world map. public Object getItem(): returns the selected country. void clearSelection(): clear the selection of a country.

Table 6.6 Description of the WorldMap filter.

The filter settings package throws FiltersException exception to warn that a problem has occurred in this package.

6.2.3. Charts package

The charts package contains the classes realizing the bar charts visualizations. The most important class here is StatisticalScatterplot and it is described by Table 6.7.

Class	StatisticalScatterplot
Description	The StatisticalScatterplot class implements the bar chart diagrams. It realizes the possibility for changing the range of values plotted on the axis and getting more details for smaller regions of the diagrams. The wine attribute in the base of the diagram can be changed.
Methods	public void Initialize(Table newTable): initializes the bar chart diagram values using the given as an argument table of a wine attribute's values. public void updateTable(Table newTable): updates the bar chart diagram values using the given as an argument table of a wine attribute's values. public boolean isInitialized(): checks if the bar chart diagram is initialized.

Table 6.7 Description of the StatisticalScatterplot class.

6.2.4. Tag clouds package

The tag clouds package is responsible for the tag clouds visualization. The basic class in this package is TextTagClouds and it is described in Table 6.8. The TextTagClouds class implements the ITagCheckBoxItemListener listener interface and in this way communicates with the tags checkboxes panel.

Class	TextTagClouds
Description	The TextTagClouds class implements the tag clouds visualization of the text of the selected tasting note on the scatter-plot. This class realizes the opportunity a word to be selected an then this word to be highlighted in the text together with all the other words with the same tag and the tag itself in the check boxes panel.
Methods	public void update(Table table): updates the fonts of the words in the text using the data from the given as an argument table. public void tagSelected(String tag): handles the event when a tag is selected in the checkboxes panel.

	public void clearCurrentTag(): clears the selection.

Table 6.8 Description of the TextTagClouds class.

6.2.5. Word tree package

The word tree package contains classes that implement the functionality of the word tree visualization. The most important classes in this package are the WordTree and the ButtonProgressBar. The WordTree class is described in Table 6.9. The ButtonProgressBar class is described in Table 6.10. It implements IProgressBarListener listener interface and in this way communicates with the scatter-plot and the word tree (Table 6.11).

Class	WordTree
Description	The WordTree class implements the functionality of the word tree visualization. It realizes the nodes focus selection, animated nodes movement and words combination construction. It also visualizes the selected root tasting note and highlights the words if contained there. It uses the IScatterplotVis interface to communicate with the scatter-plot.
Methods	public void generateTree(): generates the word tree. public void highLightText(String text): highlights the given as an argument text in the text area containing the root tasting note. public JFrame createWordTreeJFrame(): creates a word tree JFrame containing the display with the word tree. public void stopGeneration(): stops the generation of the word tree.

Table 6.9 Description of the WordTree class.

Class	ButtonProgressBar
Description	The ButtonProgressBar class combines a button for the word tree generation and a progress status bar showing the status of the word tree generation. Clicking on the button starts the word tree generation and a second click stops the word tree generation.
Methods	public void ProgressBarClicked(IScatterplotVis newScatterplot, IPropertiesInformer newPropertiesInformer, int newNoteNumber, ArrayList<String> newNoteTagsArrList, ArrayList<Integer> newSearchNotesNumbersArrList): handles the event when a button progress bar is clicked.

Table 6.10 Description of the ButtonProgressBar class.

Class	IProgressBarListener
Description	Listener interface for receiving events when a button progress bar is clicked.
Methods	public void ProgressBarClicked(IScatterplotVis newScatterplot, IPropertiesInformer newPropertiesInformer, int newNoteNumber, ArrayList<String> newNoteTagsArrList, ArrayList<Integer> newSearchNotesNumbersArrList): handles the event when a button progress bar is clicked.

Table 6.11 Description of the IProgressBarListener class.

6.2.6. Database access package

The database access package contains classes that realize the connection of the other packages with the database and the data processing. A very important prerequisite before executing SQL queries to the database is the “Wine” table to be copied from the initial to the derived database (Section 3.1). The DatabaseIO class implements the database connection and the SQL query and DML statement execution. This class is described by Table 6.12. A DatabaseAccessException is thrown by the classes in this package.

Class	DatabaseIO
Description	The DatabaseIO class realizes the connection of the other packages with the database. It implements the execution of SQL query and DML statement.
Methods	public void openConnection(): opens a connection with the database. public ResultSet executeSQL(String sqlStr): executes an SQL query using the string provided as an argument. public void executeDDLDML(String ddlDmlStr): executes an DML statement using the string provided as an argument. public void closeConnection(): closes the connection with the database.

Table 6.12 Description of the DatabaseIO class.

The ScatterplotInformer class derives and provides the necessary information to the scatter-plot. It implements the IScatterplotInformer interface. Table 6.13 describes the method of this class.

Class	ScatterplotInformer
Description	The ScatterplotInformer class implements methods that are used by the scatter-plot visualization to retrieve and process data from the database.
Methods	public Table getDataTable(): retrieves the necessary data about the tasting notes from the database and structures this data into a table. public ArrayList<String> getWordTags(): returns a list of the tags abbreviations used in the database. public Table getTastingNotesWords (int primeryKey): returns a table of a tasting note’s words depending on the primery key provided as an argument of the method.

Table 6.13 Description of the ScatterplotInformer class.

The WordTreeInformer class derives and provides the necessary information to the scatter-plot. It implements the IWordTreeInformer interface. Table 6.14 describes the method of this class. The WordTreeInformer class has one method that derives the data necessary for the word tree generation from the database. The algorithm used by the getWordTree function is described in the next section.

Class	WordTreeInformer
Description	The WordTreeInformer class implements methods that are used by the word tree visualization to retrieve and process data from the database.

Methods	public Tree getWordTree(int noteNumber, ArrayList<String> noteTagsArrList, ArrayList<Integer> searchNotesNumbersArrList): retrieves the data necessary for the word tree and structures this data into a tree structure. public void stopGeneration(): stops the execution of the queries to the database and closes the connection.

Table 6.14 Description of the WordTreeInformer class.

The WorldMapInformer class derives and provides the necessary information to the world map realization. It implements the IWorldMapInformer interface. Table 6.15 describes the method of this class.

Class	WorldMapInformer
Description	The WorldMapInformer class implements methods that are used by the world map visualization to retrieve and process data from the database.
Methods	public Hashtable<String, Integer> getCountries(): returns a hashtable containing the data about the countries and the density of wines produced there.

Table 6.15 Description of the WorldMapInformer class.

6.2.7. Word tree generation algorithm

The algorithm used for the word tree generation consists basically of retrieving the necessary information from the "Notes" table in the database and organizing this data into a treelike structure. The content of the "Notes" table was described in Section 3.1. A small part of it is given in Appendix B to give an idea of the data stored there. It can be observed that no hierarchy of the words in a tasting note is described in the "Notes" table. A great number of SQL queries need to be executed in the database to organize the data into a treelike structure. The algorithm for constructing a hierarchical structure is described by Table 6.16.

Step 1	Assure selection of a tasting note to be the root of the word tree.
Step 2	Assure selection of a set of tasting notes including the root tasting note to participate in the word tree.
Step 3	Assure selection of a set of word tags that will be the nodes of the second level of the tree.
Step 4	Find all the words from the root tasting note that correspond to the selected word tags. Add each word as a child of the node that represents the respective tag selected in the previous step.
Step 5	In this step the algorithm searches in the set of tasting notes selected in step 2. Find all the occurrences of the words added in the previous step. Find the words that follow them and that are part of the same sentence. Add their tags as children of the nodes added in the previous step. Then add the words themselves as children to their respective tags.
Step 6	Repeat step 5 until no words are found.

Table 6.16 Algorithm used for the word tree generation.

The realization of this algorithm uses many SQL queries executions and they often become very long and complex. The reason for this is that the table structure is not intended for hierarchical queries. To solve this problem the structure of the "Notes" table needs to be adapted to describe a treelike organization of the words. This task is left for further work and analyses.

6.2.8. Help package

The help package implements the user guide integration into the application. The help system is realized using the Oracle Help for Java API described in Table 6.1. The integration of the Oracle Help API is realized by the WineHelp class. It is described by Table 6.17.

Class	WineHelp
Description	The WineHelp class integrates the Oracle Help for Java API into the application.
Methods	public static WineHelp getInstance(): returns an instance of the WineHelp class. This method is used for realizing the Singleton pattern. public void showTopic(String topic): shows the topic given as an argument of the method. public void setVisible(boolean b): manages the visibility of the window containing the help.

Table 6.17 Description of the WineHelp class.

7. Results and future work

In this chapter results of the thesis are described and discussed. It contains information about the problems and solutions through the process of working on the thesis topic. The chapter also presents future work that can be done in the project's area and possible improvements of the tool and database structure.

7.1. Results

The process of working on this thesis passed through different stages and has several significant outcomes.

An important result of the problem investigation and research was a number of problem solution concepts described in Section 3.4. These abstract ideas give different points of view for possible data structuring of the available information considering the needs and purposes of the linguists. Not all of these concepts were used along the lines of the thesis but some of them can be developed for further purposes in this or another closely related area. Such possibilities will be described later in this report.

Some of the problem solution concepts were developed further. A research was made for the proper visualization methods corresponding to the respective ideas. As a result of this work a combination of visualization approaches was created in order to facilitate linguists in their analyses of wine tasting notes.

Another significant result of the thesis was the development of a versatile prototype of a visualization tool for wine tasting notes analyses. The tool offers different possibilities for text exploration and data overview integrated with each other.

Several problems occurred in the process of implementation of the visualization tool and they are described in the next section.

7.2. Problems

The main problems that occurred in the thesis project were speed and deriving the data necessary for the word tree generation.

The database stores a great number of tasting notes (Section 3.5, Table 3.1). First the whole amount of data was used by the visualizations. This made interaction with the application and queries execution in the database very slow. To solve this problem the database was divided into 10 pieces and the tool was configured to use just one of them. The path to the database file used by the application is stored as a property in the Properies.properties file, and it can be easily changed by another one of the divided pieces.

One of the outstanding problems of the prototype is deriving the data necessary for the word tree generation from the database. The algorithm used for generating the word tree data structure is described in Section 6.2.7. It works well and relatively fast for a number of tasting notes that is generally no more than 50. There are several reasons for this obstacle. The algorithm uses data from the "Notes" table containing the tasting notes from the "Wine" table segmented into words (Section 3.1). One of the reasons for the problem is the "Notes" table design as it does not store the words in a treelike structure. This causes the execution of multiple SQL queries in order to organize the data into a tree and some of the statements become too complex. Considering the great amount of data it takes a lot of time and has a bad effect on the speed of the application. Another reason is a restriction of the Microsoft Access DBMS enforcing that the maximum number of characters in a SQL statement is approximately 64 000. As a result of this restriction, many of the queries happen to become too long and slowly for execution. A partial solution for improving the speed was to add a temporary table containing only the tasting notes that participate in the word tree. This step improved the speed of the application for a small number of tasting notes. Nevertheless, if a great number of tasting notes participates in the word tree generation the temporary table becomes too large and the same problem appears. Several steps were taken in order to make the user interface more friendly in this situation. A status bar was added to indicate the status of the word tree generation. Users are also provided the opportunity to stop the process of creating the tree if it takes too long. To solve this problem it is necessary to restructure the data in the "Notes" table. A possible solution is described in the next section.

7.3. Future work

There are several improvements that can be done to develop further the wine tasting notes visualization tool. First the data in the "Notes" table can be migrated into a treelike structure. This can be done by adding two new tables describing the hierarchy of the words in the tasting notes texts.

The tag clouds visualization can be improved. There are words that occur very frequently in general language like "a", "the", "of", etc. They are visualized by the largest fonts and therefore take the attention of the users from other words that are more important for the wine tasting notes analyses. It is a good idea to be added a list of words that are not taken into account in estimating the size of their fonts.

The concept facilitating sentences structure analysis (Section 3.4.5) can be improved and developed in the future. It can be useful not just for tasting notes analyses but for analyses of languages structure in general. The structure of the database is appropriate for this visualization as it is a result of statistical data that can be easily retrieved and estimated.

The world map visualization and performance also can be improved. It can be useful to add some interactive features to the map visualization. The map can be improved by visualizing the information considering time. An interesting idea is to add an interactive control for tracing the wine produce density in different countries depending on the wines' vintages. Range slider or combo boxes can be integrated to control the time period of the data visualized on the map.

7.4. Summery and conclusion

This thesis is concerned with the research for and implementation of various visualization approaches of wine tasting notes in order to aid wine language analyses. There was a database of tasting notes designed and provided in advance and the purpose was to visualize this data in a way that would give a good insight into it.

Several solution concepts were presented in the first part of the report. They were discussed with a domain expert and the most useful of them were chosen for further development. Some of the concepts that were not applied in this thesis were left for future work as they are appropriate not only for the wine language analyses but also for language analyses in general. A research was then made for finding the proper visualization approaches to represent the selected ideas.

The second part of the thesis was concerned with the analyses, design and implementation of a wine tasting notes visualization tool. The application was intended to combine the previously discussed visualization approaches and various techniques for interaction with them. Several improvements were discussed that can be developed in the future in order to make the tool faster and more interactive.

A number of ideas were created and discussed in working on the thesis problem. Several problems arose and different deductions were made. A prototype was implemented to demonstrate the benefits of the presented ideas. The results of this thesis can serve as good starting point for future analysis and development of a powerful visualization tool for tasting notes analyses.

References

[1] Gil Morrot, Frédéric Brochet and Denis Dubourdieu (August 28, 2001), "The Color of Odors", Published online;

[2] The Prefuse Visualization Toolkit, http://prefuse.org/, Jan 24, 2009;

[3] Many Eyes, http://manyeyes.alphaworks.ibm.com/manyeyes/, March 24, 2010;

[4] The Swing Toolkit, http://java.sun.com/docs/books/tutorial/ui/overview/intro.html, March 26, 2010;

[5] HOW MUCH INFORMATION 2003?, SIMS, University of California, Berkeley, http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/, May 2, 2010;

[6] A. Kerren, J. T. Stasko, J.-D. Fekete, and C. North (Eds.), Information Visualization - Human-Centered Issues and Perspectives, Volume 4950 of LNCS State-of-the-Art Survey, Springer, 2008;

[7] A. Kerren, A. Ebert, and J. Meyer (Eds.), Human-Centered Visualization Environments, Volume 4417 of LNCS Tutorial, Springer, 2007;

[8] Card, S.K., MacKinlay, J.D., Shneiderman, B. (eds.), Readings in Information Visualization: Using Vision to Think. Morgan Kaufmann, San Francisco, CA 1999

[9] Robert Spence, Information Visualization Design for Interaction, Second Edition, Pearson Education Limited, 2007

[10] InfoVis Wiki, Film Finder, http://www.infovis-wiki.net/images/2/21/Filmfinder.png, May 4, 2010

[11] Dynamic queries, starfield displays, and the path to Spotfire, University of Maryland, http://www.cs.umd.edu/hcil/spotfire/, May 4, 2010

[12] Ahlberg, Christopher and Shneiderman, Ben, Visual Information Seeking: Tight coupling of dynamic query filters with starfield displays, Proc. of ACM CHI94 Conference (April 1994)

[13] Andreas Kerren, Information Visualization I: Course Slide 4 - Interaction Techniques, 2009, Växjö University

[14] POS tags, BioIE Wiki, http://bioie.ldc.upenn.edu/wiki/index.php/POS_tags, May 5, 2010

[15] M.V. Wattenberg and F. Viegas, The Word Tree, an Interactive Visual Concordance. Visualization and Computer Graphics, 2008.

[16] George W. Furnas, Generalized Fisheye Views, Published in Human Factors in Computing Systems CHI '86 Conference Proceedings, 16-23. 1986.

[17] Jeffrey Heer, Stuart K. Card, DOITrees Revisited: Scalable, Space-Constrained Visualization of Hierarchical Data, 2004

[18] Stuart K. Card, David Nation, Degree-of-Interest Trees: A Component of an Attention-Reactive User Interface, Published by ACM New York, NY, USA, 2002

[19] Carita Paradis, "A sweet nose of earth, smoke, cassis and cherries, Descriptions of sensory perceptions in wine-tasting notes", unpublished

[20] Geographic Visualization, Wikipedia, http://en.wikipedia.org/wiki/Geovisualization, August 20, 2010

[21] Eclipse (software), Wikipedia, http://en.wikipedia.org/wiki/Eclipse_(software), August 24, 2010

[22] Oracle Help Guide, http://www.oracle.com/webapps/online-

help/ohguide/2.0/state/content/navId.3/navSetId.ohguide/vtTopicFile.ohguide|ohg_about_about~html/vtTopicId./, August 27, 2010

[23] JDBC Overview,
http://java.sun.com/products/jdbc/overview.html, August 28, 2010

Appendix A

Tag abbreviations

Tag	Description
''	''
#	#
$	Dollar sign
,	Comma
.	Period
:	Colon
``	``
CC	Coordinating conjunction
CD	Cardinal number
DT	Determiner
EX	Existential there
FW	Foreign word
IN	Preposition or subordinating conjunction
JJ	Adjective
JJR	Adjective, comparative
JJS	Adjective, superlative
LS	List item
MD	Modal verb
NN	Noun, singular common
NNP	Noun, singular proper
NNPS	Noun, plural proper
NNS	Noun, plural common
PDT	Predeterminer
POS	Possessive
PRP	Personal pronoun
PRP$	Personal pronoun, possessive
RB	Adverb
RBR	Adverb, comparative
RBS	Adverb, superlative
RP	Particle
SYM	Symbol
TO	To
UH	Interjection
VB	Verb, base form

VBD	Verb, past tense
VBG	Verb, present participle
VBN	Verb, past participle
VBP	Verb, present tense, not 3rd person singular
VBZ	Verb, present tense, 3rd person singular
WDT	Wh-determiner
WP	Wh-pronoun
WP$	Wh-pronoun, possessive
WRB	Wh-adverb

Appendix B

Sample data contained in the "Notes" table.

Notes					
EntryN	**SentenceN**	**WordN**	**Word**	**Tag**	**LabelID**
7301	1	1	The	DT	
7301	1	2	1989	CD	
7301	1	3	red	JJ	
7301	1	4	wine	NN	
7301	1	5	exhibits	VBZ	
7301	1	6	an	DT	
7301	1	7	impressive	JJ	
7301	1	8	deep	JJ	
7301	1	9	ruby	NNP	
7301	1	10	/	NNP	
7301	1	11	purple	JJ	
7301	1	12	color	NN	
7301	1	13	,	,	
7301	1	14	a	DT	
7301	1	15	bold	JJ	
7301	1	16	,	,	
7301	1	17	cassis	NN	
7301	1	18	and	CC	
7301	1	19	smoked	VBD	
7301	1	20	herb-scented	JJ	
7301	1	21	bouquet	NNP	
7301	1	22	,	,	
7301	1	23	opulent	JJ	
7301	1	24	,	,	
7301	1	25	rich	JJ	
7301	1	26	flavors	NNS	
7301	1	27	,	,	
7301	1	28	full	JJ	
7301	1	29	body	NN	
7301	1	30	,	,	
7301	1	31	and	CC	
7301	1	32	a	DT	
7301	1	33	gutsy	JJ	
7301	1	34	,	,	
7301	1	35	chewy	JJ	
7301	1	36	finish	NN	
7301	1	37	.	.	
7301	2	1	It	PRP	

Notes					
EntryN	**SentenceN**	**WordN**	**Word**	**Tag**	**LabelID**
7301	2	2	is	VBZ	
7301	2	3	an	DT	
7301	2	4	ideal	JJ	
7301	2	5	wine	NN	
7301	2	6	for	IN	
7301	2	7	drinking	NN	
7301	2	8	with	IN	
7301	2	9	bistro-styled	JJ	
7301	2	10	dishes	NNS	
7301	2	11	over	IN	
7301	2	12	the	DT	
7301	2	13	next	JJ	
7301	2	14	4-5	CD	
7301	2	15	years.	NNP	
7401	1	1	The	DT	
7401	1	2	1990	CD	
7401	1	3	white	JJ	
7401	1	4	wine	NN	
7401	1	5	possesses	VBZ	
7401	1	6	plenty	NN	
7401	1	7	of	IN	
7401	1	8	flesh	NN	
7401	1	9	and	CC	
7401	1	10	fat	JJ	
7401	1	11	to	TO	
7401	1	12	make	VB	
7401	1	13	up	RP	
7401	1	14	for	IN	
7401	1	15	its	PRP$	
7401	1	16	lack	NN	
7401	1	17	of	IN	
7401	1	18	elegance	NN	
7401	1	19	.	.	
7401	2	1	Pair	VB	
7401	2	2	it	PRP	
7401	2	3	with	IN	
7401	2	4	spicy	NN	
7401	2	5	,	,	
7401	2	6	grilled	JJ	
7401	2	7	chicken	NN	
7401	2	8	and	CC	

Notes					
EntryN	**SentenceN**	**WordN**	**Word**	**Tag**	**LabelID**
7401	2	9	fish.	NNP	

Appendix C

User guide of the wine tasting notes visualization tool

The main screen of the application contsists of the following components: scatter-plot visialization; filters; bar chart diagrams; tag clouds visualization and tags checkboxes panel (Figure C-1).

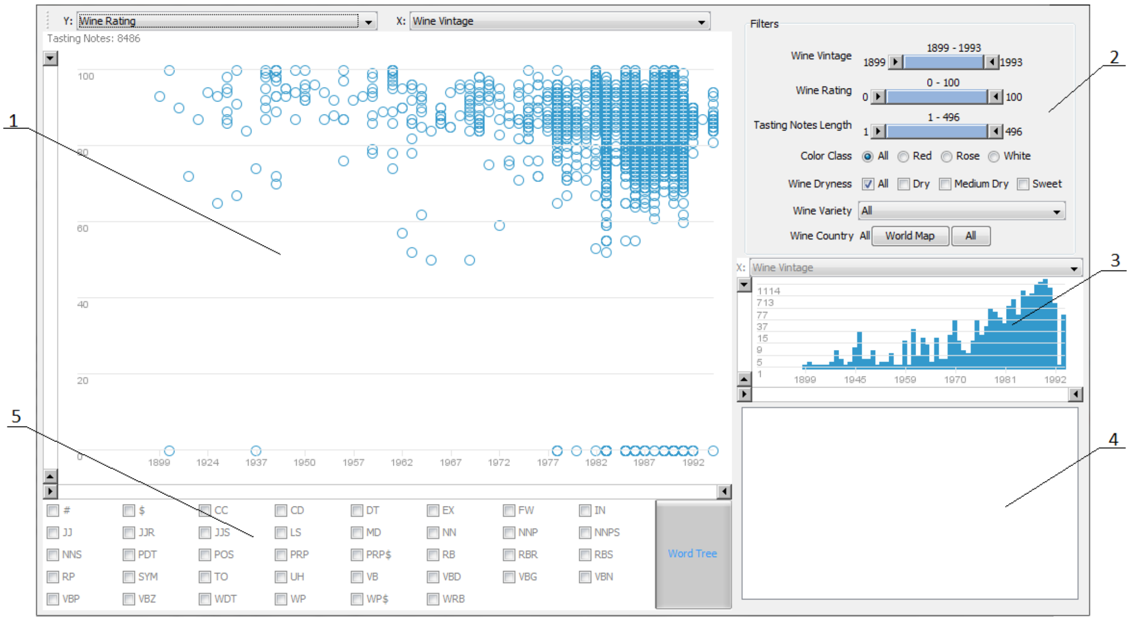

Figure C-1 A snapshot of the starting screen of the application.

Scatter-plot visualization

The scatter-plot visualizes the tasting notes distributed according to two of their wine attributes (Figure C-2). The attributes plotted on the scatter-plot's axes can be changed by the comboboxes above. The range of the visualized tasting notes can be changed by the range slider integrated to each of the scatter-plot's axes.

Only one tasting note can be selected at any time. On left mouse click on an element from the scatter plot users get a pop up list of all the tasting notes corresponding to the respective attributes and can select one of them. Then the selected tasting notes is colored in blue on the scatter-plot and in the pop up list.

The text of the currently selected tasting note is visualized by the tag clouds panel. A tooltip is added to each scatter-plot element. It provides information about the number of overlapping tasting notes on the current location. Users can see it when pointing at a tasting note.

Filters

The filters are used to filter out the elements from the scatter-plot (Figure C-3). There are five types of filters: range sliders, radio groups, checkboxes, comboboxes and a world map filter. The world map is a type of filter that gives users the opportunity to select a country from a visualized world map (Figure C-4). It provides the following functionalities realized by the tools in the toolbar on the top of the map:

1. Zooming in and zooming out – the map can be zoomed in and zoomed out by right clicking on the interesting area.
2. Panning – the map can be panned by right clicking and dragging to the interesting area.
3. Displaying full extend of the map.

4. Selecting a country – a country can be selected by right clicking on the respective area.

Figure C-2 a) Scatter-plot axes range sliders and wine attribute selection; b) Overlapping tasting notes; c) Tasting note selection.

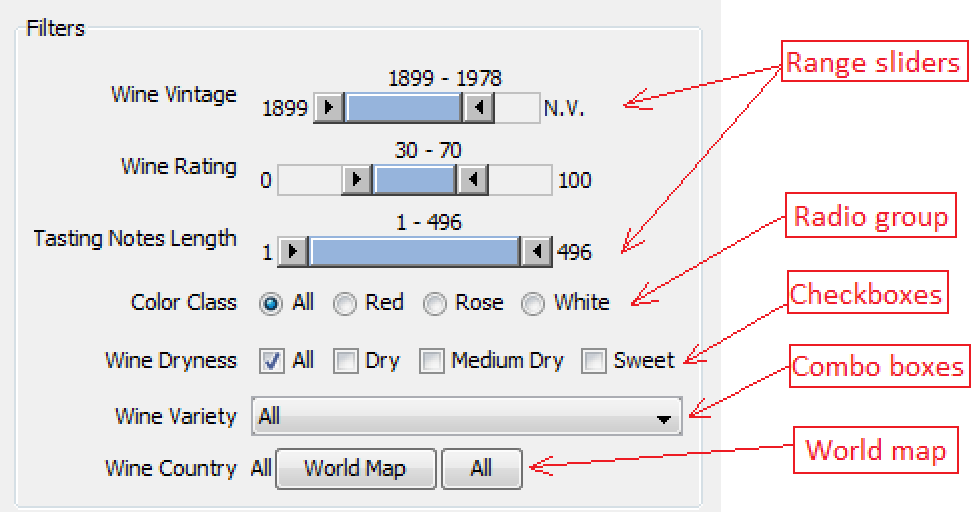

Figure C-3 Types of filters implemented in the application.

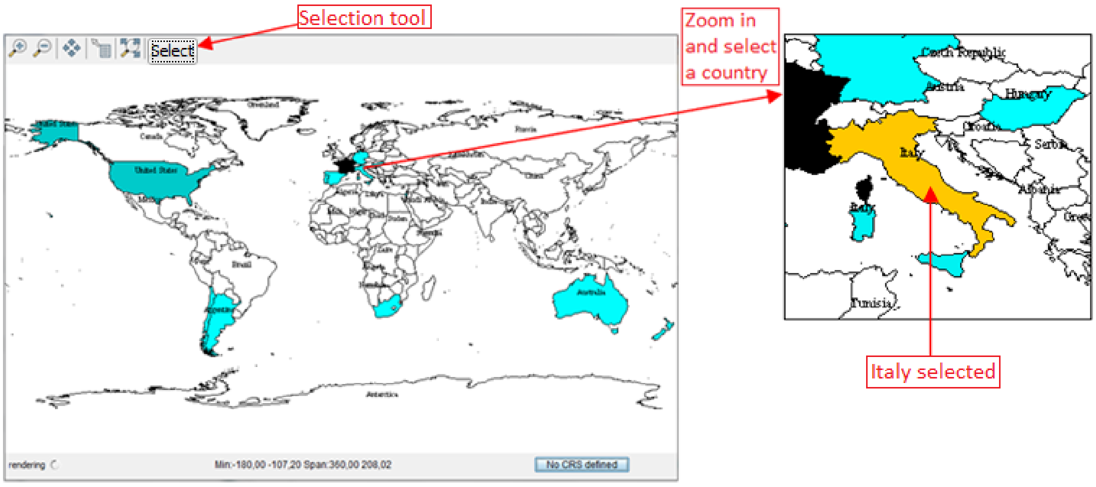

Figure C-4 World map filter.

Bar chart diagrams

The bar chart diagrams provide information about the the number of visible tasting notes corresponding to each of their wine attribute values (Figure C-5). An individual bar chart diagram is created for each particular wine attribute. Only one diagram is visualized at any time. Another one can be selected from the combobox above.

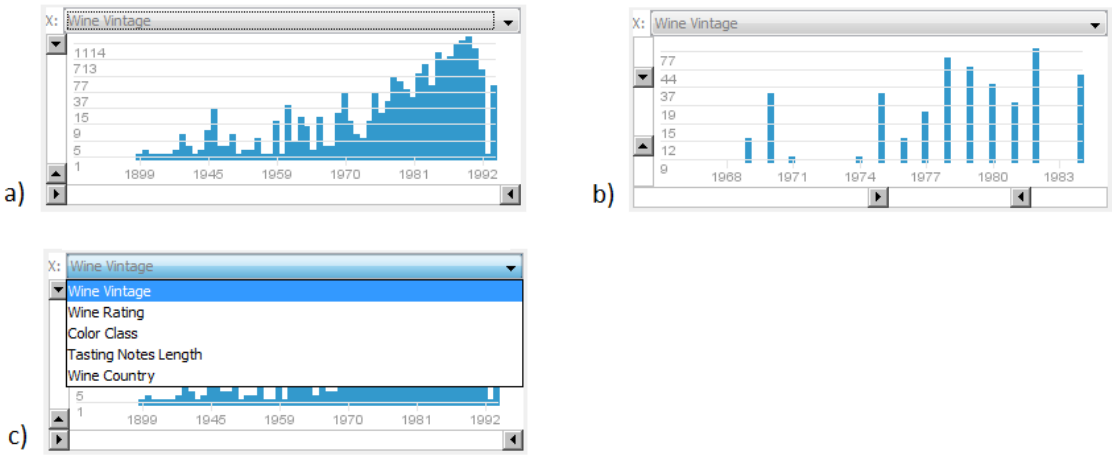

Figure C-5 Snapshots of the bar chart diagrams implemented in the application.

Range slider are integrated to the X- and Y-axes so that users can change the range of visualized attribute values and can have a closer look at a specific section of a bar chart diagram. A tooltip providing more accurate information are available on pointing at each bar from the diagram.

Tag Clouds Visualization

The tag clouds panel visualizes the text of the currently selected tasting note from the scatter-plot. It uses font size for the different words in a corpus of text to give a hint about the frequency of their usage (Figure C-6).

Each word has a tooltip that gives information about the number of times it occurs in the analyzed tasting notes and the tag it belongs to. The tooltip is shown when the user points at a word. On left mouse click on a word all the words with the same tag are highlighted in the tag clouds panel together with the tag itself in the tags checkboxes panel. The highlights can be cleared by clicking the "Clear highlights" button below.

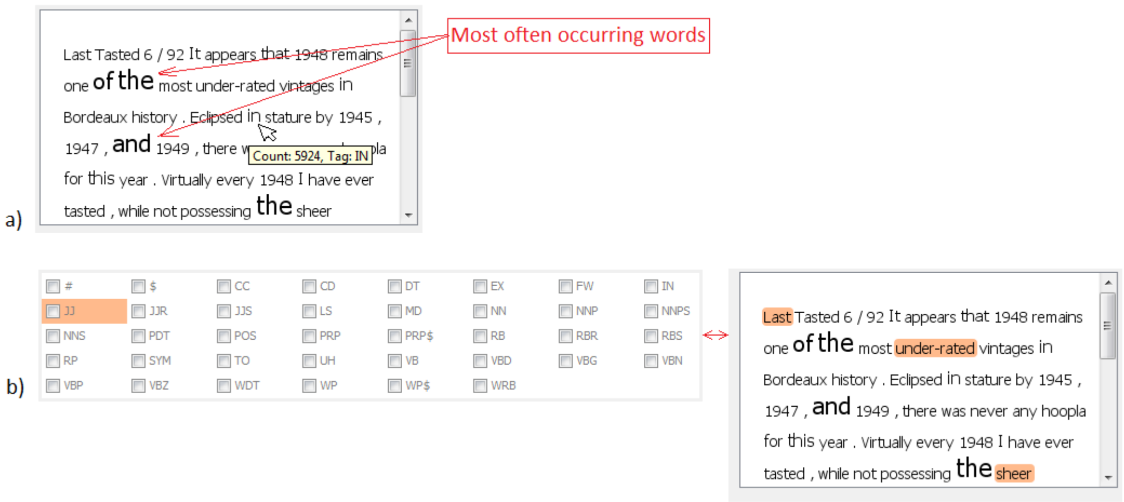

Figure C-6 Snapshots of a) the tag clouds visualization implemented in the application and b) its interaction with the tags checkboxes panel.

Tags Checkboxes Panel

The tags checkboxes panel contains word tags. Users need to check the tags they are interested in. On left mouse click the respective tag is checked and highlighted in the checkboxes panel together with the corresponding words in the tag clouds panel. The "Word Tree" button is used for the word tree visualization generation.

Word Tree Visualization

The word tree represents the text in a way that gives information about all combination of words found in a group of tasting notes (Figure C-7). The initial words are all the words from the currently selected tasting notes from the scatter-plot that belong to the checked tags from the tags checkboxes panel. The words are grouped together according to their tags. The words contained by the currently selected tasting note are colored in red and the others are in black. On the left side of the tree is visualized the selected tasting note.

The currently constructed combination of words is highlighted in the displayed text if it is contained by it.

The navigation in the tree is possible in the following way:

1. Left mouse button click on a tree node – changes the focus to the clicked word.

2. Right mouse button click with no dragging – zooms the display such that all nodes of the tree fit within the display bounds.
3. Pressing the right mouse button on the background and dragging the mouse up or down – zooms the display in and out.
4. Mouse wheel scrolling – zooms the display in and out.
5. Clicking on the background with the left mouse button and then dragging – pans the display changing the viewable region of the word tree.

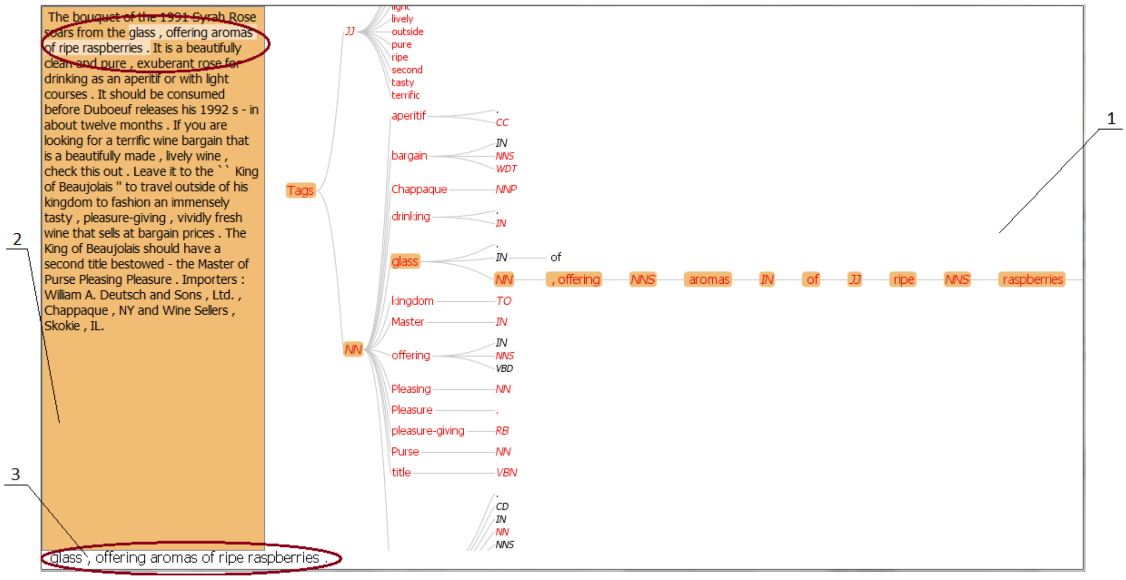

Figure C-7 Word tree visualization consisting of three basic components.

Linnæus University
School of Computer Science, Physics and Mathematics

SE-351 95 Växjö / SE-391 82 Kalmar
Tel +46-772-28 80 00
dfm@lnu.se
Lnu.se