Étude Bibliographique de Master 2

Spécialité : **AIGLE**

Personnalisation de page web : application à l'amélioration de l'accessibilité du web

par **Franck PETITDEMANGE**

Mars 2014

Sous la direction de **Marianne HUCHARD, Michel MEYNARD, Yoann BONAVERO**

Contents

1 Introduction et motivation

Le world wild web (www) est un reseau de ressource. La publication de ces ressources repose sur un langage universellement compréhensible et accepté par tous les ordinateurs : HTML, historiquement conçut pour faciliter l'échange d'article dans la communauté scientifique. La démocratisation du web à fait radicalement évoluer le contenu d'une page web, sans pour autant que le langage ne suivent ces évolutions. Ainsi les auteurs de page web ont détourné les pratiques de conception d'une page de manière anarchique. Ce manque d'homogénétité complique la compréhension du contenu publié sur le www par une machine. Faisant perdre la propriété universelle du web voulu par son créateur Tim Berners Lee :

> "La puissance du Web réside dans son universalité. L'accès à tous, quel que soit son handicap est un aspect essentiel"

Ceci introduit la motivation de ce stage et les problématiques qui en découlent.

Le sujet du stage est la personnalisation des pages web. L'objectif est de fournir des méthodes et des outils afin d'adapter une page suivant les souhaits d'un lecteur. On s'interesse à une application pour l'amélioration de l'accessibilité des pages web pour les personnes en situation de handicap visuel.

L'adaptation d'une page web implique notre première problématique : la restructuration d'une page web. On souhaite expérimenter une approche basée sur les méta-modèles. L'idée étant d'extraire la structure d'une page et d'en construire une représentation plus abstraite. Cela doit nous permettre de s'affranchir de la diversité de conception de ces dernières. A partir de cette représentation on veut lui appliquer des transformations, puis générer une nouvelle page conforme aux transformations.

La conception des pages web s'articule autour : d'un langage pour décrire la structure du document (HTML) et d'un langage pour décrire la mise en forme du document (CSS). Les pages sont constituées d'éléments hétérogènes : une page est constituée d'un ou plusieurs contenu principal, d'un menu de navigation, de publicité, etc... Chacun de ces éléments représentent une sous-structure de la page. Lorsqu'on regarde une page web depuis un navigateur, on constate que ces éléments sont structurés de façon sémantique, ils sont organisés selon leur sens. La difficulté dans la tâche de d'extraction de la structure d'une page est dû au manque d'expressivité de HTML. En effet, la norme actuelle de HTML (HTML 4), ne fournit pas de moyen de délimiter les éléments du document en fonction de leur sémantique. Par exemple, on ne peut pas délimiter de manière explicite la structure d'un menu dans une page avec ce langage. Le constat est que l'information de la structure d'une page apparaît principalement dans la mise en page. La structure d'une page est explicitée à travers l'utilisation de police, de couleur ou plus généralement d'élément visuel pour caractériser les contenus qui ont la même signification.

La seconde problématique est la définition d'un protocole d'acquisition et d'apprentissage automatique des souhaits de personnalisation d'une page par

le lecteur. On veut fournir au lecteur une interface permettant la modification d'une page. *Un modèle peut être vu comme un langage de haut niveau pour décrire une page. L'idée étant d'exploité le modèle de page transformé par l'utilisateur pour en inférer des voeux de modification. L' atout dans l'utilisation de modèle permet une capitalisation des connaissances, indépendament du support technologique, du profil d'un utilisateur.*

2 Modele de page web

2.1 Introduction

Modèle

Definition. Un modèle est une représentation simplifié d'une partie d'un système. C'est une abstraction du système étudié suivant un point de vue. Par exemple une carte routière est une abstraction d'un réseau routier, il existe plusieurs type de carte suivant ce que l'on veut étudier (chemin pédestre, chemin routier etc). L'intérêt d'un modèle est de mieux comprendre un système.

> "Pour un observateur A, M est un modèle de l'objet O, si M aide A à répondre aux questions qu'il pose sur O" (Minsky)

Métamodèle

Definition. Pour exprimer un modèle, nous avons besoin de pouvoir exprimer ces concepts. Un métamodèle, c'est un modèle qui fournit un langage pour exprimer un modèle. Littéralement, c'est un modèle de modèle.

Dans le cadre du sujet. On souhaite créer un méta-modele afin de concevoir un modèle de page web. Le metamodèle doit être suffisament expressif pour instancier un modèle de page web conforme à la vision qu'un lecteur peut avoir d'une page. Dans un premier temps on souhaite extraire la structure d'une page pour en contruire un modèle. L' avantage est d'en manipuler une representation independante de la diversité de conception des pages. Le deuxieme avantage est fournir langage de haut niveau pour decrire une page. On veut l'exploiter pour fournir à l'utilisateur un langage pour exprimer des transformations

Dans le cadre de notre sujet, on souhaite à terme l'adaptation d'une page web. Ici le système étudié est une page web. Il est important d'en cerner les concepts intrinsèques. L'objectif étant de concevoir un méta-modele suffisament riche à l'expression d'un modèle conforme à l'intention de son auteur. Mais aussi à la représentation qu'un utilisateur à d'une page, afin d'en faciliter l'expression de transformation.

Une page web possède le rôle d'affichage d'un contenu structuré et mise en forme. On peut voir une page comme une composition d'élément graphique agencé dans l'espace de la page et apportant une information. Par exemple, une page est composé d'un menu de navigation, d'une entête, d'un pied de page, de formulaire et de widget (on entend par widget des éléments de contrôle,

comme des barres de progression, des selecteurs de couleurs, etc). Chacun de ces éléments possède un type, un état et des propriétés.

Le type correspond à la nature de celui-ci. Il représente ce qu'un utilisateur attend de ce dernier. Il est attendu d'un menu de navigation qu'il nous permette de naviguer dans les page du site web.

Certain objet fournissent une interaction avec l'utilisateur qui est susceptible de modifier l'état d'un objet. Par exemple certain menu cache les différents liens de navigations, au moment où l'utilisateur survole le menu celui-ci affiche les différents liens.

Les objets possèdent également un ensemble de propriétés, quelque fois intrinsèque à leur type. Par exemple un formulaire possède des champs de saisie. D'autre propriétés peuvent être la couleur et la police du texte, la hauteur, la largeur, et la position d'une objets dans le page.

On s'interesse à un modèle capable de modéliser une page dans le but d'en abstraire la structure, les differents elements la composant, les relations entre ces éléments, leurs natures et leurs comportements. On propose d'étudier et de comparer le langage standart du publication de document sur le web, HTML, dans la norme HTML 4 et 5 mais aussi une toxonomie pour la description d'interface graphique ARIA.

2.2 HTML 4

HTML 4 [3] est un langage permettant la publication de contenu sur le web. C'est le langage standard actuelle des pages web. Il permet de structurer le contenu et d'associer une mise en forme aux contenu. Le contenu est organisé de manière hierarchique en le découpant en section et sous-section.

Structuration générique HTML 4 propose un mécanisme générique pour la composition du contenu formant la structure des pages web. Ce mécanisme gravite autour des éléments de type <DIV> leurs identifiants respective : id et classe.

DIV Signifiant division, est utilisé comme conteneur générique permettant d'organiser le contenu. Il est utilisé pour :

- regrouper les elements pour leur appliquer un style (une mise en forme particuliere).
- signaler un section ou ou sous-section.

id et class Chaque élément peut se voir attribuer un identifiant ou une classe d'appartenance. *id* assigne un nom à un élément. Ce nom est unique dans le document. *class* au contraire, assigne un ou plusieurs noms de classe à un élément; on peut dire de l'élément qu'il appartient à ces classes. Un nom de classe peut être partagé par plusieurs instances d'éléments. Les identifiants

et les classes sont des suites de caractères quelconque décidé arbitrairement par l'auteur du document.

Les éléments DIV utilisés conjointement avec les attributs id et classe sont au cœur du mécanisme générique de structuration d'un document. DIV permet de divisé le contenu d'un document en section et sous-section pour décrire sa structure. Les elements DIV ayant une sémantique neutre, c'est l'auteur du contenu qui attribut (de manière arbitraire) un nom de *class* ou un *id*. Le but étant de définir une rôle au contenu ou une mise en forme. On note le caractère implicite de la structuration.

Algorithm 1 Exemple d'attribution de rôle

Listing 1: Exemple de structuration avec HTML 4

```
<body>
        <div id="header" ></div>
        <div id="navigation_bar"/>
        <div id="main_content">
                <div class="article"></div>
                <div class="article"></div>
        </div>
        <div id="footer"></div>
</body>
```

Algorithm 2 Exemple découpage en section et sous-section

Listing 2: Exemple de structuration avec HTML 4

```
<body>
<div class="section" id="elephants-foret" >
        <h1>Les éléphants des forêts</h1>
        <p>Dans cette partie, nous abordons le sujet
moins connu des éléphants des forêts.</p>
        <div class="sous-section" id="habitat-foret" >
        <h2>L'habitat</h2>
        <p>Les éléphants des forêts ne vivent pas
dans les arbres mais au milieu d'eux.</p>
        </div>
</div>
</body>
```

Figure 1: Architecture page web HTML 4

2.3 HTML 5

HTML 5 [4] étend HTML 4. La norme HTML 4 est toujours rétroactive. HTML 5 remplace la structure générique de HTML 4 par un nouveau modèle. Ce modèle amène de nouveaux élements qui apportent une sémantique standard et explicite à la structure d'une page.

Structuration Les nouveaux élements de HTML 5 spécifie donc une sémantique standard :

- Section : represente une section générique dans un document, c'est à dire un regroupement de contenu par thématique.
- Article : représente un contenu autonome dans une page, facile l'inclusion de plusieurs sous document

- nav : représente une section de liens vers d'autre page ou des fragments de cette page
- aside : représente une section de la page dont le contenu est indirectement lié à ce qui l'entoure et qui pourrait être séparé de cet environnement
- header : représente un groupe d'introduction ou une aide à la navigation. Il peut contenir des éléments de titre, mais aussi d'autres éléments tels qu'un logo, un formulaire de recherche, etc.
- footer : représente le pied de page, ou de la section, ou de la racine de sectionnement la plus proche

Figure 2: Architecture HTML 5

2.4 ARIA

ARIA (Acessible Rich Internet Application) [2] est la spécification d'une ontologie décrivant une interface graphique. Elle fournit des informations sur la

structuration d'un document et décrit les élements qui composent l'interface au moyen d'un ensemble de rôles, d'états et de propriétés .

Rôle Les rôles permettent d'identifier la fonction de chaque elements d'une interface. Ils sont regroupé en trois catégories :

- Widget Roles : référence d'un ensemble de widget préfinis (alertdialog, button, slider, scrollbar, menu, etc)
- Document Structure Roles : décrit les structures qui organisent un document (article, definition, entête, ect)
- Landmark Roles : décrit les régions principales d'une interface graphique (main, navigation, search, etc)

Etats et propriétés ARIA permet d'associer des états et propriétés à des wigets.

Un état est une configuration unique d'un objet. Par exemple, on peut définir l'état d'un bouton par l'état *aria-checked* qui peut prendre trois propriétés suivant l'interaction avec l'utilisateur : *true* - *false* - *mixed* . Aria prévoit même un systeme d'annotation pour les objets ayant des comportements asynchrones. Par exmple, on peut annoter qu'un element se met à jour de manière autonome.

On peut associé un ensemble de propriété par exemple la valeur minimal ou maximal que l'on doit remplir dans un champs de saisit *aria-valuemin, aria-valuemax.*

2.5 Discussion

HTML 4 fourni un mécanisme de structuration générique pour la publication de document sur le web. Ce support semble ne pas être assez riche à la description d'une page web. En effet, ce dernier pose plusieurs problèmes :

- Une page possède un contenu hétérogène. Plusieurs document peuvent être décrit dans le page. Il n'y pas de moyen de délimiter le contour d'un document de manière explicite dans une page.
- Une page possède un contenu possiblement non linéaire. Impossible de l'exprimer.
- Un page web contient des élements qui ne sont pas en rapport avec le contenu d'un document, mais plûtot avec le site web. Typiquement un menu de navigation, un logo etc... Il n'est pas possible de le modéliser explicitement.

HTML 5 apporte des améliorations par rapport aux points précédents :

- Les élements section permettent de délimiter et d'exprimer explicitement les sections d'une page, et leur offre un environnement contextuel.
- On peut modéliser qu'un contenu n'est pas linéaire
- On peut modéliser certain élement propre au site web tel qu'un menu, un logo etc.

ARIA semble posséder de bonnes caractèristiques pour être le support d'un méta-modèle. HTML est un langage de publication de document, il décrit la structuration hiérarchique d'un document. Il présente des vacuités pour décrire tout le contenu que peut représenter une page web. Notamment avec le developpement des élements graphiques appelés widget. Ceux-ci possède un comportement dynamique et possède protocole de communication asynchrone, non décrit par HTML. ARIA est capable :

- de modéliser la description d'un document avec une sémantique standarts et riche (inclus les élements de sémantique de HTML 5),
- de modéliser la structure d'une page avec une sémantique standarts et riche,
- de modéliser la structure et le comportement des widget avec une sémantique standarts et riche.

3 Extraction structure

3.1 Introduction

L'extraction est la première phase du processus de restructuration. On peut la décrire comme le processus de découverte des éléments d'un tout. Ici on veut récuperer les élements consitutifs d'une page web. Les éléments à extraire de la page correspondent aux différentes brique conforme aux élements d'un méta-modele définit à priori. La problématique est dû au langage de conception des pages web qui manque de sémantique, rendant un processus d'extraction automatique difficile.

3.2 Approche segmentation visuelle

L'approche proposé par les auteurs [5] présente un algorithme de partitionnement basé sur les élements de mise en forme des pages web. Le partionnement extrait une structure qui regroupe les elements d'une page semantiquement proche en bloc. Le postulat est que les éléments d'une page possédant des caractéristiques de mise en forme proche, tels que la police, la couleur, la taille, sont sémantiquement proche.

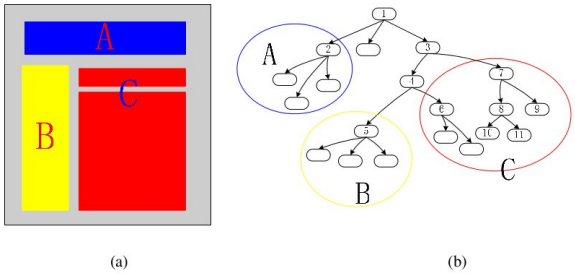

Figure 3: Exemple de partionnement, (a) page (b) DOM de la page

L'algorithme exploite le DOM[1] de la page web. Le DOM est une API pour les documents HTML (ou plus généralement XML) . Il fournit une représentation arborescente d'un document et les moyens d'acceder à son contenu et sa mise en forme.

Le processus de segmentation, figure 4, se décompose en trois phases : un processus d'extraction de blocs, un processus de detection de séparateur et un processus de reconstruction.

Le processus d'extaction détecte les éléments du niveau courant du DOM succeptible de former un contenu cohérent. Cette détection repose sur des séparateurs explicites : on sais que certains elements délimitent le contour d'un contenu (par exemple les balises <DIV>). Mais également sur une fonction de

[1] Document Object Model

distance visuelle comparant les noeuds parents et frères du noeud courant : une balise <DIV> à de plus de grande chance de délimiter un contenu sémantiquement différent du noeud parent si la couleur de fond est différente de celle du noeud parent. Pour chaque noeud l'algorithme vérifie si il forme un bloc ou non. Si oui il associe un degre de cohérence au bloc. Ce dégre de cohérence est un indicateur de l'importance sémantique du bloc. Si non, il est appliqué le même processus aux enfants du noeud. Quand tous les noeuds du bloc courant sont extraits, ils sont mis dans un pool.

Des séparateurs entre les blocs sont ensuite detectés. L'algorithme dététecte ici des séparateurs implicites, n'apparaîsant pas dans la structure HTML. Les séparateurs implicite sont les espaces entre les blocs d'une pool. Un poid est attribué à chaque séparateur suivant sont importance (par exemple, plus l'espacement entre deux blocs est grand, plus le poid sera elevé). Ce poid est un indicateur différence sémantique entre blocs adjacent. Plus le poid du séparateur est élevé en deux blocs, plus leur contenu sera sémantiquement éloigné.

Une construction hierachique des blocs est crée. Cette construction hierachique repose sur le degre de coherence attribué à chaque bloc.

Pour chaque nouveau bloc de la structure hierarchique construite, l'algorithme teste le degre de cohérence attribué par rapport à un seuil de cohérence définit. Ce seuil est définit suivant la granularité de la structure que l'on veut en sortie de l'algorithme. Si le degres de cohérence est pas supérieur au seuil de cohérence, le bloc est de nouveau partionné. La structure finale est construite après que tous les blocs soit traités.

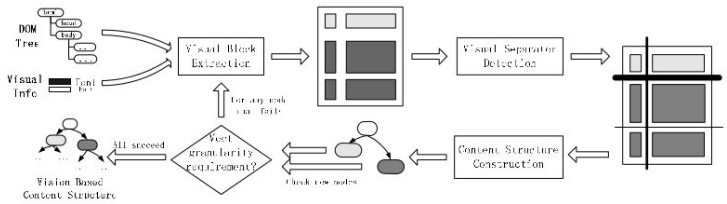

Figure 4: Algorithme de segmentation

3.3 Similarité de séquence

L'un des postulats de base en bioinformatique est qu'une séquence génomique similaire donne une protéine présentant la même fonction. En d'autre terme, des séquences ayant des similitudes (syntaxiques) est un signe de proximité fonctionnelle. Appliqué à une page web, cela signifie que par exemple la structure (syntaxique) d'un menu d'une page web A est similaire à la structure syntaxique d'une page web B. Il semble cohérent d'appliquer ce postulat à une page web. En effet la publication de contenu sur le web se standardise par l'intermédiare des scripts générant des pages de manière automatique et standardisé (Wordpress, joomla, etc). Afin d'obtenir un bon référencement, les moteurs de recherche in-

site les auteurs de pages web d'adoper une conception standard dans la conception des pages web. En construisant une séquence représentative d'un élement de notre méta-modèle, il devrait être possible de determiner la fonction d'une sequence d'une page web. Les bioinformaticiens utilisent le concept d'alignement pour déterminer la similarité de deux séquences, en d'autre terme pour savoir il elle possède la même fonction.

Definition. L' alignement est la mise en correspondance de deux séquences. Soit deux sequences X1:n et Y1:m à valeur dans le même alphabet fini Λ. Un alignement c'est une correspondance entre les lettres de la premiere séquence et celles de la deuxieme, sans en changer l'ordre, et en autorisant eventuellement des « trous ».

G A A T C _ T G A C
C A _ _ C G T _ A _

Figure 5: Alignement possible des sequences X1=GAATCTGAC,Y1=CACGTA

La mise en correspondance repose sur trois types d'opérations élementaires : la substitution, l'insertion, deletion. Plusieurs combinaisons d'alignements existent. Lorsque l'on souhaite comparer la similitude de deux sequences, la meilleur solution est celle qui minimise le nombre d' opérations d'insertions et de deletions.

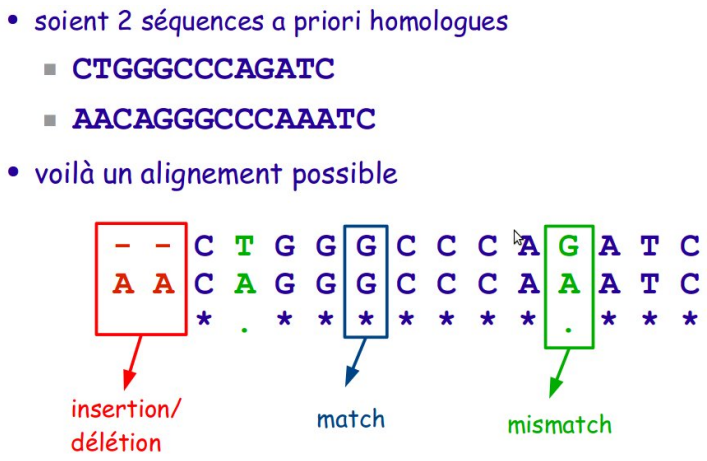

Figure 6: opérations d'alignements

Il existe deux types d'alignements : local et global.

3.3.1 L'alignement global

L'alignement global est conçu pour comparer des séquences sur toute leur longueur. Un méthode optimale pour trouver un alignement global maximal de chaîne de caractères est l'algorithme de Needleman-Wunsch [1]. La méthode de segmentation visuelle (section précedente) semble pouvoir extraire et regrouper les éléments sémantiquements proche d'une page. L'inconvénient est que l'on ne connait pas la signification de ses élements. Une approche serait de construire une séquence reprensentative d'un élement de notre méta-modèle et de comparer les sequences extraites d'une page aux sequences associée au méta-modèle.

3.3.2 L'alignement local

L'alignement local est conçu pour rechercher dans la séquence A des régions semblables à la séquence B (ou à des parties de la séquence B). Une méthode pour trouver un alignement local maximal de chaîne de caractères est l'algorithme de Smith & Waterman [7]. Cette approche ne necessite pas d'extraire de sequences dans le une page. On peut comparer directement une sequence du meta-modele à la page pour trouver la zone qui correspond.

```
Global  FTFTALILLAVAV
        F--TAL-LLA-AV

Local   FTFTALILL-AVAV
        --FTAL-LLAAV--
```

Figure 7: Comparaison sequence d'alignement globale et locale

Comme on le voit dans la figure 7 l'alignement global tente d'aligner les séquences sur toute leur longueur, tandis que l'alignement local se focalise sur les zones de forte homologie.

3.4 Similarité d'arbre

Une approche autre approche dans l'etude de similarité de structure est la comparaison d'arbre. Cette approche consiste à trouver la plus petite ou la moins coûteuse séquence d'opération d'édition (substitution, suppression et insertion) qui permet la transformation d'un arbre vers un autre.

Notons Λ un noeud vide. Une opération d'édition est ecrit $b \to c$, ou b et c sont soit un noeud, soit Λ .

- $b \to c$ est une opération de substition si $b \neq \Lambda$ et $c \neq \Lambda$,

- une opération de suppression si $b \neq \Lambda \doteq c$,
- et une opération d'insertion si $b = \Lambda \neq c$

Pour exprimer une séquence d'opération élémentaire qui transforme l'arbre, on utilise le concept de mapping, introduit [6]. Un mapping établie une correspondance un-à-un entre les nœuds de deux arbres ordonnés et qui préservent l'ordre des nœuds.

Definition. Un Mapping M de l'arbre T1 vers l'arbre T2 est un ensemble de paire ordonnée d'entier (i, j), $1 \leqslant i \leqslant n1$, $1 \leq j \leq n2$, satisfaisant les conditions suivantes, pour tous $(i1, j1), (i2, j2) \in M$:

- i1=i2 si et seulement si, j1=j2 (one-to-one condition);
- t1[i1] est à droite de t1[i2], si est seulement si, t2[j1] est à gauche de t2[j2] (preservation de l'ordre des noeuds frères);
- t1[i1] est un ancêtre de t1[i2] si et seulement si, t2[j1] est un ancêtre de t2[j2] (preservation de l'ordre des ancêtres);

Definition. Soit M un mapping entre les arbres T1 et T2 décrivant des opérations de modification. S est l'ensemble de pair $(i, j) \in M$, D l'ensemble des nœuds T1[i] n'ayant pas de paire $(i, j) \in M$, et I l'ensemble des nœuds T2[j] n'ayant pas de paire $(i, j) \in M$. Le coût du mapping est donné par $|S|p + |I|q + |D|r|$, ou p est le coût des substitution non identique, q est le coût des insertions (1), r est le coût d'une suppression (1), le coût des substitution identique est 0.

Pour connaître la similarité entre deux structures, on veut calculer une distance d'alignement. C'est à dire trouver le coût minimum du mapping pour que T1 et T2 soit isomorphe. KUO-CHUNG TAI [6] propose un algorithme de programmation dynamique pour résoudre la question de distance d'arbre en temps séquentiel $O(|T1| \times |T2| \times min(depth(T1), leaves(T1)) \times min(depth(T2), leaves(T2)))$.

3.5 Discution

4 Construction de motif

Certain elements de la section precendante repose sur la construction d'un pattern. Plusieurs approche : PAT-TREE, consensus de graphe, Modèle de Markov caché

5 Conclusion

References

[1] SAGL B.NEEDLE CHRISTUS D.WUKSCH. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Department of Biochemistry, Northwestern University, and hlwlear Medicine Service, V.A. Research Hospital Chicago, Ill. 60611, U.S.A.*

[2] World Wide Web Consortium. Accessible rich internet applications 1.0.

[3] World Wide Web Consortium. Html 4.01 specification.

[4] World Wide Web Consortium. Html 5 specification.

[5] Ji-Rong Wen Wei-Ying Ma Deng Cai, Shipeng Yu. Extracting content structure for web pages based on visual representation. *Springer Lecture Notes in Computer Science.*

[6] KUO-CHUNG TAI. The tree-to-tree correction problem.

[7] Michael Waterman Temple F. Smith. Identification of common molecular subsequences. *Journal of Molecular Biology.*