First publ. as paper in: IEEE Symposium on Visual Analytics and Technology (VAST 2008), October 21 - 23, Columbus, Ohio, USA

Visual Evaluation of Text Features for Document Summarization and Analysis

Daniela Oelke* Peter Bak* Daniel A. Keim*
University of Konstanz

Mark Last† Guy Danon‡
Ben-Gurion University of the Negev

ABSTRACT

Thanks to the web-related and other advanced technologies, textual information is increasingly being stored in digital form and posted online. Automatic methods to analyze such textual information are becoming inevitable. Many of those methods are based on quantitative text features. Analysts face the challenge to choose the most appropriate features for their tasks. This requires effective approaches for evaluation and feature-engineering.
In this paper we suggest an approach to visually evaluate text-analysis features as part of an interactive feedback loop between evaluation and feature engineering. We apply document-fingerprinting for visualizing text features as an integral part of the analytic process. Consequently, analysts are able to access interim results of the applied automatic methods and alter their properties to achieve better results.
We implement and evaluate the methodology on two different tasks, namely opinion analysis and document summarization and show that our iterative method leads to improved performance.

Index Terms: I.7.5 [Document and Text Processing]: Document Capture—Document Analysis; I.5.2 [Pattern Recognition]: Design Methodology—Feature evaluation and selection

1 INTRODUCTION

1.1 Motivation

Thanks to the web-related and other advanced technologies, textual information, ranging from news reports to literature, is increasingly stored in digital form and posted online - and textual information is still the most important source of information. Search engines such as Google have helped us to access this information but do not provide advanced tools for analysis and mining. The major challenge in computational text analysis is the gap between automatically computable text features and the users' ability to control and evaluate these features.

Traditional text mining approaches consider feature selection and extraction as their fundamental task [9]. Usually, a set of documents is described by one or several feature types. For example, a text summarization task can be carried out using term-frequencies as document features. The evaluation of the results, as delivered by the selected feature subset, is then conducted by a comparison to some ground truth provided by humans or alternative algorithms. Thus, the ground truth is used for assessing the quality of the feature subset for the described task. Methods such as confusion matrices, and F-measure based on precision and recall are often applied to assess the quality of the applied features. Evaluation is often enhanced by visualizations of the results. Conventional visualization techniques, such as diagrams and charts, can provide additional information, such as the utility of features, deviations from an expected outcome, and strengths or weaknesses of the applied feature. Such simple techniques however, do not treat visualization as an integral part of the analysis process.

The main contribution of the current research is to suggest an approach to visually evaluate text-analysis features as part of an iterative feedback loop between evaluation and feature engineering. Through this iterative process analysts are able to alter properties and functioning of algorithmic feature extraction methods and re-evaluate their results to improve the final outcome of the analytic process. Visualization in this context plays an indispensable role. It helps users to access text-properties, such as development over time, homogenous / heterogeneous sequences, location of interesting patterns, and optional combination of different features, which could not have been accessed otherwise. The proposed approach is implemented and evaluated on two different text mining tasks, namely opinion analysis and document summarization. Fingerprint visualization is applied at different stages of the analytic process and for different task purposes. The iterative process of visually evaluating documents for both tasks was shown to be beneficial and significant insights could be gained. However, suggesting new algorithmic methods and tools for opinion analysis and summarization is out of scope of the current paper. Rather, the aim of the proposed research is to suggest a new methodological framework that allows an efficient evaluation of interim results, and provides an iterative loop to improve algorithmic performance. This is achieved by applying visual analytic methods to text mining.

The paper is organized as follows: First the literature on text-mining will be reviewed. Then the visual text feature evaluation approach will be described. The approach will be applied in the subsequent sections, in which opinion analysis and document summarization will be introduced and examples shown. Finally, conclusions on the proposed research methodology are drawn and further research is suggested.

1.2 Related Work

To the best of our knowledge, so far no technique exists that visually evaluates text features with respect to their capability to represent a certain property of the text. However, there are some areas close by that our technique is based on which are text feature visualization, visual (non-text) feature evaluation, and of course, automatic (text) feature evaluation and text visualization in general. In the following we will quickly review the most important existing approaches.

Text feature visualization techniques that present a single (or a small number of) documents in detail include TileBars [12], Seesoft [5], the FeatureLens [7], and Literature Fingerprinting [16]. The four techniques differ from each other with respect to the area of application they were designed for (Software Visualization, Information Retrieval, Text Analysis), the analysis task (detection of patterns, comparison of multiple features, identification of the temporal evolution) and the properties of the visualization technique (use of structural information, level of resolution, overlapping or disjoint text-blocks etc.).
Beyond that, the visualization technique Ink Blots of Abbasi and

*e-mail: {oelke, bak, keim}@inf.uni-konstanz.de
†e-mail: mlast@bgu.ac.il
‡e-mail: guy.danon@gmail.com

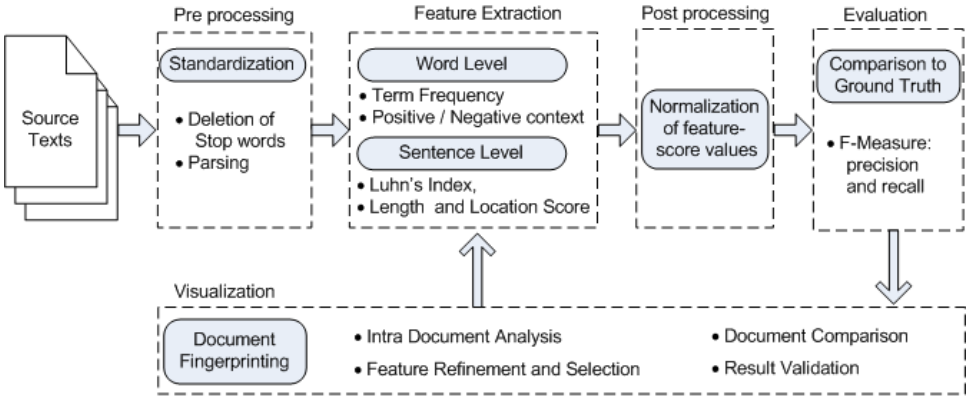

Figure 1: The pipeline for visual evaluation of text features applied for document summarization and analysis. Visualization is used to create a link between evaluation and feature engineering.

Chen [4] and the Compus system of Fekete and Dufournaud [8] have to be mentioned in the context of text feature visualization. In contrast to all the previously mentioned techniques, Compus and Ink Blots visualize a multitude of features at once in one single visualization. Due to the large amount of overplotting that both techniques accept they are restricted to features that do not provide values for each single text unit (such as word or sentence).

In addition to this, a large number of approaches for visualizing retrieval results (see e.g. VIBE [17] or InfoCrystal [26]) exists. Furthermore, a variety of techniques concentrates on the visualization of large document collections, most of them being based on dimensionality-reduction methods (see e.g. WebSOM [18], Galaxies and ThemeScape of IN-SPIRE^{TM} [27], or [10]). In contrast to the text feature visualization techniques those approaches do not visualize a single document in detail but illustrate the relations of the documents among each other.

However, the focus of this paper is not on the visual representation of a document and its properties in the first place , but instead on the evaluation of text features with respect to their predictive power for a specific text property. This is why existing approaches in the area of feature evaluation (even if not applied in the context of text) are considered as related. A fundamental distinction in this context is the one between supervised and unsupervised feature evaluation. Among the standard methods for supervised feature evaluation is the estimation of the classification accuracy by means of calculating the precision, recall, and F-measure, for example. Furthermore, a bunch of automated feature selection methods exists that are frequently applied as a preprocessing step to select a subset of features that is best suited to discriminate between the given classes (see [19] for a review on feature selection methods). In contrast to this, unsupervised feature evaluation techniques like approaches that are based on the cluster validity measures (see [11] e.g.) are also applicable when no benchmark data exists. Both for supervised as well as unsupervised feature evaluation visual approaches exist, too. In this context, the work of Keim / Schreck et. al on supervised [15] and unsupervised [24] visual feature evaluation has to be mentioned. Both approaches are based on SOMs and analyze the underlying distance distributions of the features. All of those approaches have in common that applying them to the analysis of inner-document features would mean that an important property of text would be lost, namely the information about the behavior of the feature values across the document.

2 Visual Text Feature Evaluation

In the current paper, we suggest a visual-analytics approach for feature engineering in text analysis that extends traditional approaches by applying visualization for creating a feedback loop between feature extraction and evaluation. As a result, feature engineering becomes an interactive and iterative process, in which developers (or professional analysts) evaluate the performance of text features through visualization and re-alter properties of the features accordingly.

A schematic description of the analysis process is presented in Figure 1. The feature engineering process as part of text analysis starts with a set of documents with a known ground truth that have to be preprocessed for feature extraction. Feature extraction is carried out by automatic algorithms that compute feature values for the document at different levels (word, sentence, etc). These features provide a quantitative assessment of the documents. They can point to importance of passages within the document, or classify sentences for positive / negative statements, as an example. These feature values are post-processed, in order to allow an evaluation of the features. The evaluation is usually of statistical nature, where correct and incorrect assessments are computed and compared. We propose to enrich this process by including visualization techniques in the process that can help to select the best features respectively meaningful combinations of features. Based on the visual evaluation, developers can iteratively alter properties of the features. Through iterations, the results of the feature extraction can be improved and an effective and efficient comparison between different settings of the features can be conducted. The iterations of evaluating and extracting features end, when the developer is satisfied with the extracted information. Afterwards the selected features can be used in the analysis of real data sets.

The main benefits of the proposed approach are in allowing analysts to refine and select features of interest. This is done by comparing the performance of different features on a small benchmark data set. Additionally, the technique can be used to detect correlations between text features and other parameters, such as location and time related properties of the document. This information may indicate which combination of features could be meaningful. Finally, the selected (combinations of) features can be applied on large real data sets for which no annotation exists.

Visually analyzing the data has the advantage that developers do not need to specify formally what they are looking for. Instead, the humans' perceptual abilities can be exploited. Of course, as soon as a promising feature or a combination of features has been selected automatic evaluation techniques can be used to further evaluate its performance on a larger benchmark data set (provided that a larger benchmark data set is available). One of the disadvantages of using visual methods is that they are not as scalable as automatic methods are. This problem is alleviated by the fact that for getting an idea

about which features should be selected or could be combined, it is enough to use a small data set. Automatic methods may be used to confirm or decline the assumptions on larger data sets. Another problem of the approach is that it is dependent on the availability of benchmark data. So it cannot be applied if no such ground truth is available. However, this is not only a problem for the visual methods but also for the automatic ones. In this case unsupervised evaluation methods would have to be used which is a research question on its own.

The visualization technique that is presented in this paper is based on Literature Fingerprinting that was presented in [16]. In this technique documents are represented by a pixel-based visualization in which each pixel represents one unit of text and the pixels are arranged from left to right and top to bottom which results in a compact and scalable visualization. The color of each pixel is mapped to its feature value and therefore allows to analyze the behavior of the feature values across the text in detail. Furthermore, the visualization takes the document structure into account allowing for an analysis of correlations between the feature and the structural elements. The document structure is also used for the transitions between different resolutions and to provide meaningful aggregations to the next hierarchy level.

The following sections will report two tasks, namely opinion analysis and summarization, which this proposed approach was applied to. The tasks were carried out on documents that are commonly used as text-mining benchmarks and for text-mining competitions (http://duc.nist.gov/). Also, for these documents a ground truth is already available and commonly accepted, which simplifies the presentation of the proposed approach.

3 Opinion Analysis

Besides the factual aspects of a text the expressed opinions, pragmatics and even the style can be important for a proper text understanding. In this section we concentrate on the analysis of a text with respect to the opinions that are expressed. Two examples for applications in which the analysis of the expressed opinion is of special interest are:

- A system that supports the analysis of product reviews in online shops (enabling the users to compare several products with respect to the user ratings without needing to read every single review).
- A system that supports a company in searching and analyzing (user generated) web content (like forum post, blogs, and customer reviews) to get informed about the public opinion about their products.

Building such a system involves the following four steps (based on [6]):

1. Identification of features that have been commented on
2. Classification of the comments into one of the three classes "positive", "neutral", or "negative"
3. Postprocessing steps such as detecting feature names that are used synonymously
4. Presentation of the results (visual and/or by summarization)

The approach that is presented in this section aims at providing a methodology to evaluate the features that are needed for the classification stage (step 2) of the process.
In literature two fundamentally different approaches for building the sentence classification model can be found:

- the supervised learning approach (input = preclassified sentences as training data)
- the lexical approach (input = a list of positive / negative terms)

3.1 Visual Evaluation Approach

In the following we are going to use reviews of amazon.com on a digital camera to show how our evaluation technique works. An annotated version of this data set can be found under [1]. We manually rated each sentence with respect to the expressed attitude towards the camera (positive, neutral or negative) to create a benchmark data set. The feature that we use for the automatic rating follows the lexical approach.[1] We took the necessary list of opinion words from the General Inquirer Project (as provided by [2]) and slightly adapted them. With the help of those lists, each word can be classified into positive, neutral (not in list) or negative. To get values on sentence level, for each sentence we subtract the number of negative words from the number of positive words (e.g.: If there are 2 positive words and 3 negative words in the sentence, the value for the whole sentence would be -1). In the rest of this section we use our proposed technique to evaluate this feature, with respect to its power, to assign the right class to each sentence. We show how correlations to other features can be uncovered. Through visualization, improvement can be achieved by taking into account negation and nouns as opinion words, as well.

Figure 2 shows our benchmark data set and the result of the automatic classification. Each squared pixel represents one sentence and color is mapped to the assigned class (green = positive, white = neutral, and red = negative). The color gradations in figure 2 can be interpreted as how sure the algorithm is about its rating. The pixels are grouped into the three classes. Ideally, a perfect feature would only have green pixels in the first group, white ones in the second group and red ones in the last group. Such grouping allows us to analyze whether a feature has particular problems with one of the classes. In the example in figure 2 we can see that the class of positive statements is the easiest one for the algorithm whereas there are more errors in the sections of the neutral and negative statements. Within each class the pixels are sorted by the length of the sentences. As can be seen by the decreasing confidence of the feature for classifying positive or negative sentences, there is a weak correlation between this property and the feature value. As opposed to neutral sentences, in which this correlation is negative. In case of the positive and negative statements this means that the algorithm is more confident about its decision (more correct ones and higher (darker) values), whereas in the section of the neutral statements the length of the sentences is negatively correlated to the classification accuracy. This can be explained by the fact that the probability for class opinion words is higher in longer sentences. Depending on the application scenario and on the correlation the user is interested in, other measures than sentence length can be used to sort the pixels.

To get hints for further improvements we pointedly analyzed the sentences that were wrongly classified by the algorithm. In figure 2 some pixels have been annotated with the underlying text. In the annotation words that appear in the list of positive opinion words are colored in green and the negative ones are colored in red to enable an understanding of the decision of the algorithm and reveal the problems. While analyzing the results, it can be seen that there are some systematic errors. Reason for this is that negation is not taken into account, and nouns are not included in the list of opinion words. To improve the feature, two extensions were evaluated: First, negation is taken into account by inverting the value of a word if one of the X preceding words is a negation signal word (such as "no", "not, "without" ...). We set the parameter X (the

[1] Please note that our presented evaluation technique would work equally well with the supervised learning algorithm.

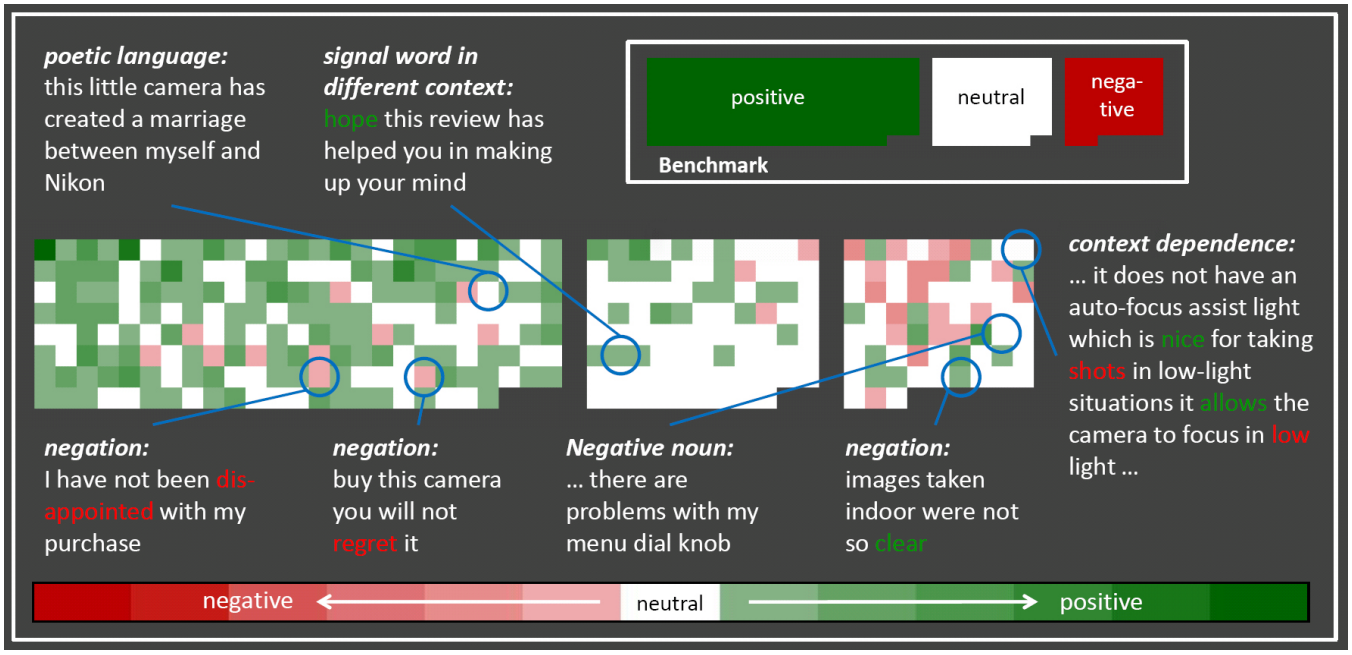

Figure 2: Visualization of product reviews for a digital camera. Each pixel represents one sentence. The sentences are grouped into positive, neutral, and negative statements (left, middle, right as shown in the benchmark visualization above). The sentences are sorted by their length allowing to analyze whether the classification accuracy correlates with this property. Color is mapped to the classification result of the algorithm. The visualization has been annotated with comments on some of the wrongly classified statements. The feature evaluated here is based on lists of positive and negative words. In the annotation words that appear in the list of positive opinion words are colored in green and the negative ones are colored in red to enable an understanding of the decision of the algorithm and reveal the problems.

Figure 3: Visualization of the changes that occur when the feature is extended by adding nouns to the list of opinion words, taking negation into account, or by using both extensions at once. It can easily be seen that the class of negative statements profits most from the changes, but that a decrease in the classification accuracy of the class of neutral statements has to be accepted.

maximum distance to the negation signal word) experimentally to 3 minimizing the failures. Second, we added nouns with negative / positive connotations (such as "problem", "error", "advantage") to our list of opinion words.

In the following we are going to evaluate whether and how those

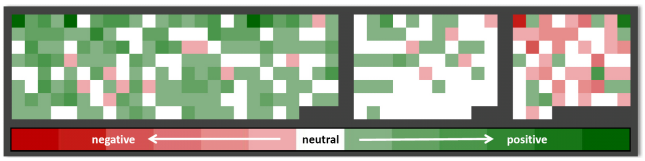

Figure 4: Visualization of the classification results when the feature has been extended by taking negation and nouns into account. In comparison to figure 2, it can be seen that especially the class of negative statements profited from the changes.

extensions result in an improvement of the classification of the sentences. Figure 3 visualizes the changes that occur when negation respectively nouns are taken into account. In this visualization all sentences whose values did not change from one version to the next one are colored in white resp. in yellow if their classification is still wrong. For the rest of the pixels we distinguished between minor and major improvements by highlighting them in light green and dark green, respectively. Correspondingly, minor deteriorations are highlighted in light red and major deteriorations in dark red. We speak of a minor improvement if the prediction was moved one step in the correct direction on the scale negative - neutral - positive, but the classification is still wrong (Example: Sentence in benchmark is defined as negative, but in the first version it was wrongly classified as positive. If in the second version the sentence became neutral, this would lead to a minor improvement. If the same sentence was correctly classified as negative in the second version, this would lead to a major improvement). One can easily see in figure 3 that both extensions result in an improvement. Especially the class of the negative statements seems to profit from the enhancements. However, it is also obvious that we introduced some new mistakes.

This is especially true for the class of the neutral statements which did not profit from the enhancements.

Finally, the third visualization in figure 3 shows the changes when both extensions are combined. As can be seen some of the errors that were introduced by one of the extensions could be eliminated by the combination of both. Figure 4 visualizes the classification result when both extensions are used. Compared to figure 2 especially the section with the negative statements has profited from our changes. However, there are still some mistakes in all three classes. Analyzing them again as in figure 2, reveals that there are different kinds of mistakes, some of which could easily be fixed. First of all, we recognized that our list of positive / negative words is not complete, which is always a problem with the lexical approach. Those words could easily be added. Furthermore, the list could be extended by context-dependent opinion words resp. the ones that do not have a positive or negative connotation in our context could be removed (like "shoot"). Other mistakes would require the usage of advanced natural language processing (NLP) algorithms, e.g. to detect change in context. Even more difficult are the cases in which the text is written in slang or in which no opinion words are used at all and knowledge about the context is required to interpret the sentence correctly (like *"You got to have flash on to get it eventhough your room is well lit"*).

4 Document Summarization

In this section, we concentrate on the evaluation of automatic summarization methods for the creation of extracts from single documents. An extract in this sense uses portions of the input text as its summary, instead of generating a semantic abstract of the text. There are several methods to weight sentences for their level of importance. The following section will first describe related approaches and features commonly used to generate extracts. Consecutively, the proposed visual evaluation approach will be introduced and demonstrated with some examples. Finally the proposed approach will be evaluated and conclusions drawn from the examples.

4.1 Related Approaches

Among the manifold features for text summarization are the following four sentence scores that are commonly used and will be visually evaluated later in this section:

- Length (based on [22]):

$$Length(sentence) = n$$

  where n = the number of words in a sentence (assuming that longer sentences are more important than shorter ones)

- Location (based on [22]):

$$Location(sentence) = \frac{1}{s_i}$$

  where s_i = position of the sentence in the document (assuming that sentences that are at the beginning of the text are more important than later ones)

- Luhn (based on [20]):

$$Luhn(sentence) = \frac{|\{w_i | freq(w_i) > 0\}|^2}{max\{W_i\} - min\{W_i\} + 1}$$

  where w_i = a word,
  $freq(w_i)$ = the term frequency of word w_i in the document,
  stopwords (and sometimes also words with a very low TF value) are set to 0,
  $W_i = \{pos(w_i) | freq(w_i) > 0\}$ with
  $pos(w_i)$ = the position of the word in the sentence
  See figure 5 for an illustration of the formula.

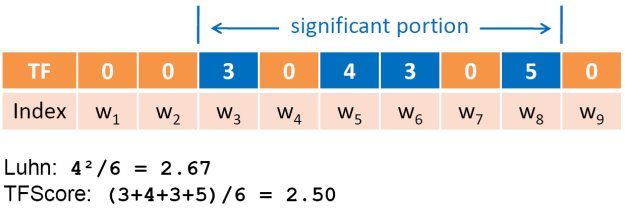

Figure 5: Illustration of Luhn's measure and TFScore. Each column in this graphic represents one word of a sentence. Both measures are based on the term frequencies of the words (TF). Stop words are assigned a term frequency of 0. The section between the first significant word (TF > 0) and the last significant word is called the significant portion of the sentence. Whereas the TFScore calculates the average term frequency in the significant portion, Luhn's measure divides the squared number of significant words by the length of the significant portion.

- TFScore (extension of [20]):

$$TFScore(sentence) = \frac{\sum_{w_i=0}^{n} freq(w_i)}{max\{W_i\} - min\{W_i\} + 1}$$

  where w_i = a word,
  n = the number of words in the sentence,
  $freq(w_i)$ = the term frequency of word w_i in the document,
  stopwords (and sometimes also words with a very low TF value) are set to 0,
  $W_i = \{pos(w_i) | freq(w_i) > 0\}$ with
  $pos(w_i)$ = the position of the word in the sentence
  See figure 5 for an illustration of the formula.

Those measures are used to assign importance weights to sentences and the extract is composed from the sentences whose weight is above a certain threshold. Appropriately defining this threshold is one of the critical parts in the process. Often simply the length of the expected extract is specified and the threshold is adjusted in a way that the desired number of sentences is returned. However, this means that the continuous (and therefore fuzzy) nature of weighting sentences is not taken into account. Consequently, sentences above a certain threshold are equally important, as are all the sentences below the threshold equally unimportant. Some statistical methods compute the utility of all possible thresholds and select the threshold for the highest utility achieved. Also, this method is sometimes extended by fuzzy computational theories that take the continuous nature of the weights into account [23]. Alternatively, some methods use empirical evaluations for determining the optimal length and quality of the generated extract. These approaches create different extracts of the same text and conduct user-studies, where the correlation between the human and automatically generated extracts are computed. These studies require huge efforts to conduct and create a measurable set of useful results. At this point it is necessary to state that no one of the known methods for text summarization is able to achieve perfect results. This is due to the difficulty to overcome individual differences in interest and the lack of appropriate methods to describe semantic text features with quantitative measures.

The most popular method to create a utility measure for the generated abstracts is computing the F-measure based on precision (p) and recall (r) as:

$$F_{(r,p)} = \frac{2rp}{r+p}$$

Consequently, the computed F-measure indicates the balance between sentences retrieved that are indeed relevant and relevant sentences that were successfully retrieved. The F-measure is optimal,

when misses (false-negative) and false alarms (false-positive) values are equally low, since precision and recall are computed as:

$$Recall = \frac{Hit}{Hit+Miss}, \quad Precision = \frac{Hit}{Hit+False\ Alarm}.$$

For more information about text summarization methods and their evaluation refer to [13] or [21].

4.2 Visual Evaluation Approach

The visual evaluation of document summarization features, as proposed in this paper, aims to aid readers to choose the best extract possible generated by different algorithms and thresholds for importance. In usual evaluation approaches, two types of sources are available. First, the original document that is weighted by an automatic feature extraction method (such as Luhn, or TF-Score), and consequently represents the feature values for all sentences, which determines its importance level for the final extract. Second, a benchmark-source that maps the expected extract to the sentences in the original document, representing the ground truth for the evaluation. Often, the benchmark source contains only binary values, 1 for important sentences that are included in the extract, and 0 for unimportant sentences that are not included in the extract.
The proposed approach is extended by a third source, which contains the deviation of the feature values from the benchmark file, aiming to exploit all the advantages of visualization. The visualization technique used in this section is also based on the fingerprinting technique described in section 1.2 and 2. The text used to describe the current approach is taken from ([3]) and describes a report to the US elections. Figure 6 represents such a visualization. First, the features values are represented as computed by Luhn's measure on sentence level. The continuous weights delivered by the feature are mapped by the gradation of the color blue. Higher values (darker blue representations) indicate higher importance and ideally these sentences should be included in the final extract. The second representation shows the benchmark file with binary values. Sentences that are marked in blue represent the expected extract for the current text. The third source is represented as a 'delta-view', which aims to show the features deviation from the benchmark for a given threshold of importance weights. A confusion matrix was calculated, having hits, false alarms, correct rejections, and misses as single values. For this representation, the threshold was set by the number of sentences expected in the extract. The colors in the delta-view represent the four options of a confusion-matrix (Hit (blue), Correct Rejection (white), False Alarm (orange) and Miss (red)). This color coding is chosen to show that the consequences of a Miss are far worth than of a False Alarm, and should therefore be visually more salient.

As for now, the Luhn's measure does not deliver results close to the expected outcome. Figure 7 shows that also the other methods, TF-Score, Length and Location, are unable to outperform Luhn's measure for summarizing the given document. TFScore shows the worst performance with zero hits for the given threshold. However, as far as the visual interpretation of the delta-view allows, a combination of Length and Luhn's measure should lead to a better performance. In addition, a step-wise decrease of the threshold (which results in an increase of the number of sentences in the extract) should allow more correct answers and not necessarily raise the number of misses and false alarms. The following figure (Figure 8) aims to show the outcome of this dynamic process. The columns of the matrix represent the Luhn's measure (first, most left column), the Length-measure (second, middle column) and the combined measure (third, most right column), whereas the combination was created by averaging the two measures. The rows of the matrix represent the step-wise decrease of the threshold value to determine importance level of the sentences. The first row shows a threshold set by the number of expected sentences indicated in

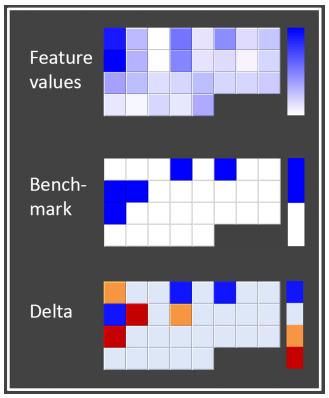

Figure 6: Visual evaluation of features for document summarization. The first row shows the feature values on sentence level as computed by Luhn's measure. Color is mapped to the importance of the sentences. The second row represents the benchmark and the third row the 'delta-view', which aims to show the features deviation from the benchmark for a given threshold of importance weights. The colors in the delta-view represent the four options of a confusion-matrix (Hit (blue), Correct Rejection (white), False Alarm (orange) and Miss (red)).

the benchmark. The second row shows the result when one sentence more is selected, the third row when two sentences more are selected. As shown in the representation, a simple combination of the two measure alone, and a simple decrease of the threshold value (without a combination) leads only to minor improvements. However, the combination together with a decrease of the threshold value leads to a significantly better result. In this representation, all expected sentences were correctly identified, and there are no missed sentences. Even though, two sentences were falsely identified as important sentences, the overall results show a clear improvement (as shown in the third column, third row of the matrix).

The interactive process of creating such an improved constellation is based on the visual evaluation of the interim stages. First, a combination was computed, and then the step-wise decrease of the threshold-value was applied to the delta-view. This was made possible by the visualization-technique, that allowed a fast and appropriate representation of the results and quick access to compare different version of the threshold setting. In addition, the iterative nature of creating such visualizations and evaluating interim results, allows to systematically approach the analysis of the current document. In order to conduct a more comprehensive evaluation of the described approach, further evaluation methods need to be carried out. To show that a strong correlation between visual representation and statistical evaluation exists, we computed the F-measure for all delta-results. A tabular representation of the F-Measure charts is shown in Figure 9. The utility of the feature, as indicated by the F-measure, is mapped as a function of all possible thresholds. The thresholds, drawn on the x-axis, range from 0 to 1, as does the utility measure, drawn on the y-axis. The area under the function indicates the overall utility, and its highest peak shows the optimal threshold of a feature. As such, analysts may benefit from this information in addition to the visualization, as discussed before. Through this graphic representation of the features' utility, analysts are able to select for each feature its optimal threshold. This information is supplementary to the fingerprint visualization. It mainly addresses an additional point of view for the evaluation. The resulting combination, even though leading to an improved performance of the feature, is only an attempt to introduce the methodological approach to improve summarization results. The evaluation of this finding requires further analysis and empirical evaluation on more datasets.

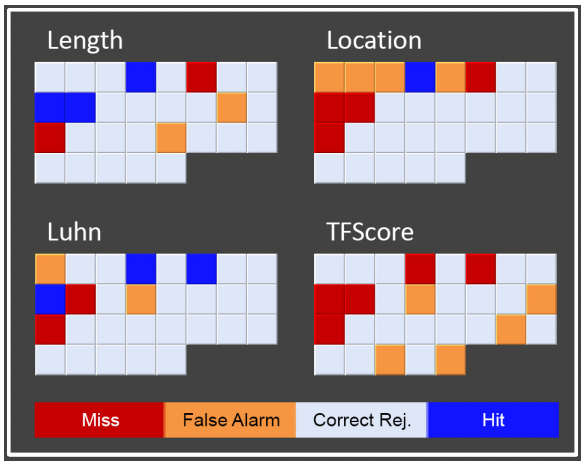

Figure 7: In the delta-view different measures can be compared. It is immediately obvious that none of the methods is able to deliver results close to the expected outcome, but that the result of Length and Luhn is better than the one of Location and TFScore.

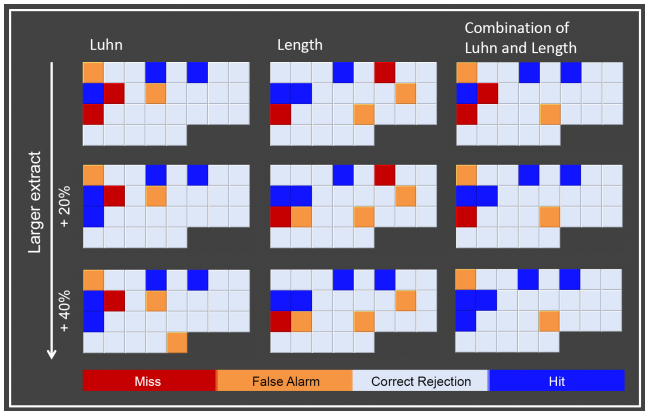

Figure 8: Visual evaluation of the combination of the two measures Luhn and Length and the step-wise increase of the number of sentences in the extract. With 40% more sentences in the extract the combined feature is able to identify all expected sentences correctly with only two false alarms.

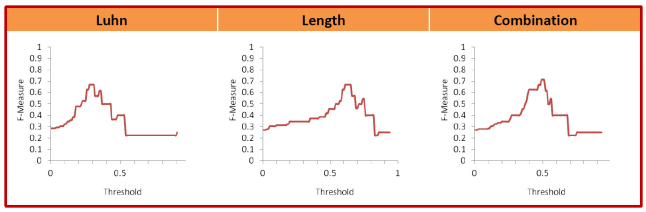

Figure 9: Utility of the applied features indicated by the F-Measure. The utility of the feature for a defined threshold is supplementary to the above visualization.

This however, is out of scope of the current work.

5 Conclusions

The field of visual analytics provides many tools and methods for visualizing documents and evaluating analytic features. The current approach attempted not to develop new text features or text visualization techniques and evaluate their quality. Rather, it aimed at establishing a methodological approach for the visual evaluation of text features for document summarization and analysis. The approach is based on interactive visualization of interim results generated during the analytic process, and consequently allows analysts to iteratively improve their final results. The approach was successfully carried out in the domains of opinion analysis and document summarization. Also, the applied visualization technique of document fingerprinting proved itself beneficial for the evaluation process.

In the domain of opinion analysis, the major task is to analyze documents, such as product reviews, for positive and negative statements. Through the applied visualization technique and through several iterations to alter properties of the feature, results of the analysis could significantly be improved. Analysts are able to use the visualization to detect failures, extend the feature's vocabulary by additional opinion words, and add additional methods, such as negation, in order to extract hidden information from the document.

Summarization is a major challenge in document analysis. Current extraction methods that aim to conduct such a task must be able to describe documents with statistical methods to capture the most important sentences in the underlying text. Research is still far away from finding such a suitable feature. The current approach attempted to find possible improvements in the functioning of the features. Through the developed approach, analysts could detect possibilities to combine features and optimize their threshold to enhance the quality of the resulting summary. Traditional statistical evaluation methods, such as F-measure, supplement the fingerprint visualization. Results gained through the visual evaluation methods undoubtedly require empirical evaluation and comparison on larger benchmark data sets (such as DUC [3]). The approach applied in this paper mainly aims to lay the foundation for the evaluation of features for summarization, rather than developing new features.

Future research must consider extending the introduced approach with more interactive elements, such as zooming and hierarchical structuring of documents. In addition, further techniques for visualization and their applicability to specific domains should be assessed. Since in some domains no benchmark data is available also techniques for unsupervised evaluation should be investigated.

Acknowledgement

This work has been partly funded by the German Research Society (DFG) under the grant GK-1042, Explorative Analysis and Visualization of Large Information Spaces, Konstanz.

References

[1] Customer review datasets. http://www.cs.uic.edu/ liub/FBS/sentiment-analysis.html, (accessed in March 08).

[2] Internet general inquirer. http://www.webuse.umd.edu:9090/, (accessed in January 08).

[3] Document understanding conference (duc), http://duc.nist.gov/, 2002.

[4] A. Abbasi and H. Chen. Categorization and analysis of text in computer mediated communication archives using visualization. In *JCDL '07: Proceedings of the 2007 conference on Digital libraries*, pages 11–18, New York, NY, USA, 2007. ACM.

[5] T. Ball and S. G. Eick. Software visualization in the large. *IEEE Computer*, 29(4):33–43, 1996.

[6] X. Ding, B. Liu, and P. S. Yu. A holistic lexicon-based approach to opinion mining. In *WSDM '08: Proceedings of the international conference on Web search and web data mining*, pages 231–240. ACM, 2008.

[7] A. Don, E. Zheleva, M. Gregory, S. Tarkan, L. Auvil, T. Clement, B. Shneiderman, and C. Plaisant. Discovering interesting usage patterns in text collections: integrating text mining with visualization. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 213–222, New York, NY, USA, 2007. ACM.

[8] J.-D. Fekete and N. Dufournaud. Compus: visualization and analysis of structured documents for understanding social life in the 16th century. In *DL '00: Proceedings of the fifth ACM conference on Digital libraries*, pages 47–55, New York, NY, USA, 2000. ACM.

[9] R. Feldman and J. Sanger. *The Text Mining Handbook*. Cambridge University Press, 2007.

[10] B. Fortuna, D. Mladenic, and M. Grobelnik. Visualization of text document corpus. *Informatica Journal*, 29(4):497–502, 2005.

[11] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. On clustering validation techniques. *Journal of Intelligent Information Systems*, 17(2-3):107–145, 2001.

[12] M. A. Hearst. Tilebars: Visualization of term distribution information in full text information access. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI'95*, 1995.

[13] E. Hovy. *Text Summarization*, pages 583–598. The Oxford Handbook of Computational Linguistics. Oxford University Press, 2005.

[14] D. A. Keim. Information visualization and visual data mining. *Visualization and Computer Graphics, IEEE Transactions on*, 8(1):1–8, 2002.

[15] D. A. Keim, F. Mansmann, and T. Schreck. Mailsom -visual exploration of electronic mail archives. In *Second Conference on Email and Anti-Spam (CEAS 2005)*. 2005.

[16] D. A. Keim and D. Oelke. Literature fingerprinting: A new method for visual literary analysis. In *EEE Symposium on Visual Analytics and Technology (VAST 2007)*, pages 115–122, 2007.

[17] R. R. Korfhage. To see, or not to see— is that the query? In *SIGIR '91: Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 134–141. ACM Press, 1991.

[18] K. Lagus, T. Honkela, S. Kaski, and T. Kohonen. Self-organizing maps of document collections: A new approach to interactive exploration. In E. Simoudis, J. Han, and U. Fayyad, editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 238–243. AAAI Press, 1996.

[19] H. Liu and H. Motoda. *Feature Selection for Knowledge Discovery and Data Mining*. Kluwer, 1998.

[20] H. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165 and 317, 1958.

[21] I. Mani. *Automatic Summarization*. John Benjamins Publishing Company, 2001.

[22] C. Nobata, S. Sekine, M. Murata, K. Uchimoto, M. Utiyama, and H. Isahara. Sentence extraction system assembling multiple evidence, 2001.

[23] R. Parasuraman. Fuzzy signal detection theory: Basic postulates and formulas for analyzing human and machine performance. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 42:636–659(24), 2000.

[24] T. Schreck, J. Schneidewind, and D. A. Keim. An image-based approach to visual feature space analysis. In *to appear in: Proceedings of 16. Int. Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG'2008)*, 2008.

[25] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *IEEE Visual Languages*, number UMCP-CSD CS-TR-3665, pages 336–343, 1996.

[26] A. Spoerri. Infocrystal: a visual tool for information retrieval & management. In *CIKM '93: Proceedings of the second international conference on Information and knowledge management*, pages 11–20. ACM, 1993.

[27] J. A. Wise, J. J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow. Visualizing the non-visual: spatial analysis and interaction with information from text documents. In *INFOVIS '95: Proceedings of the 1995 IEEE Symposium on Information Visualization*, pages 51–58, 1995.