Task Oriented Web Page Segmentation

Judith Jeyafreeda Andrew

Normandie Université

THÈSE

Pour obtenir le diplôme de doctorat

Spécialité INFORMATIQUE

Préparée au sein de l'Université de Caen Normandie

Task Oriented Web Page Segmentation

Présentée et soutenue par
JUDITH JEYAFREEDA ANDREW

Thèse soutenue le 15/12/2020 devant le jury composé de		
M. ANTOINE DOUCET	Professeur des universités, Université de La Rochelle	Rapporteur du jury
M. STÉPHANE GANÇARSKI	Maître de conférences HDR, Sorbonne Université	Rapporteur du jury
M. FABRICE MAUREL	Maître de conférences, Université Caen Normandie	Membre du jury
M. JOSÉ MORENO	Maître de conférences, Université Toulouse 3 Paul Sabatier	Membre du jury
M. EMMANUEL MORIN	Professeur des universités, Université de Nantes	Membre du jury
MME SRIPARNA SAHA	Professeur des universités, India Institute of Technology Patna	Membre du jury
M. GAEL DIAS	Professeur des universités, Université Caen Normandie	Président du jury

Thèse dirigée par STEPHANE FERRARI, Groupe de recherche en informatique, image, automatique et instrumentation

UNIVERSITÉ CAEN NORMANDIE

MIIS

GREYC
Laboratoire de recherche en sciences du numérique

Résumé en français

Avec le développement régulier de l'internet, l'accessibilité des sites web à tous est essentielle mais l'accessibilité des pages web pour les personnes malvoyantes est un défi en soi. En général, une personne voyante utilise des stratégies de lecture complexes et non linéaires. Elles sont basées en partie sur des processus cognitifs tels que le skimming (l'écrémage), qui consiste à obtenir une vue d'ensemble, et le scanning (le balayage), qui consiste à passer d'un point d'intérêt à un autre. Les processus de skimming et de scanning s'appuient sur plusieurs facteurs tels que la disposition, la structure logique et les effets typographiques qui sont disponibles dans l'environnement visuel. Cependant, ces caractéristiques ne sont pas disponibles dans l'environnement non visuel, ce qui rend le skimming et le scanning particulièrement difficiles. Le travail présenté dans cette thèse se concentre sur la segmentation des pages web pour permettre les tâches de skimming et de scanning non visuels. Le cadre applicatif de TAG THUNDER est utilisé à des fins d'expérimentation.

Dans cette thèse, une technique de clustering est choisie pour la segmentation, en vue de satisfaire les critères imposés par la tâche. À cette fin, la page web est considérée comme un Document Object Model (DOM) et les plus petits blocs visuels de l'arbre DOM sont pris comme points d'entrée pour les algorithmes de clustering. Afin de satisfaire les différents critères spécifiques à la tâche, plusieurs caractéristiques ont été introduites pour le processus de clustering. Ces caractéristiques tentent de rendre compte de l'aspect visuel et de l'aspect logique de la page web.

La technique bien établie de clustering Kmeans a été choisie pour expérimenter plusieurs adaptations guidées par la tâche. Une première variante de l'algorithme de Kmeans a été proposée, appelée F-Kmeans, qui utilise la métaphore de la force physique d'attraction des corps massifs. Cette métaphore permet aux petits éléments d'être attirés par les éléments plus grands pour constituer les clusters. Cependant, la mesure de la force d'attraction a ses propres inconvénients en raison du positionnement initial des graines qui pourrait faire que les graines soient placées sur de petits éléments qui n'en attirent pas d'autres.

C'est pourquoi nous avons proposé une nouvelle technique de regroupement guidée par la tâche, intitulée Guided Expansion (GE). Cette technique est une sorte d'expansion hiérarchique où l'expansion de chaque zone (cluster) se fonde sur des décisions locales, contrairement à la méthode Kmeans. GE utilise en particulier une distance entre éléments. Une variante exploitant la mesure de force d'attraction a aussi été testée (F-Guided Expansion).

Ces algorithmes initiaux ont été testés et comparés en utilisant des graines (seeds) placées identiquement sur la diagonale de la page web. Cependant, après les premières expériences, il a été constaté que le positionnement de ces graines initiales joue un rôle très important dans l'expansion d'une zone, indépendamment des algorithmes. C'est pourquoi plusieurs manières de positionner les graines initiales ont été expérimentées par la suite. Pour commencer,

les stratégies de lecture utilisées sur le web ont été exploitées pour placer les graines. Les stratégies de lecture "F" et "Z" sont connues pour être plus courantes lors de la lecture d'une page web (parmi d'autres stratégies). Ensuite, nous avons testé une méthode de pré-clustering pour identifier des clusters probables d'éléments afin de positionner les graines initiales. Nous avons utilisé en particulier la technique de clustering QT pour identifier les groupes probables d'éléments. Deux variantes de cette technique ont été expérimentées. Dans l'une, les clusters formés à partir de la technique QT sont utilisés comme graines (ou clusters initiaux), et dans l'autre, ce sont leurs centroïdes et non les clusters eux-mêmes qui constituent les graines. Ces variantes ont été utilisées avec l'algorithme GE.

Pour nos expérimentations, les algorithmes des différentes méthodes sont testés sur 900 pages web appartenant à trois catégories différentes : 300 pages de tourisme, 300 pages de commerce électronique et 300 pages d'actualités. Deux formes d'évaluation sont effectuées - manuelles et automatiques. Pour les évaluations manuelles, deux expérimentations ont été réalisées. Sur 50 pages web extraites du corpus d'expérimentation (20 pages web du tourisme, 12 pages web du commerce électronique et 18 pages web des actualités) des experts connaissant la tâche à accomplir ont procédé à une annotation manuelle. Les algorithmes Kmeans, F-Kmeans et GE avec le positionnement initial des graines en diagonale ont alors été manuellement comparés à cette annotation en terme de "compactness" (compacité) et de "separateness" (séparation). Pour une deuxième évaluation manuelle, les mêmes 50 pages web sont cette fois utilisées pour calculer automatiquement différente mesures classiquement exploitées en clustering : le B3F1-score, la précision, le rappel, l'ARI, la Jaccard et le FM-score. Ceci a été fait pour tous les algorithmes avec les différents positionnements des graines. Comme une annotation manuelle est nécessaire pour ce type d'évaluation, des mesures entièrement automatiques ont été développées pour permettre d'évaluer sur un plus grand nombre de pages web. Elles se fondent sur le retour d'expérience des experts, exploitant différents aspects qu'ils ont jugés jugés importants lors de leurs évaluations initiales de la "compactness" et de la "separateness" des algorithmes. Elles permettent donc d'évaluer un grand nombre de pages web sans plus avoir à comparer à un référentiel (ground truth) qui aurait été établi manuellement. En pratique, ces métriques cherchent à refléter le nombre de coupures indésirables (cuts), l'équilibre entre les différentes zones (balance) ainsi que les intersections entre ce que nous pourrions qualifier de rectangles exinscrits aux différentes zones (exterior rectangles). L'évaluation avec ces mesures automatiques montre que l'algorithme GE avec un positionnement diagonal des graines et ce même algorithme avec un positionnement des graines à l'aide des centroïdes issus du pré-clustering QT produit les meilleurs résultats.

Les différentes approches proposées dans cette thèse ont également été comparés à des travaux déjà existants tels que Block-O-matic, Box Clustering Segmentation algorithm et un travail à paraître sur un algorithme de Mutiobjective Clustering Segmentation (MCS). À cette fin, le nombre de clusters a été varié entre 3 et 8 afin de pouvoir effectuer une comparaison équitable. Les algorithmes proposés dans cette thèse se sont avérés plus efficaces que Block-

O-matic et Box Clustering Segmentation. Leur performance est comparable à celle de l'algorithme MCS. La thèse se termine donc par une description d'un travail en cours et plusieurs idées pour des travaux futurs.

Structure de la thèse Le document est composée de 3 parties. La partie 1, état de l'art, donne une vue détaillée des algorithmes déjà existants pour la segmentation de pages Web. La partie 2 explique les choix faits pour le processus de segmentation ainsi que pour les propriétés qui seront exploitées par le processus de clustering. Cette partie présente ensuite les algorithmes que nous avons développés, puis les différentes manières de positionner les germes, les graines initiales pour le processus de clustering. La partie 3 présente l'évaluation des algorithmes, tant manuelle qu'automatique. Elle inclut une évaluation permettant de comparer nos algorithmes avec certaines méthodes existantes. La thèse se conclut avec quelques lignes directrices pour de futurs travaux.

Abstract

With the regular development of the internet, the accessibility of web sites to every one is essential but accessibility of web pages for the visually disabled people is a challenge in itself. In general, a person with sight uses a complex and non-linear reading strategy. They are based in part on cognitive processes such as skimming which is to get a global overview, and scanning which is to jump from one area of interest to another. The skimming and scanning processes are based on several factors like layout, logical structure and typographic effects which are available in the visual environment. However, these features are not available in the non visual environment thus making skimming and scanning a rather difficult task. The work presented in this dissertation focuses on the segmentation of web pages for the task of non visual skimming and scanning. For the purpose of experimentation the framework of TAG THUNDER is used.

In this dissertation, a clustering technique for the purpose of segmentation is employed allowing to satisfy the task oriented criteria. For this purpose, the web page is considered as a Document Object Model (DOM) and the smallest visual blocks of the DOM tree are taken as data points for the clustering algorithms. In order to satisfy the various criteria specific to the task in hand, several features have been introduced with the clustering process. The features comply with the visual and logical aspect of the web page.

The very well established Kmeans clustering technique has been used for experimentation with task oriented adaptations. A variation of the Kmeans algorithm has been proposed called F-Kmeans which uses the metaphor of the physical force of attraction. This metaphor allows the small web elements to be absorbed by the bigger web elements. However, the force measure has its own disadvantages because of the initial positioning of seeds which could cause seeds to be placed on small web elements that do not attract other web elements.

Therefore, in this dissertation, a task-oriented clustering technique known as Guided Expansion(GE) has been developed. This clustering technique follows a sort of hierarchical expansion using the features and expansion of the zones based on local decisions unlike Kmeans. As a variation of GE the force measure as the distance measure known as F-Guided Expansion

The initial algorithms are tested using the seeds placed in a diagonal fashion along the web page. However, after initial experiments, it has been found that the positioning of initial seeds plays a very important role in the expansion of a zone irrespective of the algorithms. Thus several ways for positioning initial seeds are experimented. For starters, the reading strategies used on the web is used to place seeds. The “F” and “Z” reading strategies are known to be more common while reading a web page among other strategies. These are used to position the initial seeds. Following this, a pre clustering method to identify probable clusters of web elements that can be used to position the initial seeds is proposed. In particular, the QT clustering technique that helps in identifying the probable clusters of web elements is studied. In this

dissertation, two variations of this technique in particular are experimented. One, where the clusters formed from the QT technique is used as seeds and the other where the centroid of the clusters formed from the QT technique is used as seeds. These different ways of positioning the initial seeds has been used along with the Guided Expansion(GE) algorithm for segmenting a web page.

For the purpose of experimentation, the algorithms with the various positioning methods are tested with 900 web pages belonging to three different categories – 300 web pages from Tourism, 300 web pages from E-commerce and 300 web pages from News. For the purpose of evaluation, a two way evaluation is performed – manual and automatic. The manual evaluation is done in two ways. It is performed on 50 web pages extracted from the experimentation corpus (20 Tourism web pages, 12 E-commerce web pages and 18 News web pages) and manually annotated by experts with the knowledge of the task in hand. The algorithms Kmeans, F-Kmeans and Guided Expansion with the diagonal positioning of seeds are tested with this ground truth for compactness and separateness. A second manual evaluation, the same 50 web pages are evaluated for the cluster metrics such as B3F1 score, Precision, Recall, ARI, Jaccard and F&M score. This is done for all the algorithms with all different positioning of seeds. Since a manual annotation is required for this sort of evaluation, automatic metrics is developed to evaluate larger number of web pages. Thus based on the initial evaluations done by the experts on the compactness and separateness of the algorithms, certain task-oriented metrics for evaluation of large number of web pages are developed. This allows for evaluating a huge number of web pages without a manually evaluated ground truth to be compared with. The metrics thus defined are cuts, balance and Exterior Rectangle. This evaluation proves again that the GE with the diagonally positioning of seeds and the positioning of seeds using the centroid of the clusters formed from the QT clustering technique is best for the task at hand.

The work in this thesis is also compared with already existing works such as Block-O-matic, Box Clustering Segmentation algorithm and a yet to be published work of Mutiobjective Clustering segmentation (MCS) algorithm. For this purpose the number of clusters have been varied between 3 to 8 to be able to make a fair comparison. The algorithms proposed in this thesis have proved to be more efficient than the existing works of Block-O-matic and Box Clustering Segmentation algorithms. They perform equal to the MCS algorithm. The thesis is thus concluded with a description of an ongoing work and several ideas for future works.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors Dr. Stéphane Ferrari and Dr. Fabrice Maurel. It has been a pleasure working with them. I appreciate all their contributions of time and ideas to make my Ph.D. experience productive. I would also like to thank them for their notable contribution to my professional time at Groupe de Recherche en Informatique, Image, Automatique et Instrumentation de Caen (GREYC).

I would like to thank Dr. Gaël Dias, for his expert opinions on Clustering Techniques, his patience, and continuous help and support.

Special thanks goes to the jury and other faculty members of GREYC for their advise and suggestions. I would like to thank GREYC - CNRS UMR 6072 Laboratory for financially supporting my research.

I would also like to thank the members of my comité de suivi, Dr. Xavier Tannier and Dr. Laurence Mechin for their assistance during the thesis.

I would like to thank Mr.Mukul Dhiman for his support during the course of the thesis.

I would also like to thank my parents and my brother. They were always supporting me and encouraging me with their best wishes.

List of Publications

1. J.-J. Andrew, S. Ferrari, F. Maurel, G. Dias, and E. Giguet. Web page segmentation for non visual skimming. In 33rd Pacific Asia Conference on Language, Information and Computation (PACLIC), 2019a. Andrew et al. (2019a) [Chapter 5]
2. J.-J. Andrew, S. Ferrari, F. Maurel, G. Dias, and E. Giguet. Model-driven webpage segmentation for non visual access. In 16th International Conference of the Pacific Association for Computational Linguistics (PACLING), 2019b. Andrew et al. (2019b)[Chapter 6]
3. F. Maurel, G. Dias, S. Ferrari, J.-J. Andrew, and E. Giguet. Concurrent speech synthesis to improve document first glance for the blind. In 2019International Conference on Document Analysis and Recognition Workshops (ICDARW), volume 3, pages 10–17. IEEE, 2019. Maurel et al. (2019) [Chapter 1]

Contents

List of Tables

List of Figures

Chapter 1

Introduction

As the Internet develops, the web related applications have become one of the most significant applications of networks. A web page is very much accessible to everyone. Most users share a similar mental process when accessing informative content of web pages. The reader spots different areas of interest and seeks for specific information in identified areas using a zoom-in zoom-out strategy. Skimming and scanning are two well-known reading processes, which are combined to access the document content as quickly and efficiently as possible. The reader gets a first glance of the page content (skimming), followed by a quick search for specific information (scanning). In this work, scanning is defined as a process of searching for a specific piece of information in a web document, and skimming is defined as the action of quickly passing through a web page to get an overall impression of its content (a.k.a first glance). While skimming can be seen as an easy task in a visual environment, reproducing the document content driven by its structure in a non visual setting (e.g. visually impaired people) is a much harder problem. The skimming and scanning processes are based on several factors like layout, logical structure and typographic effects which are available in the visual environment. However, these are not available in the non visual environment thus making skimming and scanning a difficult, if not an impossible task. Figure 1.1 illustrates the difficulty of considering a web page without the visual modality. The left page allows to collect a large amount of information in a few seconds (general subject, category, central elements vs. peripheral ones ...), the right one does not offer such possibilities, due to the radical modification at the visual structure level. Yet, it is the same page depending on whether it is produced by a visual web browser or intended for the input of a conventional screen reader used by blind people on desktop computers. Thus this work is a part of a project whose goal is to allow skimming and scanning opportunities in a non visual environment.

The presentation of a webpage aims to deliver coherent information to end-users. Most browsers use the way similar to Document Object Model (DOM) to render a webpage. DOM defines the logical structure of documents and the way a document is accessed and manipulated. One important property of DOM structure models is structural isomorphism: if any two Document Object Model implementations are used to create a representation of the same

Figure 1.1: *sighted vs. blind web page perception*

document, they will create the same structure model, with precisely the same objects and relationships. The contents that are logically related or positioned cohesively are therefore grouped into the same DOM element in the source code of a webpage. This kind of correlation and cohesion have been used as an important feature for webpage segmentation. However, the front-end technologies of webpage design have rapidly developed, and new dynamic frameworks have been increasingly introduced. Webpages are now organized in a more flexible and varying manner. This undermines the correlation and cohesion nature of webpage contents, making webpage segmentation a more challenging task to perform in a visual environment and even more challenging in a non visual environment.

In order to allow experimentation on the skimming and scanning in a non visual environment, the TAG THUNDER framework has been proposed. There are various modules in the TAG THUNDER framework with the end task of allowing skimming and scanning for the blind.

1.1 Tag Thunder and Task Constraints

Tag thunder project aims at giving a blind person the advantage of skimming and scanning. Based on Maurel et al. (2019), in this work skimming and scanning are defined as two cognitive micro-processes at the basis of our ability to build efficient high level reading strategies (quickly or diagonally scan a text, evaluate the interest of a document at a glance or efficiently find known information). These two, more or less conscious written text processing abilities can be repeated in different combinations until the individual objectives are achieved. The more successful the task, the more the combination used will

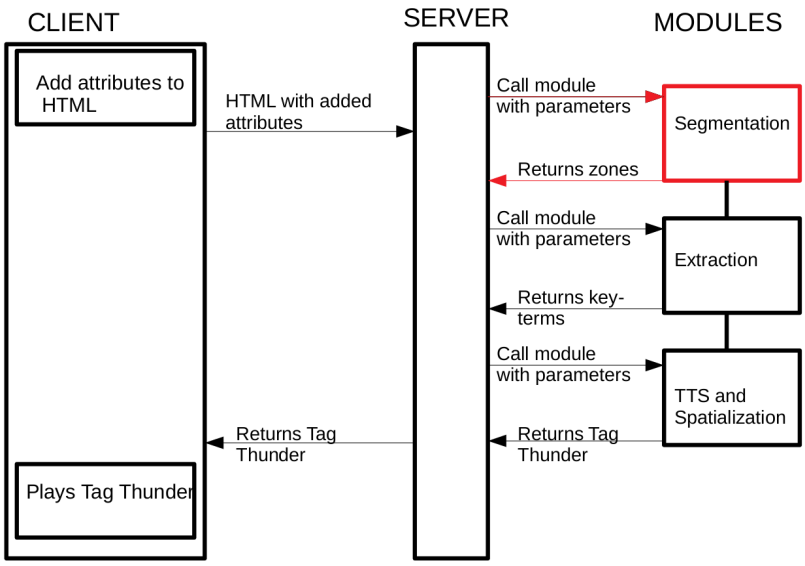

Figure 1.2: *Architecture of Tag Thunder*

be retained as a new reusable strategy. Layout and typography are crucial to the success and effectiveness of these processes. In Tag Thunder, the focus is on the skimming process : oral transposition of web pages visual structure to promote the development of blind browsing strategies based on non-visual skimming process. Lecarpentier et al. (2016) proposes the Tag Thunder idea for making the visual aspects of a web page available for the non-visual environment. The idea behind Tag Thunder is to take a web page as input and identify the representative words of the web page to create a tag cloud. The Tag cloud is then vocalized by taking into account various metaphors to produce a tag cloud, which allows the visually impaired to skim and scan a web page. Figure 1.2 shows the architecture of Tag Thunder. In the current version, the Tag Thunder Project (TTP) is implemented to work as a client-server where each module enriches the original HTML source with specific information or creates new information such as audio files containing the key terms extracted and then vocalized by the TTS (Text to Speech Synthesis)

— **Client Side**: In the client side, the HTML source of a web page is enriched with information about the bounding boxes, styles and xpath. This creates a single file with all the information which are crucial for following modules.
The client side also tags all the elements that do not have any visual effect on the rendered web page. This tagging will greatly facilitate the segmentation process.

— **Segmentation Tool**: The client sends a HTML file with the added attributes to the server. The segmentation module operates on the server side. It aims at producing coherent zones for human perception and cognition. The coherency of the zones can be seen in terms of visual or semantic or structural features.

— **Extraction tool**: The extraction module operates on the server side. This module extracts and weighs the important k-terms or produces zone descriptors, which will be integrated as a tag cloud. One or sev-

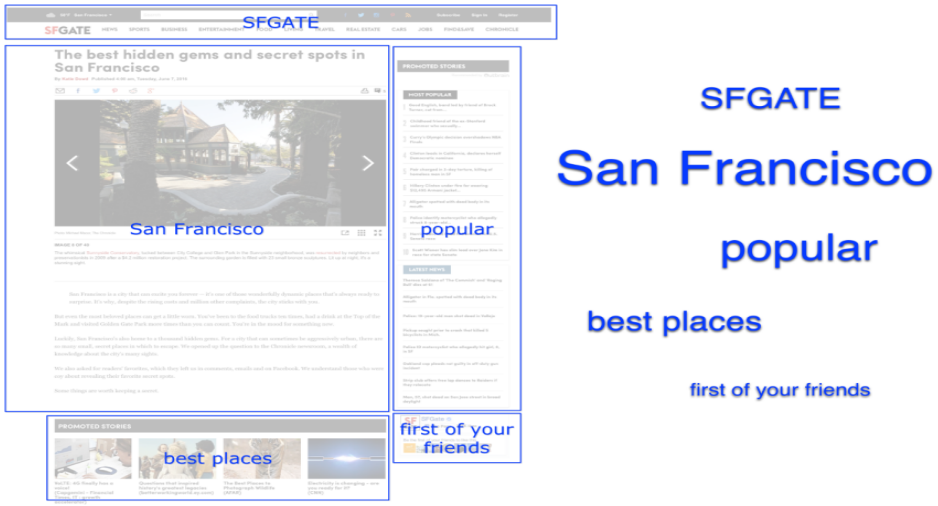

Figure 1.3: *Expected output*

eral k-terms are selected from each zone because each zone has to be represented in the Tag Thunder.

— **TTS and Spatialization**: This module is on the server side. KALI TTS is a tool developed at the University of Caen Normandie by the CRISCO laboratory Morel and Lacheret-Dujour (2001). KALI supports speech rate acceleration without loss in intelligibility and sound quality, which is a very important feature in non-visual web browsing. KALI is used to generate an audio file for each key term, choosing the voice depending cocktail party effect metaphors Lecarpentier et al. (2016). The spatialization tool organizes the spatial and temporal rendering of the audio files, producing what we call a Tag Thunder. This is then returned to the client side which is played by the user as TAG THUNDER.

This work is based on improving the first module on the server side of the Tag Thunder architecture - the segmentation module. The expected output from the segmentation module of the TAG THUNDER framework is presented by the blue boxes on the left of figure 1.3 and the words inside the blue boxes represents the expected output of the extraction module. The right side of figure 1.3, represents the web page after the extraction module of the TAG THUNDER framework generating representative words from each zone of a segmented web page forming a tag cloud (i.e) TTS, which are later vocalized using vocalization tool to fulfill the task of allowing skimming and scanning for the visually impaired.

Three aspects for non visual skimming and scanning: To help visually impaired people to be able to skim through a web page, it is necessary that the logical, visual and semantic aspects of the web page be preserved. The logical structure of the web page is obtained through the DOM structure of the web page. The visual structure is obtained using the css styling of the web page and the semantics of the web page is the textual content of the page. Thus to be able to cluster similar elements together by respecting the three aspects:

— Keeping some HTML elements together such as list items, taking the role of the sectioning elements, thus taking into account the rules proposed by HTML5.
— Elements with same background color, font color, size, weight and type (font similarities), thus following the Gestalt law of similarity.
— Elements which are close to each other, thus following the Gestalt law of proximity.

Thus based on these aspects, there are three criteria that are necessary to be taken into account. They are:

— The number of zones has to be fixed in order to foster the emergence of regularities in the output and to comply with the maximum number of concurrent oral stimuli a human-being can cognitively distinguish. Indeed, we assume that each semantically coherent zone can be summarized and simultaneously synthesized into spatialized concurrent speech acts. This criteria is discussed in detail in part II chapter 4.
— Each zone should be associated to a unique sound source spatially located in accordance with its position in the web page. Thus, each zone should be a single compact block made of contiguous web elements, and the zones should not overlap.
— Segmentation must be complete, which means that no web page element should remain outside a given zone, as the objective is to reveal the overall semantics of a document and not just parts of it

1.2 Structure of the dissertation

This dissertation is divided in 3 parts. Part 1 of the dissertation gives a detailed view of the already existing algorithms for segmentation of web pages. Part 2 of the dissertation explains the choices made for the segmentation process and the features that will be used for the clustering process. This part also explains in detail the algorithms developed, followed by the various ways to position initial seeds. Part 3 presents the evaluation of the algorithms, both manual and automatic. This part also presents the comparison of the algorithms with the existing methods of segmentation. The dissertation concludes with several ideas of future works for the task and the framework of TAG THUNDER PROJECT.

Part I: Part I of the dissertation presents a detailed overview of the existing works on web page segmentation, text segmentation and web accessibility. Firstly, several works concerning the web accessibility for the disabled are discussed. This is followed by a detailed presentation of the works on removing noise (all parts of the web page that does not contain the supposed main content) from a web page to help in the segmentation. Then the techniques available for segmentation by summarization of content and using the annotation mark ups are also considered, discussed and evaluated for the task at hand. Following this, the logical, textual and visual approaches available for the purpose of segmentation. In particular, the well established visual based Page

segmentation Algorithm (VIPS), Block-O-matic and Box Clustering Segmentation algorithms are detailed with both their advantages and disadvantages along with their comparability for the task in hand. The Block-O-matic and the Box Clustering Segmentation algorithms are later used for the evaluation and the comparisons with all the algorithms developed during the course of the dissertation. Finally, hybrid techniques for segmentation by combining different techniques or different features are discussed for their benefits.

Part II: Part II of the dissertation elaborates on the technical choices and the feature choices made for the segmentation of web pages for the task of non visual skimming and scanning. This part also introduces the algorithms developed during the course of the dissertation. Firstly, the well established Kmeans with task-oriented changes is introduced. Followed by E-Kmeans which is a variation of Kmeans using the metaphor of the physical force of attraction. Following this, a task-oriented clustering technique known as Guided Expansion(GE) is introduced. Initial experiments on these algorithms prove that the positioning of initial seeds plays a very important role in the expansion of a zone irrespective of the algorithms. Thus several ways to positioning initial seeds are studied. In particular, positioning of seeds using reading strategies used on the web and positioning of seeds using a pre clustering technique are experimented for the task in hand.

Part III: Part III presents the evaluation of the results from the algorithms. A two-phase evaluation is conducted. The first phase is a manual evaluation, where experts are asked to segment 50 web pages to create a ground truth. They are then asked to use this ground truth to evaluate the results from the algorithms for compactness and separateness. The manual evaluation also evaluates the algorithms for the general cluster metrics (B3F1, Precision, Recall, ARI, Jaccard and F&M index). This is done for the 50 web pages for which the ground truth has been established by the experts. However, in order to be able to evaluate huge number of web pages in less time a method of automatic evaluation is developed. This is phase 2 of evaluation. The metrics for this sort of evaluation has been developed with the help of the experts. Interviews have been conducted with the experts to know how the scoring for compactness and separateness was done. This helped to know when and why an expert penalised the algorithm for not performing well. Based on this interviews, 3 metrics were developed that help with evaluating any algorithm for any number of web pages without the necessity of a ground truth.

This is then followed by comparison of the algorithms developed in the thesis with already existing works (Block-O-matic and Box clustering Segmentation). They are evaluated for the general cluster metrics for 50 web pages whose ground truth were previously established by the experts. It is seen that the proposed algorithms outperforms the existing algorithms for segmentation.

The dissertation is concluded with ideas for future works. Also a short presentation of an ongoing work has been discussed.

Part I

Literature Review

Chapter 2

Zoning for accessibility

The web is designed for all people irrespective of abilities. However, when websites and technologies are designed badly they create a barrier and thus exclude people with disabilities from using them. Accessibility is essential for developers and organizations that want to create high quality websites and web tools, and not exclude people from using their products and services. There have been some research allowing people with disabilities to access the web.

2.1 Web Accessibility for the disabled

Asakawa and Takagi (2000) aims to identify visually fragmented groups of elements in a web page such that the page can be transcoded to better support accessibility for blind users (screen reader users). The authors propose to manually annotate the page to identify the role of fragments in a page. The system consists of three components - a proxy server, an annotation database and an annotation server. The proxy server transcodes a target HTML document. When the transcoding module receives a target HTML document, it sends the URL of the target HTML document to the annotation manager. The annotation manager retrieves an id list of proper annotation files and sends it to the annotation database. Finally, the transcoding module will receive the annotation files which are sent by the annotation database according to the id list, and transcodes the target HTML document using the annotation files.

Takagi et al. (2002) developed a system that has the ability to transcode complete pages on annotated sites into totally accessible pages without changing the original pages. By utilizing this algorithm, the transcoding system can automatically determine appropriate annotations based on each page's layout. They also developed a site-wide annotation-authoring tool, "Site Pattern Analyzer."

The methods described in Asakawa and Takagi (2000) and Takagi et al. (2002) are a sort of classification technique. In these algorithms, the annotation tool consists of annotations belonging to two different types - functional and

commentary. It has a predefined set of classes such as "annotation", group", "role", "alternative" etc (full list in Asakawa and Takagi (2000)), each with its own definition. However, for the task of non visual skimming and scanning, zoning a web page using pre-defined classes/categories is not possible as the goal is not to put every element of the web page into a class/category because firstly, the classes are described with the aim of enabling transcoding a web page for voice output thus it does not give a segmentation for the first glance of the output, secondly, not all web elements can be exactly put into one or the other class, there are possibilities of overlap. Also the process described by Asakawa and Takagi (2000) and Takagi et al. (2002) are fully as it requires volunteers to annotate web pages before transcoding them. This limits the number of pages to the number of pages that can be annotated and thus cannot be used for all the pages. Thus using techniques as suggested in Asakawa and Takagi (2000) and Takagi et al. (2002) are not suitable for the task at hand where the number of zones to be discovered is fixed, all web elements should belong to one and only one zone and the process to be fully automatic so that it can be used for all web pages.

2.2 By noise separation

Yi et al. (2003) proposes the use of a style tree(ST), which consists of 2 types of nodes: style node - representing the layout or presentation style and element node containing the content of the web page. The definition of noise is based on the following assumptions: (1) The more presentation styles that an element node has, the more important it is, and vice versa. (2) The more diverse that the actual contents of an element node are, the more important the element node is, and vice versa. For an element node E in the ST, if all of its descendants and itself have composite importance less than a specified threshold t, then the element node E is noisy. Yi et al. (2003) uses the fact that web pages within the same web site tend have overlapping templates. Thus Yi et al. (2003) uses the styling and templates and styling on a web site to help remove the noise and identify the main content.

Alassi and Alhajj (2013) introduces Noise Detector (ND) as an effective approach for detecting and removing templates from Web pages. ND segments Web pages into semantically coherent blocks. Then it computes content and structure similarities between these blocks; a presentational noise measure is used as well. ND dynamically calculates a threshold for differentiating noisy blocks. Barua et al. (2014) propose a technique that detects noise in news articles. The algorithm is known as StaDyNoT. The authors define static noise content and dynamic noise content. Static noise content is the content which is present in all the article web pages of a same news website, whereas, the dynamic noise contents are advertisements and irrelevant hyperlinks which keep changing from one article web page to another web page. They follow a 2 step process, one to identify static noise using the DOM tree of every neighbor web page and extracting HTML web elements - their attributes and content. These are then put into a global hash table with initial support value 1, if identical

entries are then found, the support value is increased. Once the static noise is identified, the DOM tree of the target webpage is traversed and the nodes are marked as noise if the node matches with any of the identified Static Noise Tag. The second step is to identify dynamic noise. This is done by applying Least Common Ancestor(LCA) on the resulting DOM from stage one. To do this, each node with HTML tag <a> is represented using a path string. After obtaining the set of path-strings for each anchor (<a>)node, least common ancestors of discovered path-strings is identified. These least common ancestors are also nodes in a DOM tree and are marked as a Candidate Dynamic noise Tag. However, an hyperlink node could be found inside text nodes as well and if only the LCA method is used to identify the dynamic noise nodes and thus some heuristics are used to filter them. The experiments on this algorithm is conducted on 440 news article web pages from 11 different web sites. A ground truth is manually created. StaDyNoT performs better than the other similar approaches in terms of F1score, precision and recall. This technique has been developed for the task of extracting news content from online web pages. However, for the task of non visual skimming and scanning this technique is not suitable as it removes certain content from the web page as noise while the task at hands aims at presenting the web page completely to the visually impaired person. Specifically, tourism and e-Commerce web pages have a lot of hyperlinks and advertisements which the user might want to view. Thus this technique is not suitable for our task.

Lin and Ho (2002) aims to identify informative and redundant content blocks, and then aims to make use of the features in the informative content blocks to support information extraction. Lin and Ho (2002) use the fact that a web site usually employs one or several templates to present its Web pages. A page cluster is a set of pages that are presented by the same template. If all pages of a Web site use the same template, the Web site is regarded as one page cluster. Lin and Ho (2002) assume a web site as a page cluster for their approach. The approach generates a coarse tree structure by parsing the HTML page based on <TABLE> tag. Each internal node indicates a content block that consists of one or more content strings (without HTML tags) as its leaf nodes. After parsing a page into content blocks, features of each block are simultaneously extracted. Features indicates meaningful keywords. Thus stop-words are removed. The entropy value of a feature is estimated according to the weight distribution of features appearing in a page cluster. Feature entropies contribute to the semantic measure of a content block that owns these features. I.e. the entropy value of a content block is the summation of its features entropies. Based on the entropy, the content block can be divided into two categories: redundant and informative. If the entropy of the content block is higher than a defined threshold or close to 1, the content block is absolutely redundant since most of the block's features appear in every page. If the entropy of a content block is less than a defined threshold, the content block is informative because features of the page are distinguishable from others. I.e. these features of the page seldom appear in other pages. The threshold is not easy to determine since it would vary for different clusters or sites. If the higher threshold is chosen, the higher recall rate is expected. However, the precision rate may become lower. To get a balanced recall-precision rate, a

greedy approach to dynamically determine the threshold for different training sets (page clusters or sites) is applied. If the threshold is increased, more informative features (in informative content blocks) will also be included. This greedy approach has been tested on news web pages to retrieve informative content and achieves a high precision and recall for identifying informative blocks within the same web site. This approach relies on the assumption that the template of all web pages within a web site is the same, however this is not true for all cases, specifically with tourism web pages, the template within the web site changes frequently to present the information in an attractive manner. Thus this technique is not suitable for all web pages.

Borodin et al. (2007),Mahmud et al. (2007a), Mahmud et al. (2007b) indicates that the applications such as screen readers process a web page sequentially (i.e., they first read through menus, banners, commercials, etc) therefore this makes browsing time-consuming and strenuous for screen reader users. Therefore, with the specialised audio browser called Hearsay or CSurf what they try to do is that when the user clicks on a link they aim to retrieve the target page and point the user to the relevant block to that link. They do this by segmenting a web page into a number of blocks and then identifying the context and the relevant block to that link. i.e. the advertisements, banners and menu are removed as unwanted content, which is not the aim of the task at hand. This approach resembles the "reading mode" on the firefox browser which focuses on the news content on news Websites.

Giraud et al. (2018) describes the web accessibility as a process of filtering irrelevant and redundant information to improve the usability of a website for the users. However, this might not always be the case. There could be useful information for a blind user in the links/advertisements on a web page which are discarded as "noisy" content by the above mentioned methods. On the other hand, the main goal of the task in hand is to allow the user to choose what to see and what not to see. The user might want to know about certain advertisements on a web page. Thus removing these contents as "noise" is not part of the task. Although these works provide good results for the task of identifying the main content of the web page, they do not fit the goal of the task at hand - non-visual skimming and scanning.

2.3 By summarizing (page level) /aggregating (website level)

Chen et al. (2003) aims at creating a better way for easy navigation and browsing in large web pages for the mobile phones. The idea is to provide an overview of the web page and allow the user to select a desired portion of the web page to zoom in for detail reading. The overview is like a Table of Content - provides a thumbnail on which each block of semantically related content is represented with a different color. The whole process is decomposed into two main steps which are page analysis and page splitting. The goal of

page analysis is to extract the semantic structure of an existing web page. This structure is a hierarchical representation of the web page, in which each node is a group of objects in the web page. The goal is to identify a set of nodes in the hierarchy, in which each represents a unit of information that can be managed and displayed individually. At the beginning, the whole web page is regarded as a single content block. At each iteration, the page analysis algorithm finds a best way to partition a content block into smaller ones. A set of content blocks will remain at the end of the process, which serves as the final information for page splitting.

The **page analysis** algorithm consists of the following three steps: first, the HTML DOM tree is analysed and the high-level content blocks about the locations and sizes of header, footer, side bar and body is detected; then the content inside is analyzed at each high-level content block to identify explicit separators to split the content blocks; lastly, implicit separators are detected and used to split the content blocks further. This detects the header, footer, body, the left and the right bar. However further partition is done using explicit and implicit seperators. The explicit seperators are tags such as <HR>, <TABLE>, <TD>,<DIV>and <IMG>. Implicit seperators are blank spaces. The **Page splitting** algorithm deals with deciding which blocks should be put together. This is done by using the CSS associated with the blocks. After this an index page is created with thumbnails and hyperlinks to the sub-pages. The method has been tested on 50 web sites. More than 90 percent was perfect or good. The evaluation in this work has been done by using the algorithm on 50 popular web pages and testers are asked to put them in three categories: perfect, good and error. Chen et al. (2003) has achieved an average of 55% perfect score indicating that the page analysis and splitting is perfect 55% of the times on average on the chosen 50 web pages. This algorithm relies heavily on the HTML DOM tree which causes parsing errors due to the HTML syntax errors left by the author of the web page. The evaluations presented on this algorithm have been conducted on 50 web pages which is not huge enough to make proper conclusion. Ofcourse, manually annotating web pages for ground truth is not an easy task, however, Chen et al. (2003) could have developed an approach to automatically evaluate several web pages and thus enabling better conclusions on the efficiency of the algorithm. Chen et al. (2005) is a continuation of the work in Chen et al. (2003), where the development schemes are described: client-side, proxy side or server side.

Baluja (2006) present an algorithm with the goal of allowing zones that can be zoomed in for small screen devices. Baluja (2006) show that a multi-label classification problem can be addressed through techniques based on entropy reduction and decision tree learning. They consider each DOM element of interest to be a separate class. The goal of the decision tree classifier is then to select splits on the page that help to determine which DOM-element (class) the user is looking at. The probability of a class is defined by the area(in pixels) of the DOM element that it represents. Each node has a X and Y coordinates associated with it which are called the attributes. These attributes are used to determine possible cuts. Once all possible points of cuts are determined, the one with the maximum Information Gain is chosen to be the actual cut. This

process is repeated recursively to segment the web page. The experiments are performed on content heavy and simple web pages. The segmentation is good on web pages that are structured very well but the segmentation is not perfect with content heavy web pages. However, Baluja (2006) does not provide quantitative analysis for the experiments conducted.

Yang and Shi (2007) presents the web page segmentation in terms of representing how humans usually understand web pages. It is based on the Gestalt theory, a psychological theory that can explain human's visual perceptive process. Proximity, similarity, closure and simplicity are used to simulate how humans understand the layout of web pages. Proximity refers to the distance between web elements, Similarity refers to similar web elements based on visual components, closure refers to the web elements that are part of a structure (eg: items of a list) and simplicity refers to simple structures according to symmetry, smoothness and regularity. For testing purposes, twenty web sites are chosen, starting from ones with simple layouts like amazon.com and the ones with the complex layouts like some chinese web sites. The results of this algorithm are compared with the VIPS algorithm (Cai et al. (2003b)). It shows that that when the number of output segments are small VIPS performs better as in this stage proximity plays the most important role, however when the number of output segments are increased the algorithm with the Gestalt theory outperforms the VIPS algorithm significantly.

Fernandes et al. (2011) presents the segmentation from the website perspective. They define the block graph as an auxiliary tree (SOM tree), which have the attributes as the DOM elements but with 2 extra attributes: a counter, with the number of pages where the element occurs in the site and the list of pages where it occurs. The SOM tree is refined applying heuristic rules to merge those elements, conforming blocks where the difference in their depth is below to a threshold. However, this largely concerns with the design of the website itself. It is not very relevant for the segmentation problem at hand. The segmentation method in Fernandes et al. (2011) is particularly useful to segment data-intensive Web sites, such as digital libraries, Web forums, news Web sites, electronic catalogs, or institutional sites, whose main focus is providing access to a large quantity of data and services. These sites usually contain a large set of web pages which can be organized in a few tens of groups according to the regularity of their structure. This segmentation method thus takes advantage of such regularity to automatically segment data intensive web sites. The experiments have been performed on a collection of 4,460 pages crawled from Brazilian Web portals which are composed of a recipe site, a forum site, and a news Website. The evaluations are done to compare the algorithms in Fernandes et al. (2011) and Kohlschütter and Nejdl (2008). The Adjusted RAND index evaluation shows that the algorithm in Fernandes et al. (2011) performs better than the algorithm in Kohlschütter and Nejdl (2008).

The methods presented in section 2.3, are efficient on content intensive sites as they are designed for the task of browsing in small screen. However, while it comes to sites like e-Commerce and e-Tourism where visual features dominate content these algorithms are not effective as these methods use the contents of

the HTML DOM to segment the web page and summarizes the web pages to fit the task. The task of non-visual skimming and scanning requires a technique of segmentation that is efficient on all types of web pages irrespective of the template and content.

2.4 By annotation markup

Manabe and Tajima (2015) develop a method for extracting logical hierarchical structure of HTML documents. They exploit the properties of headings such as: (1) headings appear at the beginning of the corresponding blocks, (2) headings are given prominent visual styles, (3) headings of the same level share the same visual style, and (4) headings of higher levels are given more prominent visual styles, in order to perform the task. The authors define a block as a coherent segment of a document that has its own heading describing its topic and a heading is a visually prominent segment of a document describing the topic of another segment. The authors make some observations and assumptions for the positions and visual style of heading on the web page. Their method is called HEPS (HEading-based Page Segmentation). After the pre-processing step, in which the blank nodes and the sentence breaking nodes are removed, the authors use three types of information - tag path, computed style and height of images, in order to get the visual styles of the candidate heading nodes. The candidate heading nodes can either be text or images. This produces a set of candidate heading lists. In the next step, the candidate heading are sorted, first by block depth, then by visual style and later by document order. Given a sorted list of candidate-heading lists, the authors first segment the document into top-level blocks by using the first candidate-heading list then segment these blocks by using the next candidate-heading list. For evaluation, the authors use Precision and Recall measures for heading extraction and block extraction separately. The algorithm presented by Manabe and Tajima (2015) relies on the heading nodes to segment web pages which is relevant for content heavy web pages. However, for pages such as tourism or e-Commerce where headings are not the most important visual aspect for segmentation of a web page. Thus though this algorithm is efficient for content heavy web pages, it might not work well for the web pages that do not rely on headings.

Chen et al. (2001) proposes a Function-based object Model(FOM) for the goal of web page adaptation on small screen devices. Every Object in a website serves for certain functions (Basic and Specific Function). FOM includes two complementary parts: Basic FOM based on the basic functional properties of Object and Specific FOM based on the category of Object. The authors describe the Basic Object and Composite Object. The authors go ahead to propose various different object categories such as Information Object, navigation Object, interaction Object, Decoration Object, Special function object and page object. The authors present a heuristic based approach using the basic and specific FOM. A system for web content adaptation over Wireless Application Protocol(WAP) has been developed as an application example for

their proposed model. However, the authors do not present any quantitative evaluation of their approach except they mention that the approach provides satisfactory results when compared to other techniques and is able to give the user the same browsing experience on a small screen. As no quantitative evaluations are provided it is difficult to conclude on the efficiency of this algorithm.

The algorithms described in section 2.4, use HTML tags and their properties/functions to segment web pages. The methods presented in this section use only the HTML tags and their perceived functions to segment a web page, however, they do not consider the visual features associated with the tags. Though these techniques could work well on content heavy web sites, they might not fare well on other types of web pages. In case of tourism or e-Commerce web pages, there are no headings or any particular tags that help in distinguishing between paragraphs. Thus it is essential to use the visual aspects of the web page such as position and alignment of web elements to enable good segmentation in such web pages. Thus these algorithms are not very useful to the task at hand.

In this chapter, segmentation of web pages with respect to accessibility was discussed. Accessibility refers to a web site being accessible by all of people in all sorts of devices. Section 2.1 describes web page segmentation to enable web access to the disabled while sections 2.2, 2.3 and 2.4 discuss web page segmentation for the task of allowing small screen web access. The goal of the task at hand is to allow skimming and scanning for the visually disabled people. Thus for the task, it is required that the visual features of the web page be preserved and used for segmentation process. Also it is necessary to present all the information that is on the web page without removing information in the pretext of "noise". It is also necessary to develop a method which works equally good on web pages that are content heavy and web pages that do not have much content like tourism and e-Commerce web pages.

Chapter 3

Zoning a web page

3.1 Logical Approach: web page as a graph or tree

Yin and Lee (2005) introduces a model which helps in segmenting a web page by constructing a graph and using a random walk algorithm on it. The algorithm classifies elements of Web pages into five categories which are Content (C), Related Links (R), Navigation and Support (N), Advertisement (A) and Form (F). The algorithm constructs directed graphs for each functional category based on the elements such that the sum of the weights coming out of each node is 1. The weight of edge (i, j) is the probability of a random walker at node i moving to node j at the next time instance. Based on the features of the two basic elements in consideration, a connection between them is formed indicating the increase in the likelihood that the two nodes belong to the same object. The features considered to form this connection are match (cosine similarity), Distance, Neighborhood, same tag, same edge, same parent. The authors also proposes the idea of CategoryRank (CR) to calculate the likelihood that an element in a web page belongs to certain category (kind of like Page rank). Each element will get five CategoryRank from the five graphs. Then it compares the element's CategoryRank in five graphs and classifies an element to a category with the maximum CategoryRank. Evaluation has been done against other machine learning algorithms and the performance of this approach is better than the machine learning approaches for all categories except for the navigation.

Liu et al. (2011) presents a web page segmentation algorithm based on finding the Gomory-Hu tree in a planar graph. The algorithm first gets the rendered DOM of a web page to construct weighted undirected graph. The vertices of this graph are nodes with real message, such as text, picture and video. The visual layout information of a web page is used to add edges in a graph - if two vertices are neighbors on the browser screen, an edge is added between them. The structural layout of the web page is used to add weights for the edges. The path similarity calculated using the DOM is used as edge weight between two

vertices. Then the algorithm partitions the constructed Gomory-Hu tree by computing the minimum cuts. Minimum cuts is the maximum flow between each pair of vertices to find the minimum cut between them, and constructs the minimum cut tree using these minimum cuts. These minimum cut tree thus formed maximizes the intra block similarity and minimizes the inter block similarity.

Chakrabarti et al. (2008) uses DOM as a graph and performs correlation clustering and energy-minimizing graph cuts. The correlation clustering problem starts with a complete weighted graph. The weight $v_{pq} \epsilon [0, 1]$ of an edge represents the cost of placing its endpoints p and q in two different segments; similarly, (1 - v_{pq}) represents the cost of placing p and q in the same segment. Since every edge contributes, whether it is within one segment or across segments, the segmentation cost function is automatically regularized. The algorithm used is CClus.The CClus is iterative. At each stage, a node p in the current graph is chosen uniformly at random and removed from the graph. A new cluster is created with just p in it. Next, all the nodes q such that $v_{pq} \geq 1/2$ are removed from the graph and placed in the cluster along with p. The process is repeated on the remaining graph. Chakrabarti et al. (2008) also gives the GCuts algorithm for the Energy-minimizing graph cuts. The algorithm starts with a trivial segmentation mapping of all nodes to an arbitrary visible label. Then, the algorithm proceeds in stages, called α-expansions. In each α-expansion, the algorithm picks a label $\alpha \epsilon$ L, and tries to move subsets of nodes from their current labels to α so as to lower the objective function. The optimal answer for each α-expansion can be found using the minimum s-t cut of a graph. The energy function becomes critical. After the best max-flow cut is found, nodes connected to s have their labels unchanged, while nodes connected to t have their labels changed to α. Now, α-expansions are iteratively performed for all possible labels $\alpha \epsilon$ L until convergence. Chakrabarti et al. (2008) also suggests that energy-minimization technique is more effective than the correlation clustering. Hu and Liu (2014) adds 2 extra features to the work done in Chakrabarti et al. (2008). The visual features and the content base features. The edge weights are derived from these features using a regression function learned from a set of manually labelled web pages.

The methods presented in section 3.1 uses the DOM as a sort of graph to segment web pages. Although, these approaches provide good results for the segmentation of a web page, the DOM is prone to errors due to uncontrolled page creation. Also, in order to calculate the weight of each edges certain features of the web page are taken into account, which depends on the task. Thus, for the task of non visual skimming and scanning, it is important to decide which features should be considered and which should not be. This might depend on the type of web page, the information the user wants to see or the content of the web page. These type of techniques might be useful for the task at hand if the right features and importance of the features are selected.

3.2 Textual Approach: web page as text content

Prince and Labadié (2007) is about text segmentation based on topical change. The method presented in Prince and Labadié (2007) is for information retrieval for answering queries. The algorithm is about comparing the semantic vector of the query with all the detected segments (with a transition of 2 sentences), and retrieving those segments whose angular distance with the query does not exceed a pre defined amount (in this case, 0.8, which is roughly about 45°). If two vectors make an angle of 45° and less, they are considered to be relatively close to each other. Transposed as a relationship between query and fragments, this means that the fragment is (semantically, topically) relevant to the query. The closer to 0° the angle is, the more relevant the fragment is.

Mihalcea et al. (2006) suggests multiple methods for measuring text semantic similarity, which could be used along with the algorithm presented in Prince and Labadié (2007) for segmentation. The methods for text semantic similarities are classified into Corpus-based Measures - Pointwise Mutual Information and Latent Semantic Analysis, Knowledge-based Measures - Leacock & Chodorow's method, Lesk's method, Wu and Palmer method, Resnik's method, Lin's method, Jiang & Conrath's method.

Hearst (1993) gives an algorithm to partition full length text documents. The algorithm is same one as explained in Fitzgerald (2000), but instead uses only the tf-idf method to identify the coherent blocks.

Kohlschütter and Nejdl (2008) presents a block fusion algorithm. For the algorithm, the web page is considered as a series of blocks. When an element lacks tag information(tags such as
), then it is a strong indicator for the segmental unity of text portions. Thus these types of tags are used to create blocks. The algorithm uses a measure of block density to decide whether to partition or fuse two adjacent blocks which is made by comparing them with respect to their text densities. Text density is the number of words within a particular 2-dimensional area. This idea is used with a sequence of atomic text portions, which are called blocks and the block density is defined as follows:

$$\rho(b_x) = \frac{Number\ \ of\ \ tokens\ \ in\ \ b_x}{Number\ \ of\ \ lines\ \ in\ \ b_x}$$

The algorithms presented in section 3.2 uses the textual contents for segmentation. The algorithms presented are developed with the goal of segmenting text content not specifically web pages. Thus these methods would work well for text intensive web pages like news web pages, however, these might not work well for web pages like tourism as there are usually not much text content in them.

3.3 Visual Approach: web page as an image of the page

3.3.1 Web pages as images

Chudasama et al. (2015) presents the general techniques that could be used for image segmentation. The approach suggests two phases of segmentation. In the first phase, the image is pre-processed for Edge Detection using Hybrid Fuzzy-Canny method. The second phase uses morphological methods for image segmentation. The morphological methods used include Erosion, Flooding and Dilation.

Cao et al. (2010) presents an algorithm about iterated shrinking and dividing for web page segmentation. The web page is saved as image that is pre-processed by edge detection algorithm such as Canny. Then dividing zones are detected and the web image is segmented repeatedly until all blocks are indivisible.

Mathematical Morphology(MM) is a technique used for image segmentation. MM is most commonly applied to digital images, but it can be employed as well on graphs, surface meshes, solids, and many other spatial structures. The basic MM operators are erosion, dilation, opening and closing. This technique has been used to segment web pages within the task of TAG THUNDER by using the image of the web page as the input. An example of a web page is shown in Figure 3.1. MM uses the edges and vertices of the image and thus cannot be used for web pages because of the numerous amount of edges and vertices present and the fact that the task requires a fixed number of 5 zones. Also when a page with a background image needs to be segmented, the edges of the background images are used for the detection of zones as seen in Figure 3.2. Thus the image processing technique is ruled out for the WPS for the TAG THUNDER task.

The algorithms presented in sub section 3.3.1, use the web pages as images and performs segmentation on the images. These techniques uses the edges of the images to segment the web pages. However, while the background of the web pages are images, these techniques are not very efficient as the edges of the background images are detected with the algorithms and disturbs the segmentation of the web page.

3.3.2 Visual based Page Segmentation Algorithm (VIPS)

One of the major and most algorithms which uses vision based techniques is the VIPS (Vision Based Page Segmentation) algorithm. Cai et al. (2003b) gives a detailed account of the algorithm. The web page is represented as a triple. A set of finite set of blocks (O), a finite set of separators and the relationship between the blocks. The blocks are not overlapped. Every separator has a

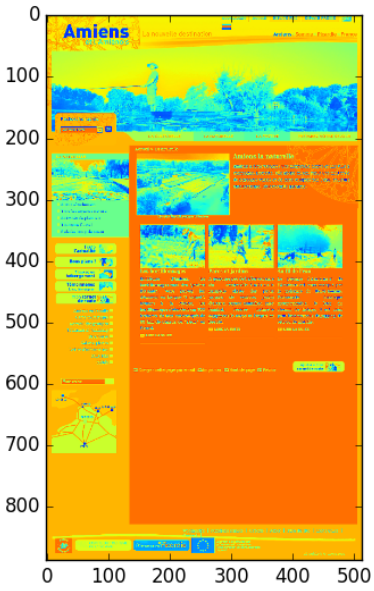

Figure 3.1: *Example of a Web Page segmented using MM*

weight indicating its visibility and all the separators in the same set have the same weight. For each block, the Degree of Coherence (DoC) is defined to measure how coherent it is. DoC has the following properties:

— The greater the DoC value, the more consistent the content within the block

— In the hierarchy tree, the DoC of the child is not smaller than that of its parent

Permitted Degree of Coherence (PDoC) is used to achieve different granularities of content structure for different applications. The vision-based content structure of a page is obtained by combining the DOM structure and the visual cues. The algorithm is in three steps: block extraction, separator detection and content structure construction. The web page is firstly segmented into several big blocks and the hierarchical structure of this level is recorded. For each big block, the same segmentation process is carried out recursively until sufficiently small blocks whose DoC values are greater than predefined PDoC are obtained. For each round, the DOM tree with its visual information corresponded to the current block is obtained from a web browser. Then, from the root node(s) of the DOM tree, the block extraction process is started to extract blocks from the DOM tree based on visual cues. Every DOM node is checked to judge whether it forms a single block or not. If not, its children will be processed in the same way. Then, the DoC value to each extracted block is assigned based on the visual property. When all blocks of the current round are extracted, they are put into a pool. Separators among these blocks are identified and the weight of a separator is set based on properties of its neighboring blocks. The layout hierarchy is built based on these separators. After constructing the layout hierarchy of the current round, each leaf node of the content structure is checked to see whether or not it meets the granularity requirement. If not, this leaf node will be treated as a sub-page and will be

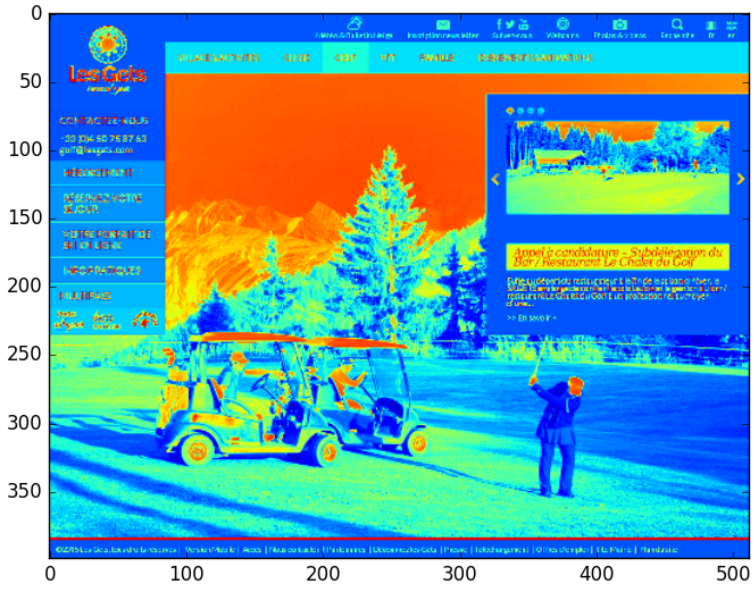

Figure 3.2: *Example of a Web Page with a background image that is segmented using MM*

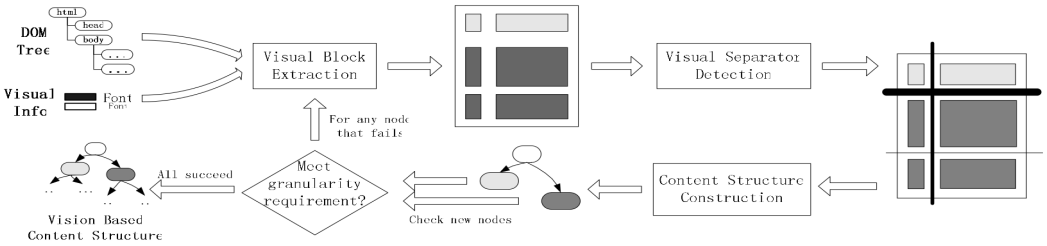

Figure 3.3: *Vision Based Page Segmentation Algorithm. Figure referenced from Cai et al. (2003b)*

further segmented similarly. All the blocks have to be processed to get the final vision based content structure of the web page. Figure 3.3 shows the process involved in the VIPS algorithm in a diagrammatic way.

Experimentation is done on 600 web pages from popular sites listed in 14 main categories of Yahoo! directory. Human evaluators then evaluate the segmented web pages using the VIPS algorithm and give a label of "perfect", "satisfactory", "fair" and "bad". 93% of web pages have been classified as segmented as "perfect" or "satisfactory".

VIPS is one of the first algorithms to use the visual information of the web page for segmentation. It does produce good results for the segmentation of web pages. However, the PDoC that is used in the algorithm is a sort of threshold that needs to be defined. This depends on the web page and depending on the set PDoC some web elements might be left without being in any zone. This is one of the disadvantages of VIPS.

3.3.3 Further works on VIPS

Cai et al. (2003a) uses the VIPS algorithm to segment a web page into semantically related content blocks from its visual presentation. The algorithm is used to increase the performance of information retrieval. Since the VIPS algorithm can group semantically related content into a segment, the term correlations within a segment will be much higher than those in other parts of a web page. With improved term correlations, high-quality expansion terms can be extracted from segments and used to improve information retrieval performance.

Cunhe LI (2010) deals about one application of VIPS. It is about extracting only the informative blocks from a web page and excluding other blocks like navigation, copyright information, privacy notices, and advertisements, which are not related to the topic of the web page. Cunhe LI (2010) is about applying the VIPS algorithm to the web pages to identify visually similar blocks. Although visually separated blocks provide a semantic partitioning of a page, a block might be too small to be considered as the source for information extraction. Therefore, different algorithms to find the similarity between the segmented visual blocks are used. The spatial, semantic and content features are considered. Based on the similarity measures, a block clustering method is used to cluster the visually segmented blocks.

Liu et al. (2006) studies the response pages returned from web databases or search engines. The work uses only visual information of the response pages when they are rendered on web browsers. Several type of visual features are analyzed in Liu et al. (2006) First, Liu et al. (2006) uses the VIPS algorithm to construct the Visual Block tree for each response page. Second, locate the data region in the Visual Block tree based on the Position features. Third, extract the data records from the data region based on the Layout Features and Appearance features. Akpınar and Yeşilada (2013) presents a technical improvement of the VIPS algorithm, adapting it to current web standards and use them in the context of new applications.

3.3.4 Box Clustering Algorithm

Zeleny et al. (2017) propose a clustering approach based on similarity of boxes. Zeleny et al. (2017) create a rendering tree with all CSS elements. They choose the basic elements using a pre-order traversal of the rendering tree, selected boxes contain information such as position, color, size and shape. Then the area graph is produced which is basically the neighborhood of each element considering the alignment. The idea is to find the most similar couples of boxes and then, to select them for merging. If at least one of the entities is a cluster, a new candidate cluster is created. If both entities are boxes, a new cluster seed is created instead. The authors compare their work with VIPS. BCS uses a Cluster Threshold (in case of VIPS it is called PDoC), which varies for each web page and brings along a risk of unclustered elements. The algorithm

is experimented with 8 different types of pages from 5 news web sites. The box clustering segmentation algorithm is evaluated for ARI and F-score with respect to algorithm acuuracy and algorithm stability. The results show that the accuracy of VIPS is slightly better, especially when processing structured pages. When processing pages with less structure, the accuracy of BCS and VIPS is comparable, in some cases BCS is even better than VIPS. In some cases the stability of BCS is almost three times better than that of VIPS.

3.3.5 Block-O-matic Algorithm

Sanoja and Gancarski (2014) is a technique which combines vision based model and geometric layout model (Block-O-Matic). A web page is processed to build three structures: content, geometric and logical structure. The outcome of the processing is a segmented web page, which is a consolidated view of the three structures above mentioned. The segmentation process of a web page is divided into three phases: page analysis, page understanding and page reconstruction. The DOM tree is obtained from the rendering of a web browser. The result of the analysis phase is the content structure of the web page. Page understanding is done by mapping the content structure into a logical structure. This mapping is performed using a granularity parameter pG. Then the web page reconstruction gather the three structures(content, geometric and logical). For experiments, a custom test collection of 400 pages crawled from dmoz.org Open Directory is used. These web pages are manually assessed to define a comparable segmentation. A set of 25 pages from each of 16 categories is then selected. These pages are then segmented automatically with both VIPS and Block-O-matic algorithms. With the ground truth and an automatic segmentation, a block in the automatic segmentation is said correctly segmented if its geometry and location are equal to only one block in the ground truth. The Block-o-Matic algorithm has a better performance than VIPS in the amount of correct blocks found over the whole collection. Both algorithms have problems when the tolerance is very low, which is normal because the geometry of blocks is not entirely exact. However, with 10px of tolerance Block-o-Matic present the best performance which means that blocks geometry is very similar. On the other hand, VIPS requires a high tolerance to observe a better performance.

Sanoja (2015) describes an application of the Block-O-matic algorithm called Pagelyzer, which compares two Web Page versions and decides if they are similar or not. The first step of the Pagelyzer produces an HTML document integrating the visual cues. In the second step, the web page is segmented using the Block-O-Matic algorithm. At the end of this step, 2 XML trees, representing the web pages are returned. In the third step, visual and structural descriptors are extracted. The structural and visual differences are merged to obtain a similarity vector used to determine if the two urls are similar or dissimilar. The experiments for Pagelyzer has been conducted on 5 categories of web pages such as blogs, enterprise, forum, picture and wiki. Sanoja (2015) compares the ground truth with machine generated segmentation to perform

evaluation in order to perform precision and recall. The BoM algorithm has a high precision for the forum and picture categories. Forum category presents the lowest error rate. The worst performance is for the enterprise category.

The Block-O-matic algorithm performs better than VIPS in identifying zones that are similar to the ground truth. However, again like the VIPS the Block-O-matic algorithm uses a threshold/granularity(pG) in the page understanding phase (mapping). This causes some web elements left without being put into any segment. Also, based on the threshold/granularity set the number of zones formed differs. None the less, this is an interesting algorithm for exploration.

3.3.6 Other vision based Techniques

Aruljothi et al. propose Web page segmentation for small screen devices using tag path clustering approach. The HTML tags are extracted from the HTML source code. Every tag has a tag path indicating its ancestors. Each tag path defines a unique visual signal. For example, a visual signal for the tag path of html/body/table is 0 0 1 0 0 0 0 and the visual signal for the tag path htmlbody/tabletr/td is 0 0 0 0 1 0 1. The authors do not exactly precise how the visual signals are computed. Then a pairwise similarity matrix is constructed based on the visual signals. Clustering is then performed using spectral clustering algorithms based on the similarity matrix. This is known as tag path clustering. After tag path clustering, the web page is segmented either by reappearance based segmentation or by layout based segmentation. The algorithm checks for a key pattern in order to be able to use the reappearance algorithm. If no key patterns are found, the layout information is used for the segmentation process. The web page is split into blocks, after segmentation process. Informative blocks are determined by evaluating the quantity of information within the blocks, which might be done by assigning an importance weight to every node. From that informative divided block, hyper-link is created and displayed on the mobile devices.

Gu et al. (2002) proposes a technique for web page content structure detection to facilitate automatic web page adaption. The authors use a projection-based algorithm to do this. The web elements are divided based on their position or merged if they are visually similar.Projection refers to the mapping of a web page into a waveform. All objects in a web page are contained in rectangular blocks. Blanks are placed between these rectangles. Thus, the projection profile is a waveform whose deep valleys correspond to the blank areas of the web page. A deep valley with a width greater than an established threshold can be considered as separator between objects. The separators detected may break a holistic object thus a merging step is performed based on visual similarities. The text content similarity, font similarities an alignment are considered for the visual similarity. Experimentation for this algorithm has been conducted on 50 popular web pages listed on http://www.yahoo.com, of which, 45 web pages have been correctly segmented.

Yang et al. (2003) presents a method to automatically analyzing semantic

structure of HTML pages based on detecting visual similarities of content objects on web pages. In this approach, the algorithm first measures visual similarities of HTML content objects. The visual similarities are the font similarities with elements with text attributes and for embedded media objects like images their description is extracted from the tag and used for the similarity measure. Then a pattern detection algorithm is applied to detect frequent patterns of visual similarity and a number of heuristics are used to choose the most possible patterns. By grouping items according to these patterns, hierarchical representation, a tree, of HTML document with "visual consistency". Experiments are conducted on 50 web pages selected from http://www.100hot.com, of which 46 pages have been processed correctly. The authors also describe an application for the algorithm which is an adaptive web content delivery. The idea behind this system is to summarize web pages to some levels that will not affect human comprehension too much in favorof download speed and client (device/browser) capability, in case of slow internet connections.

Kovacevic et al. (2002) describes a process of segmenting web pages using the visual information. The process of the visual information extraction takes place in two steps. In the first step a page is parsed using an HTML parser that extracts two different types of elements, tags(TE) and data(DE). In the second step, as soon as the TE,DE pair is extracted from the input HTML stream, it is injected into the tree builder. Tree builder applies stack machine and a set of predefined rules to build the tree that represents the HTML structure of the page. This new tree structure is called the m-tree(mT). Then the coordinates of objects are calculated using the mT as the input. This is done by the rendering module(RM). However, the RM does not support frames, does not support style sheets, does not support layered HTML. The authors go ahead to describe certain heuristics for recognition of common areas of interest given the mT of a web page. These heuristics along with the mT are used for 1000 pages and about 73 percent of the pages were successfully recognized for common areas of interest. The authors go on to describe a method to perform page classification using the RM. First, a Naive Bayes classifier is trained on all the words in the documents. Such classifier is usually constructed when not considering visual information and it provides a baseline to validate the effectiveness of the proposed data representation. In order to classify a page taking into account its visual appearance, each page from the training (test) set is processed, extracting the bag-of-words representation of its six basic constituents: header, footer, left menu, right menu, center and title and meta-tags. Then, six Naive Bayes classifiers are created where the i-th classifier is trained using the bag-of-words representations of the i-th constituents of the documents. When classifying a document, the i-th classifier assigns a score to the document equal to pi(c—d). After some tuning the following weights are assigned to each classifier: header 0.1, footer 0.01, left menu 0.05, right menu 0.04, center 0.5, title and meta-tags 0.3. Taking into account the visual appearance of the page that is provided by MT, more than 10 percent improvement has been achieved in the classification accuracy.

Xiang and Shi defines a web page as a composition of basic visual blocks and separators. The authors define two types of basic visual blocks: Nontext blocks

and text blocks. A font weight is assigned to each text block to describe its importance of visual perception. Separators between basic blocks are composed of visual lines and blank in the web pages. Visual lines usually come from borders of TABLE or HR. Three kinds of separators: horizontal, vertical, and closed separators. The authors go on to describe a method to recover semantic relations. Several basic visual blocks are logically grouped together by semantic relations into a composite block which represents a design pattern. A pattern consists of three parts: the pattern block, lines that represent semantic relations, and the child blocks that may be basic visual blocks or composite blocks. First, blocks which are separated by the smallest weighed separators are chosen. Then, for each pattern, a special recognizer based on hard and soft constraints examines these blocks to find appropriate pattern instances. Recovered patterns are merged into composite blocks.

Zhang et al. (2010) focus on finding the set of nodes that are labeled as Content Row. A content row is a set of leaf nodes of the rendered DOM tree which are horizontally aligned and are siblings. Content rows are merged if there is an overlap between them. In step two, the block headers are detected. A content row is a block header expect under certain conditions (there are heuristics). Each detected block is a separator of two semantic blocks. A semantic block is a stack of vertically aligned content rows. Vineel (2009) defines a content size and an entropy value that measures the strength of local patterns within the subtree of a node. Threshold values are defined for both measures to perform page segmentation.

The methods discussed in section 3.3 focus on using the visual aspects of a web page. Block-O-matic and VIPS rely heavily on the DOM structure of a web page, however, DOM is prone to errors due to uncontrolled page creation. Also, the number of clusters is automatically determined in these methods and thus can greatly vary from page to page. BCS (Box Clustering Segmentation) relies on a flat visual representation of the document, thus allowing great adaptability to new web contents. BCS follows a sort of hierarchical agglomerative clustering algorithm including a threshold that controls the formation of clusters. Thus the number of clusters is automatically determined and leaves some elements unclustered. However, BCS uses the flat structure of the web page instead of relying on the DOM. This technique is more reliable for the task of non visual skimming and scanning as well Zeleny et al. (2017).

3.4 Hybrid Techniques

Safi et al. (2015) presents a hybrid technique to segment a web page. This technique uses the visual, DOM and graph based strategies to segment a web page. In the visual phase of the algorithm the web page is rendered using the selenium web driver and the Mozilla FireFox browser. The visual structure of the web page is then obtained using by the injecting the css styling information within the HTML file. In the DOM based approach, filters are used to remove unnecessary nodes from the HTML file. In the graph based approach,

a block2zones clustering is used to segment the web page. In this clustering technique, each node is considered as a block. The smallest block is selected and merged with the connecting block having the highest weight until the remaining blocks is equal to the number of desired zones. The weights of the edges are calculated using the euclidean distance between the nodes/blocks. For the evaluation, 15 volunteers are asked to manually segment different kind of web pages. Each volunteer is asked to segment 8 web pages with 3,4,5 and 6 zones, considering certain criteria. The algorithm is then run on 8 web pages and segments each web page into 3,4,5 and 6 zones. The manual annotations and the automatic segmentation are compared with each other using the strong and weak criterion. The strong criterion meaning 100% identical results and the weak criterion meaning 50% identical results. With the strong criterion, The percentage of identical matching depending on the strong criterion is 15.42% with highest match while the pages are segmented into 4 zones. The matching percentage depending on the weak criterion is 47.5% with the highest match when the number of zones is 3. The algorithm presented in Safi et al. (2015) uses a hybrid technique to segment web pages, however, the process of clustering blocks uses only the euclidean distance. Thus this approach is interesting but has to integrate more features to be able to satisfy all the criteria for the task of non-visual skimming and scanning. Maurel et al. (2020) presents a detailed discussion on this method.

Nguyen et al. (2012) is about a hybrid technique combining visual and semantic features. The algorithm extracts the DOM, creates the property tree, creates the elementary blocks (depth first traversal of the property trees), identifies the main content, predicts semantics of elementary block using SVM, groups the elementary blocks using Bayesian network, CRF and rule based.

Hattori et al. (2007) is a hybrid between layout and content based segmentation. Content distance is a distance between content elements based on the number and depth of tags. Web page segmentation is based on the calculated content distance and the layout information of the web page.

Fitzgerald (2000) is one of the most famous algorithm. It combines the structural and semantic features for the segmentation process. The algorithm divides the text into appropriately sized sequence of text to calculates gap scores at the potential boundary points. Then, it uses these cohesion scores to produce depth scores for each potential boundary point that has a lower cohesion than the neighboring boundary points. The next step is the smoothing process using an average smoothing technique with a flexible window size. Then the algorithm calculates the depth score, if the depth score is high, then the relativity is low.

Rajdeepa B (2014) uses VIPS, K means and hierarchical clustering to segment web pages. First the web page is analyzed for content structure using Vision-Based content structure analysis App. Then the analyzed web page is segmented into smaller units using VIPS algorithm. Segmented web pages are clustered and noisy data in the units are removed using K-Means algorithm in clustering. Noisy data means citations, advertisements etc. After removing

noisy data the segments are again merged into single unit using hierarchical clustering method. The silent content of web page is stored in web page database.

Jiang et al. (2019) describes a two stage segmentation process that considers the visual, logical and semantic features of a web page. The method consists of a pre-processing step which included rendering, extracting and filtering of web elements.In order to keep up with the increasing use of dynamic content in web technology, Jiang et al. (2019) uses PhantomJS - a web browser framework for executing scripts and rendering web pages. The web elements are then extracted from the rendered web page. Unwanted content such as online videos, invisible elements, and overlapped elements are filtered out. The remaining web elements after the filtration is considered as the input for the two stage segmentation process. Stage 1, known as modeling and clustering, aims to fit the human reading habits. Thus a a model to measure the similarities of the elements on web pages based on visual layout and logic organization is proposed. This aims to aggregate both visual and logical features into the model to measure the similarity between DOM elements on web pages. The visual similarity is given by the visual distance, which is based on the vertical and horizontal distance between the web elements. The logical similarity is given by the logical distance, which is the distance of the nodes in the DOM. A model is then built combining these two distances. Based on this model, the similarities between elements are calculated and the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is used to cluster the elements into bigger blocks. DBSCAN does not require to set the number of clusters, and it is able to filter out noise data points. Stage 2, known as the semantic regrouping stage, the text density of the blocks are used to regroup the web elements i.e. If two adjacent blocks obtain similar text density, the blocks can be regrouped together, to aggregate related contents into the same blocks thus giving the final segmentation of a web page. Experiments for this work has been done on 3 different datasets - 70 homepages of popular websites collected in 2014, including the labeled ground truth for segmentation, 82 homepages of random websites collected in 2014, including the labeled ground truth for segmentation, homepages of websites from Alexa Topsites collected in September 2017. The method presented in Jiang et al. (2019) outperforms both BoM and VIPS on all metrics under the three individual datasets. Considering the merged dataset(combination of all three datasets), this method reaches 39.9% in precision, 42.4% in recall, and 0.518 in ARI, which improves the performance significantly in comparison with the other two methods.

Alcic and Conrad (2011) describes a general approach to web page segmentation. Alcic and Conrad (2011) describes three different measures that are used to segment web pages. Later these measures are used with various clustering techniques to segment web pages. The three distances mentioned in Alcic and Conrad (2011) are DOM-based distance, Geometric distance and Semantic distance. The DOM-based distance, uses the DOM structure of a web page. The method assigns a weight weight to each level of the DOM. Then based on the path between the two web elements the DOM distance is calculated. This method makes 2 assumptions - Adjacent sibling leaf nodes

have all the same distance and The minimal distance of leaves belonging to different parents must be greater than the maximum distance of these leafs to their siblings. The geometric distance uses the rectangular box surrounding the web content. It is the minimal distance between the rectangles. The semantic distance used the text content of the web elements. Alcic and Conrad (2011) represents all content of the web page as text - Images, videos etc. The images, videos etc. are represented using the alternative text or using their url. Then the semantic similarity is calculated using the metric stated in Lin (1998), this metric computes the semantic similarity on the concept level but in order to expand this to a text paragraph level after some pre-processing. These measures - DOM based distance, geometric distance and the semantic distance, are used with clustering techniques to segment web pages. The clustering approaches studied in Alcic and Conrad (2011) are partition clustering including K-means and K-medoid, hierarchical agglomerative clustering and Density based clustering such as DBSCAN. Each distance is used individually with each clustering methods. The evaluations in Alcic and Conrad (2011) are evaluated for the average Dunn index for different distance measures and different clustering techniques and rand statistic described in Alcic and Conrad (2011). Alcic and Conrad (2011) show that the DOM-distance measure combined with the extended DBSCAN algorithm reached the best Rand statistic value in average. Otherwise, the best separation values have been achieved by the semantic distance.

The methods described in section 3.4, combine different techniques or different features in order to achieve a good segmentation for the task. There are techniques that combine visual features with the semantic features, techniques which combine various measures such as DOM distance, geometric distance and semantic distance in order to combine or split blocks of web elements to form zone. These approaches are promising for the task at hand, however, the features and techniques have to be tuned for the task at hand. Inspired by these approaches and to satisfy the requirements for the task of allowing skimming and scanning for the visually impaired people, certain criteria, measures and techniques are being developed to perform good segmentation for the task at hand. To be able to do this there are various choices that needs to be made such as the measures that needs to used, the technique, the criteria for segmentation etc.

Part II

Implementation

Chapter 4

Choices for Web Page Segmentation (WPS)

4.1 Introduction

As discussed in section 3.4, the task of web page segmentation to allow skimming and scanning for the visually impaired people can benefit from using hybrid techniques. In this part, the features and techniques that are necessary to be combined to achieve a good segmentation for the task at hand - segmentation to allow skimming and scanning for the visually impaired people, are discussed and algorithms using the choices made are developed. In this chapter, the technique chosen will be discussed in detail followed by the choices to be used with the chosen technique. Following chapters will discuss the developed algorithms and their results on web pages.

Firstly, after ruling out the possibilities of performing classification and image processing techniques with reasons described in the previous part, the decision to choose a clustering approach for the task at hand has been made. However, for the clustering process, there are certain choices that have to be made. Firstly, the "basic elements" in a web page, i.e. the elements that will be considered as data points for clustering, should be chosen. Second, the formation of zones is highly dependent on the distance between web elements. The elements with the shortest distance should be placed in the same zone. However, not only the distance but there are several other features that influence the segmentation for the task at hand. These features have to be selected and used based on their roles played in the design of the web page. Thirdly, for the clustering process to be more effective the number and positioning of initial seeds are very important. These are again guided by the task at hand.

4.2 Technique Choice

As mentioned in section 4.1, a clustering approach is being followed to tackle the task at hand. This section describes the steps and experiments that lead to this choice. Based on the task of Tag Thunder(non visual web page skimming), there are certain constraints to be imposed for the clustering problem. The constraints are have been presented in chapter 1. The constraints can be summarized as follow:

— The number of zones to be identified should be fixed because of the task of non visual skimming and scanning, and ideally to 5 following the study made in Guerreiro and Gonçalves (2015), Manishina et al. (2016).
— Each zone should be a single compact block made of contiguous web elements, and the zones should not overlap.
— All web elements must be placed in one zone or the other.

Clustering

Based on the criteria listed above for the task of TAG THUNDER, it is possible to use clustering techniques. Within the technique of clustering, it is possible to fix the number of clusters to 5, where each cluster would represent a zone. It is also possible to ensure that all the basic elements are clustered and to impose clusters to have visual coherency. Thus we choose the clustering technique for the task of WPS within TAG THUNDER. There are several clustering approaches readily available, and in this section a overview of some of them that have been traditionally used for segmentation of pages are discussed.

Clustering in literature

Clustering approaches have been used for several tasks previously. The works, Kriegel and Zimek (2010) and Rokach and Maimon (2005) present an overview of the different types clustering techniques that could be used for various different purposes. The types of clustering dealt in Kriegel and Zimek (2010) are subspace clustering, ensemble clustering, alternative clustering, and multi view clustering. The type of clustering dealt in Rokach and Maimon (2005) are Hierarchical Methods, Partitioning Methods, Density based methods, Error Minimization Algorithms, Graph-Theoretic Clustering and Model based Clustering Methods

However, there are specific clustering techniques that have been used for segmenting web pages using various features and different types of data points. Lin et al. (2010) is about extracting a similarity matrix among pages via in-page and cross- page link structures, based on which a density-based clustering algorithm is developed, which hierarchically groups densely linked web pages into semantic clusters. Lin et al. (2010) uses a hierarchical clustering method called HSClus, which is derived from a density based network clustering al-

gorithm called SCAN. HSClus tests SCAN with different pairs of parameters, then uses a scoring function to evaluate the clustering results under different parameters and finally clusters pages by the optimal parameters.

Manjula and Chilambuchelvan (2013) takes a web page as input, constructs the DOM of the web page, uses SVM to classify the non-content blocks from the content blocks, then uses hierarchical clustering (Data structure similarity) using MDL (minimum description length) to segment web pages. Each web page is considered as a cluster to start with. Thus this follows a divisive method of hierarchical clustering.

Choi (2000) describes the algorithm C99, which is a classic algorithm for segmentation. The algorithm takes a list of tokenized sentences as input. A dictionary of word stem frequencies is constructed for each sentence. This is represented as a vector of frequency counts. The similarity between a pair of sentences is calculated with the help of the cosine similarity measure and the similarity matrix is constructed. Each value in the similarity matrix is replaced by its rank in the local region. The rank is the number of neighboring elements with a lower similarity value. The final step is clustering. This algorithm uses the divisive clustering method for clustering.

Alorf (2017) presents a comparison between Kmeans, mean shift and SLIC clustering algorithms by performing clustering on human skin color (images). The results prove that the K-means algorithm has a good performance when the number of clusters K is between 10 and 15. On the contrary, the mean shift algorithm has good performance when the bandwidth is between 0.03 and 0.06. The SLIC algorithm reaches its maximum performance at around k = 100 and the number of clusters can be increased to K = 300 without introducing a substantial amount of time. The comparisons are done based on the time complexity and performance.

4.3 Inputs for Web Page Segmentation (WPS)

The input for a web page segmentation process is the web page itself. A web page is usually written as a HTML or XML file. There can be external style sheets added to HTML using Cascading Style Sheets(CSS). This would contain the styling information. For the purpose of manipulation, there are several ways to represent a web page. A web page can be seen as Images, Document Object Model (DOM) tree or as graphs. It is necessary to choose a certain representation of the web page to be used as an input in order to perform the segmentation algorithms. For the task in hand, the DOM representation of the web page is chosen as it allows easy manipulation.

As for the Web Page Segmentation(WPS) process, a technique of clustering is used as explained in section 4.2. The clustering process used in this process has been adapted to the task of non visual web page segmentation. For this purpose, the number of clusters required from the clustering process has to be

fixed as the number of coherent zones required after the segmentation process is fixed. Also, for the purpose for clustering, the positioning of initial seeds has to be determined. The choice of the number of clusters and the position of initial seeds are discussed in the following sub sections.

4.3.1 Web Pages using DOM

The Document Object Model (DOM) is a representation of a web page that treats an XML or HTML document as a tree structure. The DOM represents a document as a logical aspect. Each branch of the tree ends in a node, and each node contains objects. DOM methods allow programmatic access to the tree; with them one can change the structure, style or content of a document. This method of representation is convenient as it allows easy access to various web elements at different levels that could be used for the clustering techniques of web page segmentation.

For the Web Page Segmentation for the task of non visual skimming and scanning, the input is the web page itself. However, the HTML elements of the web page are enriched with the CSS attributes corresponding to the element. The CSS attributes added include "data-bbox" referring to the bounding box of a particular web element, "data-style" referring to all styling features of the web element and "data-cleaned" referring to the visibility of the particular web element when rendering the web page on screen. The bounding box (data-bbox) are used as data points for the clustering process.

Basic web elements

The data points in the task of web page segmentation are the bounding boxes of the web elements along with their textual and visual properties i.e. they are rectangular in shape. However there are different ways to choose the "basic elements" from the DOM (Document Object Model). An example of a Document Object Model (DOM) is shown in figure 4.1. Figures 4.2, 4.3, 4.4 and 4.5 are the some of the possibilities to choose the basic elements. In order to foster balance between the chosen basic elements, the smallest visual blocks are chosen as basic elements (Figure 4.4).

Smallest visual block as web elements

Smallest visual block refers to the last visual block of a web page. This choice is taken as it is a between the leaf elements (Figure 4.2) and the blocks (Figure 4.3). The last block is identified using the following criteria:

— Web elements such as <p>,<h1> and <h2> are considered as last blocks.
— element is not considered as a last block and thus the children nodes of the DOM are examined.

— The styling elements are not considered as blocks such as small, big, em, strong. Thus the parent node of the DOM of the styling element is examined.
— A list item (<li>) in a list comprising of no other HTML tags is considered as a block. However, if a list item (<li>) is seen to comprise of other HTML tags such as <p> or <h1>, then the list item (<li>) will be further parsed to find the smallest visual block.
— If a element contains text with a <h1> followed more text, then the element is chosen as the block.
— For a table, each entry is considered as a block. (i.e) <th> and <td> are considered as a block.
— <form> is not considered as a block and thus the children nodes of the DOM are examined. <label> and <input> are considered as block.
— Images are considered as blocks.
— Links are not considered as a block and thus the children nodes of the DOM are examined.
— Line break elements such as <hr> and
 are considered as a block.

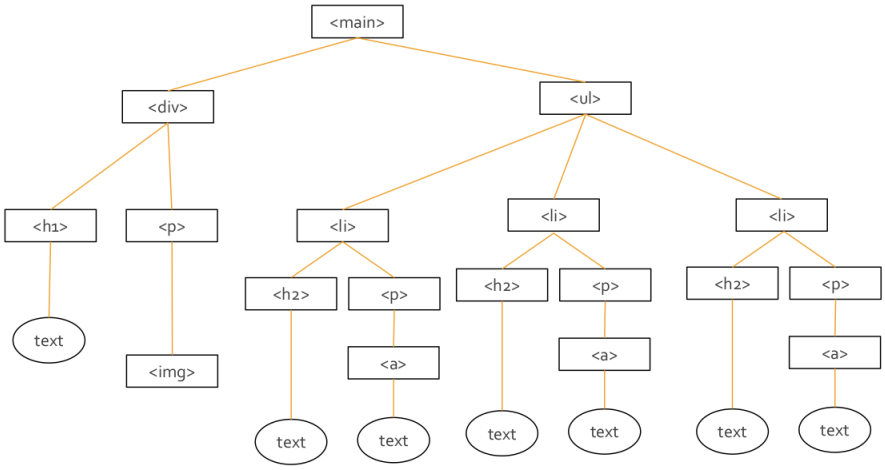

Figure 4.1: *Example of a Document Object Model (DOM)*

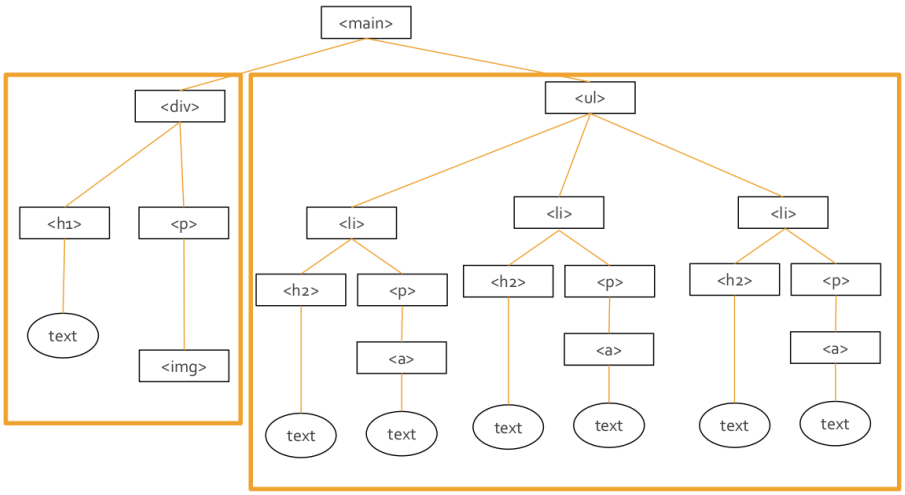

Figure 4.2: *The biggest blocks from the DOM could be chosen as basic elements*

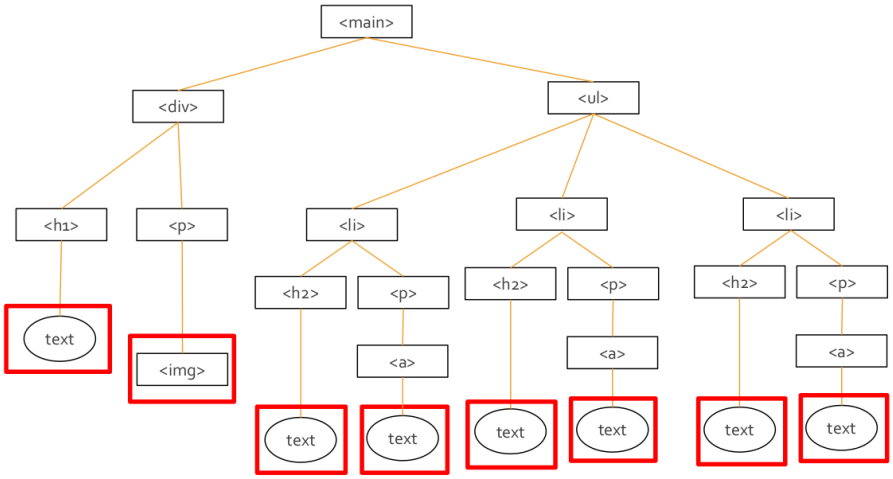

Figure 4.3: *The leaf nodes of the DOM could be chosen as basic elements*

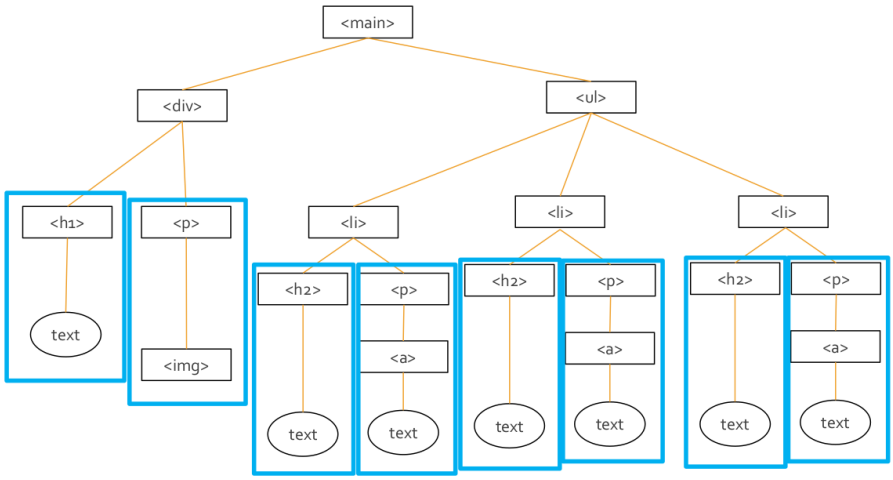

Figure 4.4: *The smallest visual blocks following the rules mentioned in subsection 4.3.1 are chosen as basic elements*

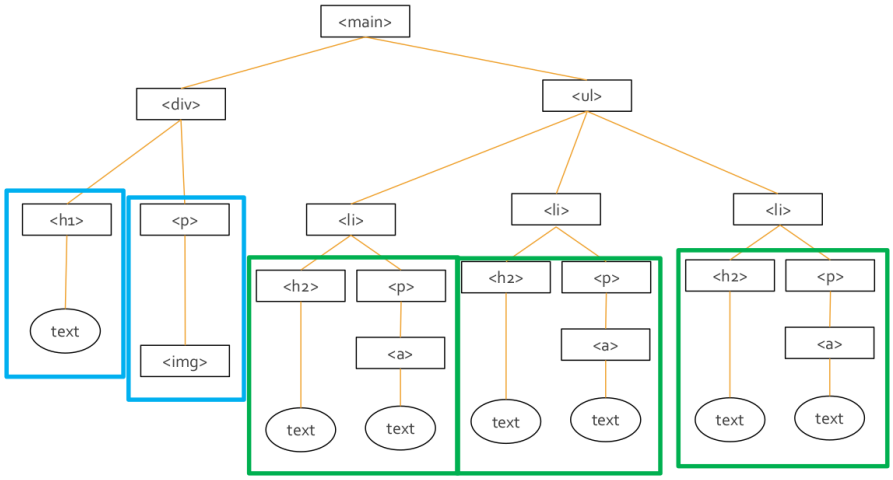

Figure 4.5: *Frequently occurring patterns identified from the DOM could be chosen as basic elements*

4.3.2 Number

There are several clustering techniques that can find the optimal number of clusters based on the given input. However, in order to enable web page skimming and scanning for the visually impaired, it is necessary to comply with the maximum number of concurrent oral stimuli a human-being can cognitively distinguish. Based on the assumption that each semantically coherent zone formed from the web page segmentation can be summarized and simultaneously synthesized into spatialized concurrent speech acts. Within this context, Guerreiro and Gonçalves (2015), Manishina et al. (2016) have shown that the cognitive load can range between five to seven different stimuli. Also it is important to keep in mind the fact that a varying the number of oral stimuli for different web pages is difficult to adapt to for a visually impaired

person (i.e) if a web page has 5 oral stimuli, the visually impaired person will expect to have 5 oral stimuli in the next web page as well as they are adapted to the fact that there is going to be 5 oral stimuli every time. Thus it is important to have a uniform the number of zones that can be identified for all web pages allowing them to be later spatialized into a TAG THUNDER. Keeping the facts in mind, the number of zones resulting from the WPS process has been fixed to 5 for the algorithms experimented in this dissertation. However, in order to be able to compare with already existing works, experiments with different number of clusters have been experimented for few web pages as well. (Chapter 9)

4.3.3 Position

As the number of seeds has been restricted to 5, it is necessary to place the seeds intelligently within the web page in order to achieve good results. The importance of positioning the seeds is proven by the first experiments conducted which are detailed in chapter 5. In chapter 5, the experiments are conducted by placing the seeds in a diagonal fashion (explained in chapter 5), the results show that position of the seeds influences the segmentation of a web page. Thus further experiments have been conducted using various techniques for positioning the seeds have been studied and detailed in chapter 6.

4.4 Features for clustering

For the purpose of clustering, certain features have to be considered in order to decide if the web element has to be put in a particular cluster or not. Based on the task at hand, the features considered should help in allowing the first glance of a web page. The features thus chosen are explained in this section.

4.4.1 Distance

As the distance is most important feature that needs to be considered for the formation of zones. There are various distance measures that can be used to calculate distance between the two web elements. A comparative study on the various distance measures used in clustering techniques are explained in Pandit et al.. For experiments in this dissertation, the euclidean distance between the data points are used as a distance measure.

For this purpose, there are 2 ways to measure the euclidean distance between two web elements - euclidean distance between the centers of the web elements or euclidean distance between the borders of the web elements. The distance between the centers is not the accurate as one basic element could be bigger than the other. On the other hand, the distance between the borders is the

most accurate distance as it represents the human perception of the visual distance between two rectangles (web elements) as in Figure 4.6. Thus euclidean distance between the borders is chosen to be used in the segmentation process.

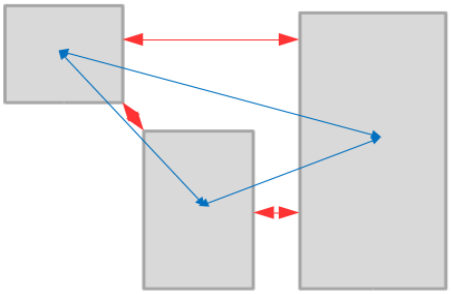

Figure 4.6: *Distance calculations - red arrows showing the border to border distance and the blue arrows showing the center to center distance*

The border to border distance (bbd_i^k) between a bbox (b_i) and a bbox (b_k). Let (x_1, y_1) and (x_2, y_2) be the top-left and bottom-right pixel coordinates of bbox (b_i) and (x'_1, y'_1) and (x'_2, y'_2) be the top-left and bottom-right pixel coordinates of the bbox (b_k), then

$$bbd_i^k = \begin{cases} 0 & \begin{cases} ((x'_1 \le x_1 \le x'_2) \vee (x'_1 \le x_2 \le x'_2) \vee (x_1 \le x'_1 \le x'_2 \le x_2)) \wedge \\ ((y'_1 \le y_1 \le y'_2) \vee (y'_1 \le y_1 \le y'_2) \vee (y_1 \le y'_1 \le y'_2 \le y_2)) \end{cases} \\ \sqrt{(x'_1 - x_2)^2 + (y'_1 - y_2)^2} & \begin{cases} (x_2 \le x'_1) \wedge (y_2 \le y'_1) \end{cases} \\ y'_1 - y_2 & \begin{cases} ((x'_1 \le x_1 \le x'_2) \vee (x'_1 \le x_2 \le x'_2) \vee (x_1 \le x'_1 \le x'_2 \le x_2)) \wedge \\ (y_2 \le y'_1) \end{cases} \\ \sqrt{(x_1 - x'_2)^2 + (y'_1 - y_2)^2} & \begin{cases} (x'_2 \le x_1) \wedge (y_2 \le y'_1) \end{cases} \\ x_1 - x'_2 & \begin{cases} (x'_2 \le x_1) \wedge \\ ((y'_1 \le y_1 \le y'_2) \vee (y'_1 \le y_2 \le y'_2) \vee (y_1 \le y'_1 \le y'_2 \le y_2)) \end{cases} \\ \sqrt{(x_1 - x'_2)^2 + (y_1 - y'_2)^2} & \begin{cases} (x'_2 \le x_1) \wedge (y'_2 \le y_1) \end{cases} \\ y_1 - y'_2 & \begin{cases} ((x'_1 \le x_1 \le x'_2) \vee (x'_1 \le x_2 \le x'_2) \vee (x_1 \le x'_1 \le x'_2 \le x_2)) \wedge \\ (y'_2 \le y_1) \end{cases} \\ \sqrt{(x'_1 - x_2)^2 + (y_1 - y'_2)^2} & \begin{cases} (x_2 \le x'_1) \wedge (y'_2 \le y_1) \end{cases} \\ x'_1 - x_2 & \begin{cases} (x_2 \le x'_1) \wedge \\ ((y'_1 \le y_1 \le y'_2) \vee (y'_1 \le y_2 \le y'_2) \vee (y_1 \le y'_1 \le y'_2 \le y_2)) \end{cases} \end{cases} \tag{4.1}$$

4.4.2 Alignment

It has been shown that aligned parts of a web page share similar layout structures, which can be used as a valuable cue for Web Page Segmentation Cai et al. (2003a), Yang and Shi (2007). Two web elements are said to be aligned if their horizontal or vertical margin lines coincide. If two web elements are aligned either in the horizontal and vertical axis, it is highly likely that they belong to the same zone.

4.4.3 Font similarities

Font styles are a part of the visual features that need to be considered for the task at hand. As the designer of the web page uses font styles to emphasis

connectivity between content and to highlight the importance of the content. Thus this feature plays an important role along with the distance and alignment features. The font styles such as font-style, font-weight, font-family etc. are taken into consideration for this feature. Two web elements with exactly the same font styles are likely to belong to the same zone as it could be the intention of the web page developer to use similar font styles to indicate relevance. However, even if one of the font styles is different it is necessary to look at other features described above. There could also be cases where a heading or a line break could be used to indicate separateness of the content but the font styles could remain the same.

There are ofcourse other visual cues that are available within the web page such as background-color, shadow effects, z-index etc. These sure are representatives of the developers idea to differentiate zones. However, they are quite difficult to manipulate. For example, a background-color of red could have several variations and this could make it difficult to differentiate between the variations. This makes it difficult to identify similarities between web elements as they could be very minimal, making the decision if they have to be put together or be separated a tough task. They could mean to belong in the same zone or they could mean to belong to different zones, depending on the difference in variations. Thus taking into account the difficulties in manipulating these sort of visual cues, for the initial experiments, the basic visual cues that clearly differentiate between web elements are taken into account.

4.5 Conclusion

Thus in the chapter it has been established that

- The technique that will be used for the segmentation of web pages for the task at hand is clustering.
- The features that will be used for this clustering are Distance, Alignment and font similarities.
- The web elements chosen as inputs for the clustering are the smallest basic block of visual elements
- The number of zones to be identified is 5
- Various positioning methods to place initial seeds will be studied along the course of the dissertation.

In the following chapters, the already well established K-means clustering technique MacQueen (1967) is explored Web Page Segmentation. This is followed by using a small change to the K-means algorithm leading to F-K-means. Later, a sort of hierarchical propagation clustering technique has been developed with the features in mind for the task at hand. These clustering techniques are later experimented with various positions of initial seeds and their results are evaluated.

Chapter 5

Algorithms

Based on the criteria in chapter 1, using clustering is the technique of choice for the task of segmentation in the TAG THUNDER project. This chapter is about the various clustering algorithm which have been experimented for the Web Page Segmentation. The two major algorithms experimented are K-means and Guided Expansion. These two algorithms are studied with various different variations. It is necessary to keep in mind that the number of clusters required (K) is 5, thus each cluster formed representing a zone.

5.1 K-means

The K-means MacQueen (1967) is a well-established algorithm, when the number of clusters(K) must be fixed in advance (K=5 for WPS for TAG THUNDER). However, some task-based adaptations are required. Firstly, for the task at hand, the data points used in the clustering process is the web elements chosen from each web page as described in chapter 4. Secondly, the assignment phase for this task is based on the shortest euclidean distance between the borders of two visual elements as discussed in chapter 4, Figure 4.6 on page 4.6. For the update step of the Kmeans algorithm, a simple averaging of the coordinates of the web elements within a cluster is done to find the center of the cluster. However, the cluster center thus formed might not actually correspond to a web element. This is called as a virtual centroid. But, averaging the features like alignment and font styles may however not be possible because the features in consideration cannot exactly be averaged. Also if the features were to be considered, there is a necessity to determine the weight of each features to be determined. All features may not be equally important and the degree of importance of each feature is not set. Thus for the purpose of initial experiments, only the border to border distance feature is used for clustering with Kmeans. Finally, the classical K-means relies on the random selection of initial seeds. However, this strategy does not adapt to this task for various reasons. Choosing random seeds does not allow for comparison between different algorithms. Also, it does not allow for comparison

between web pages as well because if the seeds are placed randomly it will not be possible to compare between web pages of the same web site. Thus the placing of the 5 initial seeds should be controlled for comparison and evaluation purposes. They are placed following the diagonal reading strategy for initial experiments, i.e. if a diagonal is drawn on the web page from top-left to bottom-right, two seeds are positioned on each extremities, one in the center and the two other ones between the extremities and the center of the diagonal as in Figure 5.1 on page 42. In figure 5.1 on page 42, the blocks represent the basic elements of the web page, the blue lines through the blocks represent the reading strategies and the red blocks indicate the chosen seeds. Finally, in the classical Kmeans the clusters formed changes and evolves at each step following the position of the centroid until the centroid of the clusters do not change anymore. However, within the task, the centroids could be a virtual web element i.e. the coordinates of the centroid might not correspond to an actual web element on the web page. Thus for this purpose, if the centroid of the any of the clusters formed in any of the steps is a virtual web element, then the closest actual web element to the virtual web element is chosen as the centroid for the clustering process.Thus the final algorithm is detailed as follows.

Input: The set of basic visual elements; K
Output: K clusters
Initialization: Select K centroid elements based on the reading strategy;
while *true* **do**
  Assign each visual element to its closest centroid based on *distance*(., .);
  Compute K new centroids as the gravity center of each cluster;
  if *centroids do not change* **then**
    break;
  end
end

Algorithm 1: K-means algorithm.

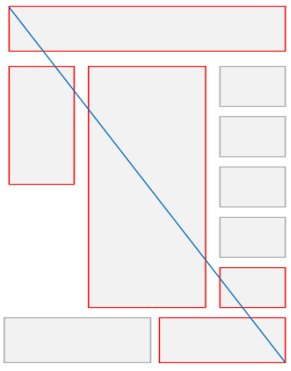

Figure 5.1: *Positioning the initial seeds in a diagonal fashion*

5.2 F-K-means

In the K-means presented in section 5.1, the assignment phase is exclusively based on the geometric distance between visual objects. F-K-means however is a variant of the K-means (Algorithm 1 on page 42). It takes into account the area covered by each basic element, the rationale being that the bigger elements are more likely to "absorb" the smaller elements than the contrary. So, if two basic elements are close to each other, their assignment function $force(b_1, b_2)$ will also depend on their differences of covered area as defined in equation 5.1 on page 43, where a_{b1} (resp. a_{b2}) is the area of the visual element b_1 (resp. b_2) and $distance(.,.)$ is the shortest euclidean distance between the borders of the basic elements. Thus the Force factor is necessarily a metaphor for the physical force of attraction between two physical bodies, the physical bodies referring to the web elements. The equation 5.1 on page 43 is thus necessarily the Newton's law of Gravitation for two physical bodies. However, the mass of the physical body is the area of the web elements in consideration and distance is the euclidean distance between their borders. Therefore, the F-K-means algorithm follows the exact same procedures as algorithm 1 (page 42), to the exception of the function used for the assignment step, which is the $force(.,.)$, i.e. the elements, which show the highest force to their centroids are selected.

$$force(b_1, b_2) = (a_{b1} * a_{b2})/(distance(b_1, b_2))^2 \tag{5.1}$$

5.3 Guided Expansion

The guided expansion (GE) algorithm can be thought of as a guided K-means, where the assignment process is made by local decisions. So, instead of assigning all basic elements to one or the other cluster in a single step like in K-means, only one basic element is assigned at a time to a cluster. This decision is controlled by a set of conditions that include the shortest euclidean distance between the borders of two elements, the alignment between elements (as explained in section 4.4.2), and their visual similarity in this particular order. The visual similarity $vsim(.,.)$ between two elements b_1 and b_2 is computed as in equation 5.4 (page 45) over their respective feature vectors $\overrightarrow{b_1}$ and $\overrightarrow{b_2}$ formed by their HTML attributes (i.e. font-color, font-weight, font-family and background-color). It is to be noted that as GE follows local decisions, unlike Kmans there is no need for updating cluster centers and thus solving the problem of having a virtual cluster center.

The GE starts with 5 web elements as seeds. The seeds are positioned in a diagonal fashion as represented in figure 5.1 on page 42 for initial experiments and other methods of positioning the initial seeds are studied in chapter 6. The expansion of the clusters take place one at a time following certain criteria as follows:

— Each seed is considered as a cluster and the web page is considered as list of web elements (W).
— A candidate set of web element(s) is formed for each cluster. For this purpose, the border to border distance is first taken into consideration. The border to border euclidean distance between the web elements and the elements of a cluster is calculated and the web element(s) with the minimum border to border distance is identified as the candidate web element(s) for that particular cluster. Note there are 5 sets of candidates, one for each cluster. The web elements of the candidate set with the minimum distance among the 5 candidate sets is chosen to be added to the corresponding cluster. This candidate set with the minimum border to border distance is the chosen set. If there are more than one candidate set with the same minimum distance to their corresponding cluster elements (in which case there are more than one chosen candidate set) or if there are more than one element in the chosen candidate set, all such candidate sets are considered for the further steps.
— The chosen candidate set(s) from the previous step are then checked for alignment. The web elements of each candidate set(s) are checked for alignment with their respective cluster elements. The web elements that are not aligned with the any of the respective cluster elements are removed from the candidate set. If any of the chosen candidate set becomes empty, it is disregarded as the chosen set.
— If there are still more than one chosen candidate set or if there are more than one element in the chosen candidate set, then the visual similarity between the elements of the candidate set(s) and their corresponding cluster is considered. If any web element from the candidate set does not share the same visual aspect with the any of the elements of their respective cluster elements, then they are removed from the set. If any of the chosen candidate set becomes empty, it is disregarded as the chosen set.
— If even after this feature check, there are still more than one chosen candidate set or if there are more than one element in the chosen candidate set, all the web elements within these sets are added to their corresponding cluster.
— The web elements(s) that have been added to one or the other cluster are removed from the list of web elements (W).
— It is important to notice that a cluster is a set of visual elements, except for the first step of the algorithm. So, when the distance and the visual similarity are computed between an element and its cluster candidate, this refers to the computation of each metric between the element and all the elements in the cluster. This situation is formalized in equations 5.2 and 5.3, where c_1 is the cluster candidate for b_1.
— The above steps are repeated until all web elements in W are clustered.

$$distance(b_1, c_1) = argmin_{b_i \in c_1} distance(b_1, b_i) \tag{5.2}$$

$$vsim(\overrightarrow{b_1}, c_1) = argmax_{b_i \in c_1} vsim(\overrightarrow{b_1}, \overrightarrow{b_i}) \tag{5.3}$$

$$vsim(\overrightarrow{b_1}, \overrightarrow{b_2}) = \sum_{i=1}^{|\overrightarrow{b_1}|} \mathbb{1}_{\overrightarrow{b_1^i} = \overrightarrow{b_2^i}} \tag{5.4}$$

The GE algorithm is presented as Algorithm 2.

5.4 F-Guided Expansion.

The Guided Expansion algorithm is extended to be able to use the $force(b_1, b_2)$ measure for segmentation. This algorithm is termed as F-GE (F-Guided Expansion).

The F-Guided Expansion (F-GE) is a variation of the Guided Expansion algorithm that has been presented in Algorithm 2 on page 48. This takes into account the area covered by each basic element. Thus, the first criterion to check between elements is the force of attraction between the basic elements, $force(b_1, b_2)$, as presented equation 5.1 on page 43, instead of the border-to-border geometric distance. Of course, this is followed by the checking of alignment between the basic elements and their visual similarities (equation 5.4 on page 45) between elements as in the GE algorithm (Algorithm 2 on page 48).

The F-GE starts with 5 web elements as seeds. The seeds are positioned in a diagonal fashion as represented in figure 5.1 on page 42 for initial experiments and other methods of positioning the initial seeds are studied in chapter 6. The expansion of the clusters take place one at a time following certain criteria as follows:

— Each seed is considered as a cluster and the web page is considered as list of web elements.

— A candidate set of web element(s) is formed for each cluster. For this purpose, the force measure as in equation 5.1 (page 43) is first taken into consideration. The force of attraction between the web elements and the elements of a cluster is calculated and the web element(s) with the maximum force of attraction is identified as the candidate web element(s) for that particular cluster. Note there are 5 sets of candidates, one for each cluster. The web elements of the candidate set with the maximum force of attraction among the 5 candidate sets is chosen to be added to the corresponding cluster. This candidate set with the maximum force of attraction is the chosen set. If there are more than one candidate set with the same maximum force of attraction to their corresponding cluster elements (in which case there are more than one chosen candidate set) or if there are more than one element in the chosen candidate set, all such candidate set(s) are considered for the further steps.

— The chosen candidate set(s) from the previous step are then checked

for alignment. The web elements of each candidate set(s) are checked for alignment with their respective cluster elements. The web elements that are not aligned with the any of the respective cluster elements are removed from the candidate set. If any of the chosen candidate set becomes empty, it is disregarded as the chosen set.

— If there are still more than one chosen candidate set or if there are more than one element in the chosen candidate set , then the visual similarity between the elements of the candidate set(s) and their corresponding cluster is considered. If any web element from the candidate set does not share the same visual aspect with the any of the elements of their respective cluster elements, then they are removed from the set. If any of the chosen candidate set becomes empty, it is disregarded as the chosen set.

— If even after this feature check, there are still more than one chosen candidate set or if there are more than one element in the chosen candidate set, all the web elements within these sets are added to their corresponding cluster.

— The web elements(s) that have been added to one or the other cluster are removed from the list of web elements (W).

— It is important to notice that a cluster is a set of visual elements, except for the first step of the algorithm. So, when the distance and the visual similarity are computed between an element and its cluster candidate, this refers to the computation of each metric between the element and all the elements in the cluster. This situation is formalized in equations 5.2 and 5.3, where c_1 is the cluster candidate for b_1.

— The above steps are repeated until all web elements in W are clustered.

5.5 Conclusion

The segmented web pages for the algorithms presented in this chapter are shown in figures 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9 and 5.10 (on pages 47, 47, 49, 49, 49, 50, 50, 50 and 5.10 respectively). The seeds have been placed in a diagonal fashion across the web page for initial experiments and other methods of positioning the initial seeds are studied in chapter 6 . It has to be noted that the positioning of seeds is independent of the algorithms. For the three algorithms, a quantitative evaluation was done with 3 experts for the two common indices in clustering, i.e. compactness and separateness Acharya et al. (2014). The procedure and the results of this evaluation are detailed in chapter 7. In general, it is noted that most algorithms evidence an horizontal segmentation strategy, i.e. vertical cluster are difficult to identify. Another issue concerning the selection of the seeds concerns the F-K-means algorithm. If some seed are associated to a small element, this cluster will hardly expand as the $force(.,.)$ metric tends to benefit larger visual elements. But at the extremities of the web pages where the initial seeds are selected, usually small elements are present. As a consequence, very small clusters are built for these seeds, while only the other three expand. This observation is confirmed by the quantitative results(presented in chapter 7) and thus explains the poor

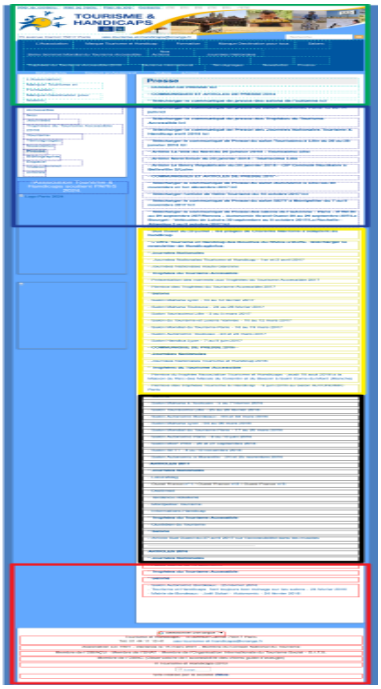

Figure 5.2: *K-means (Tourism web page)*

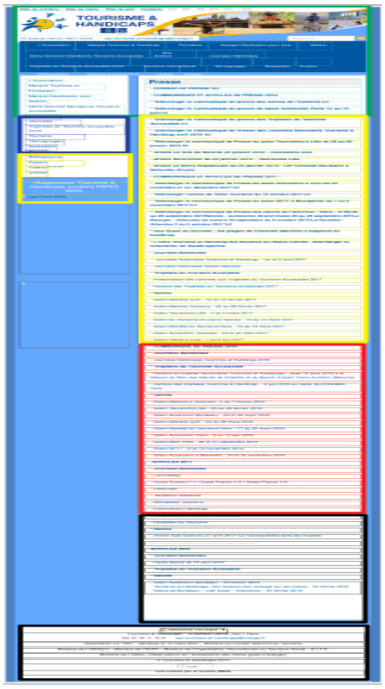

Figure 5.3: *F-K-means (Tourism web page)*

results of the F-K-means in terms of separateness. This is seen in Figure 5.3 (page 47) as in the zone marked in green and in Figure 5.11 (page 5.11) as in the zone marked in red. In fact, these issues are due to the *apriori* selection of the seeds, and in particular to the seeds positioned at the top-left and bottom-right extremities. Indeed, as these positions usually refer to headers and footers, which are horizontally shaped, the clustering process tends follow this general direction for the other three seeds. The quantitative results also prove that the algorithms are highly sensitive to the seeds position. Thus the diagonal reading strategy used to place seeds should be improved to not separate similar blocks at the very beginning of the process.

This leads to the next direction of research which is to experiment with different ways to position the initial seeds for all the three algorithms. Seeds position could be determined experimentally using eye-tracking to test if positioning seeds on points of interest, extract from a sight reading strategy, increase the segmentation quality. Seeds could also be positioned using a pre-clustering technique to identify areas where the probability to form a zone is high.

Input: The set of basic visual elements; K
Output: K clusters
Initialization: Select K centroid elements (clusters) based on the reading strategy;
while *there are unclustered elements* **do**
  Select each closest element to every cluster using $distance(.,.)$;
  Order these elements by the minimum distance to their candidate cluster;
  Remove all elements that do not evidence the smallest distance for possible assignment;
  if *there are no ties* **then**
    Assign the closest element overall to its cluster;
  end
  else if *there are ties* **then**
    Check whether the elements are vertically or horizontally aligned with at least one element of their cluster;
    Order elements by alignment;
    if *there are no ties AND one aligned element* **then**
      Assign the aligned element to its cluster;
    end
    else if *there are ties OR no aligned element* **then**
      Compute the visual similarity between the elements and their cluster using $vsim(.,.)$;
      Order elements by the maximum visual similarity to their cluster;
      Remove all elements that do not evidence the highest visual similarity for possible assignment;
      if *there are no ties* **then**
        Assign the most visually similar element to its cluster;
      end
      else if *there are ties* **then**
        Assign all elements to their cluster;
      end
    end
  end
end

Algorithm 2: Guided expansion algorithm.

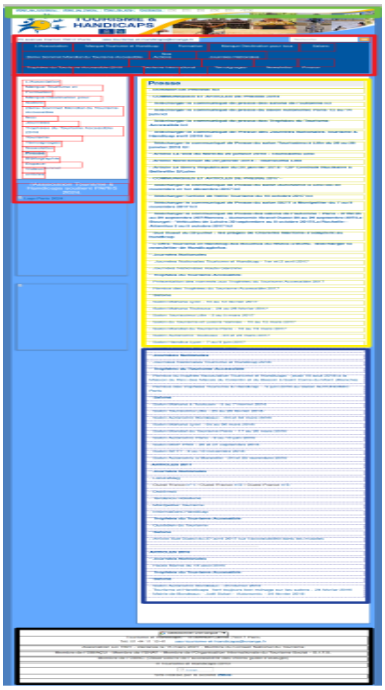

Figure 5.4: *Guided expansion (Tourism web page)*

Figure 5.5: *K-means (E-commerce web page)*

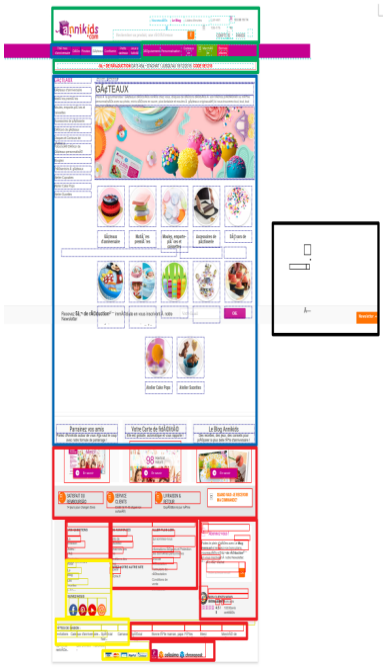

Figure 5.6: *F-K-means (E-commerce web page)*

Figure 5.7: *Guided expansion (E-commerce web page)*

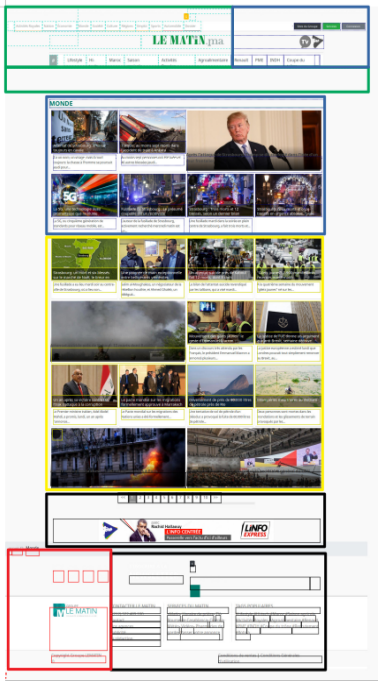

Figure 5.8: *K-means (News web page)*

Figure 5.9: *F-K-means (News web page)*

Figure 5.10: *Guided expansion (News web page)*

Figure 5.11: *A snippet of a web page segmented by F-K-means*

Chapter 6

Positioning of seeds

6.1 Various ways to position seeds

As concluded in chapter 5, the position of initial seeds plays a very important role in the segmentation process. The positioning of seeds in a diagonal fashion does not allow the formation of horizontal clusters. Thus in this chapter, the various ways to position the seeds are investigated for all the algorithms presented in chapter 5.

First experiments are done by placing the seeds in a way that it follows the various reading strategies used on the web. There have been several studies using eye-tracking to monitor the way a user scans the web page. Pernice (2017) proposes a study on the "F" reading strategy that users use while reading the Web. The observations of Pernice (2017) are summarized as follows: (1) users first read in an horizontal movement, usually across the upper part of the content area. This initial element forms the F top bar; (2) next, users move down the page a bit and then read across in a second horizontal movement that typically covers a shorter area than the previous movement, which forms the F lower bar; (3) finally, users scan the left side of the content in a vertical movement, thus forming the F stem. In particular, the authors Pernice (2017) show heat maps, which evidence the F pattern of reading on the Web. Another strategy is studied by Babich (2017). They propose a study, which shows that users read the Web in a "Z" shape fashion when the web pages are not centered around its text content. The summary of Babich (2017) is as follows: (1) first, users scan from the top left to the top right, forming an horizontal line; (2) next, down and to the left side of the page, creating a diagonal line; (3) last, back across to the right again, forming a second horizontal line. Thus the idea is to use these reading strategies to position the seeds.

Second experiments are done using a pre-clustering technique for the positioning of seeds. First, a simple clustering method is used for positioning the seeds and later the QT clustering method is used to identify areas where probability of forming a zone is high. There have been experiments done with 2 different variations of positioning the seeds using the QT algorithm which is detailed in

this chapter.

6.2 Using reading strategies for positioning of seeds

The diagonal method places the seeds on a diagonal virtually drawn on the web page from top-left to bottom-right. i.e., two seeds are positioned on each extremities, another one in the center and the two other ones between the extremities and the center of the diagonal as in Figure 5.1(page 42). In this chapter, the seeds will be positioned in a "F" and "Z" fashion motivated by the studies of Nielsen (2006), Pernice (2017) [1] and Babich (2017) [2]. These strategies are shown in figure 6.1(page 53). In figure 6.1(page 53), the blocks represent the basic elements of the web page, the red lines through the blocks represent the reading strategies and the green blocks indicate the chosen seeds. For both the "F" and "Z" reading strategies, there are two seeds that needs to be identified on the extremities of the top line. However, if there are no elements on the extremities, then two elements along the same line (top line) is chosen. This same strategy is used for the bottom line of the "Z" letter. If there is only one element in the top/bottom line, then the closest element based on the border to border distance is chosen to play the seed. Figures

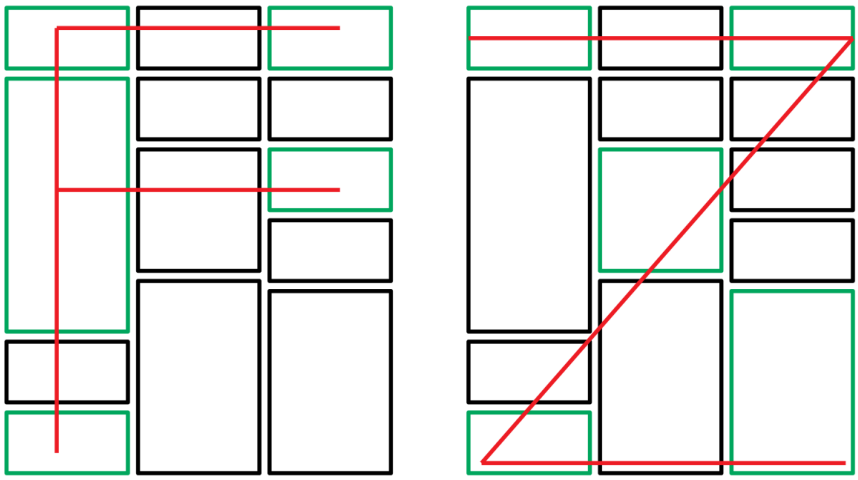

Figure 6.1: *F (left) and Z (right) strategies to position the seeds.*

6.2, 6.3 and 6.4 (pages 61, 61 and 61 respectively) show web pages of different domains segmented using Guided Expansion (GE) Algorithm by positioning the seeds in a F fashion. As there are two seeds positioned on the top, each following one end of the top horizontal line in the letter "F", the header has been put into two different zones. In figures 6.2 and 6.3 (pages 61 and 61 respectively), the header is put in both the green and yellow zones. Figures

1. Nielsen (2006), Pernice (2017) are studies from the Nielsen Norman Group, a private corporations.
2. Babich (2017) is a study from UX Planet.

6.5, 6.6 and 6.7 (pages 62, 62 and 62 respectively) show web pages segmented using the Guided Expansion (GE) algorithm by positioning the seeds in a "Z" fashion. As in the letter "F", the letter "Z" has two horizontal lines - one on top and the other on bottom. Thus there are two seeds placed on either sides of the header and either sides of the footer. Thus this tends to separate the header in two different zones ad footer in two different zones. Thus this kind of positioning tends to split basic elements that need to be in a the same zone because of two seeds being placed in the same horizontal line. Using this type of reading strategies maybe efficient for skimming and scanning of the web page for people with sight, but for segmenting a web page for visually impaired people this strategy is not very efficient.

6.3 Using pre-clustering techniques

Instead of using a single basic element as a seed, there is a possibility to use a cluster of closely located basic elements as seeds for the segmentation process. Thus in this context, two variations of positioning the seeds were developed.

6.3.1 Simple Clustering

The first method follows a simple pre clustering method as follows.

- **Step:1** The web page is considered as a list of web elements.
- **Step:2** A threshold for the search has to be set. The threshold set is one tenth of the maximum border to border euclidean distance possible on a web page.
- **Step:3** The first web element is considered
- **Step:4** A cluster is formed for the web element under consideration, using the set threshold and the euclidean border to border distance.
- **Step:5** The elements in this cluster are then deleted from the list of web elements.
- **Step:6** The first web element in the updated list of web elements is considered next.
- **Step:7** Steps 4,5 and 6 are repeated for the element until consideration, until 5 clusters are formed.
- **Step:8** The 5 clusters thus formed are used as seeds for the Guided Expansion Algorithm 2.

This method of positioning seeds for the Guided Expansion Algorithm is detailed in Algorithm 3 on page 58. A threshold is necessary to be set so as to restrict the formation of clusters in the pre-clustering step and thus allowing left out web elements to be segmented using the GE algorithm allowing the consideration of alignment and font similarities for the segmentation process. While using this method to position initial seeds, most web elements are zoned in the initial clustering process to find seeds, this reduces the use of the GE algorithm drastically and thereby decreases computational time drastically. However, on the flip side as the use of GE algorithm is minimized, the

alignment and font features of a page are not completely considered for the segmentation process. The other disadvantage is the fact that the method to position seeds does not perform an extensive search with all web elements as the first remaining web element is always considered to form the cluster, however, the probability for a good cluster might not necessarily be in this region. Thus it is important to make an extensive search to find the best clusters to be used as seeds.

6.3.2 QT clustering

In this method, the already established QT clustering technique Xin and Jiawei (2016) is used. However, there are some task-based adaptations are required. It has to be noted that the QT algorithm is chosen as it does not require any initial positioning of seeds and is able to do an extensive search on all the basic elements of the web page to form the best possible clusters.

- **Step:1** The web page is considered as a list of web elements.
- **Step:2** A threshold for the search has to be set. There have been various thresholds that are experimented in this dissertation. The thresholds are set as fractions of the maximum border to border euclidean distance possible on a web page.
- **Step:3** Following this, a cluster is formed for every single web element on the web page. (i.e.) Consider every web element on a web page in sequence, the euclidean border to border distance is calculated between the web element under consideration and all the other web elements on the web page. If the calculated distance is within the set threshold, the web elements are added to the cluster of the web element under consideration.
- **Step:4** The biggest cluster thus formed is taken as the seed and all the web elements in this biggest cluster are removed from the list of the web elements.
- **Step:5** Step 3 and 4 are repeated with the updated list of web elements, until 5 seeds are identified.
- **Step:6** If 5 seeds are possible to be identified because of the set threshold (a threshold analysis is presented in section 6.3.4), then the remaining seeds are randomly placed among the remaining web elements.

The five clusters thus identified are used as initial seeds for the Guided Expansion(GE) segmentation. Again, this algorithm uses a threshold that restricts the formation of clusters allowing remaining web elements to be segmented using the GE segmentation algorithm. This is essential as GE takes into account the alignment and font similarities between basic elements before segmenting them while the pre clustering using QT clustering technique is used to find initial seeds and thus uses only the euclidean distance between the borders of the basic elements. This algorithm is detailed in Algorithm 4 on page 59.

However, as stated already, this algorithm follows a threshold. While the threshold is small, the necessary number of 5 clusters cannot be formed on certain pages. Thus to place the remaining seeds are placed randomly from

the remaining unclustered web elements. However while using this strategy to position remaining seeds, there is a possibility for two seeds to be positioned next to each other thus restricting the growth of zones while using the GE algorithm. Also, once the QT clustering algorithm forms clusters to be used as seeds, there are very few web elements from which to choose the remaining seeds thus making it highly likely that the seeds are placed next to each other. This can been seen in Figures 6.8 and 6.9 (pages 63 and 63 respectively) which is a tourism web page. The Figure 6.9 on page 63 shows that there was a possibility to form only one cluster because of a small threshold (one fiftieth of the maximum border to border distance) and thus results in a huge green zone that expands rapidly with the GE algorithm that follows, however the position of the remaining seeds creates small zones (red, yellow, blue and black zones) as the growth of these zones have been restricted by the position of the seeds very close to each other. While in Figure 6.8 on page 63, the QT clustering allows the formation of 5 clusters because of the threshold that is used (one tenth of the maximum border to border distance) and thus allows better separation of the zones. This situation where using smaller thresholds generates one huge zone and many smaller zones can be seen also in Figures 6.11 and 6.13 (pages 64 and 65) which are E-commerce and news web pages respectively.

6.3.3 Variation of QT pre clustering

As stated in section 6.3.2, the threshold plays an important role in positioning the seeds while using a QT clustering method. The technique proposed to position the remaining seeds in cases where 5 clusters could not be formed is inefficient as it tends to place seeds next to each other affecting the growth of zones. Thus in order to solve this problem of positioning remaining seeds, another strategy is proposed. This is done as follows:

- **Step:1** The web page is considered as a list of web elements.
- **Step:2** A threshold for the search has to be set. There have been various thresholds that are experimented in this dissertation. The thresholds are set as fractions of the maximum border to border euclidean distance possible on a web page.
- **Step:3** Following this, a cluster is formed for every single web element on the web page. (i.e.) Consider every web element on a web page in sequence, the euclidean border to border distance is calculated between the web element under consideration and all the other web elements on the web page. If the calculated distance is within the set threshold, the web elements are added to the cluster of the web element under consideration.
- **Step:4** The centroid of the biggest cluster thus formed is chosen as the seed and all the web elements in this biggest cluster are removed from the list of the web elements.
- **Step:5** Steps 3 and 4 are repeated with the updated list of web elements, until 5 seeds are identified.
- **Step:6** If 5 seeds are possible to be identified because of the set threshold (a threshold analysis is presented in section 6.3.4), then the element

with the maximum average euclidean border to border distance between the already chosen seeds is taken as a seed.

The seeds thus identifies are used for the Guided Expansion(GE) segmentation algorithm. This strategy to position the remaining seeds helps to position the initial seeds in places where the probability to form zones is the highest and at the same time allows the Guided Expansion (GE) Algorithm to form efficient shape of zones. This strategy is detailed in Algorithm 5 on page 60.

The segmentation using Algorithm 5 (page 60) is shown in Figures 6.14, 6.15, 6.16, 6.17, 6.18 and 6.19 (pages 66, 66, 67, 67, 68 and 68 respectively)for the biggest and smallest thresholds. As can be seen from the figures where the threshold is one fiftieth of the maximum border to border distance, though the formation of the necessary number of 5 clusters is not possible, the seeds are positioned with maximum distance between each other. Thus this allows the expansion of zones without restrictions as opposed to algorithm 4 (page 59).

6.3.4 Threshold Analysis

Placing seeds using the simple pre-clustering technique (Algorithm 3) and QT clustering technique (Algorithms 4 and 5) uses a threshold, thus this section presents a short analysis of various possible thresholds. An analysis of the number of clusters formed using various thresholds is shown in table 6.1 (page 57). The table 6.1 ((page 57)) shows that while the threshold decreases from 1/10 to 1/50 of the maximum border to border distance between the basic elements, the number of pages that are able to form the 5 clusters decreases and also there are certain web pages in which there are no clusters formed. Note that the analysis shown in table 6.1 (page 57) are performed for 50 web pages (20 tourism web pages, 12 e-Commerce web pages and 18 news). This gives a necessity for using Algorithm 4 (page 59) or Algorithm 5 (page 60) to position more than one remaining seeds. As the threshold decreases, the necessity to place more than one seeds arises. As the algorithm 5 (page 60), proposes a more efficient way of positioning the seeds, this algorithm tends to produce better results for decreasing thresholds.

	Number of clusters					
	5	4	3	2	1	0
1/10	37	9	4	0	0	0
1/15	34	10	6	0	0	0
1/20	21	20	5	3	1	0
1/25	19	16	9	5	1	0
1/30	15	14	10	8	3	0
1/35	12	16	12	6	4	0
1/40	10	14	14	5	7	0
1/45	9	13	14	4	8	2
1/50	9	10	15	5	8	3

Table 6.1: Threshold analysis

Input: The ordered list of basic visual elements; K
Output: K clusters
Threshold $\leftarrow$ max(distance between two visual elements)/10;
$K \leftarrow 1$;
while $K \leq 5$ **do**
  Choose the first basic element, as the parent element, from the ordered list;
  Remove the first element from the ordered list;
  for *each visual element in the ordered list* **do**
    Calculate $dist(.,.)$ between the visual element and the parent element;
    if $dist(.,.) < Threshold$ **then**
      Add the visual element in the cluster of the parent element;
      Remove the visual element from the ordered list;
    end
  end
  $K \leftarrow K + 1$;
end
while *the ordered list is not empty* **do**
  Select each closest element to every cluster using $dist(.,.)$;
  Order these elements by the minimum distance to their candidate cluster;
  Remove all elements that do not evidence the smallest distance for possible assignment;
  if *there are no ties* **then**
    Assign the closest element overall to its cluster;
  end
  else if *there are ties* **then**
    Check whether the elements are vertically or horizontally aligned with at least one element of their cluster;
    Order elements by alignment;
    if *there are no ties AND one aligned element* **then**
      Assign the aligned element to its cluster;
    end
    else if *there are ties OR no aligned element* **then**
      Order elements by the maximum visual similarity to their cluster;
      Remove all elements that do not evidence the highest visual similarity for possible assignment;
      if *there are no ties* **then**
        Assign the most visually similar element to its cluster;
      end
      else if *there are ties* **then**
        Assign all elements to their cluster;
      end
    end
  end
end

Algorithm 3: Guided Expansion with simple pre-clustering.

Input: The list of basic visual elements; K
Output: K clusters
Threshold $\leftarrow max(dist(.,.))/10$;
$K \leftarrow 1$;
while $K \leq 5$ **do**
  for *each visual element in the list* **do**
    Calculate $dist(.,.)$ between the visual element and the other elements ;
    if $dist(.,.) < Threshold$ **then**
      Add the visual element in the cluster of the current element;
    end
    choose the biggest cluster as cluster K;
    Delete the elements of cluster K from the list of basic visual elements;
  end
  $K \leftarrow K + 1$;
end
while *the list of visual elements is not empty* **do**
  Select each closest element to every cluster using $dist(.,.)$;
  Order these elements by the minimum distance to their candidate cluster;
  Remove all elements that do not evidence the smallest distance for possible assignment;
  if *there are no ties* **then**
    Assign the closest element overall to its cluster;
  end
  else if *there are ties* **then**
    Check whether the elements are vertically or horizontally aligned with at least one element of their cluster;
    Order elements by alignment;
    if *there are no ties AND one aligned element* **then**
      Assign the aligned element to its cluster;
    end
    else if *there are ties OR no aligned element* **then**
      Order elements by the maximum visual similarity to their cluster;
      Remove all elements that do not evidence the highest visual similarity for possible assignment;
      if *there are no ties* **then**
        Assign the most visually similar element to its cluster;
      end
      else if *there are ties* **then**
        Assign all elements to their cluster;
      end
    end
  end
end

Algorithm 4: Guided Expansion with QT pre-clustering

```
Input: The list of basic visual elements; K
Output: K clusters
Threshold ← max(dist(.,.))/10;
K ← 1;
while K ≤ 5 do
    for each visual element in the list do
        Calculate dist(.,.) between the visual element and the other
         elements ;
        if dist(.,.) < Threshold then
            Add the visual element in the cluster of the current element;
        end
        choose the biggest cluster as cluster K;
        Delete the elements of cluster K from the list of basic visual
         elements;
    end
    K ← K + 1;
end
for All K clusters do
    if cluster K has only one element then
        seed K ← element that maximizes the average dist(.,.) to the
         centroid of the other clusters;
    end
    else if cluster K has many elements then
        seed K ← centroid of cluster K;
    end
end
while the list of visual elements is not empty do
    Select each closest element to every cluster using dist(.,.);
    Order these elements by the minimum distance to their candidate
     cluster;
    Remove all elements that do not evidence the smallest distance for
     possible assignment;
    if there are no ties then
        Assign the closest element overall to its cluster;
    end
    else if there are ties then
        Check whether the elements are vertically or horizontally aligned
         with at least one element of their cluster;
        Order elements by alignment;
        if there are no ties AND one aligned element then
            Assign the aligned element to its cluster;
        end
        else if there are ties OR no aligned element then
            Order elements by the maximum visual similarity to their
             cluster;
            Remove all elements that do not evidence the highest visual
             similarity for possible assignment;
            if there are no ties then
                Assign the most visually similar element to its cluster;
            end
            else if there are ties then
                Assign all elements to their cluster;
            end
        end
    end
end
```

Algorithm 5: Guided Expansion with QT and maximum average distance.

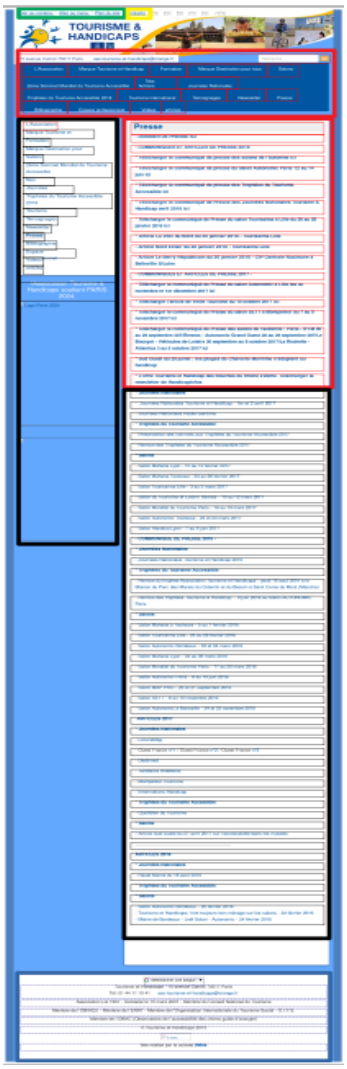

Figure 6.2: *Segmentation using by positioning the seeds in a F fashion (Tourism web page)*

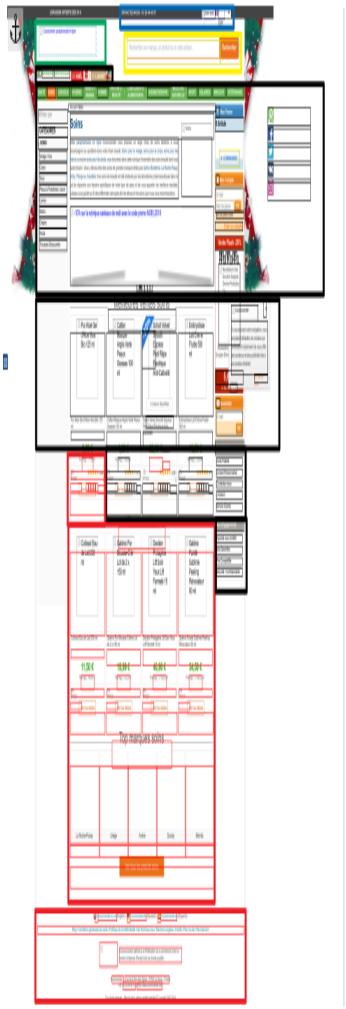

Figure 6.3: *Segmentation using by positioning the seeds in a F fashion (E-commerce web page)*

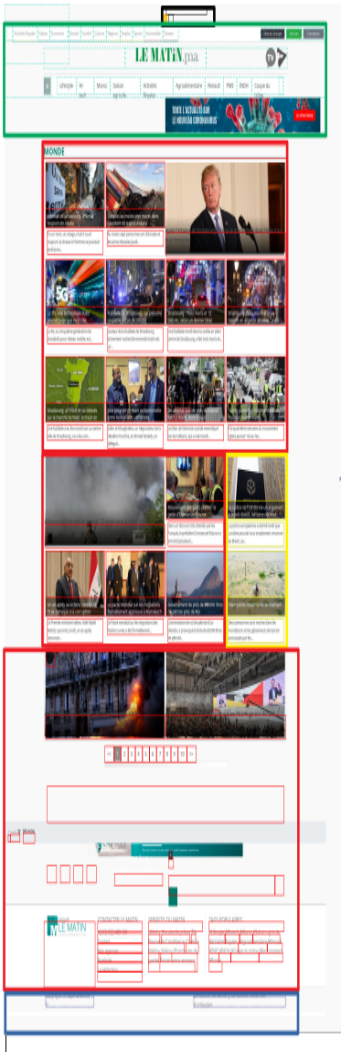

Figure 6.4: *Segmentation using by positioning the seeds in a F fashion (News web page)*

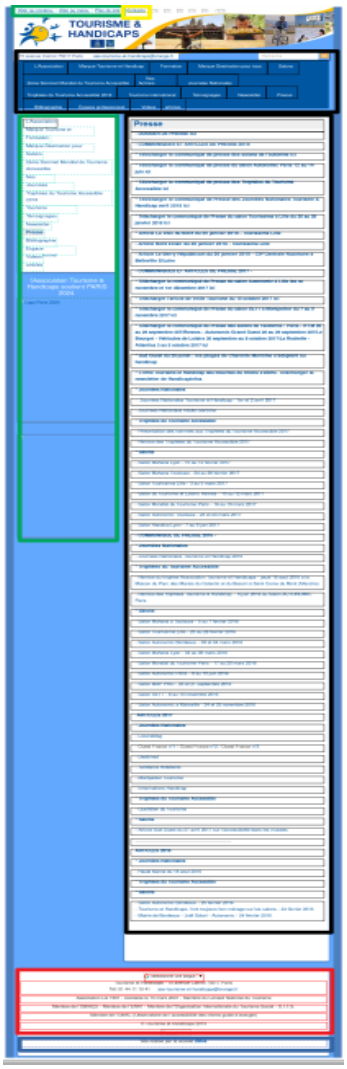

Figure 6.5: *Segmentation using by positioning the seeds in a Z fashion (Tourism web page)*

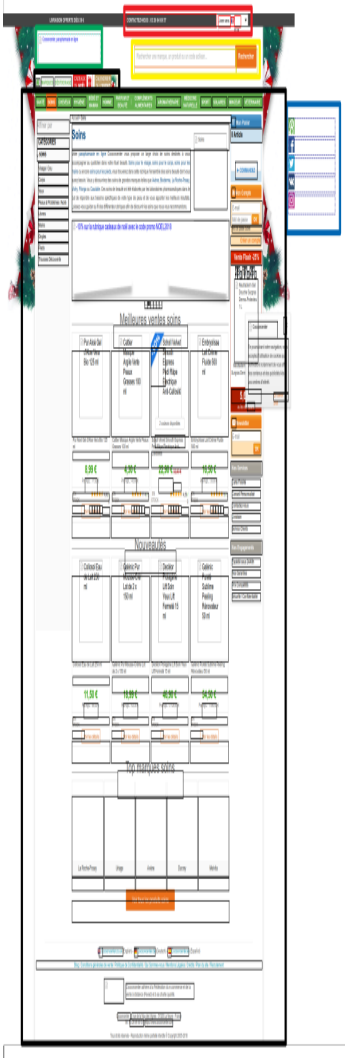

Figure 6.6: *Segmentation using by positioning the seeds in a Z fashion (E-commerce web page)*

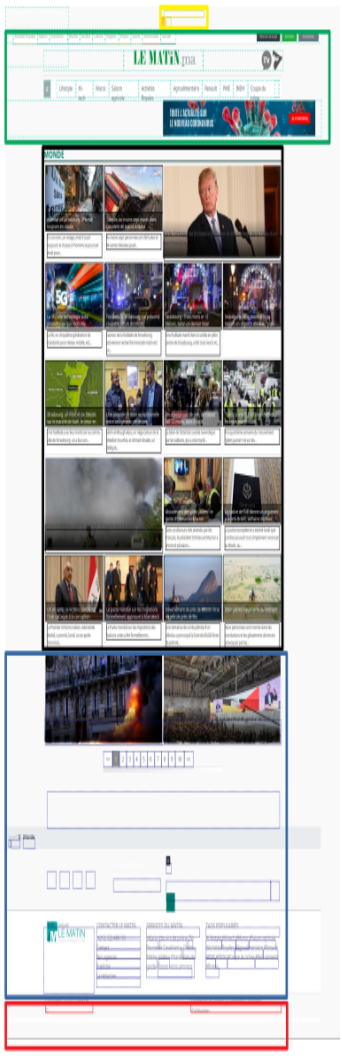

Figure 6.7: *Segmentation using by positioning the seeds in a Z fashion (News web page)*

Figure 6.8: *Segmentation with Algorithm 4 with threshold as one tenth of the maximum border to border distance(Tourism web page)*

Figure 6.9: *Segmentation with Algorithm 4 with threshold as one fiftieth of the maximum border to border distance(Tourism web page)*

Figure 6.10: *Segmentation with Algorithm 4 with threshold as one tenth of the maximum border to border distance(E-commerce web page)*

Figure 6.11: *Segmentation with Algorithm 4 with threshold as one fiftieth of the maximum border to border distance(E-commerce web page)*

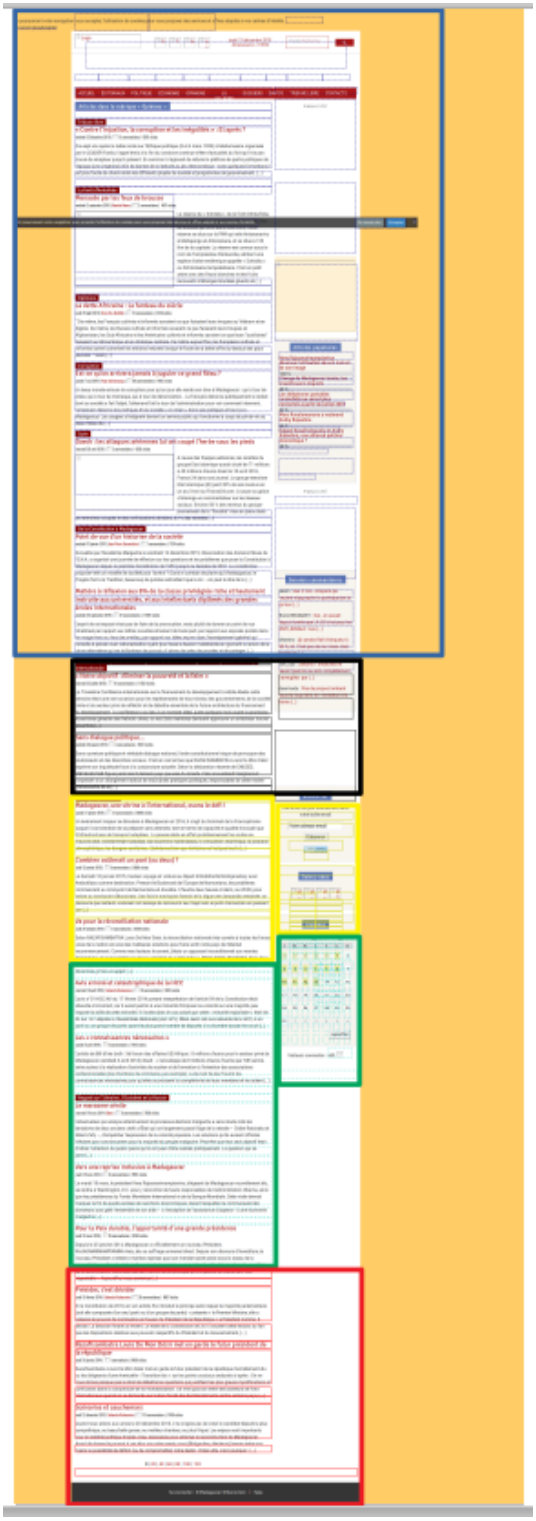

Figure 6.12: *Segmentation with Algorithm 4 with threshold as one tenth of the maximum border to border distance (News web page)*

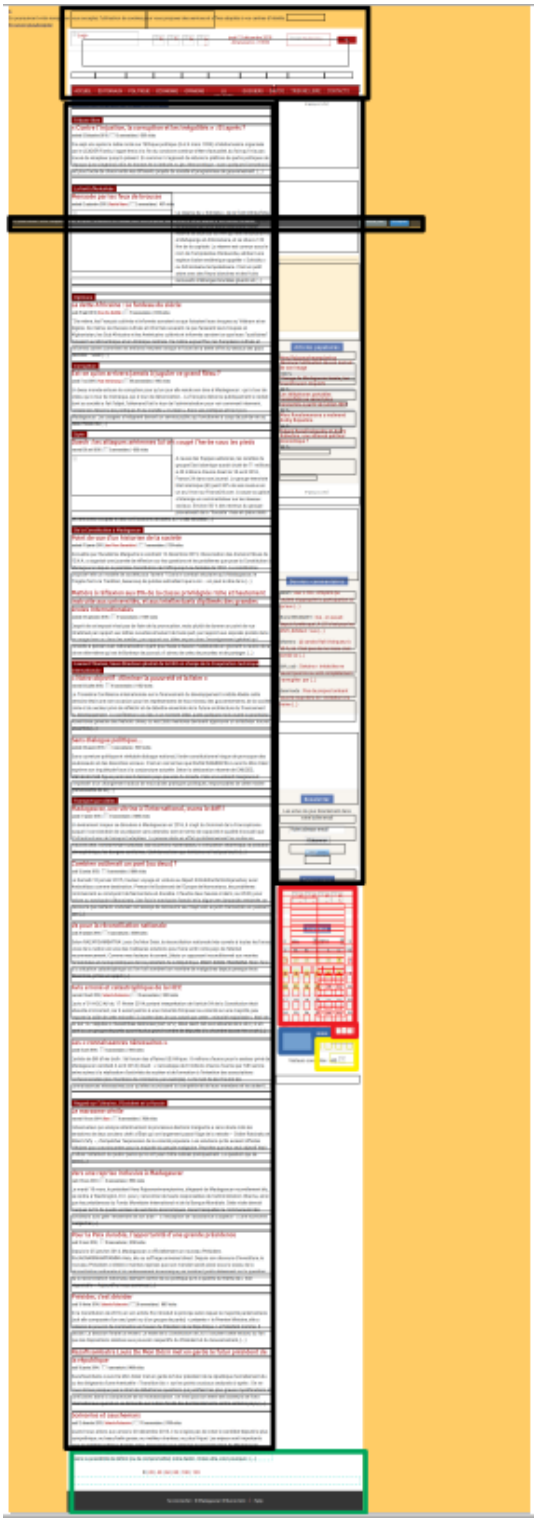

Figure 6.13: *Segmentation with Algorithm 4 with threshold as one fiftieth of the maximum border to border distance(News web page)*

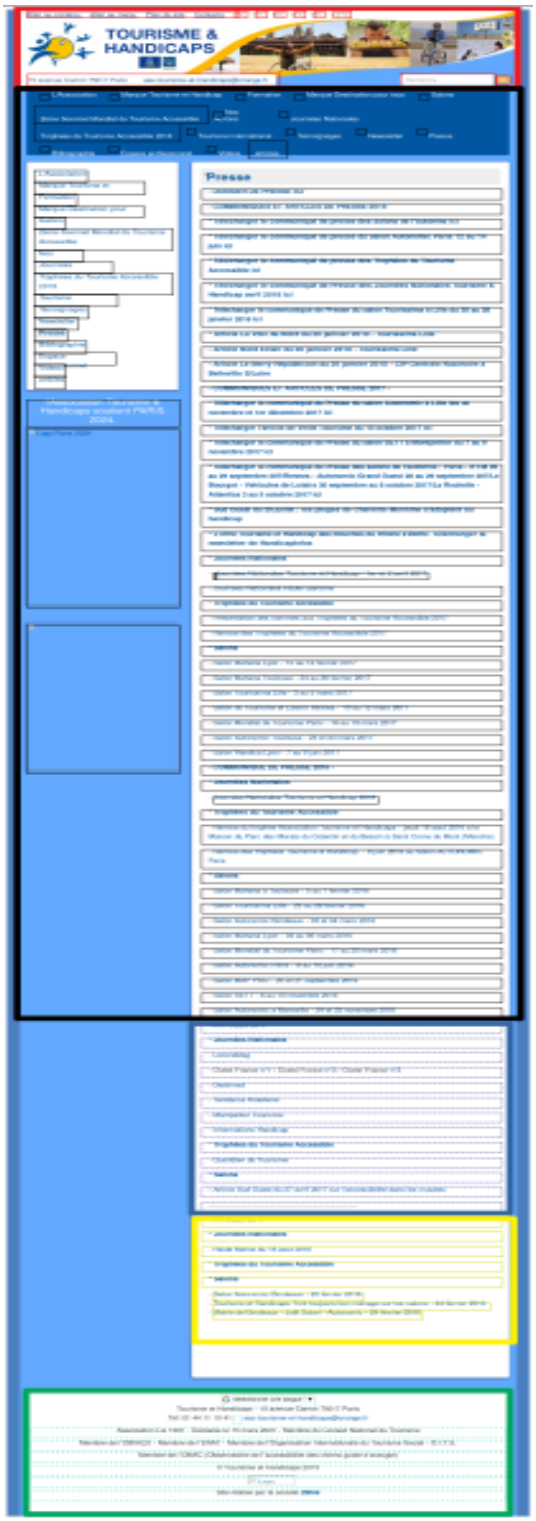

Figure 6.14: *Segmentation using Algorithm 5 with a threshold of one tenth of the maximum border to border distance (Tourism web page)*

Figure 6.15: *Segmentation using Algorithm 5 with a threshold of one fiftieth of the maximum border to border distance (Tourism web page)*

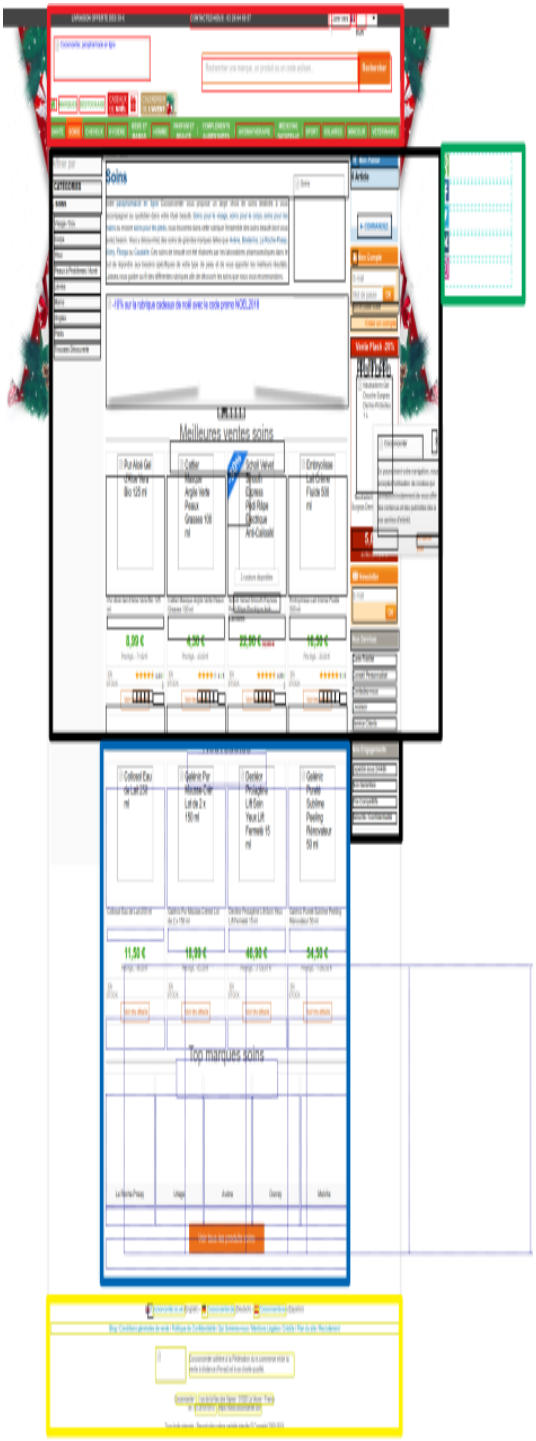

Figure 6.16: *Segmentation using Algorithm 5 with a threshold of one tenth of the maximum border to border distance (E-commerce web page)*

Figure 6.17: *Segmentation using Algorithm 5 with a threshold of one fiftieth of the maximum border to border distance (E-commerce web page)*

Figure 6.18: *Segmentation using Algorithm 5 with a threshold of one tenth of the maximum border to border distance (News web page)*

Figure 6.19: *Segmentation using Algorithm 5 with a threshold of one fiftieth of the maximum border to border distance (News web page)*

Part III

Evaluations

Chapter 7

Manual Evaluation

In this chapter, the algorithms discussed in the previous methods will be evaluated. Firstly, the algorithms K-means, F-K-means and Guided Expansion using a diagonal reading strategy are subjected to quantitative evaluation which proves the necessity to study positioning of seeds with the clustering context (as discussed in chapter 6). Followed by this, each algorithm is evaluated for the usual cluster metrics used for all clustering techniques.

7.1 Qualitative evaluations

For the purpose of evaluation a qualitative approach is taken. In this approach 3 human experts were asked to evaluate two common indices in clustering, i.e. compactness and separateness Acharya et al. (2014). Each expert must produce his/her own segmentation which is considered as the ground truth. The ground truth that the experts have created are used to help with the development of the cluster metrics (presented in section 7.2). The experts are then asked to evaluate the algorithms (Kmeans, F-Kmeans and GE with diagonally positioned seeds) in terms of compactness and separateness. The ground truth previously produced by them helps with this evaluation, they are asked to compare the segmentation produced by the algorithms and the ground truth that they have created previously and assign a score for the two metrics - compactness and separateness. Compactness is defined at the cluster level and evaluates how many of the elements within a cluster belong to a same cluster in the (individual) ground truth. Separateness is defined at the web page level and evaluates how much the proposed segmentation guarantees the separability between clusters when compared to the expert ground truth segmentation. For each web page, the expert must evaluate how much, on average, elements that should belong to the same cluster following the (individual) ground truth are separated in different clusters. Each expert must give a mark ranging from 0 (unacceptable), 1 (bad), 2 (passable), 3 (good) and 4 (perfect). Based on this protocol, the three algorithms (K-means, F-Kmeans and guided Expansion(GE)) have been tested on a total of 53 web

pages from 3 domains: Tourism (23 web pages), E-Commerce (12 web pages) and News (18 web pages). The overall results are presented in table 7.1 on page 71 and an example from each category(Tourism, E-commerce and News) of the expert manual segmentation is illustrated in figures 7.1, 7.2 and 7.3 on pages 73, 73 and 73 respectively.

		Compactness		Separateness		*GSS*	
		Avg.	$\pm\sigma$	Avg.	$\pm\sigma$	Avg.	$\pm\sigma$
K-M	E1	2.42	1.16	1.15	0.64	0.30	0.12
	E2	1.90	0.87	1.20	0.60	0.26	0.11
	E3	3.10	0.74	0.70	0.80	0.29	0.15
F-K-M	E1	2.43	1.46	0.62	0.57	0.23	0.09
	E2	1.83	1.15	0.40	0.50	0.16	0.07
	E3	3.05	1.22	0.30	0.50	0.21	0.095
GE	E1	**2.89**	1.24	**1.62**	0.93	**0.42**	0.19
	E2	**2.41**	0.81	**1.90**	0.90	**0.41**	0.16
	E3	**3.40**	0.68	**1.50**	0.90	**0.44**	0.18

Table 7.1: Overall results for K-means (K-ME.), F-K-means (F-K-ME.) and Guided expansion (GE).

Guided Expansion(GE) algorithm shows the best numbers (table 7.1 on page 71) both in terms of compactness and separateness for the 3 human experts. Compactness receives average values between passable and good, separateness receives much lower values, between passable and bad. This finding is transverse to all three algorithms. This shows that finding coherent zones that match human expectations is a hard task, while building internally semantically coherent zones is easier. Also, numbers (table 7.1 on page 71) show differences between K-means and F-K-means. Both algorithms show similar compactness, but the F-K-means evidences worst results for separateness. This result can easily be explained as the F-K-means tends to create unbalanced clusters, that are either very small or rather big. This is confirmed by the higher standard deviation in terms of compactness for F-K-means than for K-means, signifying that F-K-means tends to create very compact clusters (but small) and uncondensed big ones, thus penalizing separateness.

To statistically confirm these results, a global segmentation score (GSS) taking into account both compactness and separateness (equation 7.1 on page 71). In equation 7.1 on page 71, the evaluation scale refers to the scoring scale of separateness (*separat*) and compactness (*compact*), i.e. in our case 5 (0 to 4 grade). Results in table 7.1 on page 71 show that GE evidences statistically superior results to both K-means and F-K-means, and that K-means provides statistically higher results than F-K-means, for all three experts in all tested situations, to exception for Expert 3 when comparing K-means and F-K-means. This proves that Guided Expansion(GE) is the best algorithm amongst the three presented algorithm, followed by K-means.

$$GSS = \frac{(1 + separat) \times (1 + \overline{compact})}{|\text{evaluation scale}|^2} \tag{7.1}$$

The superiority of the GE algorithm is probably due to the introduction of the alignment constraint and font similarities inside the expansion process. Thus allowing the local assignment at each step of the algorithm which allows more fine-grained decisions when compared to both K-means and F-K-means,

which produce global assignments. However, the alignment constraint is more difficult to encode in a K-means family algorithm as alignment is a local feature.

While K-means based algorithm allows revision of clusters during the process, the GE algorithm is highly sensitive to the seeds position. Thus the diagonal reading strategy used to place seeds should be improved to not separate similar blocks at the very beginning of the process. This qualitative evaluation led to more experiments on the various ways to position the seeds as presented in chapter 6 (starting on page 52).

Conclusion: Based on the scores of manual evaluations, the GE algorithm outperforms the Kmeans and F-Kmeans in terms of the compactness and separateness. All three experts have evaluated similarly for the two metrics - compactness and separateness, which is validated by the Global Segmentation Score (GSS) as well. This evaluation also proved that the Kmeans, F-Kmeans and GE algorithms are sensitive to the initial positioning of seeds, thus leading to further experiments on the positioning of initial seeds.

It is to be noted that the task of creating a ground truth and scoring each algorithm for compactness and separateness is time consuming. It should also be noted that there are 31 variations of the algorithms that need to be evaluated, including the variations in the positioning of the seeds and the thresholds for when seeds are placed using the QT clustering strategy (S1 and S2). Thus scoring all variations of segmentation would be a tedious task. Also, the number of web pages scored using the above mentioned method is 50 and scoring more web pages manually will need more time and more human resources.

Thus in order to minimize time in scoring and to enable scoring without a ground truth, there is the necessity to develop a method that is based on the scoring strategies used by the experts but will evaluate the segmentations automatically and without a ground truth. The experts have been interviewed to understand why and when they penalized a segmentation. This has been the inspiration to develop the metrics that will be presented in chapter 8. The inspirations behind each of the metrics lies within the way the experts scored the web pages for compactness and separateness. This is explained in detail in chapter 8 and these metrics have been used to evaluate 900 web pages (explained in detail in chapter 8 from page 98).

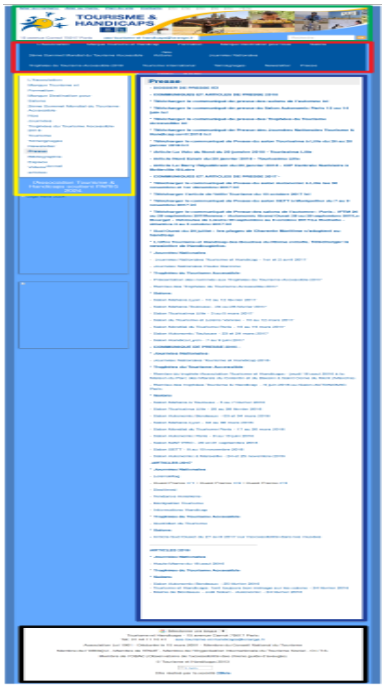

Figure 7.1: *Manual Segmentation from one of the experts (Tourism web page)*

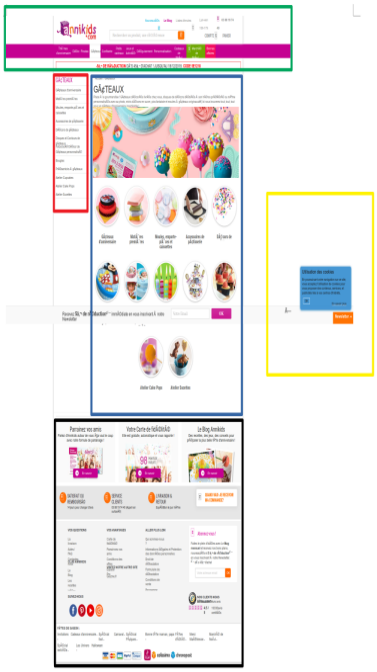

Figure 7.2: *Manual Segmentation from one of the experts (E-commerce web page)*

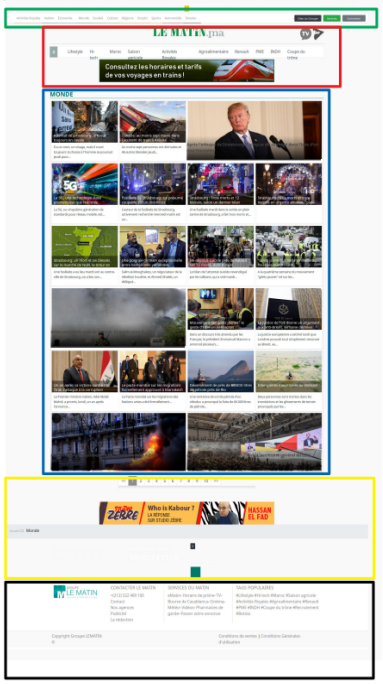

Figure 7.3: *Manual Segmentation from one of the experts (News web page)*

7.2 Cluster Metrics

The cluster metrics evaluated include Precision, recall, F1 score, ARI, Jaccard and F&M score.

Precision is defined as the number of true positives (T_p) over the number of true positives plus the number of false positives (F_p). It measures the fraction of pairs correctly put in the same cluster as in the ground truth.

$$Precision = T_p/(T_p + F_p) \tag{7.2}$$

Recall is defined as the number of true positives (T_p) over the number of true positives plus the number of false negatives (F_n). Recall is the sensitivity measure. It is the fraction of the total amount of relevant instances that were actually retrieved.

$$Recall = T_p/(T_p + F_n) \tag{7.3}$$

F1 score is the harmonic mean of the precision and recall. While **B3F1** is the weighted mean of the precision and recall.

$$F1score = 2((Precision * Recall)/(Precision + Recall)) \tag{7.4}$$

Rand Index(RI) computes a similarity measure between two clustering by considering all pairs of samples and counting pairs that are assigned in the same or different clusters in the predicted and true clustering. It is the measure of the accuracy of the clusterings (between the algorithms and the ground truth). The adjusted Rand index is the corrected-for-chance version of the Rand index. This sort of correction establishes a baseline by using the expected similarity of all pair-wise comparisons between two clusterings.

$$RandIndex(RI) = (T_p + T_n)/(T_p + F_p + F_n + T_n) \tag{7.5}$$

Jaccard Index is a similarity measure between finite sample sets. It compares the members between the two sets (one from the clustering algorithm and the ground truth) to see which members are shared and which are distinct. It is defined as the size of the intersection divided by the size of the union of the sample sets. Let A be the set of elements in cluster 1 with algorithm A and B be the set of elements in cluster 1 with algorithm B. The Jaccard Index is thus defined as:

$$JaccardIndex = |A \cap B|/|A \cup B| \tag{7.6}$$

F&M score (Fowlkes–Mallows index) computes the similarity between the clusters returned by the clustering algorithm and the benchmark clustering and is based on the pairwise approach to calculate the True Positives,True

Negatives,False Positives and False Negatives.

$$F\&Mscore = \sqrt{[T_{\mathrm{p}}/(T_{\mathrm{p}} + F_{\mathrm{p}})] * [T_{\mathrm{p}}/(T_{\mathrm{p}} + F_{\mathrm{n}})]} \tag{7.7}$$

7.3 Evaluations using cluster metrics

The table 7.2 on page 77 shows the cluster metrics for all the algorithm presented in the previous chapters. They are calculated over 50 web pages - 20 tourism web pages, 12 e-Commerce web pages and 18 news web pages. At first a ground truth is created, the ground truth created earlier by 3 experts (detailed in section 7.1 on page 70) is used here. The task in hand, segmenting web page to allow skimming and scanning for visually impaired people, has been explained to the experts. They are asked to segment web pages based on the first glance into 5 zones based on the visual features of the web page. The segmentation should also maintain coherency between the zones. The segmentation thus produced by the experts forms the ground truth. The 50 web pages that are segmented using the various algorithms presented in other chapters are then compared with this ground truth to be evaluated for the cluster metrics - B3F1 score, Precision, Recall, ARI, Jaccard and F&M score.

The algorithms evaluated are as follows:

- K-means diagonal - The K-means algorithm with K=5 and using the diagonal reading strategy as presented in Algorithm 1 on page 42.
- K-means F strategy - The K-means algorithm with K=5 and using the "F" reading strategy as explained in section 6.2 from page 53.
- K-means Z strategy - The K-means algorithm with K=5 and using the "Z" reading strategy as explained in section 6.2 from page 53.
- F-K-means diagonal - The K-means algorithm with K=5, with the Force measure to cluster web elements together and using the diagonal reading strategy as presented in section 5.2 on page 43.
- F-K-means F strategy - The K-means algorithm with K=5, with the Force measure to cluster web elements together and using the "F" reading strategy as explained in section 6.2 from page 53.
- F-K-means Z strategy - The K-means algorithm with K=5, with the Force measure to cluster web elements together and using the "Z" reading strategy as explained in section 6.2 from page 53.
- GE diagonal - The Guided Expansion Algorithm using the diagonal reading strategy as presented in Algorithm 2 on page 48.
- GE F strategy - The Guided Expansion Algorithm using the "F" reading strategy as explained in section 6.2 from page 53.
- GE Z strategy - The Guided Expansion Algorithm using the "Z" reading strategy as explained in section 6.2 from page 53.
- GE Force diagonal - The Guided Expansion Algorithm with the Force measure in the place of Euclidean distance to cluster web elements together and using the diagonal reading strategy as presented in section 5.4 on page 45.

— GE Force F Strategy - The Guided Expansion Algorithm with the Force measure in the place of Euclidean distance to cluster web elements together and using the "F" reading strategy as explained in section 6.2 from page 53.
— GE Force Z Strategy - The Guided Expansion Algorithm with the Force measure in the place of Euclidean distance to cluster web elements together and using the "Z" reading strategy as explained in section 6.2 from page 53.
— S1 - Guided Expansion Algorithm using the clusters formed by QT clustering technique as seeds as in Algorithm 4 on page 59.
— GE Pre Cluster - Guided Expansion using the clusters formed by simple pre clustering technique to place seeds as in Algorithm 3 on page 58.
— S2 - Guided Expansion Algorithm using the centroid of the clusters formed by QT clustering technique as seeds and placing the remaining seeds using the maximum average distance as in Algorithm 5 on page 60.

The fractions such as 1/10, 1/15, 1/20 etc are the thresholds set for the QT clustering algorithm. The threshold is a fraction of the maximum distance between web elements in a web page.

Table 7.2 (page Table 77) show that Guided Expansion by placing the seeds in a diagonal fashion gives the best segmentation result when compared to the ground truth. However, it can also be observed that the same kind of results can be obtained while using the Guided Expansion Algorithm using the centroids of the clusters formed from QT clustering as seeds and positioning the remaining seeds with the maximum average distance (S2, Algorithm 5 on page 60). It can also be seen that using the force measure does not yield expected results.

	B3F1 Avg.	Precision Avg	Recall Avg	ARI Avg	Jaccard Avg.	F and M Avg.
K-means diagonal	0.63	0.70	0.57	0.40	0.41	0.58
Kmeans F strategy	0.61	0.68	0.55	0.37	0.39	0.57
Kmeans Z strategy	0.62	0.68	0.58	0.40	0.42	0.59
F-K-means diagonal	0.58	0.62	0.57	0.29	0.35	0.5
F-K-means F strategy	0.57	0.60	0.58	0.28	0.36	0.53
F-K-means Z strategy	0.59	0.59	0.60	0.29	0.37	0.54
GE diagonal	0.69	0.73	0.67	0.47	0.48	0.65
GE F strategy	0.67	0.63	0.73	0.37	0.45	0.62
GE Z strategy	0.67	0.66	0.77	0.47	0.52	0.69
GE Force Diagonal	0.59	0.52	0.75	0.17	0.37	0.55
GE Force F Strategy	0.58	0.48	0.80	0.16	0.37	0.57
GE Force Z Strategy	0.60	0.49	0.81	0.20	0.40	0.59
S1 1/10	0.63	0.60	0.68	0.29	0.39	0.57
S1 1/15	0.63	0.57	0.73	0.29	0.41	0.59
S1 1/20	0.62	0.54	0.77	0.27	0.41	0.59
S1 1/25	0.62	0.54	0.76	0.25	0.41	0.59
S1 1/30	0.61	0.51	0.79	0.24	0.41	0.59
S1 1/35	0.61	0.52	0.79	0.24	0.40	0.59
S1 1/40	0.61	0.51	0.79	0.23	0.40	0.59
S1 1/45	0.61	0.51	0.80	0.24	0.41	0.60
S1 1/50	0.61	0.51	0.81	0.23	0.41	0.60
GE Pre cluster	0.63	0.69	0.59	0.37	0.40	0.57
S2 1/10	0.68	0.63	0.76	0.36	0.45	0.63
S2 1/15	0.65	0.58	0.77	0.31	0.43	0.61
S2 1/20	0.65	0.58	0.79	0.32	0.44	0.62
S2 1/25	0.66	0.59	0.79	0.34	0.45	0.63
S2 1/30	0.66	0.59	0.79	0.34	0.45	0.63
S2 1/35	0.67	0.58	0.80	0.35	0.46	0.64
S2 1/40	0.66	0.58	0.79	0.34	0.45	0.63
S2 1/45	0.67	0.58	0.81	0.35	0.46	0.65
S2 1/50	0.67	0.58	0.82	0.35	0.46	0.65

Table 7.2: Cluster Metrics

K-means vs Guided Expansion: It should be noted that K-means algorithm produces lower results (0.63 for B3F1 score, 0.70 for Precision and 0.57 for recall with a diagonal reading strategy) when compared with the Guided Expansion(GE) algorithm (0.69 for B3F1 score, 0.73 for Precision and 0.67 for recall with a diagonal reading strategy). This is due to the fact that GE uses takes into account several criteria before producing a segmentation. Firstly, the Guided Expansion Algorithm considers the border to border distance between the web elements. Secondly, while there are several elements with the same minimum distance, the algorithm considers the x or y axis alignment between the web elements. Thirdly, while there are several web elements with the same minimum distance and if the web elements are aligned by the x or y axis to the seeds, then the font similarities such as font style, font size, font color etc between the web elements. Thus at each iteration, there are only a few web elements that satisfy all these constraints that are added to a zone. On the contrary, while using K-means, only the distance/force measure is consid-

ered for the segmentation process. And at each iterations all web elements are clustered and re-clustered until a stable clusters are formed where each cluster is a zone. Thus due to the constraints encoded within the GE algorithm, this algorithm performs better than the K-means algorithm.

Force measure vs Distance measure:

It should be also noted that the force measure does not perform as expected. Kmeans with the distance measure and diagonally positioned seeds have a B3F1 score value of 0.63, a precision of 0.70 and a recall of 0.57 while Kmeans with the force measure and diagonally positioned seeds has a B3F1 score of 0.58, a precision of 0.62 and recall of 0.57. The GE with distance measure with diagonally positioned seeds has a B3F1 score of 0.69, a precision of 0.73 and a recall of 0.67 while the GE with force measure and diagonally positioned seeds has a B3F1 score 0.59, a precision of 0.52 and a recall of 0.75. In all cases, the distance measure performs better than the force measure. The force measure clusters web elements together based on the force measure mentioned in equation 5.1 on page 43. This equation takes into account the areas of the web elements into consideration. Thus if one of the seeds placed, using any of the strategies, falls on a web element with very a small area, it fails to attract other web elements and thus forming small zones. This case worsens when there are several seeds placed in small web elements. Thus this creates many small zones and one huge zone, which is not what the task requires. Thus this produces low values for the cluster metrics. However, when the euclidean distance measure is used to cluster web elements, irrespective of the size of the seeds, the zones continue to expand thus forming a comparable segmentation to the ground truth. This phenomenon is independent of the way the seeds are positioned.

F and Z reading strategies: It is also important to note that the performance of zoning decreases while positioning the seeds with the "F" and "Z" reading strategy. Kmeans with "F" reading strategy has a B3F1 score of 0.61, a precision of 0.68 and a recall of 0.55 while the positioning of seeds with the "Z" reading strategy has a B3F1 score of 0.62, a precision of 0.68 and a recall of 0.58. With respect to the GE algorithms, GE with "F" reading strategy has a B3F1 score of 0.67, a precision of 0.63 and a recall of 0.73, when the "Z" reading strategy is used, a B3F1 score of 0.67, a precision of 0.66 and a recall of 0.77 is achieved. Though this is not a huge variation from using the diagonal reading strategy (Kmeans diagonal with B3F1 score of 0.63, GE diagonal with a B3F1 score of 0.69), it shows that diagonal positioning of seeds is more efficient for the 50 web pages that are evaluated for the cluster metrics in table 7.2 on page 77. This is because of the structure of the letters "F" and "Z". There are two seeds placed on the header in case of the letter "F" and two seeds placed on the header and two on the footer in the case of the letter "Z". This causes the header to form two separate zones in the case of the letter "F" while in case of the letter "Z", the header is placed in two separate zones and the footer in two separate zones causing the formation of a huge zone in the middle. Thus this sort of segmentation does not compare well with the ground truth established by the experts causing low evaluation results for the cluster

metrics. It should be noted that he "F" and "Z" reading strategies work well for skimming/scanning a web page as the user skims/scans along the lines of the letters "F" and "Z". However, this is not exactly what is represented by the method of positioning the seeds. The seeds are positions in the corners of the letters "F" and "Z" and the lines of the letters "F" and "Z" are not used for the seeds. This is one of the reason for this sort of positioning the initial seeds to not work as expected.

S1: The results for Guided Expansion(GE) with seeds positioned using the clusters formed from the QT clustering, described in Algorithm 4 on page 59are referenced as "S1" with the thresholds as "1/10", "1/15", "1/20", "1/25", "1/30", "1/35", "1/40", "1/45", "1/50". The classical cluster metrics are presented in table 7.2 on page 77. In general, bigger Thresholds evidences better results when compared with smaller thresholds, but it is necessary to note that while the precision decreases with decreasing threshold, the recall increases with decreases threshold. A threshold of 1/10 has a B3F1 score of 0.63, a precision of 0.60 and recall of 0.68 while a threshold of 1/50 has a B3F1 score of 0.61, a precision of 0.51 and a recall of 0.81. In all cases, S1 evidences better results in terms of B3F1 score when compared with the Kmeans with different reading strategies, F-Kmeans with different reading strategies and GE Force with different reading strategies. But the B3F1score of GE with various reading strategies still scores better than S1. Indeed, the precision of S1 with all different thresholds is lower than the previously presented algorithms. The recall of S1 with the lowest threshold matches highest recall produced by the previous algorithms (F-GE Z strategy) - a recall of 0.81 has been achieved for F-GE Z strategy and S1 with a threshold of 1/50. The F and M score and Jaccard of S1 is comparable with the other algorithms. However, the ARI scores are comparable with the Kmeans Force with various reading strategies and Guided Expansion Force Algorithms with various reading strategies but the ARI of Kmeans and Guided Expansion are much better than the ARI of S1 irrespective of the threshold. In short, the bigger thresholds such as "1/10" and "1/15" evidences better results for the cluster metrics (table 7.2 on page 77). The results also show that as the threshold decreases, the results of segmentation evidences decreasing results. This is could be explained as follows: as the threshold decreases, there are several pages that could not form 5 clusters as explained in table 6.1 on page 57 and thus the remaining seeds are placed using the strategy mentioned in section 6.3.2 (page 55). While doing so, there are several cases where the seeds are positioned next to each other and thus restricts the growth of a particular zone as seen in figures 6.8, 6.9, 6.10, 6.11, 6.12 and 6.13 on pages 63, 63, 64, 64, 65 and 65 respectively. As seen in figures 6.9, 6.11 and 6.13 on pages 63, 64 and 65 respectively, when the threshold set is 1/50 of the maximum border to border distance, the possibility of forming 5 clusters is low and thus the remaining seeds are placed next to each other thus restricting the growth of a zone. It should also be noted that while using smaller thresholds, the size of the cluster formed from the QT clustering technique is small. This can be witnessed with the red, blue, yellow and black zones in figure 6.9 on page 63. In figure 6.11 and 6.13 on pages 64 and 65 respectively, it can be seen in the red, blue and yellow zones.

S2: The results for Guided Expansion(GE) with seeds positioned using the centroids of the clusters from the QT clustering technique(Algorithm 5 on page 60) are referenced as "S2" with the various threshold as fraction of the maximum border to border distance such as "1/10", "1/15", "1/20", "1/25", "1/30", "1/35", "1/40", "1/45", "1/50". As seen in table 7.2 on page 77, this strategy (S2) to tackle incomplete clustering in situations where the necessary number of of seeds could not be obtained from the QT clustering method gives better results than S1. S2 evidences better results in terms of all cluster metrics - B3F1 score, Precision, Recall, ARI, Jaccard and F&M scores. Again it should be noted that higher thresholds evidences better results than the lower thresholds. Indeed, as the threshold decreases from 1/10 to 1/50(in both the techniques), the precision decreases but the recall increases (With a threshold of 1/10 the B3F1 score of 0.68, a precision of 0.63 and a recall of 0.76 has been achieved while with a threshold of 1/50, a B3F1 score of 0.67, a precision of 0.58 and a recall of 0.82 has been achieved). This shows that while decreasing the threshold, the number of false positives increases and thus the precision decreases. On the other hand, the recall increases with the decreasing threshold by increasing the number of true positives. The precision decreases slowly with S2 when compared with S1 and the recall increases slowly with S2 when compared with S1 and thus the precision and recall stabilizes quickly with S2. Notably, S2 produces better results than S1. This is because while the possibility of forming 5 clusters is low when using the QT clustering algorithm, then the remaining seeds are placed using the average maximum distance between the formed clusters and the remaining web elements. Thus, this allows the positioning of seeds far away from each other allowing the comfortable expansion of zones. This can be seen in figures 6.14, 6.15, 6.16, 6.17, 6.18 and 6.19 on pages 66, 66, 67, 67, 68 and 68 respectively. It can been seen that in these figures the zones are bigger and more balanced when compared with the figures 6.8, 6.9, 6.10, 6.11, 6.12 and 6.13 on pages 63, 63, 64, 64, 65 and 65 respectively, because of the positioning of the seeds. While the threshold set is small, though the size of the cluster formed from the QT clustering method is small, as the centroid is taken as seed, this allows freedom for expansion of the zones based on the criteria taken into account by GE. This goes on to prove how important the positioning of the initial seeds is, while using clustering techniques for segmentation.

In short, S2 is a better segmentation algorithm than S1 and gives comparable results to the GE with the diagonal reading strategy.

7.4 Box Plots

Box Plots are used to display data based on a five-number summary. The five numbers used to build this plot are the minimum, the maximum, the sample median, and the first and third quartiles.

— **Minimum:** the lowest data point excluding any outliers.
— **Maximum:** the largest data point excluding any outliers.
— **Median (Q2 / 50th Percentile):** the middle value of the dataset.

— **First quartile (Q1 / 25th Percentile):** is also known as the lower quartile q_n (0.25) and is the middle value between the smallest number (not the minimum) and the median of the dataset.
— **Third quartile (Q3 / 75th Percentile):** is also known as the upper quartile q_n (0.75) and is the middle value between the largest number (not the maximum) and the median of the dataset

Box Plots for the *K*means, F*K*means and Guided Expansion(GE)

Figures 7.4, 7.5, 7.6, 7.7, 7.8, 7.9 and 7.10 on pages 83, 84, 84, 85, 86, 87 and 87 respective, show the box plots of the cluster metrics presented in table 7.2 for the *K*means, *K*means Force, GE and GE Force with all the reading strategies. The minimum and maximum values are the bottom and top points of the line on either sides of the box. The 25th percentile is the bottom line of the box and the 75th percentile is the top line of the box. The 50th percentile is represented using the thick line inside the box.

B3F1score: The B3F1 score for the algorithms of *K*means, *K*means Force, GE, GE Force algorithm with the various reading strategies are represented in figure 7.4. The 50th percentile shows the median, which indicates the value shown by the maximum number of web pages. In the case of B3F1 score, the median is around 0.7 for both GE with seeds positioned in the diagonal fashion and GE with seeds positioned in the "Z" fashion. However, GE with seeds in the diagonal fashion achieves the highest maximum B3F1 score (the line on top of the box) than any of the other algorithms. Also it should be noted that that GE with seeds in the diagonal fashion has less outliers than the other algorithms and also the value of B3F1 score of the outlier is much higher than any other algorithm. It should also be noted that the B3F1 score for algorithms using the Force measures are considerable lower than the once using the distance measure. Thus it can be concluded that considering the algorithms presented in figure 7.4 on page 83, based on the B3F1 cluster metric GE with seeds positioned in a diagonal fashion performs better than the other algorithms under consideration.

Fscore: The Fscore for the algorithms of *K*means, *K*means Force, GE, GE Force algorithm with the various reading strategies are represented in figure 7.5 on page 84. As mentioned in the previous paragraph on B3F1 score, 7.5 on page 84 confirms that using a force measure for clustering is less efficient than using the distance measure for reasons stated in section 7.3. The 50th and 75th percentile of the GE with seeds positioned in a diagonal fashion witnesses higher values when compared with the other algorithms under discussion. Though the minimum value achieved by the GE with diagonally positioned seeds is lower than the GE with seeds positioned in the "Z" fashion, the highest value witnessed is higher than the GE with seeds positioned in the "Z" fashion. Thus it could be said that both GE with diagonally positioned seeds and GE with seeds positioned in the "Z" fashion perform in a comparable manner in terms of the Fscore.

Precision: The Precision for the algorithms of *K*means, *K*means Force, GE, GE Force algorithm with the various reading strategies are represented in figure 7.6 on page 84. It should be noted that the highest precision is achieved using the GE with diagonally placed seeds. It can also be seen that *K*means with diagonally placed seeds performs slightly better than the GE with seeds placed in the "F" and "Z" fashion, when considering the all the parameters of a box plot(maximum, minimum, 1st, 2nd and 3rd quartile). Again it should be noted that using the distance measures performs better than using force measure.

Recall: The Recall for the algorithms of *K*means, *K*means Force, GE, GE Force algorithm with the various reading strategies are represented in figure 7.7 on page 85. While the recall cluster metric is considered, it should be noted that using force measure gives better recall than using the distance measure. The recall values for the *K*means strategies are much lower than the GE strategies. However, it should be noted that the recall value is higher while positioning the seeds in the "F" and "Z" fashion both for *K*means and GE algorithms. The recall cluster metric measures the compactness of a cluster. As explained in section 7.3 on page 75, while the seeds are placed on a web element with a small area, it fails to attract other elements thus creating small, compact zones - in some cases zones with only one web element in it. This increases the recall value for algorithms using the force measure. Thus in terms of the recall cluster metric, GE force algorithm outperforms other algorithms under consideration.

ARI: The ARI for the algorithms of *K*means, *K*means Force, GE, GE Force algorithm with the various reading strategies are represented in figure 7.8 on page 86. The ARI value of algorithms using force measure is way lower than the ARI measure of the algorithms using the distance measure. The GE with diagonally placed seeds evidences better ARI than the other algorithms. The median value(50th percentile) of the GE with seeds positioned in the "Z" fashion is slightly lower than the median value for GE with seeds positioned in the diagonal fashion. It should also be taken into consideration that the ARI values have a huge range for the GE with seeds positioned in the "Z" fashion when compared with the GE with diagonally positioned seeds i.e. the box for the GE with seeds in the "Z" fashion is longer than the box for the GE with diagonally positioned seeds. This indicates that GE with seeds in the "Z" fashion tend to have a huge standard deviation which is not desirable for the task at hand. Thus it is concluded that GE with diagonally positioned seeds has a better performance than the other algorithms under consideration in terms of the ARI cluster metric.

Jaccard: The Jaccard for the algorithms of *K*means, *K*means Force, GE, GE Force algorithm with the various reading strategies are represented in figure 7.9 on page 87. It is noted again that using distance measure increases the performance of GE and *K*means than using the force measure. It is to be noted that the Jaccard cluster metrics for the GE using distance measure is considerably higher than the Jaccard value for the other algorithms. Though the median value (50th percentile) of the GE with diagonally positioned seeds

is a bit lower than the value for GE with seeds positioned in "Z" fashion, again, like the ARI measure, the values of the Jaccard for the GE with seeds positioned in a "Z" fashion covers a huge range when compared with the GE with diagonally positioned seeds. Thus it could be concluded that GE with diagonally positioned seeds gives a comparable performance when compared with the GE with seeds positioned in the "Z" fashion, when considering the Jaccard cluster metric.

F&M score: The F&M score for the algorithms of *K*means, *K*means Force, GE, GE Force algorithm with the various reading strategies are represented in figure 7.10 on page 87. In terms of the F&M score, GE with distance measure outperforms the GE with force measure. However, *K*means with distance measure outperforms the *K*means with force measure only with a small value. Also, it can be noted that GE Force, *K*means with distance and *K* with force form a set with almost similar values. But the GE with distance measure has considerable higher Jaccard measure irrespective of the positioning of the seeds.

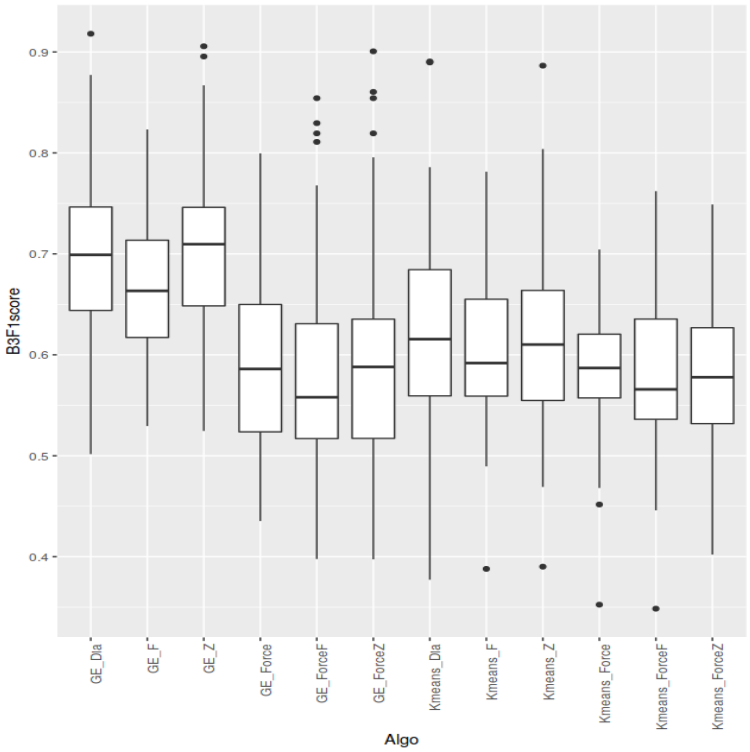

Figure 7.4: *Box plot of the B3F1 score for K means,K Force, GE, GE Force with the various reading strategies.*

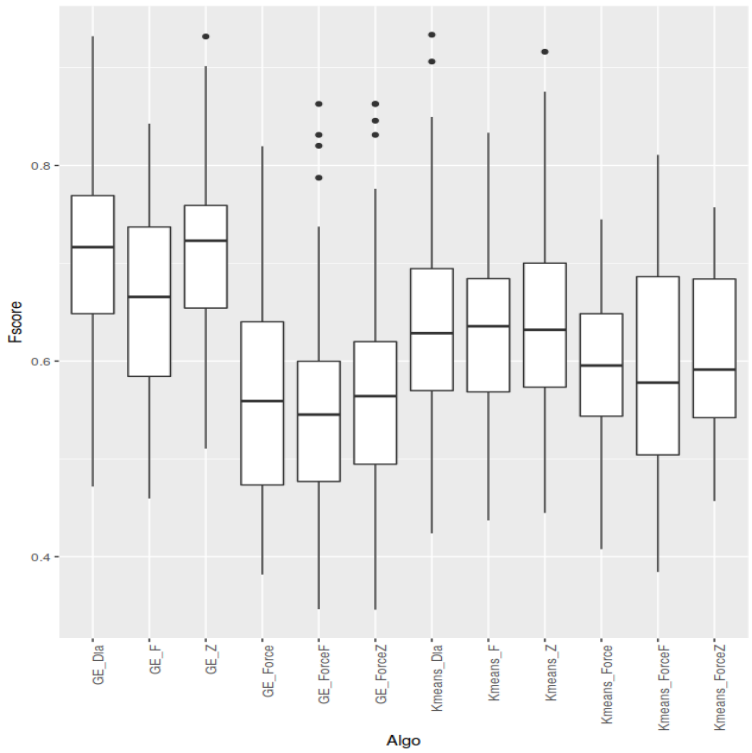

Figure 7.5: *Box plot of the Fscore for Kmeans,KForce, GE, GE Force with the various reading strategies.*

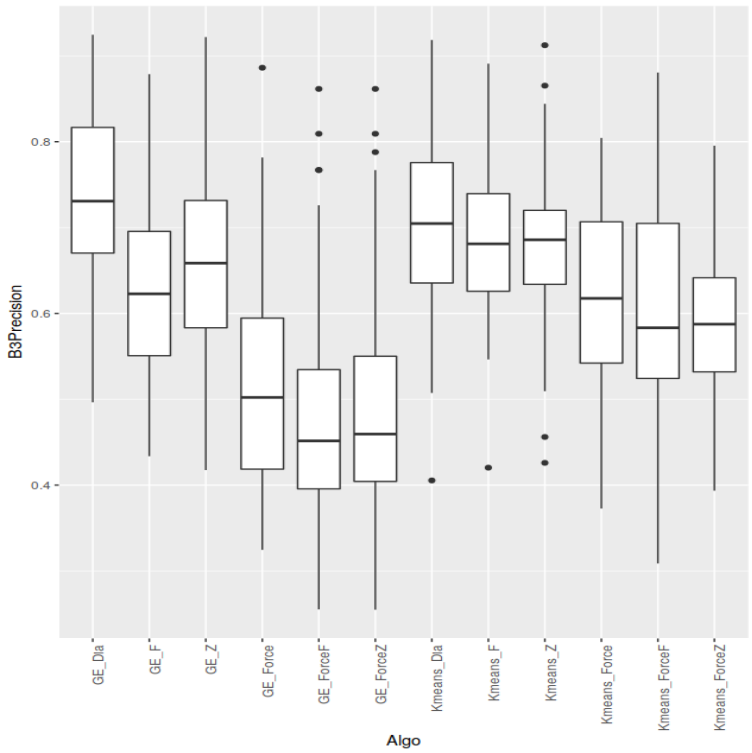

Figure 7.6: *Box plot of the Precision for Kmeans,KForce, GE, GE Force with the various reading strategies.*

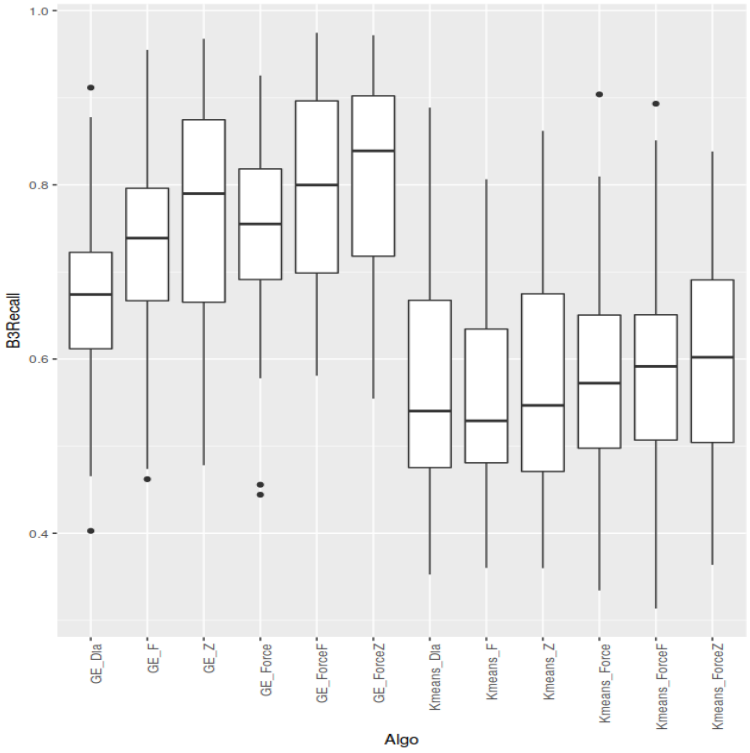

Figure 7.7: *Box plot of the Recall for Kmeans,KForce, GE, GE Force with the various reading strategies.*

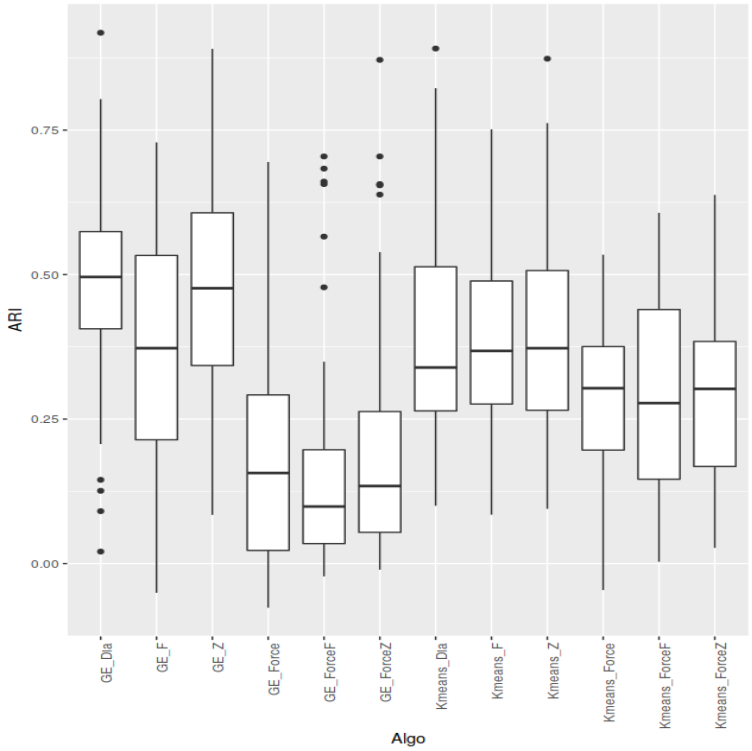

Figure 7.8: *Box plot of the ARI for Kmeans,KForce, GE, GE Force with the various reading strategies.*

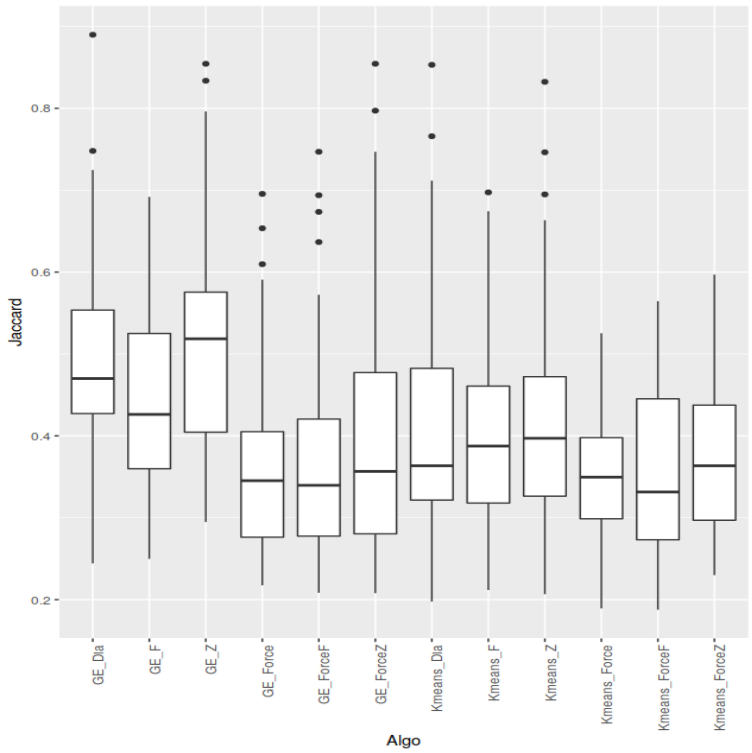

Figure 7.9: *Box plot of the Jaccard for Kmeans,KForce, GE, GE Force with the various reading strategies.*

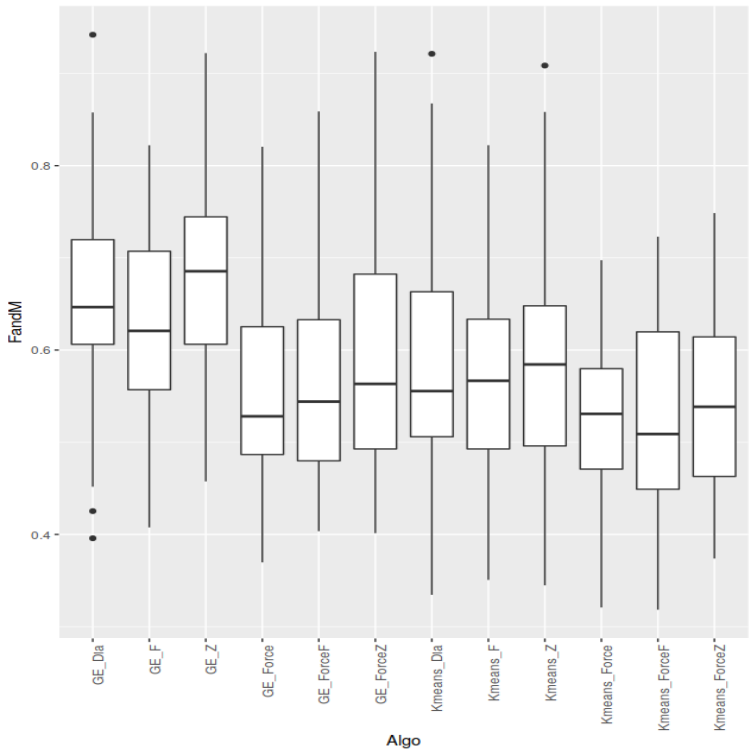

Figure 7.10: *Box plot of the F&M for Kmeans,KForce, GE, GE Force with the various reading strategies.*

Box Plots with various thresholds

Figures 7.11, 7.12, 7.13, 7.14, 7.15, 7.16 and 7.17 on pages 90, 91, 92, 93, 94, 95 and 96 respectively, show the box plots of the cluster metrics presented in table 7.2 on page 77 for the various thresholds of S1 - QT with clusters as seeds (Algorithm 4) is referenced on the figures as GE_QT_<threshold>, S2 - QT with centroids as seeds (Algorithm 5 on page 60) is referenced on the figures as GE_QT_complete_<threshold> and GE pre cluster (Algorithm 3) as presented in table 7.2 on page 77.

B3F1score: Figure 7.11 on page 90 shows the B3F1 score for the Algorithms 3 on page 58 and Algorithms 4 and 5 on pages 59, 60 respectively with various thresholds. There is a clear difference between using Algorithm 4 on page 59 and Algorithm 5 on page 60. The B3F1 score is higher when using S2 (5). This is due to the fact that S2 (Algorithm 5 on page 60) allows for the better placement of seeds and thus proper expansion of zones. It should also be noted that QT clustering technique uses thresholds to determine the formation of clusters. Thus fixing a threshold plays an important part in the positioning of the seeds. As explained before, the thresholds are a fraction of the maximum border to border distance. In both cases, S1(Algorithm 4 on page 59) and S2(Algorithm 5 on page 60), the biggest thresholds gives a better B3F1 score. However, while using a threshold of 1/25, 1/30, 1/35 of the maximum border to border distance, the Algorithm 5 on page 60 produces comparable results. This can be explained by the table 6.1 on page 57. Bigger thresholds allow the formation of the necessary clusters, 5 clusters in this task and thus allowing a better positioning of seeds. And the strategy used by Algorithm 5 on page 60 to position the remaining seeds in situations where the necessary 5 clusters could not be formed always better positioning of seeds even in case of smaller thresholds. But when the thresholds decreases towards 1/50 of the maximum border to border distance, there are a lot of pages that form only 1 cluster and some that do not form any clusters at all as seen in table 6.1 on page 57 and thus even while using the strategy mentioned in Algorithm 5 on page 60to position the remaining seeds, as there are a lot of seeds to be positioned, the segmentation is not very efficient. Algorithm **??** on page 58 referenced as Ge_precluster produces a B3F1 score which is between Algorithm 4 on page 59 and Algorithm 5 on page 60. Algorithm 3 on page 58 is a simple pre clustering using a threshold. The threshold used here is one tenth of the maximum border to border distance. However, using this sort of clustering is not very extensive i.e. it does not identify the biggest clusters, it just takes the first formed cluster within the threshold. Thus this does not allow the efficient positioning of seeds. It thus could be concluded that Algorithm 5 on page 60 with big thresholds allow the formation of better zones.

Fscore: Figure 7.12 on page 91 shows the Fscore for the Algorithms 3 on page 58 and Algorithms 4 and 5 on pages 59 and 60 respectively with various thresholds. Again it can be clearly seen that there are two groups, one with Algorithm 4 on page 59 with its various thresholds and the other with Algorithm 5 on page 60 with its various thresholds. However, it should be noted that the Fscore decreases for thresholds 1/15 and 1/20 and rises again from

thresholds 1/25 then evidencing a small decrease at threshold 1/40 and stabilizes again. This again could be explained by the strategy used to position the seeds and the number of clusters formed from QT clustering as seen in table 6.1 on page 57. On the other hand, while using Algorithm 4 on page 59, the Fscore decreases constantly with the decreasing thresholds. While decreasing the threshold, the possibility to form the necessary number of clusters is low. The strategy used in this algorithm(Algorithm 4 on page 59) to position the remaining seeds is not very efficient. This gets only worse as the threshold decreases to 1/50, where there are web pages that is not able to form clusters at all (table 6.1 on page 57). Again like in the B3F1 score, Algorithm 3 on page 58 gives a result between the two algorithms.

Precision and Recall: Figures 7.13 and 7.14 on pages 92 and 93 respectively, shows the Precision and Recall for the Algorithms 3 on page 58 and Algorithms 4 and 5 on pages 59 and 60 respectively, with various thresholds. It can be noticed that while the precision decreases the recall increases for every algorithm. The highest precision and lowest recall is achieved by the Algorithm 3 on page 58. This means that the number of false positives increases and thus decreasing the precision. This is true for the thresholds of Algorithm 4 on page 59 and Algorithm 5 on page 60 as well. As the threshold decreases, the precision decreases while increasing the recall due to increase in the number of false positives and decrease in the number of false negatives. It can be seen that while using S1(Algorithm 4 on page 59), the precision decreases rapidly while with S2(Algorithm 5 on page 60) the precision decreases gradually before stabilizing. Collectively, the precision of S2(Algorithm 5) is higher than that of S1(Algorithm 4 on page 59). This again proves that the strategy used to position the seeds with S2 (Algorithm 5 on page 60) is more efficient than the one used in S1(Algorithm 4 on page 59), as it allows better expansion of the zones. Again, among the various thresholds used for S1 and S2, the highest precision is achieved when the threshold is set to one tenth of the maximum border to border distance in a web page i.e. Bigger the threshold set, better the precision achieved.

ARI: Figure 7.15 on page 94 shows the ARI for the Algorithms 3 on page 58 and Algorithms 4 and 5 on pages 59 and 60 respectively, with various thresholds. With S1(Algorithm 4, the best ARI is achieved with the biggest threshold (1/10 of the maximum border to border distance) and then the ARI decreases gradually as the threshold becomes smaller (towards 1/50). While using S2 (Algorithm 5 on page 60, the ARI of the biggest threshold is high, however, it can be noted that the ARI value decreases for thresholds 1/15 and 1/20 of the maximum border to border distance and then increases again while the threshold is 1/25 of the maximum border to border distance before stabilizing. Overall, S1(Algorithm 4 on page 59) achieves lower ARI than S2(Algorithm 5 on page 60). Algorithm 3 on page 58 achieves a ARI equal to the ARI of the smallest threshold of the S2 (Algorithm 5 on page 60).

Jaccard: Figure 7.16 on page 95 shows the Jaccard for the Algorithms 3 on page 58 and Algorithms 4 and 5 on pages 59 and 60 respectively, with various thresholds. As with the other cluster metrics, the highest Jaccard is achieved

from S2 (Algorithm 5 on page 60) with the biggest threshold and it should also be noted that like in all other cluster metrics (except recall), S2 (Algorithm 5 on page 60) evidences better Jaccard than S1 (Algorithm 4 on page 59). Finally, Algorithm 3 on page 58 stands in between Algorithms 4 and 5 on page 59 and 60 respectively.

F and M score: Figure 7.17 on page 96 shows the F&M score for the Algorithms 3 on page 58 and Algorithms 4 and 5 on pages 59 and 60 respectively, with various thresholds. For S1 (Algorithm 4 on page 59) evidences almost equal F&M score for all thresholds. For S2 (Algorithm 5 on page 60) also evidences almost equal F&M score for all thresholds except for 1/15 and 1/20 of the maximum border to border distance for which the F&M score are lower. Again comparing S1(Algorithm 4 on page 59) and S2(Algorithm 5 on page 60), like in other cluster metrics, S2(5 on page 60) evidences higher F&M score than S1(Algorithm 4 on page 59). The F&M score of algorithm 3 on page 58 is lower than that of both S1(Algorithm 4 on page 59) and S2(Algorithm 5 on page 60).

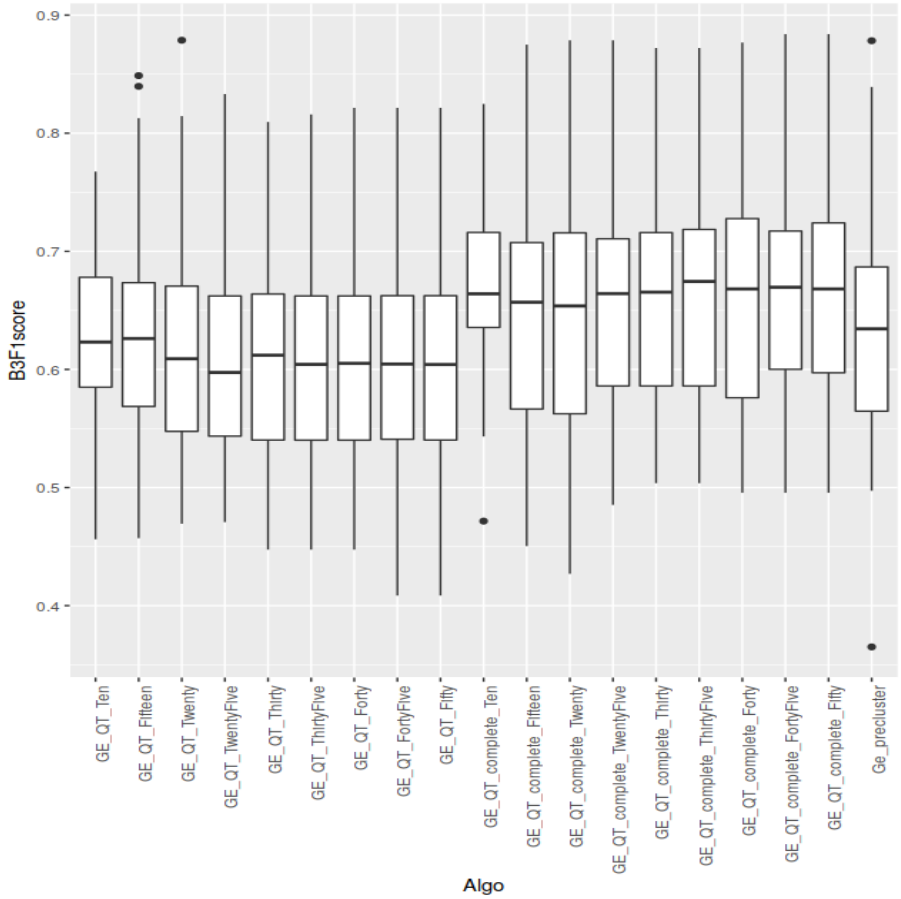

Figure 7.11: *Box plot of the B3F1 score for the various thresholds using algorithm 4 (S1 - QT with clusters as seeds) and 5 (S2 - QT with centroids as seeds), and algorithm 3 (simple pre-clustering)*

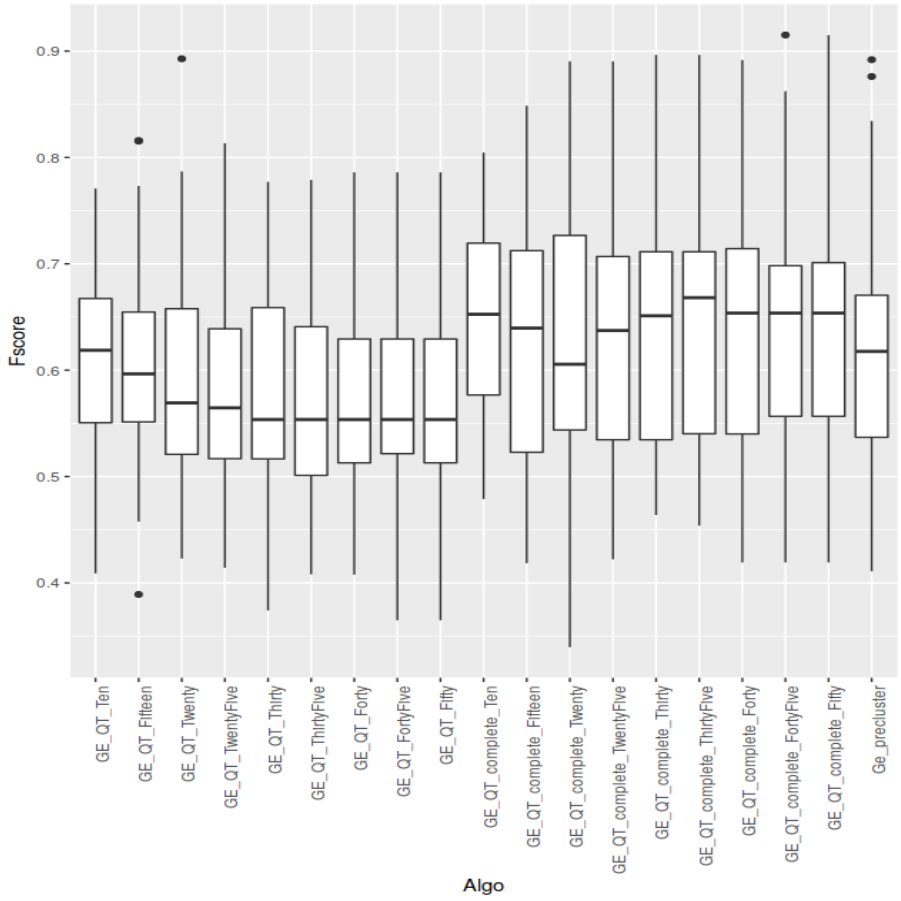

Figure 7.12: *Box plot of the Fscore for the various thresholds using algorithm 4 (S1 - QT with clusters as seeds) and 5 (S2 - QT with centroids as seeds), and algorithm 3 (simple pre-clustering)*

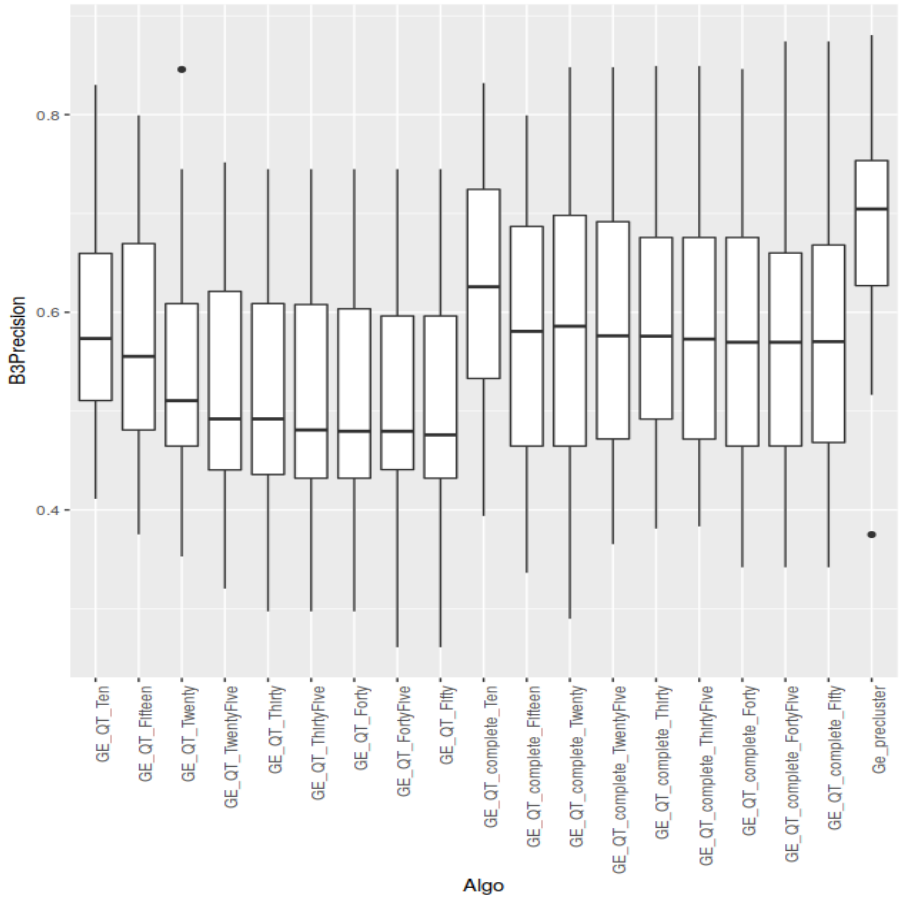

Figure 7.13: *Box plot of Precision score for the various thresholds using algorithm 4 (S1 - QT with clusters as seeds) and 5 (S2 - QT with centroids as seeds), and algorithm 3 (simple pre-clustering)*

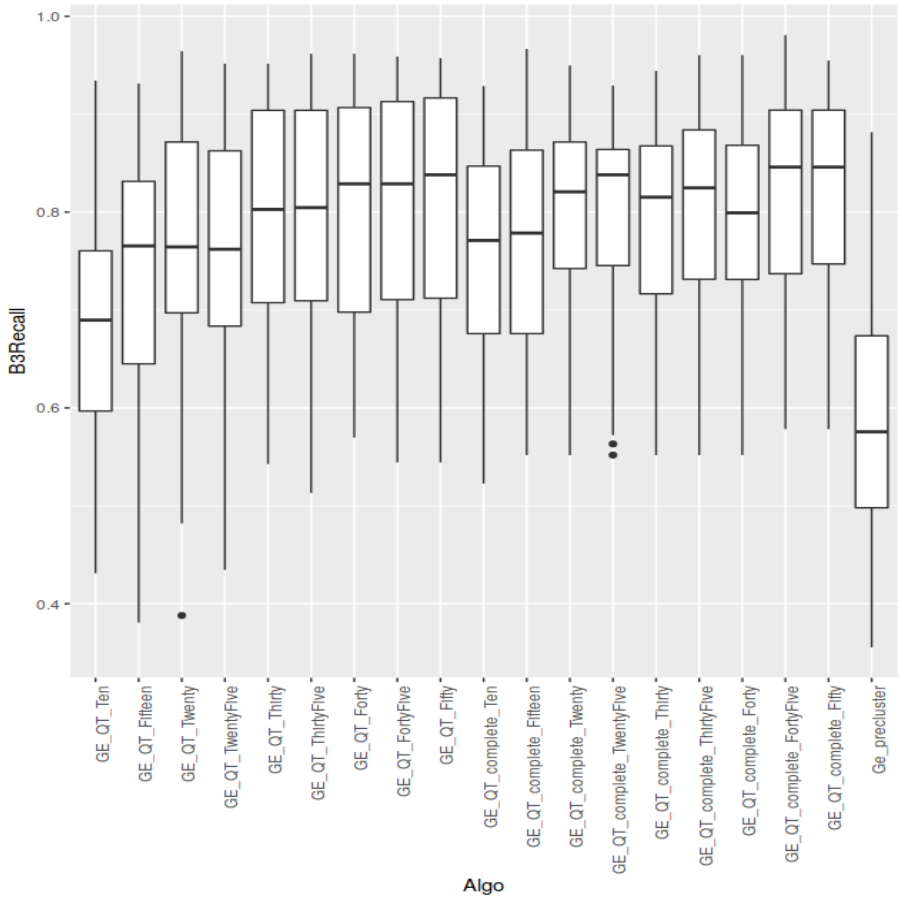

Figure 7.14: *Box plot of Recall for the various thresholds using algorithm 4 (S1 - QT with clusters as seeds) and 5 (S2 - QT with centroids as seeds), and algorithm 3 (simple pre-clustering)*

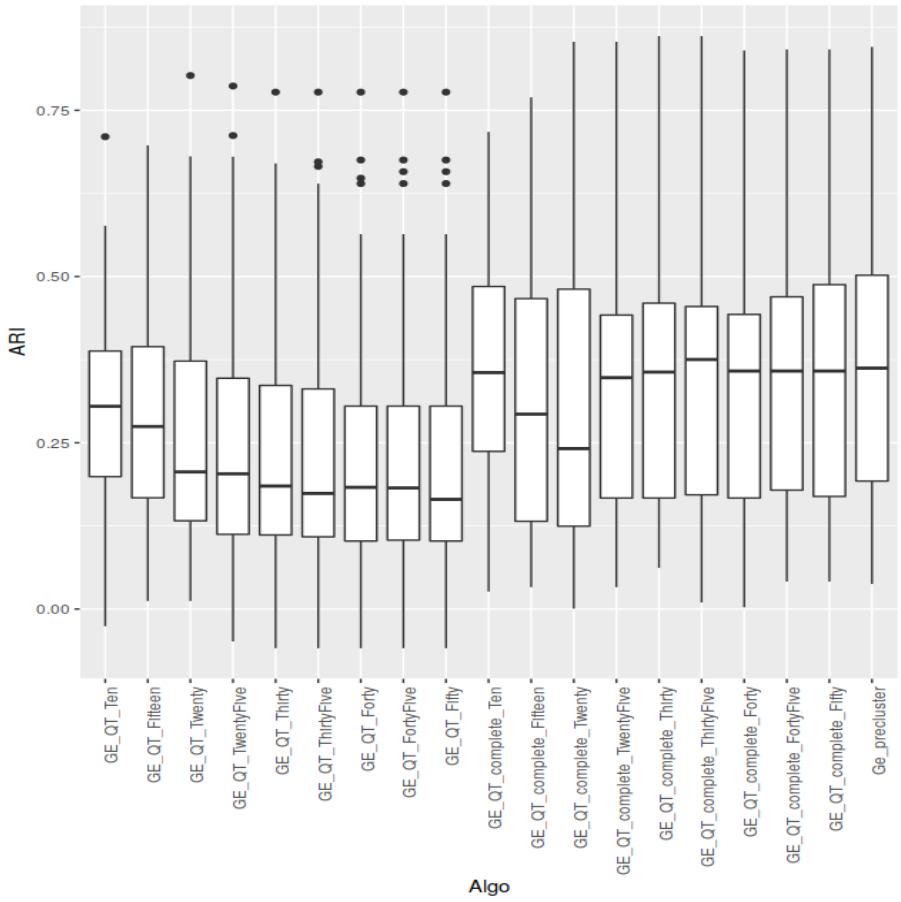

Figure 7.15: *Box plot of ARI for the various thresholds using algorithm 4 (S1 - QT with clusters as seeds) and 5 (S2 - QT with centroids as seeds), and algorithm 3 (simple pre-clustering)*

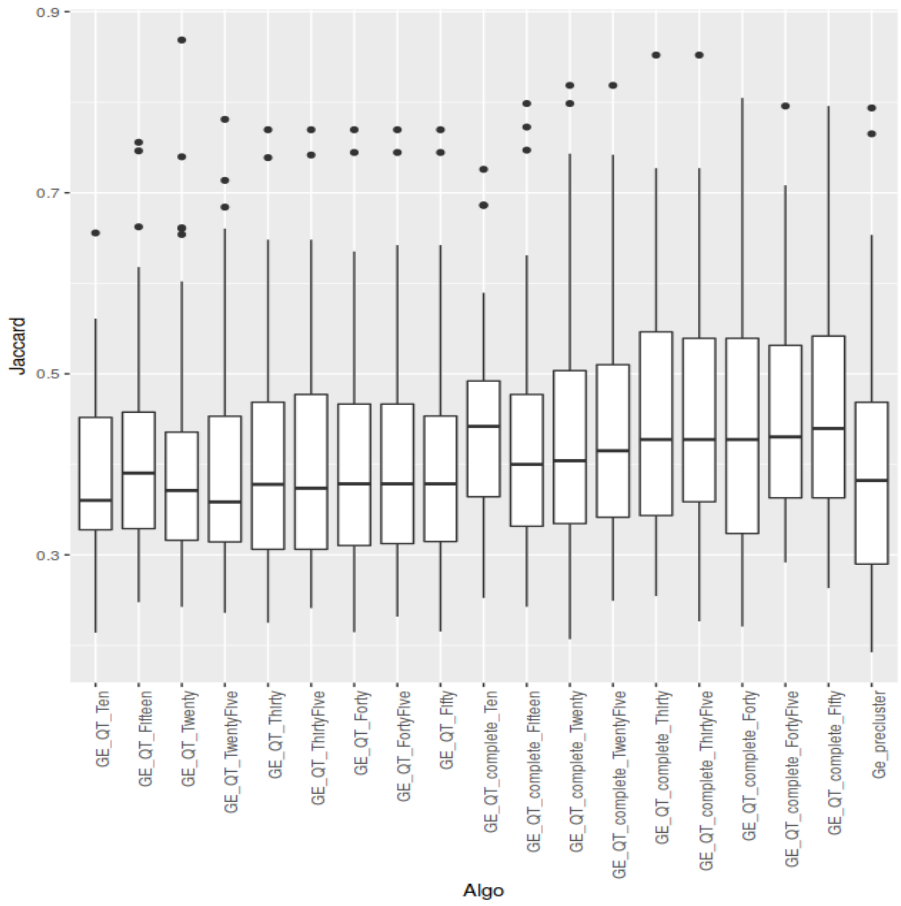

Figure 7.16: *Box plot of Jaccard for the various thresholds using algorithm 4 (S1 - QT with clusters as seeds) and 5 (S2 - QT with centroids as seeds), and algorithm 3 (simple pre-clustering)*

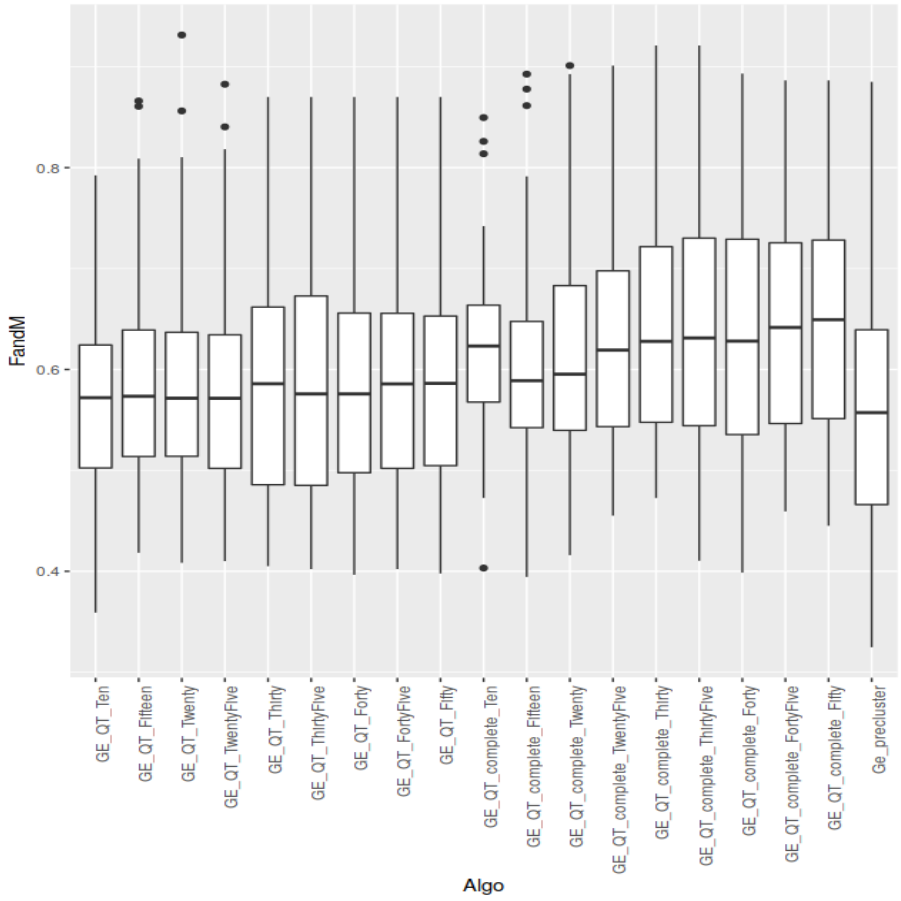

Figure 7.17: *Box plot of the F&M score for the various thresholds using algorithm 4 (S1 - QT with clusters as seeds) and 5 (S2 - QT with centroids as seeds), and algorithm 3 (simple pre-clustering)*

Conclusion: While the algorithms K-means, F-K-means and Guided Expansion(GE) with various reading strategies are compared with each other, Guided Expansion(GE)(Algorithm 2 on page 48) with the seeds position along the diagonal gives the best results in terms of all the cluster metrics described above (except Recall).

On the other hand, when GE with seeds positioned using a clustering technique (a simple clustering or QT clustering) - Algorithms 3, 4 and 5 on pages 58, 59 and 60 respectively, using Algorithm 5 on page 60 - GE with seeds positioned using the QT clustering technique where the seeds are the centroids of the clusters formed from QT clustering and in cases where the required number of clusters cannot be formed from QT clustering, the seeds are placed using the maximum average distance between the remaining web elements and the formed clusters, gives the best results in terms of the cluster metrics (except recall). The threshold that gives best result is 1/10 of the maximum border to border distance of all web elements in a web page i.e. the bigger the threshold the better the results.

When the best algorithms from both the sets are compared with each other - Algorithm 2 on page 48 with the diagonal reading strategy and Algorithm 5 on page 60 with threshold 1/10 of the maximum border to border distance, they both perform equally on all cluster metrics as can be seen from the box plots.

The metrics such as ARI, Jaccard and F&M scores compares the ground truth with the clustering algorithms one to one. However, it should be noted that the segmentations achieved by the clustering algorithms could be very good and yet very different from the ones done by the experts. This is proved by the ground truths presented in figures 7.1, 7.2 and 7.3 (pages 73, 73 and 73) and the GE with diagonally positioned seeds as in figures 5.4, 5.7 and 5.10 respectively (Pages 49, 50 and 51). The segmentations from the GE with the diagonally positioned seeds are good as indicated by the B3F1 scores, precision and recall in table 7.2 on page 77 but they are not similar to the segmentations done by the experts presented as the ground truth (figures 5.4, 5.7 and 5.10 respectively (Pages 49, 50 and 51)). Because of this difference between the ground truth and the segmentations produced by the algorithms, ARI, Jaccard and F&M scores are not the most relevant metrics for the task at hand.

Chapter 8

Automatic Evaluation

8.1 Introduction

As it is necessary to have annotated web pages that can act as ground truth in order to be able to evaluate them in terms of the cluster metric, evaluated huge number of web pages is not possible. This is due to the fact that the algorithms developed are developed with a particular task in mind and an annotated data set is not available for this particular task, also annotating several web pages is not feasible as it is expensive. Thus it is essential to develop a method of evaluation without the use of annotated web pages and this method should take into account the various criteria required for the task. Thus following the criteria and the evaluations given by the experts on the 50 web pages (stated in chapter 7) are used to design metrics that will allow an evaluation of any and every segmented web page without the need for an annotated ground truth.

The measures are designed to evaluate the logical coherency and the visual similarities. The evaluations given by the experts for the first 50 web pages are closely examined to determine when the expert appreciates the segmentation and when the expert penalises the segmentation. Questions and discussions were conducted with the experts to better understand their views on the segmentation and to enable the development of the metrics that will allow automatic quantitative evaluations. Following this process, the metrics described below were developed to evaluated a segmented web page without the need for an annotated web page.

- **Cuts:** Experts evaluated negatively clustering results when logical constraints were broken, embodied by specific HTML tag sequences such as <li> items, <title> and the following paragraph <p>, <header>, <footer> or <nav> elements. So, each time one of these logical constraints is broken, this counts for one cut, and each web page is evaluated based on its overall number of cuts. Thus the lower the number of cuts, the better the algorithm.
- **Balance:** Experts negatively evaluated strong imbalance between clusters, but also high balance between clusters. This can be motivated by the fact that a great deal of web pages contain a main (rather large)

body section, while all other zones show similar sizes. Note that this issue is usually not taken into account by classical clustering metrics such as Adjusted Rand Index or F-score. As a consequence, this notion of balance is tested over three different properties of the clusters: surface area of the cluster which is calculated as the sum of the surface areas of all web elements in a cluster, text area which is calculated as the number of characters within the cluster, and number of elements which is calculated by counting the number of web elements within the cluster. So, each web page receives an overall score that stands for the standard deviation between all clusters for each of the three balance criteria (i.e. surface, text and visual elements).

— **Exterior Rectangle:** Experts evaluated negatively when the zones were intertwined with each other, i.e. the clustering should avoid non-rectangular clusters. To evaluate this phenomenon, the number of overlaps between the outer rectangles of all clusters is calculated, i.e. the smallest rectangle including all the elements of each cluster. So, if two clusters overlap in terms of outer rectangle, this stands for the presence of a non rectangular zone, and it is counted as a nested situation.

8.2 Automatic Evaluation

The measures discussed in the previous section allows for evaluating any number of segmented web pages for the task of non visual skimming and scanning without the need for an underlying ground truth (annotated web page). In this section, 900 web pages - 300 tourism web pages, 300 E-Commerce web pages and 300 News web pages are segmented using all the algorithms presented in the previous chapters of this dissertation are evaluated. The evaluations for all 900 web pages are presented in table 8.1 on page 103. The column "cuts" refers to the average number of cuts over all 900 web pages. For the column "SA", the normalized average of the surface area covered by each zone for web page is calculated and then the average over these 900 values is computed. For the column "TA", the normalized average of the text area covered by each zone ((i.e) number of words in each zone) for web page is calculated and then the average over these 900 values is computed. The column "Ext. Rect." represents the average number of intertwined zones present over the 900 web pages.

cuts: As mentioned earlier, the cuts property refers to the breaks or cuts in the HTML elements. In a web page, while there is a cut on a web page, it counts as a cut. The numbers in table 8.1 on page 103 refer to the average cuts over all the web pages. Thus the lower the number of cuts, the better the algorithm. Thus considering this criteria for table 8.1 on page 103, the best algorithm is Guided Expansion with the diagonal reading strategy with 1.23 average cuts. This is closely followed by Guided Expansion where the seeds are placed using the QT clustering technique with a completion technique to place remaining seeds (S2) with a threshold of one tenth with 1.38 average cuts. This is evidently because of the alignment and font distances used in the Guided Expansion algorithm. This goes on to prove once again the importance of these

features for the task at hand. It should be noted that the strategy 2 of placing seeds with the QT clustering technique uses the centroids of the clusters formed as seeds and thereby allowing for the use of the alignment and font features (because of the complete use of Guided Expansion) for the expansion of a zones. While the strategy 1 of placing seeds with the QT clustering technique uses the whole clusters from the QT clustering as seeds and thereby clustering most web elements without the alignment and font features (QT clustering uses only the Euclidean distance between web elements) resulting in more cuts. With respect to the thresholds for S1 and S2, it has been shown in table 6.1 on page 57 that as the threshold decreases from 1/10th of the maximum distance to 1/50th of the maximum distance, it is not always possible to form the necessary number of clusters. This effect is also evident in the number of cuts. While the threshold is big and the necessary number of clusters/seeds could be formed from the QT technique, then the cuts are lower - S1 1/10 is 2.08 and S2 1/10 is 1.38. On the other hand, when the necessary number of clusters/seeds could not be formed from the QT technique because of the small threshold set, there is a necessity to use a different strategy to position the remaining seeds, in which case S1 uses a random approach while S2 uses the maximum average distance from the formed clusters to position the remaining seeds. This is seen to cause increase in the number of cuts - S1 1/50 is 4.00 and S2 1/50 is 3.58. It is also seen that irrespective of the thresholds, S2 performs better than S1 as S2 takes the centroids of the clusters from QT technique instead of the whole cluster as in S1 - S1 with a range of 2.08 to 4.00 and S2 with a range of 1.38 to 3.58. It should also be noted that Guided Expansion indifferent of the reading strategy used to position the seeds performs better than the *K*means algorithms, again due to the introduction of the alignment and font features and a step by step careful expansion unlike *K*means where all web elements are clustered and re-clustered at every step. There is also a high standard deviation of 6.22 for the *K*means algorithm indicating that certain web pages have way more cuts than the others causing a huge difference between the web pages.

Balance: Balance refers to the balance between the zones formed. Table 8.1 on page 103 shows this criteria using the three balance metrics which are surface area covered by the zones(SA), balance in the text content within the zones(TA) and the number of web elements within the zones (No.Of.Elements). In all cases there are similar observations between clusters. This is shown in figures 8.1, 8.2 and 8.3 on pages 102, 102 and 102 respectively, which are box plots for the Surface Area, Text Area and Number of Elements for the *K*means with diagonally placed seeds, F-*K*means with diagonally positioned seeds, GE with diagonally placed seeds, F-GE with diagonally positioned seeds, GE with the clusters from the QT clustering as seeds(S1) with threshold 1/10, GE with a simple pre cluster (GE pre cluster (Algorithm 3 on page 58)), GE with the centroids of clusters from the QT clustering as seeds(S2) with threshold 1/10 [1]. The red line through the boxes show that Surface Area(SA), Text

1. These algorithms are chosen as representatives from all the 31 variations of the algorithms for the box plots. These box plots are made for 50 web pages (23 tourism web pages, 12 E-commerce web pages and 18 news web pages). These choices have been made for better visual representation of the plot.

Area(TA) and Number of elements have the same pattern, indicating that all three metrics of balance gives the same observations. This is also evident from table 8.1 on page 103 for all the other algorithms as well.

Guided Expansion using the strategy 2 for positioning the seeds show the highest imbalance ranging from 25.98 to 31.00 for surface area, 26.12 to 29.25 for Text Area and 25.75 to 30.01 for the no.of.elements. *K*means algorithms show the lowest imbalance with the lowest imbalance when the seeds are positioned diagonally - 11.94 for surface area, 10.45 for text area and 10.95 for no.of.elements. It is also important to note that using a pre-clustering step with GE increases (GE Pre cluster) the balance between the zones in a huge way - 18.51 for surface area, 17.35 for text area and 15.04 for no.of.elements. However, imbalance between the zones does not really mean bad segmentation. As seen from the manual segmentation presented in figures 7.1 (page 73), 7.2 (page 73) and 7.3 (page 73), there is huge imbalance between the zones specially between the main contents and the menu region showing that humans do not necessarily look for balance while segmenting a web page. However, considering the framework of TAG THUNDER, where the contents of the zones formed in the segmentation process is used to form tag clouds which are vocalized forming a tag thunder as explained in chapter 1, it is better to have a good balance between the contents of the zones enabling the better formation of a tag thunder.

Exterior rectangle: This criteria measures how much the different zones are intertwined with each other. It can be seen from figures 7.1 (page 73), 7.2 (page 73) and 7.3 (page 73) that most of the times times when humans prefer to have rectangular zones. However, in many cases "L" shaped zones are preferred as well (like red zone in figure 5.4 on page 73). From table 8.1 on page 103, it is seen that GE with a diagonal reading strategy has the lowest intertwined zones - 0.63, followed by GE using the simple pre-clustering method to position seeds - 0.65. Using the clusters as seeds from the QT clustering method(S1) has high value for Exterior Rectangle value when the threshold is high (1.80 and 1.84 for S1 1/15 and S1 1/20 respectively) but the value decreases when the threshold is low (1.19 and 1.13 for S1 1/40 and S1 1/45 respectively). This is because when the threshold is low, the positioning of the remaining seeds (in case of impossibility to form the required number of seeds) causes some small zones with one/few web elements which does not interact with other zones at all and thus lowering the value of the exterior rectangle metric. The use of centroids as seeds from QT clustering and a different strategy to position initial seeds(S2) has almost similar values for the exterior rectangle irrespective of the threshold and the values are lesser than using the clusters as seeds(S1). This is again because this method allows for the positioning of seeds far from each other and in the most probable expansion areas within a web page. This helps with the proper expansion of each zones allowing rectangular or "L" shaped zones and zones that are not intertwined.

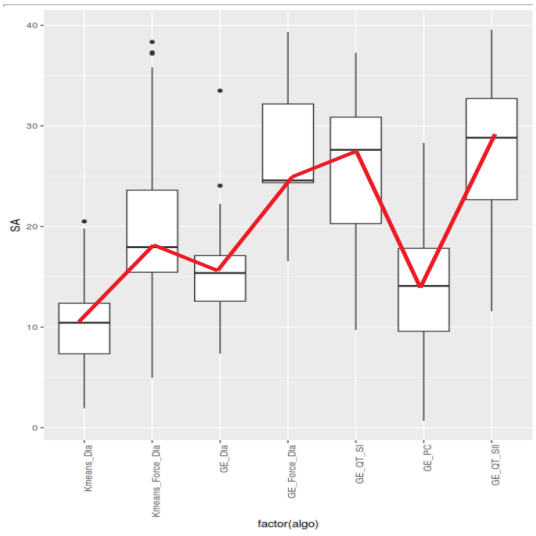

Figure 8.1: *Box plot for the Surface Area metric*

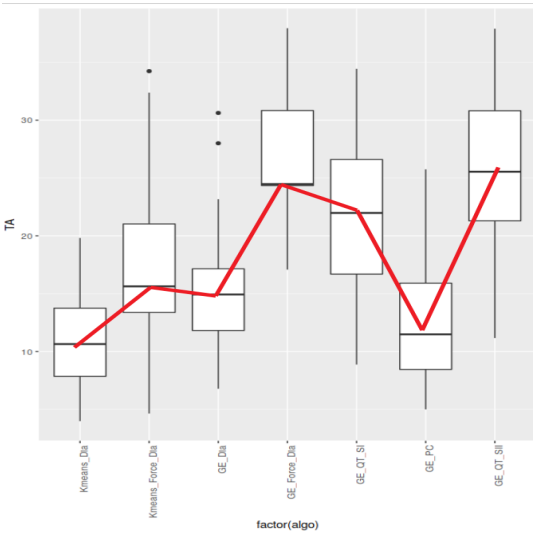

Figure 8.2: *Box plot for the Text Area metric*

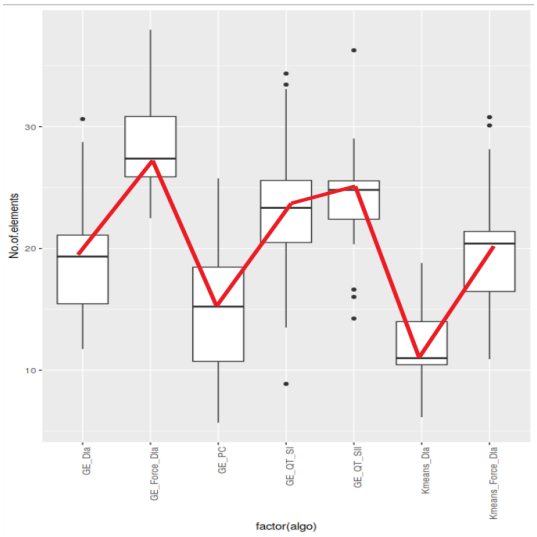

Figure 8.3: *Box plot for the No.of.Elements metric*

	Nb. of Cuts Avg. $\pm\sigma$	SA Avg stdev. $\pm\sigma$	TA Avg stdev. $\pm\sigma$	No. Of. Elements Avg stdev. $\pm\sigma$	Ext. Rect. Avg. $\pm\sigma$
K-means diagonal	3.20 ± 6.22	11.94 ± 6.73	10.45 ± 4.90	10.95 ± 8.01	1.41 ± 1.42
Kmeans F strategy	3.52 ± 4.72	13.18 ± 7.03	11.69 ± 4.93	12.83 ± 7.90	1.09 ± 1.32
Kmeans Z strategy	3.32 ± 3.90	14.13 ± 6.56	13.20 ± 5.05	14.85 ± 8.45	1.72 ± 1.04
F-K-means diagonal	3.43 ± 6.50	21.05 ± 8.78	19.35 ± 9.52	20.79 ± 10.78	1.30 ± 1.43
F-K-means F strategy	3.76 ± 6.71	21.46 ± 9.92	19.52 ± 9.59	21.87 ± 8.90	1.04 ± 1.18
F-K-means Z strategy	3.50 ± 4.01	23.90 ± 8.88	23.07 ± 10.09	24.45 ± 9.67	1.45 ± 1.14
GE diagonal	1.23 ± 1.15	20.15 ± 6.70	24.61 ± 6.16	19.67 ± 7.17	0.63 ± 1.18
GE F strategy	1.85 ± 1.23	23.22 ± 7.80	23.45 ± 6.89	25.43 ± 7.10	1.21 ± 1.05
GE Z strategy	1.90 ± 1.13	24.45 ± 7.45	23.49 ± 7.04	24.79 ± 8.04	1.32 ± 1.04
GE Force Diagonal	2.45 ± 2.13	24.30 ± 7.64	23.89 ± 6.56	27.89 ± 7.53	1.56 ± 1.11
GE Force F Strategy	2.95 ± 2.34	26.77 ± 6.63	27.77 ± 7.56	30.09 ± 8.12	1.67 ± 1.32
GE Force Z Strategy	2.67 ± 2.22	25.67 ± 7.11	26.56 ± 7.22	28.90 ± 7.68	1.43 ± 1.13
S1 1/10	2.08 ± 4.81	25.19 ± 8.40	21.96 ± 7.46	23.34 ± 7.56	1.31 ± 1.56
S1 1/15	2.43 ± 4.12	27.90 ± 8.64	26.90 ± 6.09	26.13 ± 5.99	1.80 ± 1.34
S1 1/20	2.57 ± 4.02	28.09 ± 9.02	28.56 ± 9.45	29.56 ± 9.72	1.84 ± 1.23
S1 1/25	3.06 ± 3.90	30.67 ± 9.89	28.90 ± 7.09	29.46 ± 8.07	1.28 ± 1.34
S1 1/30	3.33 ± 3.51	31.45 ± 8.41	29.78 ± 8.42	30.09 ± 7.69	1.29 ± 1.58
S1 1/35	3.90 ± 3.47	30.90 ± 9.34	31.09 ± 9.05	30.09 ± 8.62	1.25 ± 1.61
S1 1/40	3.78 ± 2.98	27.89 ± 7.98	28.64 ± 8.92	27.95 ± 8.69	1.19 ± 1.07
S1 1/45	3.96 ± 2.48	28.45 ± 8.74	31.31 ± 8.03	29.74 ± 7.89	1.13 ± 0.87
S1 1/50	4.00 ± 2.13	29.56 ± 8.89	25.73 ± 7.89	27.86 ± 7.60	1.32 ± 1.11
GE Pre cluster	2.72 ± 6.07	18.51 ± 8.44	17.35 ± 7.78	15.04 ± 8.71	0.65 ± 1.15
S2 1/10	1.38 ± 2.41	25.98 ± 8.29	26.12 ± 8.36	25.75 ± 8.45	0.84 ± 1.33
S2 1/15	1.60 ± 2.30	27.89 ± 9.00	28.70 ± 8.67	27.18 ± 8.10	0.90 ± 1.34
S2 1/20	1.87 ± 2.40	28.45 ± 8.05	28.63 ± 8.32	27.90 ± 8.21	0.90 ± 1.43
S2 1/25	2.05 ± 3.12	32.36 ± 8.33	29.20 ± 8.26	30.86 ± 8.24	0.80 ± 1.30
S2 1/30	2.54 ± 2.20	31.52 ± 7.85	29.43 ± 8.13	29.90 ± 8.06	0.83 ± 1.22
S2 1/35	3.13 ± 2.93	30.96 ± 8.59	29.16 ± 8.44	30.12 ± 8.33	0.81 ± 1.14
S2 1/40	3.43 ± 3.13	30.88 ± 8.43	29.12 ± 8.48	30.11 ± 7.22	1.22 ± 1.35
S2 1/45	3.52 ± 3.15	31.16 ± 8.45	29.24 ± 8.53	30.16 ± 7.89	0.86 ± 1.36
S2 1/50	3.58 ± 3.10	31.00 ± 8.57	29.25 ± 8.45	30.01 ± 8.39	0.89 ± 1.38

Table 8.1: Automatic Evaluation

8.3 Statistical tests

To confirm the results, a statistical test is performed. A Dunn Test is performed for this purpose. A Dunn's Test Dunn (1961) can be used to pinpoint which specific means are significant from the others. Thus, the Dunn's Multiple Comparison Test is a post hoc, non parametric test, which is done to determine which groups are different from others. In order to verify the differences between algorithms in terms of statistical significance this test is used. Table 8.2 on page 105 shows the result of this Dunn Test. Algorithms which belong to the same group are not significantly different from each other for that particular metric. The algorithms from the different groups are significantly different from each other for that particular metric. The analysis presented in table 8.2 on page 105 are performed for the data set of 900 web pages (300 web pages on tourism, 300 web pages on e-Commerce and 300 web pages on News). Note that the threshold for this test has been set to 5% (i.e) if 5% of the web pages have very different value for any two algorithms for the metric in consideration, then the two algorithms are said to be significantly different.

Cuts: The GE with diagonally positioned seeds and GE with seeds positioned using the centroids of the clusters from the QT clustering technique (S2) with big thresholds (1/10, 1/15, 1/20) form a single group. This means that these algorithms are not significantly different from each other, in accordance with table 8.1 on page 103. The strategy of placing seeds using the clusters from QT technique as seeds (S1) for GE with the bigger thresholds such as 1/10, 1/15 are not significantly different from using the centroids of the clusters from QT

technique (S2) for GE with thresholds such as 1/25 and 1/30. In general, using bigger thresholds such as 1/10 and 1/15 are significantly different from using smaller thresholds such as 1/35 and 1/40 irrespective of S1 (using clusters from QT technique as seeds for GE) or S2 (using the centroids of the clusters from QT technique as seeds for GE). This indicates that the thresholds used for the QT clustering technique that is used for positioning the seeds, irrespective of S1 or S2, plays a very important role in the segmentation of web pages.

Balance: The groups from the Dunn test for both the surface area and text area are identical. With respect to the metric "No.Of.Elements", the GE using a simple pre-clustering (algorithm 3 on page 58), produces identical results to the *K*means algorithms with all its reading strategies. This is the only change that is noticed when comparing with the other balance metrics (SA and TA). This once again proves that all balance metrics produce a similar evaluation. This is in accordance with the table 8.1 on page 103. This indicates that both the surface area and the text area indicates the same thing in terms of balance. GE with force measure (F-GE) with irrespective of the reading strategies belongs to the same group as using the QT clustering technique (cluster or centroids of cluster) for positioning the seeds for GE with the biggest threshold (1/10). This indicates that they are not significantly different from each other. Similarly, GE irrespective of the reading strategies are not significantly different from each other in terms of balance. *K*means irrespective of the reading strategies are not significantly different from each other in terms of balance. F-*K*means irrespective of the reading strategies are not significantly different from each other in terms of balance. This means that positioning the seeds either diagonally or in a "F" fashion or in a "Z" fashion does not any significant difference in terms of the balance between the zones.

Exterior Rectangle: It is seen that GE using the clusters as seeds from the QT clustering technique (S1) irrespective of the thresholds belong to the same group. Similarly, GE using the centroids of the clusters from the QT technique as seeds (S2) irrespective of the threshold belong to the same group. Thus this indicates that the thresholds for the QT technique (for both S1 or S2) is not important for the exterior rectangle. It can be seen that GE with diagonally positioned seeds and GE with the simple pre-clustering technique (GE Pre Cluster (GE P)) belong to the same group indicating that they are not significantly different from each other. F-*K*means with diagonally positioned seeds and with seeds positioned in the "F" fashion belongs to the same group, F-GE with diagonally positioned seeds and with seeds positioned in the "F" fashion belongs to the same group and *K*means with diagonally positioned seeds and with seeds positioned in the "F" fashion belongs to the same group. This indicates that both diagonal and "F" ways of positioning seeds are not significantly different in terms of exterior rectangle. But the "Z" way of positioning the seeds is significantly different from the other two ways (Diagonal and F), for *K*means, F-*K*means and F-GE, in terms of exterior rectangle.

Criterion	Groups	
Cuts	1	{GE D, S2 1/10, S2 1/15, S2 1/20}
	2	{S1 1/10, S1 1/15, S1 1/20, S2 1/25, S2 1/30}
	3	{S1 1/25, S1 1/30, S2 1/35, S2 1/40, S2 1/45, S2 1/50}
	4	{F-GE D, F-GE F, F-GE Z, GE P}
	5	{K-means D, K-means Z}
	6	{K-means F, F-K-means D, F-K-means Z}
	7	{F-K-means F}
	8	{GE F, GE Z}
	9	{S1 1/35, S1 1/40, S1 1/45, S1 1/50}
Surface Area	1	{K-means D, K-means F, K-means Z}
	2	{F-K-means D, F-K-means F, F-K-means Z, GE P }
	3	{GE D, GE F, GE Z}
	4	{F-GE D, F-GE F, F-GE Z, S1 1/10, S2 1/10}
	5	{S1 1/15, S1 1/20, S1 1/25, S1 1/40, S1 1/45, S1 1/50, S2 1/15, S2 1/20}
	6	{S1 1/30, S1 1/35, S2 1/25, S2 1/30, S2 1/35, S2 1/40, S2 1/45, S2 1/50}
Text Area	1	{K-means D, K-means F, K-means Z}
	2	{F-K-means D, F-K-means F, F-K-means Z, GE P }
	3	{GE D, GE F, GE Z}
	4	{F-GE D, F-GE F, F-GE Z, S1 1/10, S2 1/10}
	5	{S1 1/15, S1 1/20, S1 1/25, S1 1/40, S1 1/45, S1 1/50, S2 1/15, S2 1/20}
	6	{S1 1/30, S1 1/35, S2 1/25, S2 1/30, S2 1/35, S2 1/40, S2 1/45, S2 1/50}
No.Of.Elements	1	{K-means D, K-means F, K-means Z, GE P}
	2	{F-K-means D, F-K-means F, F-K-means Z }
	3	{GE D, GE F, GE Z}
	4	{F-GE D, F-GE F, F-GE Z, S1 1/10, S2 1/10}
	5	{S1 1/15, S1 1/20, S1 1/25, S1 1/40, S1 1/45, S1 1/50, S2 1/15, S2 1/20}
	6	{S1 1/30, S1 1/35, S2 1/25, S2 1/30, S2 1/35, S2 1/40, S2 1/45, S2 1/50}
Exterior Rectangle	1	{GE D, GE P}
	2	{F-Kmeans D, F-Kmeans F, GE F, S2 1/10, S2 1/15, S2 1/20, S2 1/25, S2 1/30, S2 1/35, S2 1/40, S2 1/45, S2 1/50}
	3	{F-Kmeans Z, GE Z, S1 1/10, S1 1/15, S1 1/20, S1 1/25, S1 1/30, S1 1/35, S1 1/40, S1 1/45, S1 1/50}
	4	{F-GE D, F-GE F, Kmeans D, Kmeans F}
	5	{Kmeans Z, F-GE Z}

Table 8.2: Dunn test analysis for the 31 algorithms over the 4 different metrics. Algorithms within a group show no statistical difference between them. Rank evidences the performance order for each criterion.

8.4 Evaluation by category

As mentioned earlier the set of web pages used for evaluations belong to 3 different categories - Tourism, e-Commerce and News. In the section 8.2, the evaluation was performed irrespective of categories. However, there could be a certain algorithm better suited for a certain category of web pages. Thus in this section, the evaluation is done for each specific category. The evaluation performed follows the automatic evaluation criteria as stated in 8.2 as there not sufficient manually annotated web pages from each category that can act as ground truth to perform the evaluation of cluster metrics as stated in 7.2.

8.4.1 Tourism

The table 8.3 on page 106 gives the automatic evaluation metrics for the 300 tourism web pages. Tourism web pages aims at attracting tourists to the place. The web pages thus uses several images and necessary information about the place. However, these pages might not use strict alignment constraints for displaying their content. This is evident with the values for the cuts metrics specifically for the GE algorithms with all possible positioning of seeds - 2.80, 3.12 and 3.54 for GE diagonal, GE F and GE Z respectively, which is higher than the Kmeans with all reading strategies (2.00,2.51 and 2.67 for diagonal, F and Z strategies respectively). Though this is a small difference in the number, as the number of web pages tested is 300, this small difference could make a significant difference. Since the alignment might not be evident for all tourism web pages, the GE algorithm does not efficiently apply for these type

of web pages as the GE considers the alignment and font similarity features for the segmentation process. On the contrary, the *K*means algorithms have less values for cuts though the only feature used for this clustering is the distance. On the other hand, the Exterior Rectangle metric is smaller for the GE algorithm with the seeds positioned diagonally. This indicates again that the GE algorithms produces zones which are mostly rectangular in shape and do not overlap. In terms of balance, the *K*means still manages to preserve the balance between the zones in terms of both surface area and text area - 14.65 for surface area, 11.39 for text area and 13.21 for the no.of.elements with respect to *K*means Diagonal. As mentioned previously, imbalance does not indicate bad segmentation but within the framework for TAG THUNDER it is preferred to have a balance between the zones to help vocalization. Considering the above mentioned metrics, it can be concluded that for tourism web pages, because of their structure and motive, *K*means algorithms fare well over GE algorithms. The GE with diagonal reading strategy also produces very close results to the *K*means algorithm with all reading strategies.

	Nb. of Cuts Avg. $\pm\sigma$	SA Avg stdev. $\pm\sigma$	TA Avg stdev. $\pm\sigma$	No.OF.Elements Avg stdev. $\pm\sigma$	Ext. Rect. Avg. $\pm\sigma$
K-means diagonal	2.00 ± 2.36	14.65 ± 7.29	11.39 ± 4.84	13.21 ± 5.05	1.41 ± 1.31
*K*means F strategy	2.51 ± 2.92	14.56 ± 6.52	12.00 ± 4.76	13.67 ± 6.45	1.27 ± 1.34
*K*means Z strategy	2.67 ± 2.91	15.14 ± 7.10	13.89 ± 5.41	13.90 ± 4.76	1.75 ± 1.11
F-*K*-means diagonal	2.83 ± 3.43	23.54 ± 8.10	19.11 ± 7.22	22.33 ± 6.83	1.53 ± 1.38
F-*K*-means F strategy	3.08 ± 3.81	21.76 ± 7.56	18.93 ± 7.56	19.77 ± 6.99	1.07 ± 1.25
F-*K*-means Z strategy	3.45 ± 3.41	25.64 ± 6.78	20.09 ± 7.13	23.23 ± 5.90	1.34 ± 1.10
GE diagonal	2.80 ± 1.40	19.02 ± 6.71	15.69 ± 5.10	17.77 ± 5.55	0.80 ± 1.15
GE F strategy	3.12 ± 1.23	20.90 ± 7.80	24.89 ± 7.10	22.34 ± 7.77	1.98 ± 1.12
GE Z strategy	3.54 ± 1.11	21.90 ± 5.45	22.55 ± 5.75	20.59 ± 6.10	1.13 ± 1.13
GE Force Diagonal	2.89 ± 1.39	20.09 ± 5.34	23.78 ± 5.10	22.22 ± 5.78	1.11 ± 1.11
GE Force F Strategy	3.33 ± 1.21	24.57 ± 7.45	23.90 ± 7.65	23.89 ± 7.08	1.43 ± 1.03
GE Force Z Strategy	3.56 ± 1.12	25.49 ± 7.83	23.79 ± 6.99	24.13 ± 7.22	1.68 ± 1.34
S1 1/10	1.94 ± 2.79	26.00 ± 7.84	22.17 ± 6.76	25.09 ± 6.33	1.65 ± 1.56
S1 1/15	2.04 ± 2.56	23.09 ± 7.03	21.65 ± 5.89	22.21 ± 5.95	1.57 ± 1.45
S1 1/20	2.45 ± 2.09	22.67 ± 6.98	22.33 ± 6.54	22.89 ± 6.11	1.53 ± 1.21
S1 1/25	2.21 ± 1.97	22.42 ± 7.02	23.41 ± 7.42	22.00 ± 7.32	1.56 ± 1.22
S1 1/30	2.28 ± 1.52	23.44 ± 7.67	24.03 ± 7.53	24.43 ± 7.78	1.34 ± 1.99
S1 1/35	2.07 ± 1.34	23.34 ± 6.89	23.96 ± 7.64	23.21 ± 7.18	1.35 ± 1.38
S1 1/40	2.03 ± 1.43	24.78 ± 7.69	24.03 ± 7.46	23.30 ± 7.34	1.93 ± 1.22
S1 1/45	2.03 ± 1.45	26.98 ± 7.85	22.54 ± 6.77	24.43 ± 7.30	1.44 ± 1.00
S1 1/50	2.00 ± 1.23	24.09 ± 7.22	22.66 ± 6.33	22.08 ± 6.30	1.32 ± 0.89
GE Pre cluster	2.94 ± 3.15	20.23 ± 7.42	17.63 ± 6.48	18.99 ± 6.44	0.71 ± 1.15
S2 1/10	2.56 ± 2.33	26.92 ± 7.80	23.73 ± 6.92	24.33 ± 7.11	1.65 ± 1.69
S2 1/15	2.60 ± 2.30	27.89 ± 7.89	24.79 ± 6.45	25.24 ± 6.33	1.54 ± 1.23
S2 1/20	2.70 ± 2.42	28.87 ± 7.91	25.75 ± 7.17	27.88 ± 7.29	1.58 ± 1.61
S2 1/25	2.78 ± 2.32	29.05 ± 7.90	26.42 ± 7.33	27.77 ± 6.39	1.50 ± 1.61
S2 1/30	2.79 ± 2.57	29.42 ± 7.99	26.71 ± 7.60	25.33 ± 7.55	1.35 ± 1.41
S2 1/35	2.87 ± 2.84	28.57 ± 8.35	26.10 ± 7.81	24.09 ± 7.22	1.29 ± 1.33
S2 1/40	2.96 ± 2.89	28.57 ± 8.36	25.92 ± 8.12	26.23 ± 8.09	1.35 ± 1.47
S2 1/45	2.57 ± 2.16	28.75 ± 8.03	26.20 ± 7.73	24.39 ± 7.36	1.46 ± 1.68
S2 1/50	2.57 ± 2.07	28.32 ± 8.10	26.03 ± 7.74	27.05 ± 7.44	1.56 ± 1.71

Table 8.3: Automatic Evaluation for Tourism web pages

8.4.2 E-Commerce

E-Commerce web pages are designed to showcase objects and to attract people to get to buy those objects. There are several images depicting various objects and texts describing the objects showcased. Thus these type of web pages tend to have equal amount of images and text. Some web site developers prefer to align object in their web page while the others do not. Some creators use a

lot to images while the others are more descriptive in nature. In general, these design options are dependent on the type of objects the web page is associated with. The automatic evaluation metrics for the 300 web pages that have been segmented using the algorithms presented in the previous chapters are listed in table 8.4 on page 108. With respect to the cuts, GE with its various reading strategies evidences the lowest scores- 1.50, 1.64 and 1.78 for the diagonal, F and Z positioning of seeds respectively. On the other hand, the *K*means algorithm with its various reading strategies have the highest values - 5.41, 6.56 and 6.75 for the diagonal, F and Z positioning of seeds respectively. This indicates that the distance is not the only feature that is necessary for the segmentation of e-commerce web pages and the fact that including alignment and font similarities improves the segmentation in a considerable manner. With respect to the balance metric, the *K*means algorithms evidences the lowest imbalance with a surface area of 11.73, text area of 10.74 and 11.06 for the no.of.elements for *K*means with seeds positioned diagonally. However, the balance metrics for GE diagonal closely follows the *K*means algorithms with a surface area of 19.36, text area of 18.57 and no.of.elements of 18.54. With respect to the exterior rectangle measure, GE with seeds positioned using the centroids formed from the QT clustering technique(S2) outperforms the other methods with a value of 0.46 for a threshold of 1/10. GE with seeds positioned diagonally has a value of 0.86 while *K*means with seeds positioned diagonally has a value of 1.40. It is clear that GE with its various positioning of seeds outperforms the *K*means algorithms with its various positioning of seeds and thus proving that GE produces less intertwined zones. For the above factors, it could be concluded that for an e-commerce web page, GE is the most suited algorithm.

	Nb. of Cuts Avg. $\pm\sigma$	SA Avg stdev. $\pm\sigma$	TA Avg stdev. $\pm\sigma$	No.Of.Elements Avg stdev. $\pm\sigma$	Ext. Rect. Avg. $\pm\sigma$
K-means diagonal	5.41 ± 9.12	11.73 ± 6.22	10.74 ± 5.38	11.06 ± 6.02	1.40 ± 1.38
Kmeans F strategy	6.56 ± 4.05	14.85 ± 7.30	13.32 ± 5.08	12.67 ± 5.58	1.00 ± 1.47
Kmeans Z strategy	6.75 ± 5.10	15.63 ± 7.10	16.77 ± 5.44	15.45 ± 5.33	1.22 ± 1.67
F-K-means diagonal	4.78 ± 6.59	22.98 ± 9.38	23.42 ± 11.15	22.90 ± 9.00	1.14 ± 1.48
F-K-means F strategy	5.17 ± 5.75	25.13 ± 11.08	23.59 ± 10.85	23.24 ± 9.22	1.19 ± 1.28
F-K-means Z strategy	5.67 ± 5.04	26.90 ± 9.45	28.77 ± 10.05	26.22 ± 9.03	1.30 ± 1.45
GE diagonal	1.50 ± 1.18	19.36 ± 7.08	18.57 ± 6.73	18.54 ± 6.32	0.86 ± 1.50
GE F strategy	1.64 ± 1.20	20.07 ± 7.10	22.33 ± 6.93	19.34 ± 5.99	0.90 ± 1.23
GE Z strategy	1.78 ± 1.45	21.30 ± 6.90	22.23 ± 7.32	21.22 ± 7.12	1.11 ± 1.22
GE Force Diagonal	2.34 ± 1.23	28.90 ± 7.60	29.06 ± 7.43	29.90 ± 7.34	1.45 ± 1.09
GE Force F Strategy	2.89 ± 1.22	30.56 ± 7.89	30.14 ± 6.77	28.96 ± 6.33	1.99 ± 1.02
GE Force Z Strategy	3.33 ± 1.11	31.31 ± 6.56	30.90 ± 7.88	30.28 ± 7.07	1.67 ± 1.55
S1 1/10	2.28 ± 6.13	27.43 ± 8.70	22.80 ± 7.93	24.89 ± 7.35	1.13 ± 1.52
S1 1/15	2.31 ± 5.89	26.44 ± 7.99	23.65 ± 7.47	24.67 ± 6.83	1.09 ± 1.33
S1 1/20	2.55 ± 6.10	25.49 ± 6.55	24.00 ± 6.45	24.67 ± 6.77	1.00 ± 1.21
S1 1/25	2.67 ± 5.60	27.09 ± 5.55	25.33 ± 5.98	26.39 ± 5.45	1.33 ± 1.53
S1 1/30	2.65 ± 4.65	28.44 ± 6.74	27.37 ± 5.58	27.33 ± 5.39	1.12 ± 1.09
S1 1/35	2.75 ± 6.04	30.78 ± 6.75	31.43 ± 6.98	30.05 ± 6.78	1.21 ± 1.45
S1 1/40	2.86 ± 5.33	31.45 ± 5.64	30.44 ± 5.78	31.24 ± 7.42	1.11 ± 0.99
S1 1/45	3.03 ± 4.80	32.98 ± 6.55	30.59 ± 5.99	30.00 ± 6.34	1.04 ± 1.21
S1 1/50	3.00 ± 4.44	33.70 ± 5.99	32.76 ± 6.94	33.33 ± 7.02	1.11 ± 1.11
GE Pre cluster	1.78 ± 2.46	20.87 ± 9.03	20.01 ± 8.02	20.20 ± 7.74	0.77 ± 1.40
S2 1/10	2.03 ± 3.36	31.82 ± 8.50	30.24 ± 8.96	32.64 ± 8.28	0.46 ± 0.93
S2 1/15	2.43 ± 2.89	28.98 ± 9.07	28.74 ± 9.67	28.30 ± 9.04	1.02 ± 0.89
S2 1/20	2.30 ± 4.50	31.20 ± 7.70	29.71 ± 8.73	28.78 ± 8.63	0.64 ± 1.06
S2 1/25	2.42 ± 3.92	33.20 ± 8.17	30.08 ± 8.68	28.07 ± 8.45	0.42 ± 0.89
S2 1/30	2.55 ± 4.86	33.56 ± 7.20	30.60 ± 8.23	28.18 ± 8.29	0.58 ± 1.00
S2 1/35	3.44 ± 5.13	31.91 ± 8.22	29.60 ± 8.44	27.76 ± 8.68	0.57 ± 0.86
S2 1/40	3.06 ± 5.53	32.78 ± 8.09	30.42 ± 8.32	28.67 ± 8.28	0.47 ± 0.92
S2 1/45	2.64 ± 5.44	32.64 ± 7.95	30.12 ± 8.50	28.43 ± 8.13	0.64 ± 1.14
S2 1/50	2.52 ± 5.62	32.41 ± 8.02	30.16 ± 8.27	28.38 ± 8.15	0.70 ± 1.17

Table 8.4: Automatic Evaluation for e-Commerce web pages

8.4.3 News

News web pages are generally heavy on content and also have good alignment features as alignment is good for presentation of news by category. The font features are also obvious in these sort of web pages as they help in distinguishing between various news snippets and to catch the attention to a particular news snippet. This is clearly visible from table 8.5 on page 109. In terms of the cuts, the Guided Expansion (GE) with the seeds positioned diagonally have the lowest value of 0.21. GE with all the other positioning methods also have quite a low value of cuts when compared with the K means algorithms. GE with seeds positioned using the centroids of the clusters from the QT clustering technique (S2) has a value of 1.33 (for a threshold of 1/10) being the second lowest value. Thus it is quite clear that the introduction of the alignment and the font features helps in producing a better segmentation. In terms of balance, the GE with diagonal positioning of seeds has a balance of 19.08 for surface area, 18.48 for the Text area and 19.05 for the no.of.elements. This is not the best in terms of balance but not the worst either. The GE with seeds which are the centroids of the clusters formed from QT technique (S2) has a high imbalance among GE algorithms (28.09 for surface area, 27.16 for text area and 27.68 for the no.of.elements with a threshold of 1/10. 32.12 for surface area, 31.32 for text area and 31.37 for the no.of.elements with a threshold of 1/50) . GE with seeds as clusters from the QT clustering technique (S1) also has a high imbalance (22.43 for surface area, 21.00 for text area and 21.05 for the no.of.elements with a threshold of 1/10. 32.57 for surface area, 32.89 for text area and 32.67 for the no.of.elements with a threshold of 1/50). With

respect to the news web pages the main content generally is huge and forms a huge part of the web page thus causing imbalance between the zones from the segmentation process. The exterior rectangle metric is the best for the GE with the diagonally positioned seeds with a value of 0.26. For a news web page, the main contents, the menus, the header and footer are well separated such that the probability of formation of intertwined zones are less. Again, it has to be noted that the alignment and font features help in formation of zones that are not intertwined. Thus with all these metric, it could be concluded that GE algorithm with the diagonally positioned seeds is much better for news web pages. This can be evidenced in figures 5.8, 5.9 and 5.10 on pages 50, 50 and 51 respectively. The segmentation produced by the GE algorithm (figure 5.10 on page 51) produces the best segmentation for the news web page under consideration.

	Nb. of Cuts Avg. $\pm\sigma$	SA Avg stdev. $\pm\sigma$	TA Avg stdev. $\pm\sigma$	No.Of.Elements Avg stdev. $\pm\sigma$	Ext. Rect. Avg. $\pm\sigma$
K-means diagonal	2.26 ± 4.68	9.67 ± 5.72	9.32 ± 4.58	9.54 ± 5.15	1.51 ± 1.52
Kmeans F strategy	3.81 ± 6.27	10.43 ± 6.36	9.92 ± 4.36	10.45 ± 5.39	1.01 ± 1.15
Kmeans Z strategy	3.41 ± 5.63	14.34 ± 6.45	10.34 ± 6.77	13.63 ± 6.45	1.34 ± 1.85
F-K-means diagonal	2.74 ± 8.16	17.02 ± 7.24	15.86 ± 8.03	15.56 ± 7.89	1.23 ± 1.41
F-K-means F strategy	3.08 ± 9.02	17.84 ± 9.42	16.33 ± 8.67	16.89 ± 8.43	0.87 ± 0.98
F-K-means Z strategy	3.12 ± 10.09	19.00 ± 8.89	19.77 ± 9.00	19.13 ± 9.12	1.38 ± 1.85
GE diagonal	0.21 ± 0.70	19.08 ± 6.35	18.48 ± 6.12	19.05 ± 6.09	0.26 ± 0.66
GE F strategy	2.34 ± 3.10	20.00 ± 6.45	20.78 ± 7.10	20.27 ± 6.53	0.98 ± 1.00
GE Z strategy	2.33 ± 3.11	20.67 ± 6.43	23.45 ± 6.90	22.89 ± 6.56	1.00 ± 0.94
GE Force Diagonal	3.23 ± 2.11	24.56 ± 5.67	23.45 ± 7.13	24.67 ± 7.27	1.45 ± 1.22
GE Force F Strategy	3.22 ± 2.13	25.25 ± 5.66	25.46 ± 7.34	25.56 ± 7.25	1.32 ± 1.18
GE Force Z Strategy	3.67 ± 2.44	27.75 ± 6.65	26.65 ± 6.90	27.17 ± 6.58	1.65 ± 1.21
S1 1/10	2.10 ± 4.89	22.43 ± 7.85	21.00 ± 7.53	21.05 ± 7.53	1.16 ± 1.55
S1 1/15	2.25 ± 3.32	23.45 ± 6.89	22.89 ± 7.21	23.02 ± 6.52	1.11 ± 1.03
S1 1/20	2.88 ± 4.02	25.23 ± 6.52	26.43 ± 6.55	25.25 ± 6.39	1.04 ± 1.22
S1 1/25	3.11 ± 3.78	27.56 ± 6.04	27.45 ± 5.59	27.63 ± 6.37	1.23 ± 0.89
S1 1/30	3.33 ± 4.02	28.53 ± 6.00	28.69 ± 5.50	28.50 ± 5.64	1.11 ± 0.90
S1 1/35	3.49 ± 4.45	31.24 ± 6.23	31.23 ± 6.05	31.58 ± 6.17	1.03 ± 1.17
S1 1/40	3.64 ± 3.98	31.76 ± 5.86	31.99 ± 5.75	31.86 ± 5.78	0.87 ± 0.92
S1 1/45	3.75 ± 4.01	32.45 ± 5.68	33.65 ± 6.08	33.19 ± 6.31	0.98 ± 1.11
S1 1/50	3.88 ± 3.56	32.57 ± 6.34	32.89 ± 6.02	32.67 ± 6.26	1.01 ± 1.22
GE Pre cluster	3.40 ± 9.41	15.34 ± 7.84	14.68 ± 7.77	15.56 ± 7.33	0.48 ± 0.83
S2 1/10	1.33 ± 9.68	28.09 ± 7.77	27.16 ± 7.79	27.68 ± 7.45	0.48 ± 0.93
S2 1/15	1.48 ± 8.09	29.06 ± 6.89	30.89 ± 8.54	30.03 ± 8.37	0.52 ± 1.00
S2 1/20	1.55 ± 6.65	31.16 ± 8.32	30.18 ± 8.25	30.74 ± 8.45	0.55 ± 1.04
S2 1/25	1.89 ± 7.07	31.67 ± 8.39	30.84 ± 8.04	30.86 ± 8.58	0.53 ± 1.05
S2 1/30	1.58 ± 7.50	31.51 ± 7.85	30.82 ± 7.91	30.56 ± 7.94	0.59 ± 1.09
S2 1/35	1.24 ± 6.56	32.36 ± 8.67	31.58 ± 8.16	31.43 ± 8.13	0.58 ± 1.02
S2 1/40	2.30 ± 8.07	31.18 ± 8.34	30.76 ± 8.16	30.71 ± 8.38	0.62 ± 1.04
S2 1/45	2.58 ± 10.31	31.95 ± 8.83	31.18 ± 8.53	31.27 ± 8.34	0.54 ± 1.01
S2 1/50	2.07 ± 8.17	32.12 ± 8.96	31.32 ± 8.40	31.37 ± 7.78	0.46 ± 0.93

Table 8.5: Automatic Evaluation for News web pages

8.5 Conclusion

The metrics presented in this chapter have been designed based on the evaluation done by the experts (presented in section 7.1 of chapter 7). This has been realised by questioning the experts to know their intentions when they penalise a segmentation. The metrics thus developed reflect the number of cuts, balance and the exterior rectangle for the segmentation as presented and explained in this chapter.

Overall, when the 900 web pages are considered (300 tourism, 300 E-Commerce

and 300 News), the metrics show that GE perform well in terms of cuts when used with the diagonal positioning of seeds. The balance metric is strong for the Kmeans algorithms indicating that these algorithms give a better balance between the zones. The balance metric for GE with seeds positioned diagonally stands second in terms of producing balanced zones. In terms of the exterior rectangle, the GE produces best results when the seeds are positioned diagonally. Thus over all, it could be said that GE with diagonally positioned seeds performs well over 900 web pages.

In this chapter each category of web pages are examined separately to know which algorithm suits the particular category better. It has been proved that the Kmeans algorithms better suit the tourism web pages because of their structure.The GE algorithm with diagonal positioning of seeds suit better for the E-Commerce and News web pages followed by GE with seeds positioned with the QT algorithm with remaining seeds positioned with the maximum average distance (Algorithm 5 on page 60). Again, this is because of the structure of these web pages having more alignment and font similarities making use of all the features that the GE algorithm takes into account.

Chapter 9

Comparison with previous works

9.1 Introduction

The task of non visual skimming and scanning requires the number of clusters to be 5, such that each cluster can be converted to a soundscape representing a zone and enabling the comparisons between the algorithms presented in the dissertation. Thus in the previous chapters the number of clusters to be formed has been set to 5. However, the previous works done on web page segmentation do not have the criteria of the number of clusters. And thus in order to be able to evaluate the segmentation with the previously existing work, in this chapter the number of clusters are varied.

While using the diagonal reading strategy, the number of seeds positioned can be increased as the diagonal does not have any intersections to place seeds. The way to position 4,6 and 8 seeds along the diagonal are shown in figures 9.1, 9.2 and 9.3 on pages 112, 112 and 112 respectively. In these experiments, the "F" and "Z" reading strategies are not used as these reading strategies have not produced results better than the diagonal reading strategy. Since the diagonal reading strategy produce the best results, only this strategy is used for experimentation by varying the number of clusters.

The number of clusters can also be changed while using pre clustering techniques (Algorithms 3,4,5 on pages 58,59,60 respectively) - as the pre clustering techniques used in these algorithms do not require positioning of any seeds, this allows for varying the number of clusters.

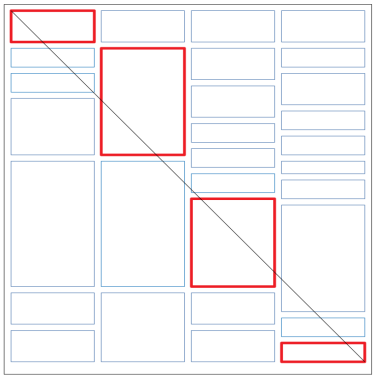

Figure 9.1: *Positioning 4 seeds along the diagonal of a web page.*

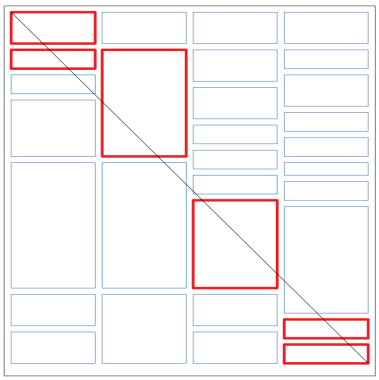

Figure 9.2: *Positioning 6 seeds along the diagonal of a web page.*

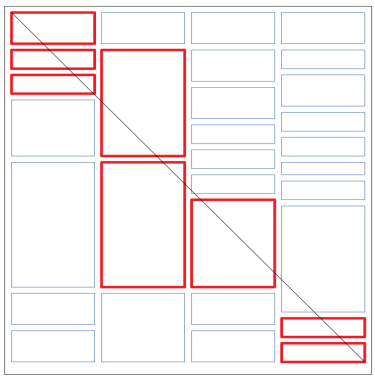

Figure 9.3: *Positioning 8 seeds along the diagonal of a web page.*

9.2 Experiments

9.2.1 Manual segmentation

In order to do this evaluation, 50 web pages from the corpus - 20 tourism web pages, 12 e-Commerce pages and 18 news web pages, are manually segmented by experts with the number of clusters(k) as 2 to 8 depending on the requirements on each web page without any task in mind. The 50 web pages

Number of Clusters	Number of Web Pages
2	0
3	4
4	7
5	16
6	13
7	4
8	6

Table 9.1: Analysis for the manual segmentation of 50 web pages

considered are the same as the ones considered for the manual evaluation with 5 clusters (section 7.1 on page 70) and the experts are the same as well.

Based on the segmentation done manually by the experts with varying number of clusters, the statistics for the number of clusters required for each of the 50 web pages is shown in Table 9.1 on page 113. i.e. 4 out of the 50 web pages required 3 zones when they were manually segmented. For the task at hand, non visual skimming and scanning, the number of clusters (k), has been fixed to 5 for various reasons with respect to the task as explained in chapter 4. However, from table 9.1 on page 113, it can be seen that this is a valid assumption to make as 16 out of 50 web pages require 5 zones.

This sort of manual segmentation helps in comparing the algorithms presented in the dissertation with the already existing works specifically Block-O-matic(presented in section 3.3.5) and Box Clustering Segmentation(BCS) (presented in section 3.3.4). These manually segmented web pages are also compared with the ongoing work, whose presentation follows in the further subsection.

9.2.2 Multi-objective Clustering Segmentation (MCS)

Web Page Segmentation experiments have been performed in collaboration with some other colleagues [1] at the GREYC lab and the Indian Institute of Technology Patna, Patna, 801103 Bihar, India [2] using a K-means-based multi-objective clustering (MCS) approach Ramesh Jayashree et al. (2020). This approach does not fix the number of clusters to be identified but executes Kmeans multiple times with varying number of K (number of clusters). The quality of the different partitionings are then measured with respect to some cluster validity index, and the partitioning, which corresponds to the optimal value is selected. With respect to the positioning of initial seeds, an method that will select specific regions on the web page that maximizes the overall distance between the seeds is used. As such, both the number of clusters and the positionings of the seeds go through an evolutionary process that must maximize the overall quality of the subsequent partitioning based on

1. Researchers at GREYC: Myself, Gaël Dias, Fabrice Maurel and Stéphane Ferrari.
2. Researches from Patna, India: Srivatsa Ramesh Jayashree and Sriparna Saha

concurrent objectives.

The Multi-objective Clustering Segmentation (MCS) algorithm starts by creating a random population of assignments, where each assignment consists of a set of random cluster centers varying in K.A chromosome encodes a set of different cluster centers, i.e., a possible assignment and represents an assignment. A specific instance of the K-means algorithm is executed for each chromosome. Each chromosome is then evaluated based on a set of objective functions. A set of chromosomes is then selected to participate in the offspring reproduction process. This selection id done based on the non-dominated sorting genetic algorithm (NSGA-II) Deb et al. (2002). Before reproduction, a Self-organizing map(SOM) is then created such that solutions which are similar are mapped to neurons next to each other. This SOM helps in pruning the set of assignments to maintain the equilibrium and a certain degree of diversity. The pruned selected set of assignments is chosen to cross over and a new population is obtained. While the number of iterations is not reached, the new population is added to the old population and the process continues. When the number of iterations is reached, a set of Pareto-optimal solutions is obtained and a single solution is chosen using priority sorting.

With respect to the assignment step of Kmeans visual, logical and semantic distances are used. The visual distances includes border to border distance and alignment distance. The logical distance include DOM path distance and the DOM tag distance. The semantic distance is the textual similarity between two wen elements using the Doc2Vec method. With respect to the update step of the Kmeans algorithm, a virtual web element (rectangular box) is defined by its pixel coordinates averaged over the coordinates of the web elements that were assigned to it during clustering, and the continuous vector summarizing the textual contents of the web elements assigned to it.

To chose a chromosome to reproduce, several objective functions are used. (1) Davies-Bouldin Index (DB Index) - to measure the compactness and separateness of a partition. (2) DB-Border -DB index which is based on the border to border distance. (3) DB-Text - DB index to define the textual similarity to measure the compactness and separateness (4) SIA: Alignment Objective - A silhouette index to measure the pairwise alignment between the web elements (5) Cuts - number of HTML cuts between web elements.

9.2.3 Algorithms for segmentation

Kmeans (algorithm 1 on page 42) with diagonal positioning of seeds, Guided Expansion (algorithm 2) with diagonal positioning of seeds, Guided Expansion by using seeds from a simple pre clustering (algorithms 3 on page 58), Guided Expansion where the clusters formed from the QT clustering is used as seeds (algorithm 4) and Guided Expansion when the centroid of the clusters formed by the QT clustering technique is used as seeds and the remaining seeds are placed using the maximum average distance from the already existing seeds (algorithm 5 on page 60) are then used to segment the selected 50 web pages

knowing the number of zones required for each web page i.e. if a web page is segmented into 6 zones in the manual segmentation, then 6 zones are demanded from the above mentioned algorithms as well. It has to be noted that the experiments for varying the number of zones using the algorithms 3, 4 and 5 on pages 58, 59 and 60 respectively are done using a threshold of one tenth of the maximum border to border distance in a web page as this threshold has proved to produce the maximum results for the various cluster matrices as detailed in table 7.2.

9.3 Results

The results from the evaluation for all the cluster metrics are presented in table 9.2. From table 9.2 on page 116, it is clear that the Guided Expansion Algorithm with all its variants and the Kmeans algorithm produce superior results to the works presented in the literature (BOM and BCS) for all the cluster metrics. GE with diagonally positioned seeds has achieved the highest B3F1 score of 0.69 among the algorithms presented in this dissertation whereas BOM has achieved the score of 0.60 and BCS a score of 0.57. With respect to precision, GE with diagonally positioned seeds proves to have the highest value of 0.73 among the algorithms in the dissertation while BOM has a precision of 0.50 and BCS has a precision of 0.45. With respect to recall, GE with the centroids of the clusters from QT clustering technique as seeds (S2) (algorithm 5 on page 60) achieves the highest value of 0.76 while BOM achieves a value of 0.70 and BCS has a recall value of 0.60. Similarly, with respect to ARI, Jaccard and F&M GE with diagonally positioned seeds achieves values higher than BOM and BCS.

In table 9.2 on page 116, with respect to the multiobjective segmentation (MCS) approaches, Alignment (A), Geometric distance (G), Cuts (C) and Textual similarities (T) are the objectives considered for optimization. The abbreviations in table 9.2 on page 116 such as AGCT shows the order in which the objectives are considered for optimization. The table 9.2 on page 116shows that multiobjective segmentation approach gives better results than the proposed algorithms in the dissertation. AGCT/AGTC achieves the highest values for all cluster metrics (B3F1 - 0.77, Precision - 0.71, Recall - 0.87, ARI - 0.62, Jaccard - 0.63, F&M - 0.77). While TGAC has the lowest values among the MCS algorithms. Although, AGCT/AGTC and CAGT/CATG achieves better values for cluster metrics than the algorithms presented in this dissertation, GTAC and TGAC have comparable values. Based on the results of GTAC and TGAC, it can be noticed that the textual features are less discriminant when compared to the other features. While the results of AGCT/AGTC proves the importance of the alignment feature in the segmentation process.

	Algorithms	B3F1	Precision	Recall	ARI	Jaccard	F&M
Algorithms	K-means diagonal	0.63	0.70	0.57	0.40	0.41	0.58
	GE diagonal	0.69	0.73	0.67	0.47	0.48	0.65
	S1 1/10	0.63	0.60	0.68	0.29	0.39	0.57
	GE Pre cluster	0.63	0.69	0.59	0.37	0.40	0.57
	S2 1/10	0.68	0.63	0.76	0.36	0.45	0.63
RW(*)	BOM Sanoja and Gancarski (2014)	0.60	0.50	0.70	0.26	0.41	0.60
	BCS(**) Zeleny et al. (2017)	0.57	0.45	0.60	0.21	0.38	0.56
MCS	AGCT/AGTC	0.77	0.71	0.87	0.62	0.63	0.77
	CAGT/CATG	0.76	0.70	0.86	0.58	0.60	0.75
	GTAC	0.68	0.62	0.76	0.43	0.48	0.65
	TGAC	0.65	0.59	0.70	0.40	0.46	0.63

Table 9.2: Cluster Metrics for evaluation with already existing works. (*) RW stands for Related Work (**) Results have been computed using Zeleny et al. (2017)'s toolbox, but some rendering errors were present and only 13 web pages could be segmented; thus results are shown only for these examples.

Conclusion: The experiments performed with varying the number of clusters proves that the algorithms presented in this dissertation outperforms the already existing works on web page segmentation, specifically the Block-O-matic and the Box Clustering Segmentation (BCS) algorithm. This indicates that the designed not only suit the task in hand but also outperforms the works already available in the literature. However, the MCS algorithms performs better than the proposed algorithms in terms of the cluster metrics. But this approach uses an extensive search to identify the best positions to place the seeds and an extensive approach to optimize the features. Because of this extensive search approach, this algorithm needs a long computational time to segment a page and thus does not efficiently suit the task at hand. Because of this, it is very difficult to integrate these algorithm with a framework like TAG THUNDER for real time use for a task of non-visual skimming and scanning.

Chapter 10

Conclusions and Future works

10.1 Conclusions

In this dissertation, various algorithms for the segmentation of web pages for the specific task of non visual skimming and scanning were proposed, experiments were conducted and evaluations were performed. The algorithms and the subsequent findings will be concluded in this chapter leading to an overlook on the future direction of research.

As mentioned in the previous chapter, web page segmentation is the process of identifying coherent zones within a given web page. The task at hand in this dissertation is allowing skimming and scanning for the visually impaired people. For this purpose, the framework of TAG THUNDER, described in chapter 1, which has been specifically designed for this task, is used for experimentation and evaluations. The algorithms designed are for the "segmentation" module of the TAG THUNDER framework (detailed in chapter 1). Indeed, there are various criteria and aspects, described in chapter 1, which are considered to facilitate the task of non visual skimming and scanning.

Based on the works reviewed in the part I, their strengths, their weaknesses and their tasks, a clustering approach has been chosen. Multiple hybrid clustering algorithms for segmentation of web pages with task specific features has been designed and experimented in this dissertation. The hybrid algorithms described are the classical Kmeans, Kmeans with a force measure(F-Kmeans) and a hierarchical propagation technique called Guided Expansion. However, it has been proved, by means of a qualitative evaluation (chapter 7)) done by three experts, that the positions of the initial seeds play a very important role in the segmentation algorithms proposed. Thus various methods to position the initial seeds, based on the reading strategies on the web and the task at hand, are proposed and experimented, detailed in chapter 6.

The algorithms are evaluated in part III. The algorithms are evaluated for the usual cluster metrics based on a ground truth created using manual annotations (chapter 7). Also, the algorithms are evaluated automatically for the

metrics designed specifically for the task at hand. From table 7.2 on page 77, it is proven that the Guided Expansion algorithm with a diagonally placed seeds and Guided Expansion with seeds placed using the QT clustering technique(S2), give the best results. It can also be seen from the table 7.2 on page 77 that the Guided Expansion Algorithm gives better results irrespective of the way the initial seeds are placed, when compared to the classical Kmeans and the F-Kmeans algorithms. As stated in chapter 7, the superior performance of Guided Expansion due to the introduction of the certain features that are considered for the segmentation process. This local assignment at each step of the Guided Expansion algorithm allows more fine-grained decisions when compared to both K-means and F-K-means.

It has also been found that using a force measure to segment web pages creates small zones with one or two web elements in them. This, as explained in previous chapter, is again due to the initial position of the seeds i.e. if the initial seed falls on a small web element, the force of attraction it offers is very small and thus not allowing opportunities for expansion of zones.

The positioning of seeds using reading strategies (F and Z) on the web have not worked very well for the task at hand. Indeed while placing the seeds in a F or Z fashion on a web page places two seeds on the header (in case of both F and Z) and two seeds on the footer (in case of letter Z), causing a cut in the header and footer, which are meant to be kept together to maintain the coherency. Thus, although this was a very interesting approach to experiment and analyse, the results are not very satisfying.

It has also been found that using a QT clustering technique(S1 and S2), to position the initial seeds, helps identify the area where the probability of expansion is high, which can be used as initial seeds, and thus being more efficient in forming coherent zones. It has been noticed that using the entire clusters formed in the QT clustering technique as seeds(S1), is not very efficient as with decreasing thresholds small clusters (frequently with just one element, called singletons) are formed after the QT algorithm. These singletons appear to be next to each other, thus restricting the growth of the zones when a Guided Expansion is used with them as seeds. In order to prove this, a threshold analysis for the QT clustering technique has been performed, detailed in table 6.1 on page 57. It has been identified that as the threshold decreases, the required number of seeds are not possible to be identified,thus causing the formation of singletons in S1, i.e. there are several web pages where the required number of 5 seeds are not possible to be identified as the threshold decreases from one tenth of the border to border distance to one fiftieth of the border to border distance. This lead to finding a method to position the remaining seeds while using a QT clustering technique. Thus a method of using the maximum average distance between the already formed clusters has been proposed (S2). Though this method is efficient for positioning remaining seeds, as the threshold decreases, there are more than one seed that need to be positioned using the maximum average distance. This minimizes the use of QT technique to identify seeds. Thus making bigger thresholds produce better results because of the better positioning of initial seeds.

Comparison with works in the literature has also been performed. However, for this purpose, a varying number of cluster is used. Although this change violates a necessary requirement for the task, it has to be done so as to maintain uniformity with the existing work and to help comparison. The Block-O-matic algorithm and the Box Clustering segmentation algorithm from literature work have been used as baseline for comparison. It has been found that algorithms proposed in this dissertation outperforms both Block-O-matic and Box clustering algorithm for all cluster metrics (table 9.2 on page 116).

10.2 Future Works

Although the algorithms presented in this dissertation suit the task at hand and their performance are good for the segmentation of web pages, there are a great deal of future work directions that could be proposed.

Ongoing Work: The GE algorithm could be enriched using semantic features with the textual features of the web pages. For this purpose, few experiments have been started. The text similarities have been introduced using the basic cosine similarities between the vectors represented by the text within the web elements. Two blocks with text similarity of 0.75 or above are considered to be put together in the same segment. Initial experiments use the diagonal reading strategy to position seeds and the number of zones required has been set to 5 taking the task of non visual skimming and scanning into consideration.

The first experiments have been evaluated with the 50 web pages (20 tourism web pages, 12 E-Commerce web pages and 18 news web pages) for which the ground truth has been developed (detailed in chapter 7). The evaluations were performed for the cluster metrics (section 7.2 of chapter 7). The evaluation is presented in table 10.1 on page 119. From table 10.1 on page 119, it can been noted that while using the textual features the ARI, Jaccard and the F&M index have improved very much. The B3F1 score has also increased a bit (0.72). This shows that while using GE with the additional textual features, the segmentation produced is more close to the human segmented web pages (ground truth). This shows results closer to using the Multi-objective Clustering Segmentation (MCS) algorithm.

	B3F1 Avg.	Precision Avg	Recall Avg	ARI Avg	Jaccard Avg.	F and M Avg.
GE diagonal	0.69	0.73	0.67	0.47	0.48	0.65
GE diagonal with text	0.72	0.70	0.69	0.75	0.77	0.65

Table 10.1: Cluster Metrics for GE with diagonal reading strategy with distance, alignment, font similarities and text similarities as the features considered in that particular order.

Though the table 10.1 on page 119 shows interesting results, the experimenta-

tion have been performed only for 50 web pages. More experiments need to be conducted. The impacts made by textual features on different categories of web pages needs to be explored as well. The textual similarities between the web elements has been calculated using by converting the content of the web elements to vectors and the cosine similarity of the vectors are used as the textual similarity. However, more powerful models may be used, such as specifically-tuned transformer-based language models like BERT Devlin et al. (2018). Text density features could also be introduced as proposed by Kohlschütter and Nejdl (2008) in the Block Fusion algorithm. Embedding maps Yang et al. (2017) that combine visual and textual information into some latent space could also be an experimented. Thus this proves to be an interesting future direction for research.

Perspectives on features: Apart from adding textual features to the GE algorithm, there are several possibilities for research to better the algorithms. Firstly, in terms of the algorithms, the Kmeans algorithms can be enriched using the other features like alignment and font similarities. However, as discussed there is a the issue of averaging the features of all web elements belonging to a cluster to be used as the centroid for the future iterations. For this purpose, an actual web element that is the closest to the virtual centroid (formed from averaging the web elements) could be chosen as the centroid for future iterations. The Kmeans algorithm could be modified to include the textual and visual features. Both the Kmeans and the GE algorithms can be enriched using other features for segmentation as well. Particularly, the visual cues used in the algorithms can be enriched using other cues from the css such as background-color and textures. Also, in the experiments and algorithms presented in this dissertation, the "cuts" metric, representing the logical aspect of the web page, has been used for the evaluation and not as a feature for the clustering as well. However, the MCS algorithms have proved that the "cuts" is an important feature to consider for the clustering process itself and not just for the evaluations. There could be some weights attached to each feature to better know how each feature influences the clustering process. Experiments could be done using these features along with the Kmeans and Guided Expansion (GE).

Perspectives on positioning of seeds: With respect to the positioning of seeds, the initial seeds could be positioned in areas with the highest visual dissimilarities. While using the force measure, the seeds could be placed on web elements that have more or less the same surface area such that the force of attraction works equally with all the seeds. Other pre clustering methods to position the seeds could be tried as well.

Thirdly, with respect to the quantitative/automatic evaluations metrics, other metrics to measure the compactness and separateness of the clusters could be designed. Metrics to measure the semantic similarities and similarities of visual cues between and within the formed clusters could be developed. The balance aspect could be enriched to represent more closely the human segmentation of

web pages.

Also, with respect to experiments with other categories of web pages, it has been noted that certain algorithms suit well for certain web pages and thus there could be some machine learning techniques to identify the most suited algorithm for the specific web page. Apart from the three categories (tourism, e-commerce and news), there are several other categories of web pages available on the internet and thus more experiments on different categories of web pages could be done. The algorithms presented in the dissertation are language independent, however, when the textual aspect is integrated within the algorithms, then experiments could be done on web pages with different languages.

TAG THUNDER project: Finally, with respect to the project of TAG THUNDER, there could be an interaction between the zones. i.e. the user could be allowed to zoom into a zone and that particular zone could be segmented again for the purpose of identifying the information the user requires. Experiments with the visually impaired people should be performed as well.

Perspectives on Evaluation: There needs to be two types of evaluation - one for each module and the other for the whole project of TAG THUNDER. In this thesis, there have been several measures suggested for the evaluation of the segmentation module. There could also be evaluation metrics designed for the textual features that have been explained in the "ongoing work" 10.2 (page 119). Two evaluation metrics could be proposed for this purpose. The first one should measure the semantic coherence within each segment and the second one should be able to score/penalize when texts are cut in between. For the purpose of evaluating the project of TAG THUNDER, experiments were scheduled with the visually impaired young people of IJA Toulouse in April 2020. Unfortunately, this had to be cancelled due to the COVID-19 pandemic. This will be scheduled again soon to test the whole framework of TAG THUNDER.

Bibliography

S. Acharya, S. Saha, J. G. Moreno, and G. Dias. Multi-objective search results clustering. In *25th International Conference on Computational Linguistics (COLING)*, pages 99–108, 2014.

M. E. Akpınar and Y. Yeşilada. Heuristic role detection of visual elements of web pages. In *International Conference on Web Engineering*, pages 123–131. Springer, 2013.

D. Alassi and R. Alhajj. Effectiveness of template detection on noise reduction and websites summarization. *Information Sciences*, 219:41–72, 2013.

S. Alcic and S. Conrad. Page segmentation by web content clustering. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, WIMS '11, New York, NY, USA, 2011. Association for Computing Machinery. ISBN 9781450301480. doi: 10.1145/1988688.1988717. URL https://doi.org/10.1145/1988688.1988717.

A. Alorf. *K-Means, Mean Shift, and SLIC Clustering Algorithms: A Comparison of Performance in Color-based Skin Segmentation*. PhD thesis, University of Pittsburgh, 2017.

J.-J. Andrew, S. Ferrari, F. Maurel, G. Dias, and E. Giguet. Web page segmentation for non visual skimming. In *33rd Pacific Asia Conference on Language, Information and Computation (PACLIC)*, 2019a.

J.-J. Andrew, S. Ferrari, F. Maurel, G. Dias, and E. Giguet. Model-driven web page segmentation for non visual access. In *16th International Conference of the Pacific Association for Computational Linguistics (PACLING)*, 2019b.

M. S. Aruljothi, M. S. Sivaranjani, and S. Sivakumari. Web page segmentation for small screen devices using tag path clustering approach.

C. Asakawa and H. Takagi. Annotation-based transcoding for nonvisual web access. In *Proceedings of the fourth international ACM conference on Assistive technologies*, pages 172–179. Citeseer, 2000.

N. Babich. *Z-Shaped Pattern For Reading Web Content*, 2017. URL https://uxplanet.org/z-shaped-pattern-for-reading-web-content-ce1135f92f1c. Last access on September 2019.

S. Baluja. Browsing on small screens: recasting web-page segmentation into an efficient machine learning framework. In *Proceedings of the 15th international conference on World Wide Web*, pages 33–42. ACM, 2006.

J. Barua, D. Patel, and A. K. Agrawal. Removing noise content from online news articles. In *Proceedings of the 20th International Conference on Management of Data*, pages 113–116. Computer Society of India, 2014.

Y. Borodin, J. Mahmud, I. V. Ramakrishnan, and A. Stent. The hearsay non-visual web browser. In *Proceedings of the 2007 International Cross-disciplinary Conference on Web Accessibility (W4A)*, W4A '07, pages 128–129, New York, NY, USA, 2007. ACM. ISBN 1-59593-590-8. doi: 10.1145/1243441.1243444. URL http://doi.acm.org/10.1145/1243441.1243444.

D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma. Extracting content structure for web pages based on visual representation. In *Proceedings of the 5th Asia-Pacific Web Conference on Web Technologies and Applications*, APWeb'03, pages 406–417, Berlin, Heidelberg, 2003a. Springer-Verlag. ISBN 3-540-02354-2. URL http://dl.acm.org/citation.cfm?id=1766091.1766143.

D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma. Vips: a vision-based page segmentation algorithm. 01 2003b.

J. Cao, B. Mao, and J. Luo. A segmentation method for web page analysis using shrinking and dividing. *International Journal of Parallel, Emergent and Distributed Systems*, 25(2):93–104, 2010.

D. Chakrabarti, R. Kumar, and K. Punera. A graph-theoretic approach to webpage segmentation. In *Proceedings of the 17th international conference on World Wide Web*, pages 377–386. ACM, 2008.

J. Chen, B. Zhou, J. Shi, H. Zhang, and Q. Fengwu. Function-based object model towards website adaptation. In *Proceedings of the 10th international conference on World Wide Web*, pages 587–596. ACM, 2001.

Y. Chen, W.-Y. Ma, and H.-J. Zhang. Detecting web page structure for adaptive viewing on small form factor devices. In *Proceedings of the 12th international conference on World Wide Web*, pages 225–233. ACM, 2003.

Y. Chen, X. Xie, W.-Y. Ma, and H.-J. Zhang. Adapting web pages for small-screen devices. *IEEE internet computing*, 9(1):50–56, 2005.

F. Y. Choi. Advances in domain independent linear text segmentation. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 26–33. Association for Computational Linguistics, 2000.

D. Chudasama, T. Patel, S. Joshi, and G. I. Prajapati. Image segmentation using morphological operations. *International Journal of Computer Applications*, 117(18), 2015.

J. C. Cunhe LI, Juan DONG. Extraction of informative blocks from web pages based on vips. 01 2010.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

O. J. Dunn. Multiple comparisons among means. *Journal of the American statistical association*, 56(293):52–64, 1961.

D. Fernandes, E. S. de Moura, A. S. da Silva, B. Ribeiro-Neto, and E. Braga. A site oriented method for segmenting web pages. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 215–224. ACM, 2011.

B. Fitzgerald. Implementation of an automated text segmentation system using hearst's texttiling algorithm, 2000.

S. Giraud, P. Thérouanne, and D. D. Steiner. Web accessibility: Filtering redundant and irrelevant information improves website usability for blind users. *International Journal of Human-Computer Studies*, 111:23–35, 2018.

X.-D. Gu, J. Chen, W.-Y. Ma, and G.-L. Chen. Visual based content understanding towards web adaptation. *AH*, 2:164–173, 2002.

J. Guerreiro and D. Gonçalves. Faster text-to-speeches: Enhancing blind people's information scanning with faster concurrent speech. In *17th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS)*, pages 3–11, 2015.

G. Hattori, K. Hoashi, K. Matsumoto, and F. Sugaya. Robust web page segmentation for mobile terminal using content-distances and page layout information. In *Proceedings of the 16th international conference on World Wide Web*, pages 361–370. ACM, 2007.

M. A. Hearst. Texttiling: A quantitative approach to discourse segmentation, 1993.

J. Hu and Y. Liu. *Analysis of Documents Born Digital*, pages 775–804. 01 2014. ISBN 978-0-85729-858-4. doi: 10.1007/978-0-85729-859-1_26.

Z. Jiang, H. Yin, Y. Wu, Y. Lyu, G. Min, and X. Zhang. Constructing novel block layouts for webpage analysis. *ACM Trans. Internet Technol.*, 19(3), July 2019. ISSN 1533-5399. doi: 10.1145/3326457. URL https://doi.org/10.1145/3326457.

C. Kohlschütter and W. Nejdl. A densitometric approach to web page segmentation. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1173–1182. ACM, 2008.

M. Kovacevic, M. Diligenti, M. Gori, and V. Milutinovic. Recognition of common areas in a web page using visual information: a possible application in a page classification. In *null*, page 250. IEEE, 2002.

H.-P. Kriegel and A. Zimek. Subspace clustering, ensemble clustering, alternative clustering, multiview clustering: What can we learn from each other. *Proceedings of MultiClustKDD*, 2010.

J.-M. Lecarpentier, E. Manishina, F. Maurel, S. Ferrari, and G. Dias. Tag Thunder: Web Page Skimming in Non Visual Environment Using Concurrent Speech. In *7th Workshop on Speech and Language Processing for Assistive Technologies (SLPAT 2016) associated to INTERSPEECH 2016*, San Francisco, United States, 2016. URL https://hal-preprod.archives-ouvertes.fr/hal-01496711.

C. X. Lin, Y. Yu, J. Han, and B. Liu. Hierarchical web-page clustering via in-page and cross-page link structures. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 222–229. Springer, 2010.

D. Lin. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2*, ACL '98/COLING '98, page 768–774, USA, 1998. Association for Computational Linguistics. doi: 10.3115/980691.980696. URL https://doi.org/10.3115/980691.980696.

S.-H. Lin and J.-M. Ho. Discovering informative content blocks from web documents. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 588–593. ACM, 2002.

W. Liu, X. Meng, and W. Meng. Vision-based web data records extraction. In *Proc. 9th international workshop on the web and databases*, pages 20–25, 2006.

X. Liu, H. Lin, and Y. Tian. Segmenting webpage with gomory-hu tree based clustering. *JSW*, 6:2421–2425, 2011.

J. MacQueen. Some methods for classification and analysis of multivariate observations. In *15th Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297, 1967.

J. Mahmud, Y. Borodin, D. Das, and I. Ramakrishnan. Combating information overload in non-visual web access using context. In *Proceedings of the 12th international conference on Intelligent user interfaces*, pages 341–344. ACM, 2007a.

J. U. Mahmud, Y. Borodin, and I. Ramakrishnan. Csurf: a context-driven non-visual web-browser. In *Proceedings of the 16th international conference on World Wide Web*, pages 31–40. ACM, 2007b.

T. Manabe and K. Tajima. Extracting logical hierarchical structure of html documents based on headings. *Proceedings of the VLDB Endowment*, 8(12): 1606–1617, 2015.

E. Manishina, J.-M. Lecarpentier, F. Maurel, S. Ferrari, and B. Maxence. Tag Thunder : Towards Non-Visual Web Page Skimming. In *18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*, 2016.

R. Manjula and A. Chilambuchelvan. Hauling templates from web pages using clustering techniques. *International Journal of Engineering Sciences & Emerging Technologies*, 5(2):119–126, 2013.

F. Maurel, G. Dias, S. Ferrari, J.-J. Andrew, and E. Giguet. Concurrent speech synthesis to improve document first glance for the blind. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 3, pages 10–17. IEEE, 2019.

F. Maurel, G. Dias, W. Safi, J.-M. Routoure, and P. Beust. "layout transposition for non-visual navigation of web pages by tactile feedback on mobile devices". journal micromachines 2020, 11(4), 376; special issue tactile sensing technology and systems. *Micromachines*, 11:376, 04 2020. doi: 10.3390/mi11040376.

R. Mihalcea, C. Corley, C. Strapparava, et al. Corpus-based and knowledge-based measures of text semantic similarity. In *AAAI*, volume 6, pages 775–780, 2006.

M. Morel and A. Lacheret-Dujour. " kali", synthèse vocale à partir du texte: de la conception à la mise en oeuvre. *Traitement automatique des langues*, 42:193–221, 2001.

C. K. Nguyen, L. Likforman-Sulem, J.-C. Moissinac, C. Faure, and J. Lardon. Web document analysis based on visual segmentation and page rendering. In *Document Analysis Systems (DAS), 2012 10th IAPR International Workshop on*, pages 354–358. IEEE, 2012.

J. Nielsen. *F-Shaped Pattern For Reading Web Content*, 2006. URL https://www.nngroup.com/articles/f-shaped-pattern-reading-web-content-discovered/.

S. Pandit, S. Gupta, et al. A comparative study on distance measuring approaches for clustering.

K. Pernice. *F-Shaped Pattern of Reading on the Web: Misunderstood, But Still Relevant (Even on Mobile)*, 2017. URL https://www.nngroup.com/articles/f-shaped-pattern-reading-web-content. Last access on September 2019.

V. Prince and A. Labadié. Text segmentation based on document understanding for information retrieval. In *International Conference on Application of Natural Language to Information Systems*, pages 295–304. Springer, 2007.

P. M. Rajdeepa B. segmenting web pages using correlation clustering and reducing noisy data using simple k means algorithm. 02 2014.

S. Ramesh Jayashree, G. Dias, J. J. Andrew, S. Saha, F. Maurel, and S. Ferrari. Web page segmentation using self-organized multi-objective clustering. Journal paper under submission, 2020.

L. Rokach and O. Maimon. Clustering methods. In *Data mining and knowledge discovery handbook*, pages 321–352. Springer, 2005.

W. Safi, F. Maurel, J.-M. Routoure, P. Beust, and G. Dias. Web-adapted supervised segmentation to improve a new tactile vision sensory substitution (tvss) technology. *Procedia Computer Science*, 52:35–42, 12 2015. doi: 10.1016/j.procs.2015.05.014.

A. Sanoja. Web page segmentation, evaluation and applications. 01 2015.

A. Sanoja and S. Gancarski. Block-o-matic: A web page segmentation framework. In *Multimedia Computing and Systems (ICMCS), 2014 International Conference on*, pages 595–600. IEEE, 2014.

H. Takagi, C. Asakawa, K. Fukuda, and J. Maeda. Site-wide annotation: reconstructing existing pages to be accessible. In *Proceedings of the fifth international ACM conference on Assistive technologies*, pages 81–88. ACM, 2002.

G. Vineel. Web page dom node characterization and its application to page segmentation. In *2009 IEEE International Conference on Internet Multimedia Services Architecture and Applications (IMSAA)*, pages 1–6. IEEE, 2009.

P. Xiang and Y. Shi. Recovering semantic relations from web pages based on visual cues.

J. Xin and H. Jiawei. *Quality Threshold Clustering*, pages 1–2. Springer US, Boston, MA, 2016. ISBN 978-1-4899-7502-7.

X. Yang and Y. Shi. Web page segmentation based on gestalt theory. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 2253–2256, 2007.

X. Yang, E. Yumer, P. Asente, M. Kraley, D. Kifer, and C. Lee Giles. Learning to extract semantic structure from documents using multimodal fully convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5315–5324, 2017.

Y. Yang, Y. Chen, and H. Zhang. Html page analysis based on visual cues. In *Web Document Analysis: Challenges and Opportunities*, pages 113–131. World Scientific, 2003.

L. Yi, B. Liu, and X. Li. Eliminating noisy information in web pages for data mining. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 296–305. ACM, 2003.

X. Yin and W. S. Lee. Understanding the function of web elements for mobile content delivery using random walk models. In *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web*, WWW '05, pages 1150–1151, New York, NY, USA, 2005. ACM. ISBN 1-59593-051-5. doi: 10.1145/1062745.1062913. URL http://doi.acm.org/10.1145/1062745.1062913.

J. Zeleny, R. Burget, and J. Zendulka. Box clustering segmentation: A new method for vision-based web page preprocessing. *Information Processing & Management*, 53(3):735–750, 2017.

A. Zhang, J. Jing, L. Kang, and L. Zhang. Precise web page segmentation based on semantic block headers detection. In *6th International Conference on Digital Content, Multimedia Technology and its Applications*, pages 63–68. IEEE, 2010.

Appendix A

List of Abbreviations

WPS	Web Page Segmentation
HTML	Hyper Text Markup Language
DOM	Document Object Model
MM	Mathematical Morphology
VIPS	Visual based Page Segmentation
BOM	Block-O-Matic
BCS	Box Clustering Segmentation
GE	Guided Expansion
QT	Quality Threshold
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
ARI	Adjusted Rand Index
F&M	Fowlkes–Mallows index
MCS	Multi-objective Clustering Segmentation

Task Oriented Web Page Segmentation

Résumé

Avec le développement régulier de l'internet, l'accessibilité des sites web à tous est essentielle mais l'accessibilité des pages web pour les personnes malvoyantes est un défi en soi. En général, une personne voyante utilise une stratégie de lecture complexe et non linéaire, comme le "skimming", qui consiste à obtenir une vue d'ensemble, et le "scanning", qui consiste à passer d'un domaine d'intérêt à un autre. Les processus d'exploration et de balayage sont basés sur plusieurs facteurs tels que la mise en page, la structure logique et les effets typographiques qui ne sont pas disponibles dans l'environnement non visuel, ce qui rend l'exploration et le balayage plutôt difficile. Le travail présenté dans cette thèse se concentre sur la segmentation des pages web pour rendre possible ces tâches de "skimming" et "scanning" non visuels. Le cadre de TAG THUNDER est utilisé à des fins d'expérimentation. Dans cette thèse, nous proposons une approche par clustering pour la segmentation, afin de satisfaire les critères imposés par la tâche. La technique bien établie de clustering Kmeans a été choisie pour expérimenter plusieurs adaptations guidées par la tâche. Une première variante de l'algorithme de Kmeans a été proposée, appelée F-Kmeans, qui utilise la métaphore de la force physique d'attraction des corps massifs. Nous proposons aussi une nouvelle technique de regroupement guidée par la tâche, intitulée Guided Expansion (GE). Cette technique est une sorte d'expansion hiérarchique où l'expansion de chaque zone (cluster) se fonde sur des décisions locales, contrairement à la méthode Kmeans. GE utilise en particulier une distance entre éléments. Une variante exploitant la mesure de force d'attraction a aussi été testée (F-Guided Expansion). Les algorithmes ont été testés avec différentes positions de graines initiales en suivant les stratégies de lecture utilisées sur le web et en utilisant également des techniques de pré-classement pour identifier les zones probables. Pour les expérimentations, les algorithmes avec les différentes méthodes de positionnement sont testés avec 900 pages web appartenant à trois catégories différentes - 300 pages web du tourisme, 300 pages web du commerce électronique et 300 pages web des actualités. L'évaluation se fait de deux manières - manuelle et automatique. Pour l'évaluation manuelle, un corpus de référence (ground truth) a été créé pour 50 pages web et des mesures de clustering standard sont utilisées pour l'évaluation. Sur la base de l'avis d'experts, des mesures automatiques ont été créées pour permettre l'évaluation automatique sur de grands corpus sans besoin de référence. Dans les évaluations manuelles et automatiques, GE avec des graines positionnées en diagonale s'avère surpasser les autres algorithmes.

Mots clés : Segmentation des pages web, clustering, accès non visuel

Abstract

With the regular development of the internet, the accessibility of web sites to every one is essential but accessibility of web pages for the visually disabled people is a challenge in itself. In general, a person with sight uses a complex and non-linear reading strategy such as skimming which is to get a global overview, and scanning which is to jump from one area of interest to another. The skimming and scanning processes are based on several factors like layout, logical structure and typographic effects which are unavailable the non visual environment thus making skimming and scanning a rather difficult task. The work presented in this dissertation focuses on the segmentation of web pages for the task of non visual skimming and scanning. For the purpose of experimentation the framework of TAG THUNDER is used. In this dissertation, a clustering technique for the purpose of segmentation is employed allowing to satisfy the task oriented criteria. The very well established Kmeans clustering technique has been used for experimentation with task oriented adaptations. A variation of the Kmeans algorithm has been proposed called F-Kmeans which uses the metaphor of the physical force of attraction. A task-oriented clustering technique known as Guided Expansion(GE) has been developed. This clustering technique follows a sort of hierarchical expansion using the features and expansion of the zones based on local decisions unlike Kmeans. As a variation of GE the force measure as the distance measure known as F-Guided Expansion. The algorithms have been tested with different positions of initial seeds following reading strategies used on the web and also using pre-clustering techniques to identify probable zones. For the purpose of experimentation, the algorithms with the various positioning methods are tested with 900 web pages belonging to three different categories – 300 web pages from Tourism, 300 web pages from E-commerce and 300 web pages from News. The evaluation is done in two ways - manual and automatic. For manual evaluation, a ground truth has been created for 50 web pages and standard cluster metrics are used for evaluation. Based on expert opinion, automatic metrics have been created to enable evaluation of huge corpus. In both the manual and automatic evaluations, GE with diagonally positioned seeds proves to outperform other algorithms.

Key Words: Web Page Segmentation, Clustering, Non visual access