

Interprétation rapide et non visuelle de pages Web : TAL et Tonnerre de mots

Jean-Marc Lecarpentier Fabrice Maurel Stéphane Ferrari Alexandre Beudin
GREYC, CNRS, Campus Côte de Nacre, Boulevard du Maréchal Juin, 14032 CAEN cedex 5
{firstname.lastname}@unicaen.fr

Résumé. L'accès rapide à un contenu précis dans une page web est généralement réalisé par une étape dite d'écrémage (ou *skimming*) pour avoir une vue globale, suivi d'une stratégie de balayage (ou *scanning*) pour repérer les informations voulues. Ces deux étapes naturelles pour une navigation visuelle n'existent pas dans le cas d'un accès non visuel à la page donnée. Nous présentons dans cet article le travail en cours sur le concept de tonnerre de mots (ou *tag thunder*), traduction à l'oral du concept de nuage de mots. Après avoir segmenté une page web en blocs, les mots-clés significatifs sont extraits de chaque bloc et pondérés afin d'obtenir un nuage de mots. Parmi ces mots, les mots-clés navigants permettent à l'utilisateur de naviguer en profondeur dans les divers blocs de la page. L'ensemble est transformé en tonnerre de mots par vocalisation. La présentation de ce travail en cours permet aussi de présenter les diverses pistes de recherche envisagées.

Abstract.

Tag thunder : providing skimming and scanning techniques in non visual situations

Skimming and scanning are two different strategies for *speed reading* and are used when accessing web pages. Skimming is used when we discover a page and it allows a reader to get a first glance of its content. Scanning usually follows, in order to get a more precise idea of the content or to find a specific content. However, these techniques are not available in non visual situations. This article introduces the concept of tag thunder, which aims to provide speed reading techniques similar to skimming and scanning to visually impaired people. A tag thunder is the oral transposition of a tag cloud, where keywords are presented with typographic effects depending on the word's importance. Within a tag thunder, each keyword is pronounced differently to enable the user to perceive its importance. Navigation keywords enable the user to select a corresponding block. This work in progress is based on three steps. First the web page is segmented to extract its main blocks. Then, for each block, keywords are extracted to create tag clouds. Finally, tag clouds are converted to tag thunders and presented to the user. Our solution is implemented within a Firefox extension and opens several research tracks.

Mots-clés : accessibilité, oral, tag thunder, nuage de mots, extraction, segmentation.

Keywords: accessibility, tag thunder, tag cloud, extraction, segmentation.

1 Introduction

La plupart des internautes découvrant une page web réagissent selon le même schéma : obtention d'une vue globale de la page suivie d'une recherche active et rapide d'informations spécifiques. La première étape, dite stratégie d'écrémage ou *skimming* permet d'appréhender les différentes zones de la page, leur disposition, etc. La seconde étape dite stratégie de balayage ou *scanning* permet à l'internaute de déterminer si la page consultée correspond effectivement à ses attentes en terme de contenu. Dans les deux cas, la réussite et la rapidité de ces stratégies dépendent fortement de la structuration de la page en zones bien identifiées (Dias & Conde, 2007), de la typographie et de la mise en forme des contenus (titres, éléments graphiques, zones distinctes, couleurs, etc). Pour les internautes en situation non visuelle, ces stratégies ne sont pas possibles (Ahmed *et al.*, 2012b) : par exemple la figure 1 montre ce qu'un voyant et un non-voyant perçoivent au premier abord après le chargement d'une page web. Des stratégies palliatives non visuelles existent comme par exemple l'accélération du débit pour les *screen readers*¹ (Borodin *et al.*, 2010), les commandes permettant de sauter d'un titre ou d'un lien à un autre, la lecture du début et fin de chaque paragraphe, etc. Malgré cela, ces stratégies sont encore loin d'égaler celles d'un utilisateur voyant (Bigham *et al.*, 2007).

1. Lecteur d'écran : système de synthèse de la parole couplé à une plage Braille (par exemple logiciel JAWS)

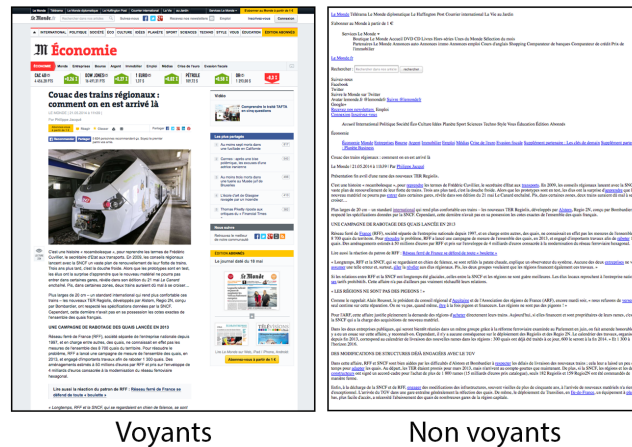


FIGURE 1 – Comparaison entre le visuel d'une page web et sa perception par un non-voyant avec les lecteurs d'écran



FIGURE 2 – Exemple de nuage de mots ou *tag cloud*

Cet article se focalise sur les possibilités d'accès non visuel, global et rapide à l'information. Nous proposons ici la transposition à l'oral du concept de nuage de mots (*tag cloud*) qui représente un ensemble de mots-clés plus ou moins mis en avant visuellement, comme le montre la figure 2². À l'instar des saillances dispositionnelles et typographiques du *tag cloud*, cette stratégie, appelée tonnerre de mots (ou *tag thunder*), veut organiser les mots en produisant des saillances spatiales et « typophoniques ».

La solution proposée dans cet article utilise des techniques de recherche d'information et de traitement automatique de la langue pour segmenter la page web en blocs, déterminer les mots-clés de chaque bloc et leur importance et enfin utilise un générateur vocal pour le rendu oral. Cette proposition est articulée comme suit. La partie suivante présente des travaux existants pour faciliter l'accès aux contenus dans des situations non visuelles. Nous présentons ensuite les concepts de tonnerre de mots et de mot-clé navigant, ainsi que les divers paramètres à prendre en compte. La partie suivante décrit les stratégies utilisées pour la segmentation et l'extraction des mots-clés qui constituent les tonnerres de mots. Elle est suivie d'une présentation de l'architecture logicielle qui implémente notre solution. Enfin, la dernière partie présente le travail restant à réaliser et les perspectives envisagées.

2 État de l'art

Des stratégies de *skimming* pour les non-voyants utilisent des techniques de résumé de texte (Ahmed *et al.*, 2012a). La page web est tout d'abord transformée en une séquence linéaire de texte puis des techniques classiques de sacs de mots permettent d'extraire des unités sémantiques. Cependant, cette approche montre ses limites pour la perception rapide des contenus de pages Web : la phase de linéarisation ne conserve pas le rôle important de la typo-disposition pour une bonne appréhension des contenus et la mise en place d'une boucle perception/action efficace (Dias & Conde, 2007). D'autres

2. Image de Anand S, <https://flic.kr/p/5BFE3V>, licence CC BY 2.0

propositions ont tenté d'intégrer cette observation et proposent des solutions avec différentes modalités d'accès au texte et à sa mise en forme (synthèse de la parole, retours tactiles ou vibrotactiles). La suite de cet état de l'art présente les principaux travaux qui ont nourri notre réflexion.

Pour améliorer la sensibilité des systèmes de synthèse de la parole à partir de textes (TTS), (Maurel *et al.*, 2003) proposent un modèle d'oralisation par reformulation en l'appliquant à divers phénomènes syntaxiques, typographiques ou dispositionnels, appelés Architecture Textuelle par les auteurs (Virbel, 1989). Les textes jugés les plus importants sont reformulés sous différentes formes discursives pour obtenir un premier jeu de stratégies orales. Ils sont ensuite réduits avec des marques prosodiques adéquates pour produire un nouveau jeu de stratégies qui sont alors plus compactes). Grâce à cette approche, une amélioration sensible de la mémorisation et de la compréhension est obtenue lors de l'oralisation par un TTS de documents fortement structurés. Cependant, (Maurel *et al.*, 2006) montre que la charge cognitive est plus difficilement gérée que lors d'une lecture visuelle de ces mêmes documents.

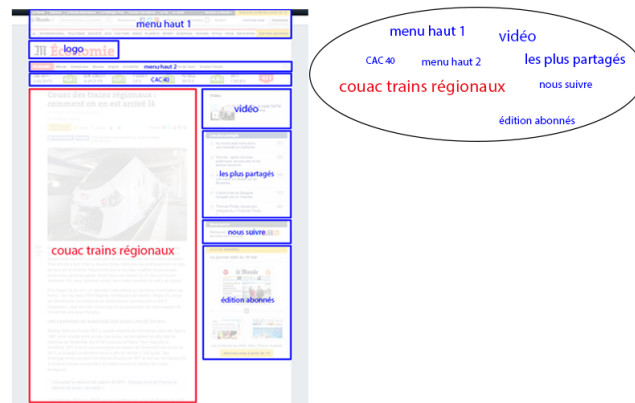
Le projet *Accessibility through Simplification & Summarization* (Parmanto *et al.*, 2005) (AcceSS) utilise des stratégies de simplification et de résumé. La simplification retire de la page les contenus jugés moins importants, donnant ainsi un accès plus rapide au contenu jugé important. Le résumé, quant à lui, donne un aperçu du contenu similaire à celui perçu visuellement. Le principe de base est de créer une nouvelle page pour chacune des sections (tête-de-page, bas-de-page, menu gauche, contenu textuel...). Pour cela, une méthode de marquage, basée sur des «modèles de genres» (presse, e-commerce, blog...) ou d'heuristiques, génère un ensemble de repères placés sur les éléments de la page. Un point central de navigation (page *Guide Dog*) est créé pour servir de pivot et de sommaire entre toutes les pages créées. Une première expérimentation avec le logiciel JAWS montre des résultats positifs. L'accès à une information particulière est plus rapide et les utilisateurs ont apprécié la navigation dans la page *Guide Dog* et plus généralement la version du site transformée par AcceSS. Une des limites de cette solution est l'utilisation d'une méthode de pattern matching pour identifier les sections de la page selon le genre qui est encore faillible. D'autre part, il n'y a pas de simplification au niveau du texte, donc une difficulté toujours grande dans le cas de contenus textuels importants.

SeEBrowser (Semantically Enhanced Browser) est un navigateur web audio spécialisé pour non-voyants (Michail & Christos, 2007). Des annotations sémantiques permettent une navigation par raccourcis. Des ontologies sur les pages web sont tout d'abord définies par annotation manuelle. Elles définissent ainsi des relations entre les différents éléments de ces pages et correspondent à la hiérarchie de la page web. L'ontologie produite est alors utilisée comme un arbre de navigation à plusieurs niveaux de granularité. Les nœuds sont appelés *Browsing Shortcuts* (BSs). Lorsque l'utilisateur charge une page web, il peut demander la liste de ses BSs puis les parcourir pour les écouter via des raccourcis claviers. Si le BS courant contient d'autres BSs, un son spécifique est émis et accessible. L'expérimentation montre que les BSs offrent un gain de temps statistiquement significatif pour des tâches de recherche d'information (scanning) et que l'utilisation de la hiérarchie produite évite la surcharge d'information. Le temps pour trouver une information reste cependant très long car les utilisateurs écoutent tous les BSs avant d'en sélectionner un. Une piste de solution serait de réordonner les BSs par ordre d'importance et de ne garder que les plus importants, mais l'inconvénient étant alors une perte d'information.

HearSay (Borodin *et al.*, 2010) est un navigateur web non visuel multi-modal développé à l'Université de Stony Brook (New York, USA) depuis 2004. Il accepte en entrée la reconnaissance vocale, le clavier, les interfaces tactiles et permet des sorties audio, écran et braille. Ses fonctionnalités sont nombreuses :

1. un module de segmentation analyse la structure et la mise en forme géométrique des pages web afin de les segmenter. Les utilisateurs peuvent alors naviguer entre les objets textuels reliés sémantiquement, par exemple le titre d'un article et son résumé. Le système utilise une variation de l'algorithme VIPS de Microsoft (Cai *et al.*, 2004) ;
2. un système d'annotation pour que les utilisateurs puissent ajouter du texte alternatif aux images et autres contenus ;
3. des algorithmes de différence comparent les pages visitées par l'utilisateur et repèrent les changements. Cela permet de passer rapidement sur les informations répétées (comme les menus) mais aussi d'obtenir les mises à jour dynamiques d'une page web ;
4. un analyseur de contexte qui permet de détecter le contenu principal d'une page web (par exemple le début d'un article) et d'identifier les informations pertinentes pour l'utilisateur en utilisant le contexte présent autour du lien cliqué.

L'expérimentation de HearSay montre une amélioration significative du temps nécessaire à des non-voyants pour trouver le contenu principal d'une page grâce au module d'analyse de contexte. Le module *Dynamo* aide fortement les utilisateurs à localiser les changements dynamiques de contenu et leur évite de lire les informations répétées (menus par ex.), permettant ainsi de se concentrer sur le contenu principal. Globalement, les autres fonctionnalités proposées par le logiciel ont aussi apporté un confort de navigation aux utilisateurs. Malgré ces points positifs, deux aspects restent en deçà de la

FIGURE 3 – Exemple de segmentation d'une page en blocs et création d'un *tag cloud*

lecture visuelle. Tout d'abord l'accès à la structure de la page n'est pas complet puisqu'il ne permet d'atteindre que le contenu principal ou jugé pertinent selon le contexte. Les autres éléments de la page restent accessibles séquentiellement, ce qui peut être long et laborieux. Ensuite, le *skimming* n'est pas permis sur la page, et le manque d'accès rapide et efficace à l'information constitue à nos yeux le deuxième aspect manquant à HearSay.

De plus en plus de travaux utilisent désormais des stratégies tactiles (Ziat *et al.*, 2007) (Giudice *et al.*, 2012) (Ahmed *et al.*, 2011). En intégrant des patterns dans les pages web, (Issa *et al.*, 2010) rend perceptible au toucher certains objets de la page et les relations qu'ils entretiennent. Les documents transformés sont alors proposés aux non-voyants sur un papier thermogonflable qui les met en relief. En positionnant le papier sur un écran tactile, on peut alors demander l'oralisation des objets présentés. L'inconvénient de cette méthode étant la nécessité d'avoir à disposition un papier spécial, (Safi *et al.*, 2014a) propose un concept proche utilisant la perception vibrotactile avec un dispositif constitué d'un gant et d'une tablette. Les contrastes lumineux produits par la tablette sont captés au survol des zones et génèrent des vibrations dans le gant. Ces approches sont intéressantes pour les stratégies de *scanning* mais sont insuffisantes pour véritablement réaliser du *skimming*.

Malgré les nombreux travaux, les stratégies de *skimming* doivent encore être améliorées pour permettre avoir un aperçu rapide du contenu d'une page web dans les situations non visuelles. La partie suivante présente les concepts de « tonnerre de mots » et de mot-clé navigant pour accéder au contenu d'une page web.

3 Tonnerre de mots ou *tag thunder*

Le *tag thunder* est un concept original reposant sur la capacité de l'être humain de pouvoir concentrer son oreille sur une source de son parmi d'autres émettant du son simultanément (Cherry, 1953). Le *Cocktail Party Effect* (Guerreiro, 2013) en est un exemple. On peut concentrer son attention sur une personne alors même que plusieurs personnes parlent simultanément autour de nous. La prononciation d'un mot important par une autre personne, par exemple notre nom, attirera immédiatement notre attention. L'idée est donc la suivante : dans un tonnerre de mots, l'utilisateur sera capable de détecter les mots-clés qui l'intéressent.

Nous considérons une page web comme un ensemble de blocs. La figure 3 illustre cette division sur une page d'article de journal. Chacun de ces blocs peut alors être identifié par un mot ou une phrase représentant son type, sa structure (métadonnée) ou son contenu (mot-clé). L'ensemble de ces mots ou phrases forme ainsi une sorte de nuage de mots (*tag cloud*), c'est-à-dire une représentation visuelle simplifiée du contenu et de la structure de la page web.

Ces blocs peuvent eux-mêmes contenir plusieurs blocs. Par conséquent, ils peuvent être définis de façon récursive comme des ensembles de blocs. De nouveaux mots ou phrases peuvent alors être extraits pour former de nouveaux nuages de mots. La combinaison du premier nuage et de ses « sous-nuages » constitue ainsi un arbre de nuages de mots, comme le représente la figure 4.

Les nuages de mots restent cependant des éléments visuels. Nous introduisons l'idée de « tonnerre de mots » ou *tag thunder* qui est une sorte de nuage de mots oralisé. Comme les mots ou phrases d'un nuage de mots sont distribués

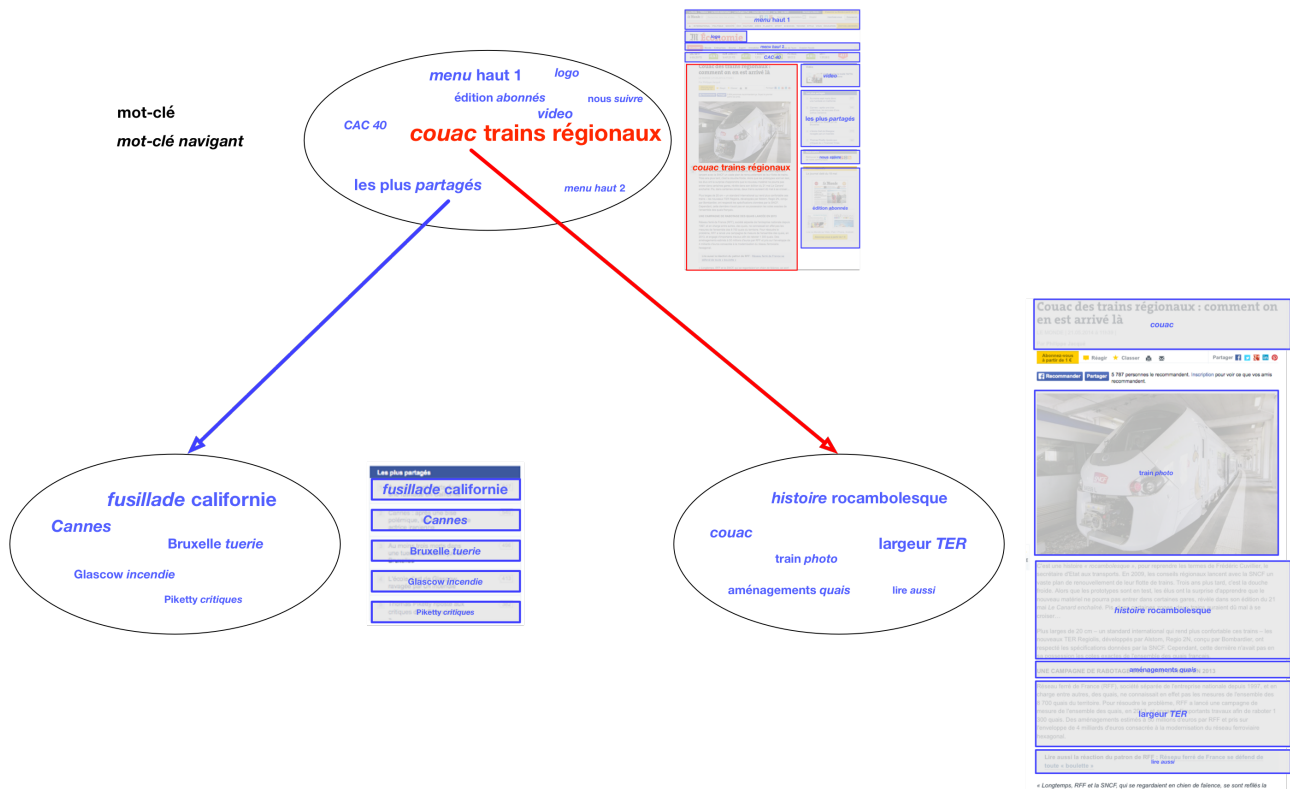


FIGURE 4 – Arbre de nuages et blocs correspondants

dans un espace avec différents effets visuels (e.g. taille de police, couleur, etc.), les mots constituant un tonnerre de mots sont distribués dans le temps avec différents effets audio (e.g., volume, débit, hauteur, prosodie, voix, répétition, multi-canaux). En utilisant une version orale de l'arbre présenté figure 4, on obtient alors un arbre de tonnerres de mots qui est une représentation de la structure de la page et de son contenu.

Nous voulons proposer une navigation dans cet arbre, permettant ainsi une navigation intra-page, c'est à dire une navigation en profondeur dans les blocs de la page. Le tonnerre de mots doit donc :

1. donner un aperçu du contenu du bloc textuel présenté (en fonction de la position dans l'arbre)
2. permettre la navigation dans l'arbre des blocs de la page

L'objectif final est que l'utilisateur puisse interagir avec le système en prononçant un mot-clé particulier entendu dans le tonnerre.

Nous définissons ainsi la notion de mot-clé navigant (MN ci-après). Un mot-clé navigant est un mot particulier du tonnerre de mots qui peut être « activé » vocalement. L'utilisateur peut ainsi prononcer le MN pour accéder au contenu correspondant au bloc identifié par le MN prononcé. Cela permet donc d'accéder au tonnerre de mots représentant un ensemble de sous-blocs s'il en contient, sinon directement à son contenu textuel (cf. figure 4). Les MN doivent être uniques, aisément identifiables et suffisamment pertinents pour représenter le contenu ou la nature de leur bloc. Il est donc nécessaire d'avoir non seulement des stratégies d'extraction de mots-clés, mais aussi des stratégies pour choisir les mots-clés navigants. Cependant, la notion d'unicité ne se réfère pas nécessairement uniquement au texte mais elle peut être liée au couple composé du texte et de sa prosodie.

Enfin, le tonnerre de mots peut être composé de mots-clés importants pour la compréhension du texte et de MN. Ainsi, l'utilisateur peut rapidement avoir un aperçu d'un bloc de texte et avoir les MN qui permettent à la fois d'appréhender le contenu et de naviguer. La figure 4 montre des exemples de noeuds composé de MN (en italique) et de mots-clés non navigants mais importants pour la compréhension du contenu. La partie suivante présente l'architecture de notre solution qui, pour l'instant, permet de créer des tonnerres de mots composés uniquement de MN.

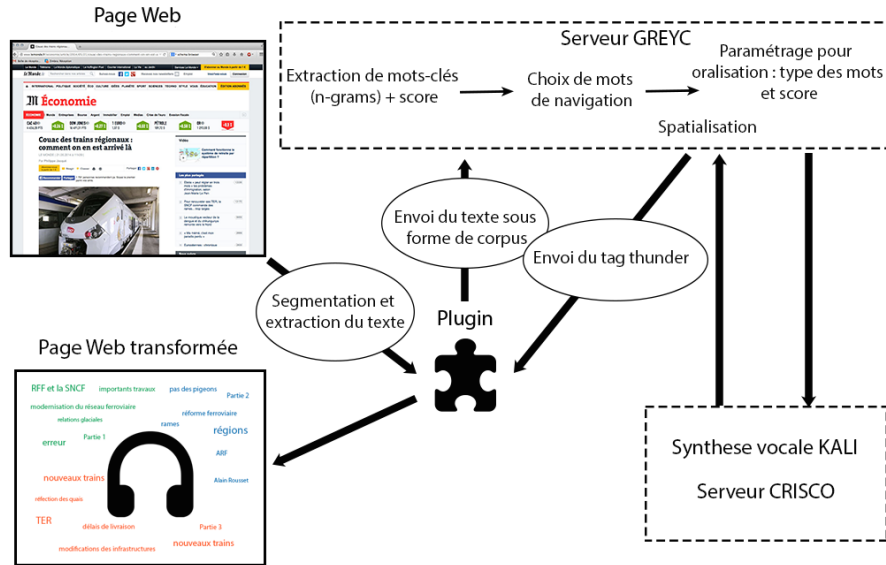


FIGURE 5 – Architecture logicielle

4 Architecture

Pour tenter de reproduire le processus d’aperçu rapide d’une page web qu’un utilisateur voyant réalise, nous devons identifier les diverses zones de la page. Puis pour chaque zone, il faut déterminer les mots importants ou représentatifs de cette zone. Cette partie présente les stratégies mises en oeuvre pour segmenter la page consultée et les techniques utilisées pour extraire un nuage de mots de chaque partie. Enfin, nous présentons comment le tout est oralisé pour être présenté à un utilisateur non-voyant.

L’architecture de notre solution est basée sur trois briques logicielles qui gèrent la segmentation de la page, l’extraction de mots-clés pour créer les nuages de mots et enfin l’oralisation des tonnerres de mots. Une extension pour le navigateur Firefox (FF ci-après) a été développée et gère les différentes briques. La figure 5 illustre l’architecture mise en place et les briques utilisées, détaillées ci-après.

4.1 Segmentation de la page en blocs

Différentes approches existent pour la segmentation de la page en blocs de texte. (Sanoja & Gançarski, 2014) propose une méthode basée sur l’analyse de l’arbre DOM (Document Object Model) représentant la page web. (Cai *et al.*, 2003) utilise l’apparence de la page dans le navigateur (approche basée vision) pour en extraire les zones. D’autres techniques existent : techniques de traitement d’image (Cao *et al.*, 2010), basées sur la structure sémantique (Adda, 2013) ou encore une résolution par graphe (Liu *et al.*, 2011). La segmentation est actuellement réalisée avec une méthode hybride que nous avons développée (Safi *et al.*, 2014b). Lorsque la page est chargée, l’extension FF applique l’algorithme de segmentation et envoie le résultat à un service web chargé de l’extraction de mots clés.

4.2 Extraction des mots-clés et nuages de mots

L’extraction de mots-clés est calculée en utilisant un algorithme basé sur TF-IDF (Term Frequency-Inverse Document Frequency). Cette mesure utilisée en Recherche d’Information, permet de faire ressortir les mots spécifiques à un contexte donné. En effet, la fréquence des termes dans le texte est pondérée par l’inverse de leur fréquence dans un corpus, ce qui fait remonter les plus spécifiques à ce texte, donc les plus pertinents.

Inspiré du travail de (Witten *et al.*, 1999), l’algorithme d’extraction de mots-clés développé ici s’appuie sur la mesure TF-IDF ainsi que sur la position des termes dans le texte pour calculer un score (e.g., dans le texte « bonjour tout le monde

», le mot « tout » est en 3^eme position sur 4, soit 0,75). Une version améliorée est proposée par (Jiang *et al.*, 2009) en utilisant le score TF-IDF et la position avec *Ranking SVM* plutôt que la position simple.

Le calcul de l’IDF (Inverse Document Frequency) a été réalisé sur un corpus de textes constitué de 953 551 articles sur 20 années du journal Le Monde (de 1987 à 2006). Le fait que le corpus s’arrête en 2006 introduit des biais et des silences dans l’extraction de mots-clés mais a permis de mettre en place une version fonctionnelle d’un prototype. Nous y reviendrons dans la dernière partie de cet article.

Un des points essentiels du travail de création du tonnerre de mots est la détermination des MN. Comme mentionné plus haut, ceux-ci doivent être uniques. Cette contrainte introduit une étape complémentaire à l’extraction des mots-clés. En effet, une fois les mots-clés extraits, il faut s’assurer de leur unicité et choisir d’autres MN en cas de conflit. Cette étape peut alors introduire des biais sur le choix des MN en « enlevant » des mots importants dans d’autres parties de la page. En particulier dans le cas où les blocs à décrire traitent du même sujet, la difficulté réside alors à obtenir des mots-clés suffisamment explicites mais distincts les uns des autres.

Une fois les nuages de mots-clés déterminés pour chaque bloc de la page, l’extension FF les reçoit alors et doit créer le tonnerre de mots. Cette étape est aussi réalisée via l’utilisation d’un web service.

4.3 Génération du tonnerre de mots

Des différences de voix, de volume, de prosodie, de hauteur ou de débit peuvent être utilisées dans un même tag thunder en simultanée étant donné que notre système auditif peut les discerner (Bregman, 1994). Leur synchronisation (Turgeon *et al.*, 2005) et leurs fréquences (Darwin *et al.*, 2003) peuvent influencer positivement ce discernement. Le travail de recherche consiste donc à définir et évaluer des modèles de *tag thunder*. L’audibilité et la rapidité de lecture du tonnerre de mots, ou encore la facilité d’identification des mots-clés d’intérêt par et pour l’utilisateur sont des critères d’évaluation possibles.

Notre solution intègre la synthèse vocale Kali (Morel & Lacheret-Dujour, 2001), conçue par le laboratoire du CRISCO (Université de Caen Basse-Normandie). Cinq étapes, prétraitement, analyse syntaxique, génération de la prosodie, phonémisation et traitement acoustico-phonétique permettent de prononcer le texte donné en entrée. Son intelligibilité et sa vitesse d’élocution élevée en font une synthèse vocale très appréciée des personnes non-voyantes. En effet, elles sont habituées à un débit élevé de parole, ce qui leur donne un gain de temps considérable lors de la lecture de documents. Kali est rendue disponible par une API prenant en paramètre le texte à synthétiser, la langue (français ou anglais), la voix, la prosodie, le volume, le débit et la hauteur. Le traitement est presque immédiat et elle retourne ensuite l’URL du fichier WAVE créé. L’extension FF envoie les textes à oraliser à l’API Kali, reçoit les URLs des sons et peut ainsi les jouer à l’utilisateur.

5 Perspectives et conclusion

Nous proposons ici une solution pour faciliter le *skimming* et le *scanning* de pages web dans des situations d’utilisation non visuelles. Après avoir segmenté la page en divers blocs, chaque bloc est analysé afin d’en extraire des mots clés navigants que l’utilisateur pourra énoncer pour sélectionner la sous-partie correspondante. Enfin, notre solution intègre un générateur vocal pour oraliser les MN de la page.

Nous travaillons actuellement à l’amélioration de notre solution, en particulier sur le corpus pour calculer. En effet, les articles les plus récents datant de 2006, un nombre non négligeable de mots sont inconnus du corpus. De plus, le corpus étant uniquement constitué d’articles de presse, le langage utilisé est propre à ce contexte. Une amélioration en cours est de baser le calcul de l’IDF sur un ensemble de pages issues du site consulté, éventuellement combiné à un corpus généraliste.

D’autres champs d’expérimentation sont ouverts par le fait qu’il s’agit non seulement d’extraire de simples mots-clés, mais de déterminer les mots-clés navigants. En effet, lorsque les divers blocs de la page traitent du même sujet, il faudra réussir à déterminer des MN distincts mais suffisamment précis pour ne pas avoir de biais dans les nuages extraits.

Une autre amélioration concerne l’utilisation des formats audio 5.1 et 7.1 afin d’utiliser plus de canaux pour la représentation spatiale d’une page web. L’utilisation de techniques binaurales est aussi envisagée³. La synthèse Kali utilisée n’est

3. Technique de spatialisation sonore. Voir http://fr.wikipedia.org/wiki/Enregistrement_binaural

| Effet visuel | Effet sonore |
|---|-----------------------------------|
| disposition du texte | spatialisation du son |
| saillance typographique | volume et fréquence de répétition |
| navigation en profondeur dans les blocs | mot-clé navigant et type de voix |

TABLE 1 – Équivalents entre effets visuels et effets sonores

pas compatible avec l'utilisation de langages normalisés d'oralisation. L'utilisation d'un TTS compatible permettrait alors de travailler directement à l'aide de formats VoiceXML+SSML.

De même que pour la génération des listes de mots-clés navigants, les paramètres possibles pour la restitution du *tag thunder* sont nombreuses. Comme mentionné plus haut, le débit, la prosodie, etc. sont divers facteurs sur lesquels nous devons mener de multiples expérimentation pour déterminer la meilleure solution pour qu'un utilisateur puisse faire son choix. En particulier, nous voulons tester le fait que la spatialisation du son reflète la spatialisation des textes, c'est à dire que des mots issus d'un même bloc de texte auront des caractéristiques spatiales similaires. De même, les effets typographiques constituent un champ d'expérimentation, pour lequel nous pensons jouer sur les paramètres de volume et de fréquence de répétition des mots. Le tableau 1 résume le parallèle entre les appréhensions visuelles et non-visuelles que nous voulons expérimenter.

En ce qui concerne les structures de données, la modélisation de ce que nous avons appelé ici l'arbres de nuages reste à formaliser afin de pouvoir utiliser des modèles et/ou algorithmes existants.

Enfin, la solution proposée doit être évaluée avec un protocole expérimental pour vérifier si les stratégies de *skimming* et de *scanning* non visuel sont améliorées. La combinaison de cette solution avec d'autres systèmes, en particulier vibrotactile, permettrait alors de concevoir des systèmes d'aides multimodaux.

Références

- ADDA N. F. S. R. G. (2013). Pré-segmentation de pages web et sélection de documents pertinents en questions-réponses. *TALN-RÉCITAL 2013*, p. 479.
- AHMED A. A., YASIN M. A. & BABIKER S. F. (2011). Tactile web navigator device for blind and visually impaired people. In *Applied Electrical Engineering and Computing Technologies (AEECT), 2011 IEEE Jordan Conference on*, p. 1–5 : IEEE.
- AHMED F., BORODIN Y., PUZIS Y. & RAMAKRISHNAN I. (2012a). Why read if you can skim : towards enabling faster screen reading. In *International Cross-Disciplinary Conference on Web Accessibility - W4A2012*, Article No. 39.
- AHMED F., BORODIN Y., SOVIAK A., ISLAM M., RAMAKRISHNAN I. & HEDGPETH T. (2012b). Accessible skimming : Faster screen reading of web pages. In *25th Annual ACM Symposium on User Interface Software and Technology (UIST)*, p. 367–378.
- BIGHAM J. P., CAVENDER A. C., BRUDVIK J. T., WOBROCK J. O. & LANDER R. E. (2007). Webinsitu : A comparative analysis of blind and sighted browsing behavior. In *9th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*, p. 51–58.
- BORODIN Y., BIGHAM J. P., DAUSCH G. & RAMAKRISHNAN I. (2010). More than meets the eye : A survey of screen-reader browsing strategies. In *International Cross Disciplinary Conference on Web Accessibility (W4A)*, p. 1–10.
- BREGMAN A. S. (1994). Auditory scene analysis : The perceptual organization of sound.
- CAI D., HE X., MA W.-Y., WEN J.-R. & ZHANG H. (2004). Organizing WWW Images Based on the Analysis Of Page Layout And Web Link Structure. *Microsoft Research Asia, Beijing, China*.
- CAI D., YU S., WEN J.-R. & MA W.-Y. (2003). *VIPS : A vision-based page segmentation algorithm*. Rapport interne, Microsoft technical report, MSR-TR-2003-79.
- CAO J., MAO B. & LUO J. (2010). A segmentation method for web page analysis using shrinking and dividing. *International Journal of Parallel, Emergent and Distributed Systems*, **25**(2), 93–104.
- CHERRY E. C. (1953). Some experiments on the recognition of speech, with one and with two ears. *Journal of the acoustical society of America*, **25**(5), p. 975–979.

- DARWIN C. J., BRUNGART D. S. & SIMPSON B. D. (2003). Effects of fundamental frequency and vocal-tract length changes on attention to one of two simultaneous talkers. *The Journal of the Acoustical Society of America*, 114, p. 2913.
- DIAS G. & CONDE B. (2007). Accessing the web on handheld devices for visually impaired people. In K. WĘGRZYN-WOLSKA & P. SZCZEPANIAK, Eds., *Advances in Intelligent Web Mastering*, volume 43 of *Advances in Soft Computing*, p. 80–86.
- GIUDICE N. A., PALANI H. P., BRENNER E. & KRAMER K. M. (2012). Learning non-visual graphical information using a touch-based vibro-audio interface. In *Proceedings of the 14th international ACM SIGACCESS conference on Computers and accessibility*, p. 103–110 : ACM.
- GUERREIRO J. (2013). Using simultaneous audio sources to speed-up blind people's web scanning. In *10th International Cross-Disciplinary Conference on Web Accessibility (W4A)*, p. 1–2.
- ISSA Y. B., MOJAHID M., ORIOLA B. & VIGOUROUX N. (2010). Analysis and evaluation of the accessibility to visual information in web pages. In *Computers Helping People with Special Needs*, p. 437–443. Springer.
- JIANG X., HU Y. & HANG L. (2009). A ranking approach to keyphrase extraction. *SIGIR'09*.
- LIU X., LIN H. & TIAN Y. (2011). Segmenting webpage with gomory-hu tree based clustering. *Journal of Software*, 6(12), 2421–2425.
- MAUREL F., MOJAHID M., VIGOUROUX N. & VIRBEL J. (2006). Documents numériques et transmodalité. *Document numérique*, 9(1), 25–42.
- MAUREL F., VIGOUROUX N., RAYNAL M. & ORIOLA B. (2003). Contribution of the transmodality concept to improve web accessibility. *Assistive Technology Research Series*, 12, 186–193.
- MICHAIL S. & CHRISTOS K. (2007). Adaptive browsing shortcuts : Personalising the user interface of a specialised voice web browser for blind people. In *Data Engineering Workshop, 2007 IEEE 23rd International Conference on*, p. 818–825 : IEEE.
- MOREL M. & LACHERET-DUJOUR A. (2001). Kali, synthèse vocale à partir du texte : de la conception à la mise en oeuvre. *Traitement Automatique des Langues* 42, p. 193–221.
- PARMANTO B., FERRYDIANSYAH R., SAPTONO A., SONG L., SUGIANTARA I. W. & HACKETT S. (2005). Access : accessibility through simplification & summarization. In *Proceedings of the 2005 international cross-disciplinary workshop on web accessibility (W4A)*, p. 18–25 : ACM.
- SAFI W., MAUREL F., ROUTOURE J.-M., BEUST P. & DIAS G. (2014a). Blind browsing on hand-held devices : Touching the web... to understand it better. In *Data Visualization Workshop (DataWiz 2014) associated to 25th ACM Conference on Hypertext and Social Media (HYPERTEXT 2014)*.
- SAFI W., MAUREL F., ROUTOURE J.-M., BEUST P. & DIAS G. (2014b). A hybrid segmentation of web pages for vibro-tactile access on touch-screen devices. In *3rd Workshop on Vision and Language (VL 2014) associated to 25th International Conference on Computational Linguistics (COLING 2014)*, p. 95–102.
- SANOJA A. & GANÇARSKI S. (2014). Block-o-matic : A web page segmentation framework. In *Multimedia Computing and Systems (ICMCS), 2014 International Conference on*, p. 595–600 : IEEE.
- TURGEON M., BREGMAN A. S. & ROBERTS B. (2005). Rhythmic masking release : effects of asynchrony, temporal overlap, harmonic relations, and source separation on cross-spectral grouping. *Journal of Experimental Psychology : Human Perception and Performance*, 31(5), p. 939.
- VIRBEL J. (1989). The contribution of linguistic knowledge to the interpretation of text structures. In *Structured documents*, p. 161–180 : Cambridge University Press.
- WITTEN I. H., PAYNTER G. W., FRANK E., GUTWIN C. & NEVILL-MANNING C. G. (1999). KEA : Practical Automatic Keyphrase Extraction. *NRC/ERB-1057*.
- ZIAT M., GAPENNE O., STEWART J. & LENAY C. (2007). Haptic recognition of shapes at different scales : A comparison of two methods of interaction. *Interacting with Computers*, 19(1), 121–132.