เครื่องมือท่องเว็บไซต์อัตโนมัติบนอุปกรณ์เคลื่อนที่สำหรับผู้พิการทางสายตา โดยใช้การจำแนกเนื้อหาบนเว็บไซต์

A Web Navigation Tool on Mobile Device for Visually impaired Persons using Web Main Content Detection

กฤษฎิ์ บรรณะชัยศิริสุข (Krit Bannachaisirisuk)* และ สุกรี สินธุภิญโญ (Sukree Sinthupinyo)*

บทคัดย่อ

เนื่องจากในปัจจุบันเราอยู่ในยุคดิจิทัล ที่ซึ่งเทคโนโลยีกลายเป็นส่วนหนึ่งในชีวิตประจำวัน ข่าวสาร สาระ และความบันเทิงมากมายนั้นสามารถหาได้จากบนอินเทอร์เน็ตอยู่ในรูปแบบของเว็บไซต์ที่มีการเกิดใหม่ขึ้นในทุกๆ วัน และมีจำนวนเกือบหนึ่งพันล้านเว็บไซต์ที่ใช้งานอยู่ในขณะนี้ แต่ผู้พิการทางสายตายังคงมีความลำบากในการเข้าถึงเนื้อหาหลักบนเว็บไซต์ งานวิจัยนี้มุ่งที่จะสร้างเครื่องมือเพื่ออำนวยความสะดวกให้กับผู้พิการทางสายตา ให้สามารถเข้าถึงเว็บไซต์ผ่านทางอุปกรณ์พกพาที่มีราคาถูก โดยระบบจะทำหน้าที่ในการรับคำค้นหาจากผู้ใช้ และนำไปค้นหาหัวข้อที่เกี่ยวข้องพร้อมทั้งสร้างรายการเว็บไซต์จำนวนหนึ่ง จากนั้นเมื่อผู้ใช้เลือกเว็บไซต์เครื่องมือจะทำการค้นหาเนื้อหาหลักด้วยการนำวิธีคุณลักษณะข้อความแบบตื้น (Shallow Text Feature) ร่วมกับการใช้คำค้น เพื่อนำไปสร้างตัวคัดแยกด้วยหลักการป่าแบบสุ่ม (Random Forest) เมื่อได้ตัวคัดแยกและทำการเรียนรู้จึงจะสามารถนำไปใช้เพื่อทำการคัดแยกเนื้อหาหลักภายในหน้าเว็บ แล้วทำการนำเสนอในรูปแบบของตัวอักษรที่ถูกเรียบเรียงใหม่ ทำให้ผู้พิการทางสายตาสามารถเข้าถึงเนื้อหาได้ผ่านทางโปรแกรมอ่านหน้าจอ (Screen Reader) ที่ทำหน้าที่อ่านข้อความที่ปรากฏให้ผู้พิการทางสายตามีความสามารถในการเข้าถึงเนื้อหาผ่านทางเสียงสังเคราะห์จากระบบ

การวัดผลของงานวิจัยชิ้นนี้คือการสร้างแอปพลิเคชันที่จะสามารถช่วยเหลือผู้พิการทางสายตา และมีประสิทธิภาพเพียงพอที่จะนำไปใช้งานได้จริง ผ่านการทดลองโดยใช้สมาร์ตโฟนระบบปฏิบัติการไอโอเอสและสามารถช่วยในการเข้าถึงเนื้อหาหลักอย่างรวดเร็วมากขึ้นกว่าวิธีการในปัจจุบัน

คำสำคัญ: เครื่องมือท่องเว็บไซต์ ผู้พิการทางสายตา การจำแนกเนื้อหา

Abstract

Even though, an endless resource of information1 is currently available over the internet, but the visually impaired persons are still not able to easily access to the content of a website because of disability. In this paper, we introduce a web navigation tool on mobile devices for helping those to be able to get into the main content of website faster. With the help of shallow text features integrated with keywords from the user, a classifier will be constructed by using the random forest method. The classifier will then be applied to remove the boilerplate and extract actual content from the webpage. Screen reader is also required here for reading the extracted main content aloud to the user who cannot navigate using their sight.

The expected outcome from this research is to create a mobile application that improves the accessibility of visually impaired person to get into the actual content faster and easier than existing methodologies.

Keywords: Web Navigation Tool, Visually Impaired, Content Detection.

1. บทนำ

ในการที่จะช่วยผู้พิการทางสายตาให้สามารถเข้าถึงเนื้อหาของเว็บไซต์ได้อย่างรวดเร็วนั้น นอกจากวิธีการ

* ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
* Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University.

คัดแยกเนื้อหาแล้ว งานวิจัยจะต้องพึ่งเทคโนโลยีช่วยเหลือ (Assistive Technology) ที่เรียกว่า เครื่องมืออ่านหน้าจอ (Screen Reader) ซึ่งถูกพัฒนาขึ้นมาเพื่อใช้ในการแปลงตัวอักษรที่อยู่บนหน้าจอออกมาเป็นรูปแบบของเสียงสังเคราะห์ จะทำให้ผู้ที่มีความพิการทางสายตาสามารถเข้าถึงข้อมูลเหล่านั้นได้ ในการพัฒนาหน้าเว็บไซต์ที่จะทำให้เครื่องอ่านหน้าจอมีประสิทธิภาพสูงสุด ควรที่จะต้องพัฒนาตามแนวทางที่ชื่อว่า Web Content Accessibility Guidelines 2.0 (WCAG 2.0) ถูกกำหนดขึ้นโดยองค์กร World Wide Web Consortium (W3C) (www.w3.org/) โดยแนวทางการพัฒนานี้จะทำให้หน้าเว็บไซต์สามารถรองรับการเข้าถึงจากเครื่องมืออ่านหน้าจอได้ดีขึ้น เนื่องจากมีการวางโครงสร้างของเอกสารเว็บไซต์ (HTML Document Object Model) ที่แบ่งออกเป็นระดับแบบโครงสร้างต้นไม้ ทำให้ผู้พิการสามารถข้ามไปยังจุดที่ต้องการได้อย่างง่ายดายและรวดเร็วมากยิ่งขึ้น แต่อย่างไรก็ตามมีเว็บไซต์เพียงจำนวนเล็กน้อยเท่านั้นที่ถูกพัฒนาตามหลักการนี้ ตามที่มีระบุอยู่ใน [1] ทำให้เครื่องมืออ่านหน้าจอยังคงขาดความสามารถที่จะทำให้ผู้ใช้สามารถเข้าถึงเนื้อหาหลักได้อย่างรวดเร็ว ซึ่งการที่ผู้พิการจะต้องใช้เครื่องมืออ่านหน้าจอในการฟังทุกๆ หัวข้อในหน้าเว็บไซต์ มีผลทำให้ต้องเสียเวลาและมีความยุ่งยากในการค้นหาเนื้อหาด้วยตัวเอง นอกจากนั้นหน้าเว็บไซต์อาจจะประกอบไปด้วยตัวอักษรมากมายที่ไม่ได้เกี่ยวข้องกับเนื้อหาที่ผู้ใช้ต้องการเข้าถึง เช่น เมนู โฆษณา หรือ บทความที่เกี่ยวข้องอื่นๆ ดังนั้นการนำวิธีการค้นหาและจำแนกเนื้อหาหลักของหน้าเว็บไซต์จึงเป็นสิ่งที่สำคัญที่จะนำมาใช้เพื่อค้นหาและเรียบเรียงข้อความมากมายในหน้าเว็บไซต์ออกมาเป็นรูปแบบใหม่ที่จะทำให้ผู้พิการทางสายตาสามารถเข้าถึงตัวเนื้อหาหลักได้อย่างรวดเร็ว

วิธีการที่จะทำการคัดแยกเนื้อหาที่แท้จริงจากหน้าเว็บนั้นสิ่งที่จำเป็นจะต้องเข้าใจอย่างหนึ่งคือโครงสร้างของหน้าเว็บไซต์ที่ถูกสร้างขึ้นจากภาษา HTML (HyperText Markup Language) โดยจะอยู่ในรูปแบบของโครงสร้างโมเดลแบบต้นไม้ (HTML Document Object Model Tree) ซึ่งไม่ได้ประกอบไปด้วยเพียงเนื้อหา และส่วนต่างๆ ที่แสดงออกมาให้เห็นบนหน้าเว็บเท่านั้น แต่ยังประกอบไปด้วยภาษาโปรแกรมต่างๆ เช่น JavaScript, Style Sheet เป็นต้น โดยส่วนต่างๆ เหล่านี้ไม่ได้มีผลสำคัญต่อผู้พิการทางสายตาอันเนื่องมาจาก ผู้ใช้นั้นจะไม่สามารถเข้าถึงส่วนที่เป็นการแสดงผลเหล่านั้นได้ ดังนั้นส่วนเหล่านี้จึงควรที่จะถูกคัดแยกออกจากส่วนที่เป็นเนื้อหาหลักของหน้าเว็บ จากการวิจัยค้นพบว่ากระบวนการที่เรียกว่าคุณลักษณะข้อความแบบตื้น (Shallow Text Feature) เป็นวิธีการที่เหมาะสม รวดเร็วและแม่นยำที่สุดในขณะนี้ ที่จะนำไปใช้เพื่อจำแนกเนื้อหาหลักในหน้าเว็บ

ในงานวิจัยนี้ ผู้วิจัยจึงตั้งใจในการนำเสนอแอปพลิเคชันบนสมาร์ตโฟนระบบไอโอเอสที่จะทำให้ผู้พิการทางสายตามีความสามารถในการเข้าถึงเนื้อหาหรือข้อมูลต่างๆ บนหน้าเว็บไซต์ได้อย่างรวดเร็วมากขึ้นกว่าวิธีการในปัจจุบัน

2. วรรณกรรมที่เกี่ยวข้อง

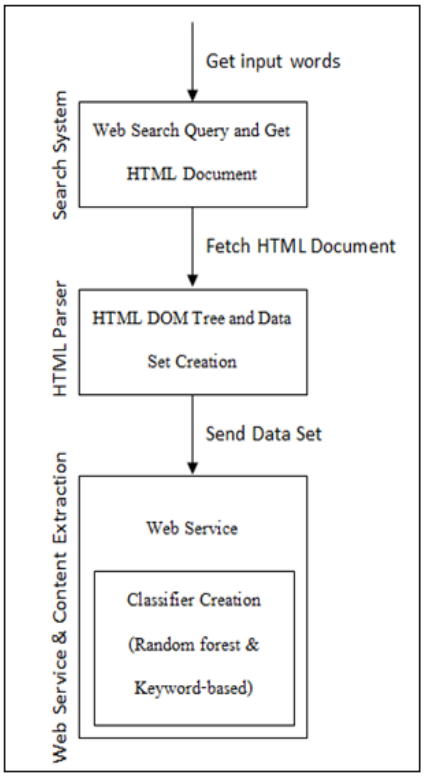
มีงานวิจัยของ Revilla และ Crow ที่เกี่ยวกับการแสดงการประเมินประสิทธิภาพการเข้าถึงเว็บไซต์ของผู้พิการทางสายตาโดยทำการวัดจากเวลาที่ใช้ในการเข้าถึงเนื้อหา [2] ซึ่งจะนำมาใช้ในการวัดประสิทธิภาพของงานวิจัยนี้ โดยได้มีการแสดงถึงเวลาที่ใช้ในเว็บแต่ละประเภทนั้นมีความแตกต่างกัน อีกทั้งมีความเกี่ยวข้องกับความซับซ้อนของเนื้อหา ถึงแม้เว็บไซต์เหล่านั้นมีการพัฒนาตามหลัก WCAG 2.0 แล้วก็ยังไม่สามารถทำให้การเข้าถึงเนื้อหาของผู้พิการทางสายตาได้ดีขึ้นจากงานวิจัยของ Power [3] มีการทำการทดลองและได้พบว่าเว็บไซต์ที่ไม่ได้พัฒนาตามแนวทางจะมีจำนวนปัญหาอยู่ถึง 102 ปัญหา หลังจากที่ถูกพัฒนาโครงสร้างเว็บตาม WCAG 2.0 ปัญหาจะลดลงเหลือเพียง 73 ปัญหา ที่ระดับ A (Level A) ปัญหาอันดับหนึ่งคือ เนื้อหาที่เจอนั้นไม่ใช่เนื้อหาที่ต้องการกล่าวคือ เนื้อหาเหล่านั้นเป็นส่วนอื่นๆ ที่ไม่ใช่เนื้อหาหลัก และถัดมาคือการที่หาเนื้อหาไม่เจอบนหน้าเว็บนั้นๆ เลย

วิธีที่จะคัดแยกเนื้อหาบนหน้าเว็บจะต้องเข้าใจถึงโครงสร้างของหน้าเว็บไซต์เนื่องจากงานวิจัยจะทำการเรียกตัวเว็บไซต์และทำการจำแนกจากแหล่งข้อมูลนั้นผ่านตัวคัดแยกเนื้อหาของผู้จัดทำ และมีงานวิจัยที่ได้กล่าวถึงการจำแนกองค์ประกอบของเว็บไซต์ [4] ด้วยวิธีการที่เรียกว่า วิศวกรรมย้อนรอย (Reverse Engineer) ซึ่งเป็นการนำ HTML Document ที่ได้รับจากการส่งคำร้องไปที่ที่อยู่นั้นๆ จากนั้นทำการแปลงให้อยู่ในรูปแบบของ โครงสร้างโมเดลแบบต้นไม้ จากนั้นนำเนื้อหาภายในโหนดลูก ออกมาแสดง

ไว้ในโหนดแม่ในกรณีที่โหนดแม่ไม่มีเนื้อหาใดๆ เพื่อเป็นการนำเสนอในรูปแบบหัวข้อของแต่ละโหนด ต่อมางานวิจัยของ Tomohiro Manabe [5] ได้กล่าวถึงวิธีการจัดกลุ่มเพื่อสร้างโมเดลต้นไม้จากเอกสารเว็บไซต์ในรูปแบบการจัดลำดับเช่นกัน วิธีการนี้จะช่วยทำให้ผู้พิการสามารถใช้เครื่องมืออ่านหน้าจอ ในการข้ามไปยังหัวข้อต่างๆ ที่น่าสนใจภายในเว็บไซต์นั้นได้อย่างรวดเร็วมากขึ้น ซึ่งตรงกับการวิจัยของ Takagi [6] ได้กล่าวไว้ว่าผู้พิการทางสายตานั้นใช้เวลาในการทำสิ่งเดียวกับผู้ที่มองเห็นมากกว่าถึง 10 เท่า จึงเป็นที่น่าคิดว่าผู้พิการนั้นมีความลำบากในการใช้งานเว็บไซต์อยู่มากเพียงใด ถึงแม้วิธีนี้จะทำให้ผู้พิการทางสายตาสามารถเข้าถึงเนื้อหาได้รวดเร็วมากขึ้นแล้วก็ตาม แต่ก็ยังคงต้องใช้เวลาในการเข้าถึงทุกๆ หัวข้อที่ปรากฏจนกว่าจะไปถึงส่วนที่เป็นเนื้อหาที่แท้จริง จึงมีการศึกษาต่อเพื่อที่จะหาวิธีการแยกเนื้อหาหลักโดยตรง โดยใช้การเรียนรู้ของเครื่อง (Machine Learning) มาประยุกต์ในงานวิจัย และวิธีการที่น่าสนใจคือวิธีคุณลักษณะแบบตื้น (Shallow Text Feature) ที่ถูกคิดค้นขึ้นโดย Christian Kohlschütter [7] เป็นการนำต้นไม้ตัดสินใจ (Decision Tree) มาใช้ในการคัดแยก ส่วนที่ไม่ต้องการออกจากเนื้อหาของเว็บ โดยวิเคราะห์จากค่าเฉลี่ยความยาวของคำ และความยาวของแต่ละประโยค พิจารณาจากช่องว่าง (White Space) หรือมหัพภาค (Full-stop) ที่ถูกใช้แบ่งแต่ละคำหรือประโยคออกจากกันในภาษาอังกฤษ ผู้วิจัยได้กล่าวอีกว่าโดยส่วนใหญ่แล้วนั้นเนื้อหาหลักในหน้าเว็บไซต์มักจะถูกล้อมรอบไปด้วยส่วนอื่นๆ ที่ไม่ใช่เนื้อหาหลักหรือที่เรียกว่าบอยเลอร์เพลต (Boilerplate) ไม่ว่าจะเป็นส่วนหัวด้านบน (Header) ส่วนท้ายด้านล่าง (Footer) หรือด้านซ้ายและขวาที่มักจะเป็นส่วนของเมนู (Navigation) เพื่อใช้ไปยังหน้าอื่นๆของเว็บไซต์ นอกจากนี้ Christian ได้แนะนำวิธีในการคำนวณเพื่อหาความหนาแน่นของคำอีกวิธีหนึ่งที่มีชื่อว่า คุณลักษณะความหนาแน่น (Densitometric Features) [8] ซึ่งเป็นการคำนวณจากการนำจำนวนของตัวอักษรที่อยู่ในบล็อกนั้นๆ หารด้วยจำนวนบรรทัด ซึ่งจำนวนบรรทัดจะหาได้จากการนำตัวอักษรในบล็อกนั้นมาตัดทอนด้วยจำนวนตัวอักษรต่อบรรทัดที่เรากำหนด (ในภาษาอังกฤษกำหนดให้อยู่ระหว่าง 80 - 90 ตัวอักษรต่อบรรทัด) จากนั้นจึงนำข้อมูลเข้าสู่ต้นไม้ตัดสินใจ เพื่อทำการค้นหาเนื้อหาหลักจากความหนาแน่น ซึ่งเป็นโหนดปลายสุดของต้นไม้ตัดสินใจ (Leaf Node) Lundgren [9] ได้นำวิธีคุณลักษณะแบบตื้นใช้ในการคัดแยกเนื้อหาข่าวจากเว็บไซต์ข่าวและคิดค้นวิธีเพิ่มความแม่นยำในการสร้างตัวคัดแยก (Classifier) ด้วยการใช้วิธีป่าแบบสุ่ม (Random Forest) ซึ่งเป็นการสร้างต้นไม้ตัดสินใจจำนวน 100 ต้นและกำหนดค่าเริ่มต้นแบบสุ่มให้กับทุกๆ การตัดสินใจ จากนั้นให้ทำการเรียนรู้ตัวคัดแยกจากเซตตัวอย่าง (Sample Set) จะทำให้ต้นไม้ตัดสินใจแต่ละต้นจะมีค่าการตัดสินใจต่างกัน เมื่อนำข้อมูลที่ต้องการคัดแยกเข้าไปในกลุ่มต้นไม้ตัดสินใจนี้ แต่ละต้นจะทำการโหวตเพื่อให้ได้มาซึ่งประเภทของข้อมูลนั้น จากการทดลองของ Lundgren ได้ให้ผลลัพธ์ความแม่นยำสูงถึงร้อยละ 96.4 จากการเรียนรู้กับเว็บไซต์ข่าวสารจาก Google News จำนวนกว่า 621 เว็บไซต์ จึงเป็นการบ่งชี้ว่าเป็นวิธีการที่มีประสิทธิภาพสูงที่สุดและมีความซับซ้อนที่น้อยกว่าวิธีการอื่นๆ ในปัจจุบัน

3. วิธีการดำเนินงาน

ในงานวิจัยนี้เครื่องมือ ได้ถูกออกแบบให้มีส่วนประกอบหลักทั้งหมด 4 ส่วนคือ ระบบค้นหา (Search System) ตัวแปลงเอกสารเว็บไซต์ (HTML Parser) เว็บบริการ (Web Service) และการคัดแยกเนื้อหาหลัก (Content Extraction) โดยมีความสัมพันธ์กันของแต่ละส่วนประกอบ ดังในภาพที่ 1



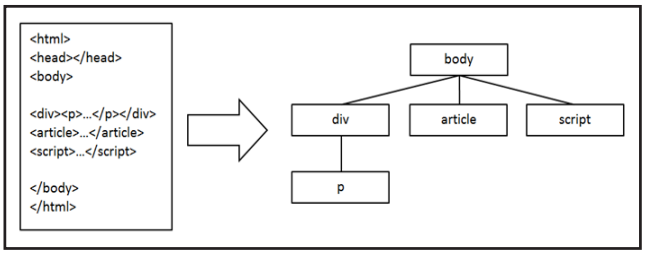
ภาพที่ 1 *ลักษณะการทำงานของเครื่องมือ*

3.1. ระบบค้นหา (Search System)

ระบบค้นหาจะมีหน้าที่ในการนำคำค้นหาจากผู้ใช้และนำไปทำการค้นหาหัวข้อข่าวที่มีความเกี่ยวข้อง โดยนำระบบค้นหาเว็บไซต์ Google.com มาใช้เพื่อให้ได้ผลลัพธ์ที่มีประสิทธิภาพ ได้ข่าวสารที่รวดเร็วและถูกจัดลำดับตามความเหมาะสมกับคำค้นหา

3.2. ตัวแปลงเอกสารเว็บไซต์ (HTML Parser)

ในการส่งคำร้องขอเว็บไซต์ไปยังที่อยู่ที่กำหนด เครื่องมือจะได้รับเอกสารที่อยู่ในภาษา HTML (Hypertext Markup Language) ซึ่งอยู่ในรูปแบบจำลองอ็อบเจกต์เอกสารหลักจากที่เครื่องมือได้รับเอกสารแล้วจะทำการแปลงเอกสารเว็บไซต์ให้อยู่ในรูปของโมเดลอ็อบเจกต์ต้นไม้ (Object Tree Model) [10] ดังที่แสดงในภาพที่ 2



ภาพที่ 2 *การแปลง HTML DOM เป็น Object Tree*

จากนั้นจะทำการค้นหาผลลัพธ์ที่ได้จากการใช้คำค้นหาจากสองส่วนคือ ส่วนที่เป็นหัวข้อ (Topic) และส่วนที่เป็นรายละเอียดย่อ (Description) และทำการจัดเรียงและแสดงผลออกมาใหม่เป็นรายการแบบลิสต์บนแอปพลิเคชันดังภาพที่ 3 หลังจากที่ผู้ใช้ได้รายการของผลการค้นหาและทำการเลือกหัวข้อที่ต้องการแล้ว เครื่องมือจะทำการแปลงเอกสารเว็บไซต์อีกรอบ โดยจะนำอ็อบเจกต์ต้นไม้ไปทำการคัดกรองเพื่อค้นหาองค์ประกอบ (Element) [11] ที่ไม่สามารถเป็นเนื้อหาได้โดยอ้างอิงกับองค์ประกอบทั้งหมดบนเว็บไซต์ขององค์กร W3C1 และพิจารณาจากแท็ก (Tag) ซึ่งเป็นตัวกำหนดประเภทของอิลิเมนต์ จากนั้นจะถูกนำไปสร้างรายการอ็อบเจกต์ (Object List) และกำหนดให้แต่ละอ็อบเจกต์มีเอกลักษณ์ (Identity) เพื่อที่ใช้ในการอ้างถึงภายหลัง แล้วทำการสร้างข้อมูลจำเพาะ (Data Set) ซึ่งประกอบไปด้วยความหนาแน่นของตัวอักษร อัตราส่วนลิงก์ และเพิ่มเติมในส่วนของอัตราส่วนของคำค้นหาที่ปรากฏ แล้วส่งให้กับเว็บบริการทำการวิเคราะห์เพื่อหาส่วนเนื้อหา ซึ่งจะกล่าวถึงในหัวข้อถัดไป

ภาพที่ 3 *การรายการค้นหาจาก GoogleNews เป็นรายการผลลัพธ์การค้นหาบนเครื่องมือ*

3.3. เว็บบริการ (Web Service)

เว็บบริการเก็บส่วนที่ใช้ในการประมวลผลและวิเคราะห์ข้อมูลที่ถูกส่งมาจากเครื่องมือ เพื่อใช้ในการคัดแยกเนื้อหาหลัก โดยการแยกการทำงาน ออกจากตัวเครื่องมือที่อยู่กับผู้ใช้ เป็นการลดการใช้ทรัพยากรของเครื่องมือ รวมถึงเป็นการรวมส่วนของการคัดแยกเนื้อหาหลักเป็นส่วนกลางไว้ในที่เดียว และตัวคัดแยกเนื้อหาจะถูกพัฒนาด้วยการใช้ข้อมูลเรียนรู้จากแหล่งเดียวกันเท่านั้น

3.4. การคัดแยกเนื้อหาหลัก (Content Extraction)

ส่วนของการคัดแยกเนื้อหาหลัก จะถูกนำไปอยู่บนเว็บบริการ ซึ่งประกอบด้วยการใช้ต้นไม้ตัดสินใจที่มี การตัดสินคลาสของข้อมูลจากความหนาแน่นของตัวอักษร (1) โดยคำนวณจากการนำจำนวนคำในบล็อก b_x หารด้วยจำนวนบรรทัดที่คิดจากการนำจำนวนตัวอักษรหารด้วยความยาวของตัวอักษรต่อหนึ่งบรรทัด (ในภาษาอังกฤษ กำหนดไว้ที่ 80 ตัว

$$Q(b_x) = \frac{Number\ of\ tokens\ in\ b_x}{Number\ of\ lines\ in\ b_x} \quad (1)$$

ในภาษาไทยทางผู้วิจัยได้กำหนดไว้ที่ 120 ตัว [8] ความหนาแน่นของลิงก์ปรากฏคำนวณตัวอักษรภายใต้แท็ก A ต่อจำนวนตัวอักษรทั้งหมดที่อยู่ภายในบล็อก b_x (2)

$$a(b_x) = \frac{Number\ of\ A\ tags\ in\ b_x}{Number\ of\ any\ tags\ in\ b_x} \quad (2)$$

และอัตราส่วนของคำค้นหาที่ปรากฏ ถูกคำนวณจากการนับจำนวนคำค้นหาที่ปรากฏ ต่อจำนวนทั้งหมดในหน้าเว็บ (3)

$$K = \frac{Number\ of\ keywords\ appear\ in\ b_x}{Number\ of\ total\ keywords\ appear} \quad (3)$$

จากนั้นจึงนำข้อมูลที่ได้ไปใช้วิธีการป่าสุ่มที่เป็นการสร้างต้นไม้ตัดสินใจจำนวน 100 ต้น แล้วกำหนดค่าเริ่มต้นแบบสุ่มให้กับทุกการตัดสินใจ และนำกลุ่มของต้นไม้ตัดสินใจ (Forest) ไปทำการเรียนรู้จากข้อมูลทดสอบ (Training set) ซึ่งทางผู้วิจัยได้เตรียมจากเว็บไซต์จริงที่มาจากการใช้คำค้นหาจำนวน 10 คำจากการสอบถามจากผู้พิการทางสายตาและนำผลลัพธ์ 5 เว็บไซต์แรกของแต่ละคำค้นหา ทั้งหมด 50 เว็บไซต์ มีส่วนประกอบกว่า 14,000 ส่วน มาใช้ในการสร้างตัวตัดสินใจในการค้นหาเนื้อหาหลัก โดยการวิเคราะห์จะเกิดจากการค้นหาส่วนประกอบที่ได้รับเสียงโหวตมากที่สุดจากการใช้วิธีการป่าแบบสุ่ม ซึ่งตามการวิจัยพบว่ามีแนวโน้มที่จะเป็นเนื้อหามากที่สุด จากนั้นระบบจะทำการตอบคำร้องมาที่เครื่องมือของผู้ใช้ และทำการแสดงผลเนื้อหาหลักนั้นให้เครื่องมืออ่านหน้าจอทำหน้าที่ในการอ่าน

ในการวัดผลทางผู้วิจัยนำคำค้นหาจากผู้พิการทางสายตามาใช้สร้างกรณีทดสอบ โดยมีคำค้นหา จำนวน 20 คำ และรายการผลลัพธ์ของแต่ละคำค้นหา จำนวน 5 เว็บไซต์แรก เป็นจำนวนทั้งสิ้น 100 เว็บไซต์ทำการทดสอบกับเครื่องมือและนำมาเปรียบเทียบกับเว็บไซต์จริงเพื่อสรุปเป็นผลลัพธ์ของงานวิจัยในการค้นหาเนื้อหา

4. ผลการดำเนินงาน

เครื่องมือที่ถูกพัฒนาจะประกอบไปด้วย 3 หน้าหลัก และถูกออกแบบเพื่อรองรับระบบการเข้าถึงจากระบบ Voice Over ซึ่งเป็นเครื่องมืออ่านหน้าจอของระบบปฏิบัติการไอโอเอสที่สามารถอ่านหน้าจอให้แก่ผู้ใช้ได้อย่างมีประสิทธิภาพ

4.1. หน้าค้นหา

หน้าจอที่ใช้ในการรับคำค้นหาจากผู้ใช้ผ่านการใช้แป้นพิมพ์หรือ การใช้ระบบแปลงเสียงเป็นตัวอักษรซึ่งเป็นฟังก์ชันที่มีอยู่ในระบบการเข้าถึงบนระบบไอโอเอส โดยมีเพียงช่องเดียวเพื่อลดความสับสนของผู้ใช้งาน โดยสามารถใส่ได้ทีละหลายคำ และใช้ช่องว่าง (Whitespace) ในการคั่นแต่ละคำ คำค้นหาจะถูกเรียงลำดับความสำคัญ เพื่อนำไปใช้ในการค้นหาผลลัพธ์ ดังในภาพที่ 4

ภาพที่ 4 *หน้าจอหน้าค้นหาของเครื่องมือ*

4.2. หน้ารายการค้นหา

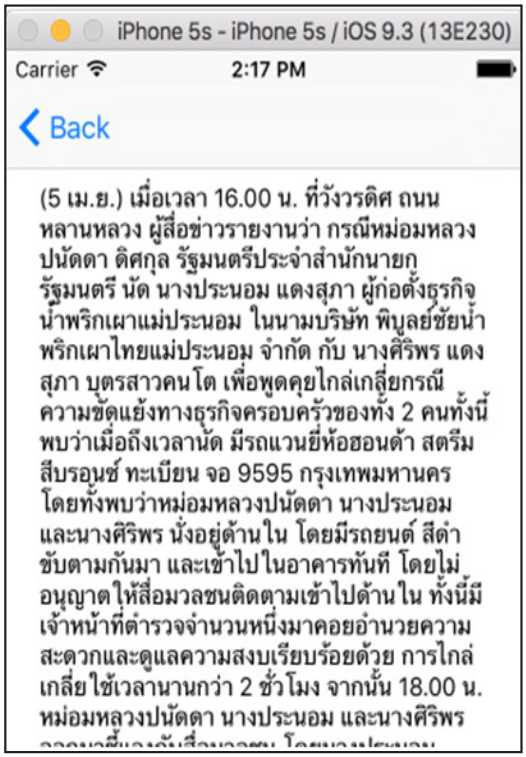
หน้าจอนี่จะทำหน้าที่แสดงผลลัพธ์ที่ได้จากการนำคำค้นหาของผู้ใช้ไปค้นหาโดยใช้เว็บไซต์ Google.com ซึ่งเกิดจากการนำเอกสารเว็บไซต์ไปดัดแปลง และแสดงผลออกมาในรูปแบบใหม่ เพื่อให้ผู้ใช้สามารถเข้าถึงได้ง่ายขึ้น ดังในภาพที่ 5

ภาพที่ 5 *หน้าจอผลลัพธ์การค้นหา*

4.3. หน้าแสดงเนื้อหา

หน้าแสดงเนื้อหาหลักเป็นหน้าผลลัพธ์ที่ได้หลังจากผู้ใช้ทำการเลือกหัวข้อจากหน้ารายการค้นหา จากนั้นเครื่องมือจะทำการแปลงเอกสารเว็บไซต์จากที่อยู่ที่ ถูกส่งมาจากหน้า

รายการค้นหา และสร้างไฟล์ข้อมูลที่บรรจุข้อมูลเกี่ยวกับทุกอ็อบเจกต์ นำไปทำการวิเคราะห์และนำผลลัพธ์ที่ได้จากเว็บบริการเครื่องมือจะระบุได้ว่าส่วนใดที่มีความเป็นไปได้ที่จะเป็นเนื้อหาหลักของหน้าเว็บไซต์นั้นมากที่สุด และนำส่วนนั้นมาแสดงเป็นส่วนแรก และเครื่องมืออ่านหน้าจะทำการอ่านเนื้อหาดังกล่าว ในภาพที่ 66



ภาพที่ 6 *หน้าจอผลลัพธ์เนื้อหาหลักที่ได้บนเครื่องมือ*

จากการทดลองทำให้พบว่าเครื่องมือสามารถค้นหาเนื้อหาหลักได้ด้วยความแม่นยำที่ 97.26% และเมื่อนำไปทดสอบกับบนอุปกรณ์จริงเปรียบเทียบวัดผลจากจำนวนขั้นตอน (Step) ที่ใช้ในการเข้าถึงเนื้อหา เครื่องมือของผู้วิจัยสามารถลดจำนวนขั้นตอนการเข้าถึงด้วยวิธีการทั่วไปซึ่งอยู่ที่ 12 ขั้นตอนลดลงมาอยู่เหลือเพียง 7 ขั้นตอน

5. สรุป

เครื่องมือนี้เป็นการนำองค์ความรู้พฤติกรรมการเข้าถึงเนื้อหาบนเว็บไซต์ของผู้พิการทางสายตามาพัฒนาแอปพลิเคชันบนสมาร์ตโฟนเพื่อเข้าไปช่วยเหลือผู้พิการทางสายตาในการเข้าถึงเนื้อหาโดยอาศัยหลักการในการจับส่วนประกอบของข้อมูลบนหน้าเว็บไซต์ ร่วมกับวิธีสร้างตัวคัดแยกจากวิธีการแบบป่าสุ่ม (Random Forest) โดยใช้วิชาการเรียนรู้ของเครื่อง (Machine Learning) ซึ่งจะช่วยให้ผู้พิการทางสายตาเข้าถึงเนื้อหาอย่างถูกต้องและรวดเร็วกว่าการเข้าถึงเว็บไซต์ด้วยวิธีในปัจจุบัน รวมทั้งเป็นการคำนึงถึงความสำคัญของผู้พิการที่ต้องการเข้าถึงข้อมูลเช่นเดียวกับผู้คนทั่วไป เพียงแต่ขาดโอกาสที่จะได้รับและผู้สนับสนุนที่เพียงพอจากสังคมในปัจจุบัน ทางผู้วิจัยจึงเห็นว่าเทคโนโลยีน่าจะเป็นคำตอบที่จะสามารถเข้าไปช่วยเหลือผู้คนเหล่านั้นด้วยการพัฒนาเครื่องมือนี้ยังมีแนวทางการพัฒนาต่อได้ โดยทำให้แอปพลิเคชันสามารถรองรับเว็บไซต์เวอร์ชั่น 2.0 ซึ่งได้มีการกล่าวถึงปัญหาไว้ใน [12] เช่นกัน

6. เอกสารอ้างอิง

[1] S. Jarungsouk. “Developing Instructional Materials on Problem-Solving Ability.” *Eau Heritage Journal*. Vol. 8.

[1] V. L. Hanson and J. T. Richards. “Progress on website accessibility.” *ACM Transactions on the Web*, Vol. 7, No. 1, Article 2, pp. 30, March 2013.

[2] L. F. Revilla, J. Crow. “Interpretation of Web Page Layouts by Blind Users.” JCDL’10, *Gold Coast, Queensland, Australia*, June 21–25, 2010.

[3] C. Power, “Guidelines are Only Half of the Story: Accessibility Problems Encountered by Blind Users on the Web.” *CHI 2012*, Austin, Texas, USA, May 5–10, 2012.

[4] Pauli P. Y. Lai. “Application of Content Adaptation in Web Accessibility for the Blind.” *20th International World Wide Web Conference*, Hyderabad, India, 28-29 March, 2011.

[5] T. Manabe. “Extracting Logical Hierarchical Structure of HTML Documents Based on Headings.” *41st International Conference on Very Large Data Bases*, Kohala Coast, Hawaii, 31st August - 4th September 2015.

[6] H. Takagi, S. Saito, K. Fukuda, and Asakawa, “Analysis of navigability of Web applications for improving blind usability.” *ACM Transactions on Computer-Human Interaction. Interact*. Vol. 14, No. 3, Article 13, pp. 37, September 2007.

[7] C. Kohlschütter, P. Fankhauser, and W. Nejdl. "Boilerplate Detection using Shallow Text Features." *WSDM'10*, New York City, New York, USA, 4–6 February, 2010.

[8] C. Kohlschütter and W. Nejdl. "A Densitometric Approach to Web Page Segmentation." *CIKM'08*, Napa Valley, California, USA, 26–30 October, 2008.

[9] E. Lundgren, P. Papapetrou, and Lars Asker. "Extracting news text from web pages: an application for the visually impaired." *PETRA '15*, Island of Corfu, Greece, 01 - 03 July, 2015.

[10] P. M. Joshi and S. Liu. "Web Document Text and Images Extraction using DOM Analysis and Natural Language Processing." *DocEng'09*, Munich, Germany 16–18 September, 2009.

[11] S. Gupta, G. Kaiser, D. Neistadt and P. Grimm. "DOM-based Content Extraction of HTML Documents." *12th International World Wide Web Conference*, May 2003.

[12] J. Hailpern. "WEB 2.0: Blind to an Accessible New World." *WWW'09*, Madrid, Spain, 20–24 April, 2009.