Speech2Mindmap: Testing the Accuracy of Unsupervised Automatic Mindmapping Technology With Speech Recognition

Bumsoo Lee
School of Mechanical Engineering,
Georgia Institute of Technology,
801 Ferst Drive, MRDC 3340,
Atlanta, GA 30332
e-mail: blee300@gatech.edu

Brian Feldman
School of Mechanical Engineering,
Georgia Institute of Technology,
801 Ferst Drive, MRDC 3340,
Atlanta, GA 30332
e-mail: bfeldman3@gatech.edu

Katherine Fu
School of Mechanical engineering,
Georgia Institute of Technology,
801 Ferst Drive, MRDC 4508,
Atlanta, GA 30332
e-mail: katherine.fu@me.gatech.edu

This research aims to augment human cognition through the advancement and automation of mindmapping technologies, which could later support human creativity and virtual collaboration. Mindmapping is a visual brainstorming technique that allows problem solvers to utilize the human brain's ability to retrieve knowledge through similarity and association. While it is a powerful tool to generate concepts in any phase of s or design, the content of mindmaps is usually manually generated while listening or conversing and generating ideas, requiring a high cognitive load. This work introduces the development of a speech-driven automated mindmapping technology, called Speech2Mindmap. The specifics of the Speech2Mindmap algorithm are detailed, along with two case studies that serve to test its accuracy in comparison to human-generated mindmaps, using audio recorded speech data as input. In the first case study, the Speech2Mindmap algorithm was evaluated on how well it represents manually generated human mindmapping output. The second case study evaluated the reliability of the Speech2Mindmap algorithm and examined the best performing methods and conditions to achieve the greatest similarity to human-generated mindmaps. This research demonstrates that the Speech2Mindmap algorithm is capable of representing manually generated human mindmapping output and found the best performing methods and conditions to generate a mindmap that is 80% similar, on average, to human-generated mindmaps. [DOI: 10.1115/1.4052282]

Keywords: artificial intelligence, collaborative design, conceptual design, creativity and concept generation, design process

1 Introduction

As science and technology advances, the importance of creativity in engineering design increases [1–3], but it seems harder to come up with novel or creative ideas. Mark Twain argues that "There is no such thing as a new idea. We simply take a lot of old ideas and put them into a sort of mental kaleidoscope." One of the most innovative people of all time, Steve Jobs, once said "Creativity is just connecting things." Mindmapping is a very popular brainstorming method that connects and clusters multiple ideas to generate new innovative ideas.

Mindmapping is a visual brainstorming technique that allows problem solvers to utilize the human brain's ability to retrieve knowledge through similarity and association. While it is a powerful tool to generate concepts in any phase of problem-solving or design, the content of mindmaps is usually manually generated while listening or conversing, inducing a high cognitive load for the documentarian(s)—who must split their attention between the creation of the mindmap artifact and the generation of ideas. The end goal of the work presented here is to augment human cognition through the advancement and automation of mindmapping technologies, which will enhance and enable human creativity and virtual collaboration while saving energy and time. The specific objective of this work is to automatically generate mindmaps of team brainstorming sessions based on audio speech data. The contribution of the work presented here is the demonstration of a new technique for completely automating mindmapping based solely on audio data of brainstorming sessions, with preliminary validation of fidelity to human-generated mindmaps through computational comparison case studies. Integrating speech into automated mindmapping technology opens up opportunities for a real-time responsive mindmapping tool which will enable capturing ideas synchronously, ideas that might be otherwise lost in the process of manual documentation or conversational flow. This work presents a novel method to develop a mindmap from a conversational transcript, text that is not necessarily grammatically well-articulated. The scope of this study does not include an evaluation of the technology's effects on design cognition, but this will be evaluated in future work.

2 Background

2.1 Introduction to Mindmapping. Mindmapping was first introduced by Tony Buzan in the 1970 s [4]. T. Buzan and B. Buzan describe the mindmap as a tool to help capture one's thinking process or mental model through representation with words, drawings, and colors [4]. A mindmap starts with a main keyword or idea in its center and then branches out to other related keywords or ideas forming a hierarchical structure. Buzan and Buzan suggest using a single word for each related idea, creating flexibility for explorative thinking, and one pictorial representation of each related idea, engaging human vision to expand knowledge retrieval [4,5]. Mindmapping enhances the cognitive abilities to think logically, memorizing, retrieving memory, learning, drawing associations, and thinking creatively through graphical visualization [4,6] Even with these advantages, many people are hesitant to use mindmaps while brainstorming because of the

Contributed by the Design Theory and Methodology Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received January 16, 2021; final manuscript received June 18, 2021; published online September 21, 2021. Assoc. Editor: Daniel A. McAdams.

Bright colored ball
Electronic Grid with ball emitter
Sound horn in ball
Exploding ball
Golf lessons
GPS System
Scent-Human
Scent-Dog
Virtual golf
Pressure sensitive ground
String attached to ball
⋮

Smoke trail
Shorter golf course
Putt-Putt golf
Spotters paced every 10 m
Colored golf course
Trajectory calculation system
Robotic arm hits ball
Mini-camera in ball
Light emitting ball
Ball shoots flare
Plexiglasside walls on golf course
Funnel shaped golf course
Speaker in ball; use microphone to call yourself
⋮

Fig. 1 Classical Brainstorming Example Result for the design problem: "Design a system to detect a golf ball hit from the tee box" [9,10]

perceived amount of mental effort and artistic skill required for illustrating ideas [7].

Mindmapping employs information visualization techniques to allow humans to use perception to capture information, recognize patterns, compare similarities and differences, draw associations, communicate ideas, and generate new ideas or knowledge. Mindmapping is a popular method of enhanced brainstorming and concept generation among designers and researchers [8]. As an example, consider the scenario below. A team is working to generate early-stage concepts for the following design problem: "Design a system to detect a golf ball hit from the tee box" [9]. With classical brainstorming [10], the ideas from the team might resemble a list like the one in Fig. 1.

Now consider the same design problem, but using a mindmap to generate concepts, as depicted in Fig. 2 [9]. By recognizing patterns and groupings in the ideas generated, team members can generate new interconnected concepts, exploring the idea space more broadly than classical brainstorming. Mindmapping helps humans to understand and to solve problems in a logical and structural manner by visually dividing a central idea into its components [11]. It has been shown to be an effective tool for taking notes, learning new knowledge, organizing structural data, summarizing documents, analyzing problems, finding new inspiration, and generating new ideas [8,11–15]. With the abundance of data and complexity of knowledge from modern technology, individuals lacking resource-based learning techniques can experience cognitive overload, resulting in an inability to process information [16]. According to a study of the impact of cognitive load on creative thinking performance by Redifer et al., higher cognitive load in participants negatively impacted their creative thinking performance [17]. Using automated mindmaps during brainstorming or idea generation reduces cognitive load by enabling an automated graphical overview of one's thought process with verbal and/or visual representations, presenting associations and relationships between ideas, expanding one's working memory through the recall of ideas instead of memorizing, and by relieving worries related to visual aesthetics and drawing ability [8,16].

2.2 Mindmapping Technology. In their simplest form, mindmaps can be generated manually on paper. However, paper-based and whiteboard mindmaps are restrictive because they do not

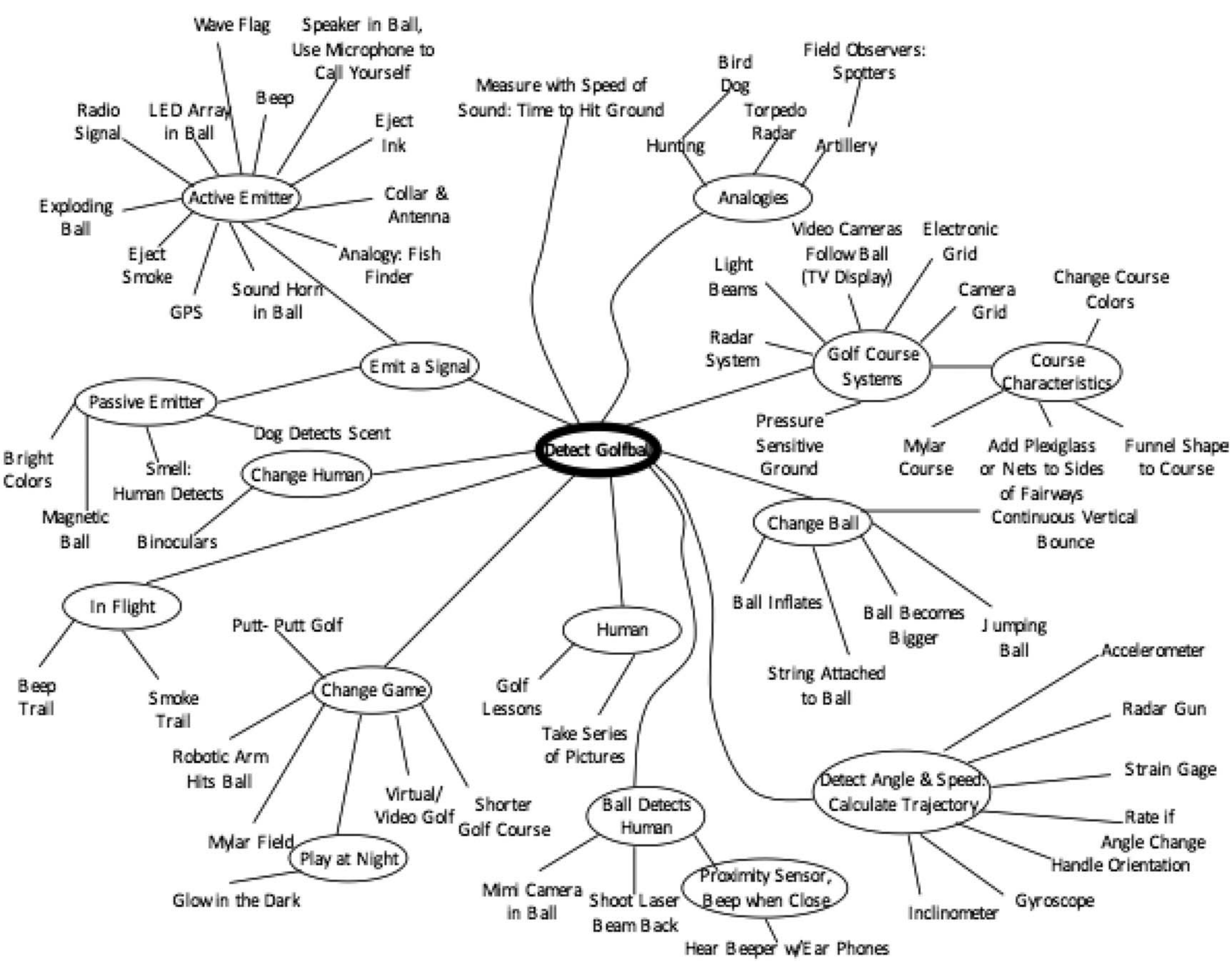

Fig. 2 Mindmapping example result for the design problem: "Design a system to detect a golf ball hit from the tee box" [9,10]

	Ref	Tool Name	Digital	Free of Location	Software /Web	Auto / Manual	Input Source	Support Image node	Free	Open Source	Filter & Rearrange	Synchronous Editing	AI Assist
Traditional / User Generated		White Board	X	X	N/A	M	Marker	O	O	N/A	X	△	X
Traditional / User Generated		Post-it	X	X	N/A	M	Pencil & Pen	O	O	N/A	O	O	X
Traditional / User Generated		Paper	X	X	N/A	M	Pencil & Pen	O	O	N/A	X	△	X
Digital / User Generated	(FreeMind, 2016)	FreeMind	O	O	Software	M	Keyboard & Mouse	O	O	O	X	X	X
Digital / User Generated	(XMind, 2016)	XMind	O	O	Software	M	Keyboard & Mouse	O	△	O	Search	X	X
Digital / User Generated	(MindManager, 2018)	MindManager	O	O	Software	M	Keyboard & Mouse	X	X	X	Search, Filter	X	X
Digital / User Generated	(iMindMap, 2018)	iMindMap	O	O	Software	M	Keyboard & Mouse	O	X	X	X	X	X
Digital / User Generated	(Creately, 2018)	Creately	O	O	Software	M	Keyboard & Mouse	O	△	X	X	O	X
Digital / User Generated / Multi-User Editing	(ThinkMap, 2018)	ThinkMap	O	O	Software	M	Keyboard & Mouse	X	X	X	O	X	△
Digital / User Generated / Multi-User Editing	(MeisterLabs, 2018)	MindMeister	O	O	Web	M	Keyboard & Mouse	O	△	X	Search	O	X
Digital / User Generated / Multi-User Editing	(Cacoo, 2018)	Cacoo	O	O	Web	M	Keyboard & Mouse	O	X	X	X	O	X
Digital / User Generated / Multi-User Editing	(Lucidchart, 2018)	Lucidchart	O	O	Web	M	Keyboard & Mouse	O	X	X	X	O	X
Digital / User Generated / Multi-User Editing	(Coggle)	Coggle	O	O	Web	M	Keyboard & Mouse	O	△	X	X	O	X
Digital / User Generated / Multi-User Editing	(Shih et al., 2009)	GroupMind	O	O	Software	M	Keyboard & Mouse	O	X	X	X	O	X
Digital / User Generated / Multi-User Editing	(Ang, Rzadca, & Datta, 2010)	ShareMind	O	O	Plug-in	M	Keyboard & Mouse	O	O	O	X	O	X
Digital / Automatic	(Abdeen et al., 2009)	M2Gen	O	O	Software	A	Text File	O	△	X	X	X	X
Digital / Automatic	(Elhoseiny & Elgammal, 2012)	English2MindMap	O	O	Software	A	Text File	O	△	X	X	X	X
Digital / Automatic	(Kudelić et al., 2012)	MindMap Gen Software	O	O	Software	A	Text File, Webpage	O	△	X	X	X	X

Legend O : Supports △ : Semi supports X : Does not support

Fig. 3 Benchmarking current mindmapping technology

enable re-spacing of content when certain areas become densely populated. Writing each idea or concept on a sticky note is a common workaround for low-tech mindmaps, as it allows adjustable spacing without having to redraw an entire mindmap. The current consumer market has a number of technologies that support digital mindmapping. Figure 3 below provides a summary of these technologies, characterizing their capabilities and availability to the public in comparison to traditional, non-digital methods for mindmapping.

Some digital mindmapping tools allow multiple users to edit the mindmap simultaneously or synchronously. However, most of these tools do not fully support real-time synchronous editing, have delays or are not able to detect conflicting input between users. There has been a development of a few mindmapping tools that incorporate a semantic network database. ThinkMap has a synonym suggestion feature that is offered by VisualThesaurus [18], and a mindmap algorithm developed by Chen et al. incorporates ConceptNet [19] so that the algorithm serves as a computer collaborator to create a mindmap with a human user [20]. Most digital mindmapping tools that support synchronous multi-user editing, with the exception of SharedMind [21], are not free nor open source. Accessibility and cost of design tools is an important factor in democratizing design. This may be of interest as we move toward more open access journals, open sharing of data sets and code, and even open sharing of tools and software.

Lin and Faste identify the speed of workflow and efficiency as a primary advantage of digital mindmapping over hand drawn maps [8], as digital mindmapping utilizes a keyboard and mouse to input and organize data. Most digital mindmapping technology allows users to use shortcut keys, colors, font size, and font style in order to design themes based on their needs. Shih et al. recognize that electronic brainstorming systems, such as digital mindmapping software, tend to benefit groups, eliminating production blocking and evaluation apprehension, which enables the generation of more creative ideas; this is due to the ability to work individually, to input data in parallel, and to enter ideas anonymously [22]. Although current electronic brainstorming systems might facilitate the generation of more ideas, some studies show they take more time than traditional techniques; users remained more satisfied with a traditional, face-to-face brainstorming method over current electronic brainstorming options that were tested [23,24].

Based on the benchmarking of digital and non-digital mindmapping techniques shown in Fig. 3, there is a clear opportunity to enhance the functionality of an integrated mindmapping technology to better support collaborative cognition.

2.3 Automated Mindmapping Technology. Few attempts have been made to develop automated mindmapping technology by researchers. Elhoseiny and Elgammal developed a framework algorithm that takes plain text and generates a single or multi-layer mindmap by using semantic processing, concept ontology processing, natural language processing, and an image web searching algorithm, in order to present a summary and visualize the information of the text [7,11]. The study was limited to English text about historical figures with a maximum length of 250 words. Abdeen et al. present automatic mindmap generating software that uses a morphological analyzer, parser, syntax analyzer, semantic analyzer, and mindmap converter to convert English text into a single layer mindmap, which does not have any keyword clustering or categorization [25]. The limitations of this software are that it uses short text and can only generate a single layer mindmap with limited natural language processing technology. Neither approach processes text that is derived from the speech of multiple team members, which is a significantly different and difficult challenge.

Kudelic et al. developed an automatic mindmap generator that uses a text-mining algorithm to parse and extract various lengths and topics of English text sources [12,13]. The mindmapping software can generate a mindmap within seconds, regardless of length and source of the text, such as a webpage or .doc file. The software was tested for speed, accuracy of recognizing text topics, selection of the correct keyword for the node, and drawing relevant associations among nodes. Kudelic et al. note the software works satisfactorily but has inaccuracies in selecting the correct keyword for the node, connecting related nodes, and engaging with synonyms. Like other mindmapping technology, this algorithm did not process synchronous, multi-person speech data. Beel and Langer present potential opportunities and possible applications of mindmaps for use in expert search methods, summarizing documents, keyword-based search engines, and recommending system mechanisms [15].

To automatically generate a record (in this case, a mindmap) of a conversation, meeting, or team brainstorming session, a number of technologies must be employed. First, a reliable Speech-To-Text (STT) transcription is needed, followed by text processing to extract meaningful keywords and associations among keywords from the text. Finally, a visualization technique is needed to create a meaningful structure showing the relationships among the textual content. Technologies associated with each of these components are discussed next.

Speech-to-Text. Hartman discusses several apps on smartphones or tablets that can transcribe voice data into text without the need to create an audio file, such as Dragon, QuickVoice2Text, and Speech Notes. He suggests that these STT apps could largely improve the productivity of occupational therapy practice [26]. Google Cloud, Apple Siri, Microsoft Azure, and Amazon AWS are all commercial systems that interpret voice data for processing into actions or transcription into text. Speech recognition technology is on the rise and continuing its refinement. In this study, most popular STT technologies, such as Google Cloud, Microsoft Azure, and Amazon AWS, will be compared.

Keyword Extraction. Keyword extraction techniques automatically identify a word or a phrase that sufficiently summarizes or describes the subject of a given sentence or document. Keyword extraction and document summarization techniques are used frequently, with various approaches and methods as shown in Fig. 4 [27,28]. This research will be only focusing on unsupervised keyword extraction since the mindmapping tool will be developed as an unsupervised automated tool.

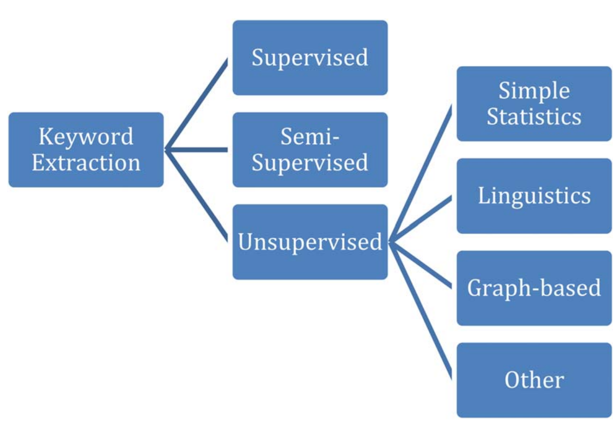

Fig. 4 Various keyword extraction approaches

N-gram statistics, word frequency, term frequency-inverse document frequency (TF-IDF) model, and PAT Tree are examples of *Simple Statistic Approaches*. This is one of the simplest approaches, as it does not require any training data. However, since it extracts a keyword based on the frequency of the word, it tends to filter out the actual keyword when the document is a professional text, such as medical documents or technology-focused academic journals, because the word appears only once or twice in the text [28,29]. One of the most common keyword extraction approaches is a *Linguistic Approach*. This approach analyzes linguistic properties or features of the words, sentences, and paragraphs in order to extract relations and determine the keyword. As an example, Cheong et al. used Subject–Verb–Object (SVO) triplet syntactic rule to acquire a functional keyword from a plain text [30]. In spite of its frequent use, this approach is very language dependent, and targeted text needs to be carefully grammatically oriented.

Since, a mindmap is ultimately a graph that is structured with keywords and branches, where keywords represent nodes (also called vertices) and branches represent edges or arcs that connect nodes in basic graph theory, it is more efficient to utilize applications and analyses that are graph network based. As the graph network analysis has been used in a variety of fields of study, it has also been increasingly used for modeling and analyzing natural language [31–36]. Currently, *Graph-based* keyword extraction approaches are being explored among a number of researchers. The graph-based data structure has the ability to follow semantic and lexical memory representations closely through an encoding of the natural meaning and cohesive structure of the corpus [28,29,36–40]. A semantic network graph is structured with nodes and edges, which represent words and associations among words accordingly. Relations among words can be established by co-occurrence, syntax, semantic similarity, and others [32]. Therefore, the graph-based keyword extraction method can effectively analyze and organize words, relationships, and structural information of a given document mathematically.

Keyword to Structure. Computational text analysis tools, also called natural language processing (NLP), can compute the similarity between words, a quantification of semantic similarity that can be used to associate words into categories. Graph-based data mining approaches can be used for categorizing keywords. The advantage of using a graph-based approach is that it will not only provide keywords but also create a graphical structure that is used to analyze keywords. Graph-based data mining can model relationships and structural information effectively providing computation related to term weight and ranking. This allows for better information retrieval. For example, semantic network analysis is used to visualize a patent database [34,35], to evaluate and represent the interconnectivity of design ideas [41], and to explore and exploit existing ideas to generate new ideas [42] by analyzing a graph-based data structures that represent semantic relations between words, concepts, and documents, using directed or undirected graphs. Graphical representation of the data structure, the mindmap, is built on the graphical structure that the graph-based method provides. With the given structure and keywords, more unrecognized connections between keywords will be explored through co-occurrence analysis, keyword similarity comparison, and semantic or lexical network analysis (WordNet and ConceptNet) [20,31], and word vector representations analysis of Wikipedia, Google news, or journal databases (Word2Vec and FastText) [20,30,40]. These methods can also be used to suggest new keywords or connections among existing keywords and networks.

3 Methodology

3.1 Unsupervised Automated Speech2Mindmap Algorithm Framework. The objective of this research is to develop a method to visualize ideas from a recorded speech data in a meaningful way, to reduce designers' cognitive load, and to enable more innovative

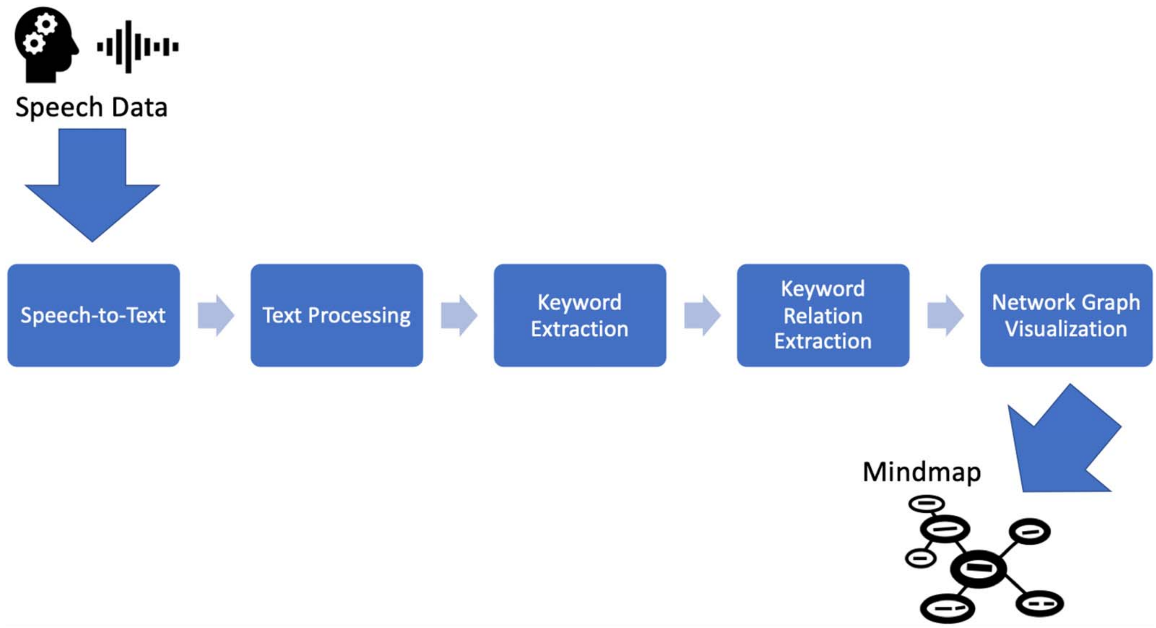

Fig. 5 Speech2Mindmap Algorithm data flow diagram

inspiration. Generating a mindmap automatically, unsupervised, using recorded speech data requires several stages of data processing. Figure 5 presents the general outline of a data flow for the Speech2Mindmap algorithm. It consists of the following five data processing stages: (1) Speech-To-Text (STT), (2) text processing, (3) keyword extraction, (4) keyword relations extraction, and (5) network graph visualization.

Data processing starts with converting recorded speech data into text. Converting speech data into text is one of the most important processes since it is the basis for all other subsequent processes. Speech recognition technology is continuing to be refined and improved, but it has limitations currently. Since improving or developing speech recognition technology is not the objective of this study, several STT products that are available today are tested and compared.

There are several off-the-shelf STT products, including Google Cloud, Microsoft Azure, Amazon AWS, IBM Watson, Dragon Speech Recognition, and others. Among these various STT products, the products that provide real-time STT features, available in a PYTHON environment, and have a large speech database with a good reputation were selected, which were Google Cloud, Microsoft Azure, and Amazon AWS, along with human transcription. A quantitative analysis was done to determine the best product to use in this study.

Processing speech-based text has significantly different challenges from processing written text. Due to the initial data input being conversational speech data, it may contain grammar errors, unnecessary words, or unnecessary phrases. Therefore, the transcribed text goes through a basic text processing stage before it goes through the keyword extraction process. The basic text processing lemmatizes the text and removes all the stopwords and onomatopoeias, which helps to improve the extraction of more important keywords from given input data.

A specific class of method called TextRank is used to extract keywords and translate them into a mindmap. TextRank is an unsupervised graph-based ranking algorithm that was first introduced by Mihalcea and Tarau in 2004 [43]. A system like TextRank solves the problem of terminology extraction and construction of domain-specific dictionaries. TextRank is used to extract keywords from each sentence, along with the relationship information of those keywords in order to translate that information into a mindmap.

TextRank keyword extraction automatically identifies a set of terms that best describe the document based on the words' lexical and syntactic features and co-occurrence relation, which is controlled by the distance between word occurrences: two vertices are connected if their corresponding lexical units co-occur within a window of maximum N words, where N can be set anywhere from 2 to 10 words. TextRank also uses a weighted ranking system that is used to determine the most important word based on the rank of the word in the corpus. The edges are weighted to emphasize the importance and the strength of the relationship between two words. The strength $WS(N_i)$ between two words, N_i and N_j, and weight w_{ij} are added to the corresponding edge with k iteration is defined as follows [37]:

$$WS(N_i) = (1-d) + d * \sum_{N_j \epsilon \mathrm{In}(N_i)} \frac{w_{ij} * WS(N_j)}{\sum_{N_k \epsilon \mathrm{Out}(N_j)} w_{jk}} \qquad (1)$$

where d is a damping factor (usually set to 0.85, but able to be set between 0 and 1) [37,44]. The weights of the relationship can be defined by measurements of co-occurrence, cosine distance, query-sensitive similarity, and other methods. One of the advantages of TextRank is the freedom in determining how weights are going to be measured, as it is possible to define weights with combinations of relationships. Another advantage is that TextRank is not language dependent. It does not require deep linguistic knowledge; therefore, it does not require pre-training to analyze a particular natural language.

Based on extracted relations from TextRank, other methods to define a relationship (edge) between two words, such as word similarity value using WordNet and word vectorization using FastText, are explored to optimize the relationship between words and to find more meaningful and unrecognized relationships. Once these keywords and edges have successfully and reliably been extracted, visualization is achieved by employing an open source visualization tool called NetworkX.

3.2 Evaluation Method. To test the reliability and the accuracy of the mindmap created by the Speech2Mindmap algorithm, two different case studies were performed with Georgia Institute of Technology's engineering or design-related major students. The purpose of these case studies was to investigate how similar an unsupervised computer-generated mindmap can be to a human-generated mindmap, as well as testing the reliability of the

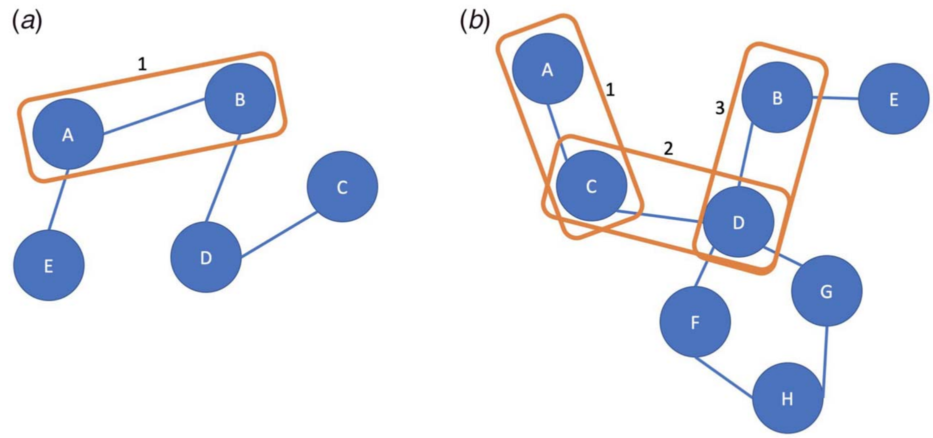

Fig. 6 Edge comparison with edge window size 3: (*a*) human-generated mindmap and (*b*) computer-generated mindmap

algorithm in generating a mindmap that is efficient and similar enough to stand in for a human mindmap generator.

3.3 Case Study 1: Testing General Agreement Between Human- and Computer-Generated Mindmaps

3.3.1 Participants. The Speech2Mindmap algorithm was evaluated on how well it represents a general thinking process of the human brain while brainstorming and drawing a mindmap. In this case study, a pre-recorded brainstorming session audio file was used to create five human-generated mindmaps and a computer-generated mindmap. The case study was conducted with five student participants who were attending Georgia Institute of Technology, Atlanta, GA, USA. Participants were composed of one female and four males who were all between the ages of 20 and 29 years, studying design or engineering, with two pursuing bachelor's degrees and three pursuing PhDs.

3.3.2 Design Problem. The design problem used for this brainstorming session was to "*find ways to reduce hospital days for patients with small bowel obstruction*."[1] This pre-recorded brainstorming session was selected because it had a clear design problem statement, a proper interaction between group members, and a mixture of general vocabulary and professional jargon.

3.3.3 Study Procedure. Five participants individually created their own mindmap while listening to the same brainstorming audio file. The computer-generated mindmap was created by using the Speech2Mindmap algorithm, based on the same audio file content. In order to test the reliability of the extraction of keywords and keyword relations, human transcribed texts were used to create the computer-generated mindmap, since the STT technology is not perfect. The mindmaps created by five participants and the algorithm are shown in the Appendix. The correlation between five human-generated mindmaps and a computer-generated mindmap was explored and analyzed. In order to test how well a computer-generated mindmap can represent a human brainstorming session, the computer-generated mindmap was evaluated to determine how many nodes and edges matched the human-generated mindmaps.

3.3.4 Node to Node Comparison Procedure. The Speech2-Mindmap algorithm was designed to discover nodes and edges that humans might not have thought of or recognized. In the direct node to node comparisons between the human and computer-generated mindmaps, we looked at how much the computer-generated mindmap included the nodes of the human-generated mindmap, comparing exact word to word. For example, if a computer-generated mindmap has all the nodes of a human-generated mindmap, then it is considered a 100 percent match, even if the computer-generated mindmap has extra nodes.

3.3.5 Edge to Edge Comparison Procedure. For the edge comparison in this study, the comparison was done between a single (topological, not geometric) length edge of a human-generated mindmap to an edge of a computer-generated mindmap with an edge window size. An edge window size is an accepted edge length that is considered as a match edge between two given nodes. As shown in Fig. 6, the edge window size between nodes A and B in the human-generated mindmap is 1, but in the computer-generated mindmap, it is 3. In this study, while comparing edges between the human-generated mindmaps and the computer-generated mindmap with exact word matching nodes, an edge of a computer-generated mindmap with a window size less than or equal to 3 is accepted as a match. Therefore, even if both the human-generated and the computer-generated mindmaps have nodes A, B, and E, edges between nodes A and B were considered as a match. However, the edges between the nodes A and E are not considered a match, since the topological edge length between nodes A and E in computer-generated mindmap is 4, compared to a topological length of 1 in the human-generated mindmap.

To test the general agreement between the computer-generated mindmap and the human-generated mindmaps, correlations between the degree of agreement among the human-generated mindmaps and the existence of the node or edge in computer-generated mindmap were explored.

3.4 Case Study 2: Testing the Automated Mindmap Algorithm's Reliability

3.4.1 Participants. For the second case study, the reliability of the Speech2Mindmap algorithm was tested. A total of 13 groups with three people in each group and a total of 39 students participated. The participants were majoring in design or engineering-related major at Georgia Institute of Technology, Atlanta, GA. Participants were composed of 18 men and 21 women, with 24 in Bachelors, five in Masters, and ten in PhD degree programs. Twelve participants were between the ages of 18 and 19 years, thirteen participants were between the ages of 20 and 22 years, nine participants were between the ages of 23 and 25 years, and five participants were between the ages of 26 and 29 years. Participants were recruited on a voluntary basis and received monetary compensation of $10 dollars for their involvement in this research.

[1] https://www.youtube.com/watch?v=xhsmihuESKY&ab_channel=StanfordBiodesign

3.4.2 Study Procedure and Design Problem. Participants were randomly assigned to groups of three to perform a 15-min brainstorming session using a given design problem. The given design problem was to "*Design a system to detect a golf ball hit from the tee box*" [9,10]. Each group was asked to create their own mindmap with as many ideas as they could think of in the given time. While creating a mindmap, participants were asked to use only the words they spoke and to use one word per node in the mindmap. They were given a whiteboard wall, a stack of post-it notes, and whiteboard markers, to use in generated in the mindmap. The brainstorming sessions were videotaped and audio recorded for further analysis. The final mindmaps generated by the teams were photographed. Audio data of the sessions were post-processed through the Speech2Mindmap algorithm to generate unsupervised computer-generated mindmaps. To measure the similarity and the reliability of the computer-generated mindmap, a number of metrics were employed. The mindmaps that were generated by each group were analyzed and compared to the computer-generated mindmaps to examine how many keywords and associations they had in common. Participants also filled out a feedback survey about their basic demographics, their prior experience in design, and their experience of the brainstorming session for this research study.

3.4.3 Granular Similarity Metric. In both case studies, in order to evaluate the similarity between a computer-generated mindmap and a human-generated mindmap, a direct edge metric, called the Granular Similarity, introduced by Jamieson and coauthors [45,46], was used. According to Jamieson et al., the Granular Similarity metric was able to identify the improvement of mindmap similarity between two mindmap generators. The Granular Similarity value (Eq. (2)) is a similarity value that is calculated by comparing directly edge to edge and node to node between two mindmaps [45,46]

$$\text{Granular Similarity} = \frac{\text{MatchE}}{\text{MatchN} + \text{MatchE} + \text{MissN}} \tag{2}$$

The Granular Similarity value is calculated by statistical records of matching nodes (MatchN), matching edges (MatchE), missing nodes (MissN), and missing edges (MissE). The Granular Similarity value, Eq. (2), was slightly modified to include all matching and missing edges and nodes to calculate the similarity value that fit the needs of this study, as shown in Eq. (3)

$$\text{Granular Similarity} = \frac{\text{MatchN} + \text{MatchE}}{\text{MatchN} + \text{MatchE} + \text{MissN} + \text{MissE}} \tag{3}$$

The Granular Similarity value was calculated for each human group generated mindmap, comparing to the corresponding computer-generated mindmap. Also, average Granular Similarity values among five mindmaps in the first case study and 13 groups in the second case study were calculated to determine the general agreement, performance, and reliability of the Speech2Mindmap algorithm.

3.4.4 Comparison of Speech-To-Text Technologies. In order to improve the average Granular Similarity value between a human-generated mindmap and a computer-generated mindmap, different variations of variables in Speech2Mindmap algorithm were explored. First, different STT technologies were compared. As mentioned earlier, the accuracy of the STT technology affects the result of the computer-generated mindmap. The text that was transcribed by each STT technology is the basis for the Speech2Mindmap algorithm process. Therefore, the three most popular STT technologies, Google Cloud, Microsoft Azure, and Amazon AWS, were compared, along with human transcription.

3.4.5 Comparison of Keyword Extraction Damping Factor Values. Another way to improve the average Granular Similarity value between a human-generated mindmap and a computer-generated mindmap is to improve the number of matching nodes. In order to improve the number of matching nodes, the method of extracting keywords has to improve in a way that mimics manually generated human mindmapping output. As explained previously, the Speech2Mindmap algorithm uses TextRank to extract keywords. With the same given nodes and weights, as shown in Eq. (1), the damping factor d is a variable that affects the result of the keyword extraction. The damping factor is usually set to 0.85, but it can be set anywhere between 0.0 and 1.0 [37]. Therefore, we tested how the damping factor of the TextRank affects the average Granular Similarity value. The damping factor values that were examined were 0.80, 0.9, and 1.0.

3.4.6 Comparison of Keyword Relation Extraction Techniques. Similar to increasing the number of matching nodes, increasing the number of matching edges improves the average Granular Similarity value as well. The edge formation starts from the backbone of the TextRank, a graph that is created based on word co-occurrence. Besides the co-occurrence relation extraction method, different relation extraction methods that will be combined were explored to improve the overall relation extraction method in the Speech2Mindmap algorithm. In this study, two additional extraction methods, FastText and WordNet similarity value, were explored with four different combinations as follows:

(1) Co-occurrence;
(2) Co-occurrence + WordNet;
(3) Co-occurrence + FastText; and
(4) Co-occurrence + WordNet + FastText.

4 Results and Discussion

After analyzing the mindmaps that participants created, we have noticed that participants did not use the tree-like hierarchical structure but used many interconnected edges between nodes and the same words in different areas of the mindmap. Therefore, the mindmaps structures did not necessarily follow a tree-like hierarchical structure, but rather had a hybrid of a tree-like structure and a web-like structure, as shown in Fig. 7, which is very similar to the computer-generated mindmap, as shown in Fig. 8.

As a first step to reduce the designer's cognitive load induced by the multitasking required to listen, document, and generate ideas at once, and to create more opportunities for more creative ideas, the Speech2Mindmap algorithm was designed. A mindmap created by Speech2Mindmap is shown in Fig. 9, as an example. Speech2Mindmap algorithm is composed of five steps of data processing: (1) Speech-To-Text (STT), (2) text processing, (3) keyword extraction, (4) keyword relation extraction, and (5) network graph visualization. The collected data for two studies were sufficient to explore a number of ways to examine and analyze the outcomes. All the data sets that were performed One-Way ANOVA significant analysis did not violate any assumptions including the normal distribution and the homogeneity of variance. The normality test and Levene's test were performed on all data sets. The main purpose of these two case studies was to examine how well Speech2Mindmap represents a human mind reliably while brainstorming only based on speech input.

The general agreement between human-generated mindmap and computer-generated mindmap was examined. Different system variable variations and methods were explored in the three main data processing steps: (1) Speech-To-Text (STT), (2) keyword extraction, and (3) keyword relation extraction, to improve the similarity between the human- and computer-generated mindmaps.

4.1 Case Study 1: General Agreement Between Human- and Computer-Generated Mindmap. In this case study, five human-generated mindmaps were compared to a mindmap that was produced by the Speech2Minmap algorithm, in order to test the general agreement between human and computer-generated mindmaps, all generated based on the same pre-recorded brainstorming session audio file.

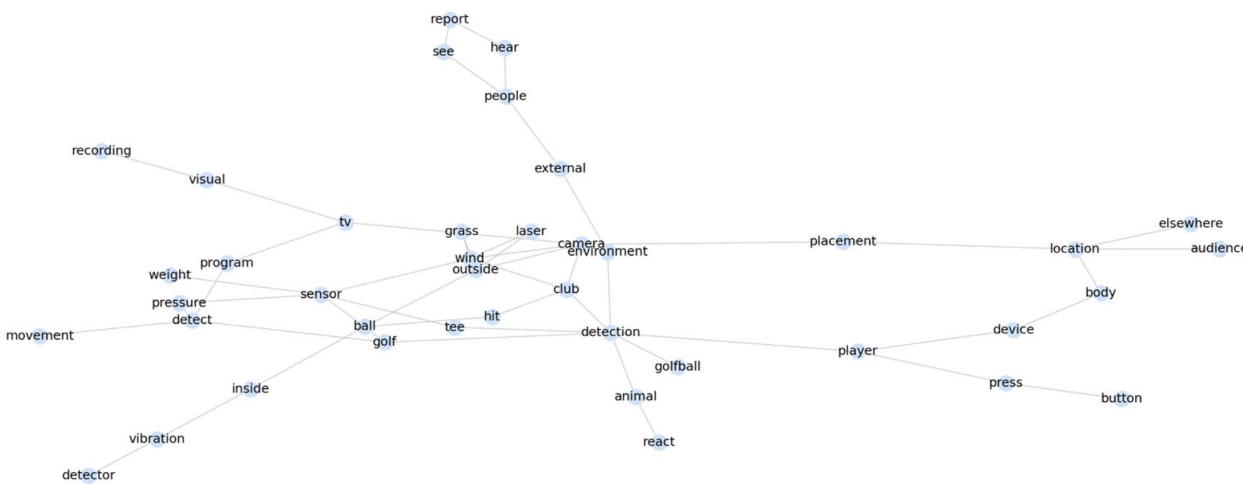

Fig. 7 An example mindmap created by human participants

4.1.1 Node to Node and Edge to Edge Comparison Results. The general agreement between the human-generated mindmaps and the computer-generated mindmap was examined by analyzing each node and each edge directly. Combining all five human-generated mindmaps, there were a total of 65 nodes and 127 edges, where the computer-generated mindmap contained a total of 51 nodes and 154 edges. The computer-generated mindmap included 36 nodes out of 65 human-generated nodes, and 46 edges out of 127 human-generated edges, resulting in a 55.38% and 36.22% match accordingly. The match percentage seems very low when it is directly compared node to node and edge to edge. However, since the algorithm is not designed to represent all human minds but to represent the general agreement among human minds, analyzing the degree of agreement among human mindmaps is important. Therefore, the nodes and the edges that were created by participants were categorized by the degree of agreement among human-generated mindmaps, and the computer-generated mindmap was then examined to determine whether it included each category.

As shown in Fig. 10, as the degree of agreement among humans increased, the percentage of nodes or edges included in the computer-generated mindmap increased. To formally test this correlation, a linear regression analysis was performed, while all the assumptions for this analysis were not violated. Two variables, the degree of agreement among human mindmaps and the accordance with computer-generated mindmap are continuous variables with a linear relationship and with no significant outliers. Also, this data set has independence of observations and shows homoscedasticity, according to Durbin–Watson statistic value being equal to 1.5, which is within the normal range from 1.5 to 2.5, and having a residual value of 0, respectively. Lastly, looking at the normal P–P plot, the residual of the regression line indicated an approximately normal distribution, as seen in Fig. 11.

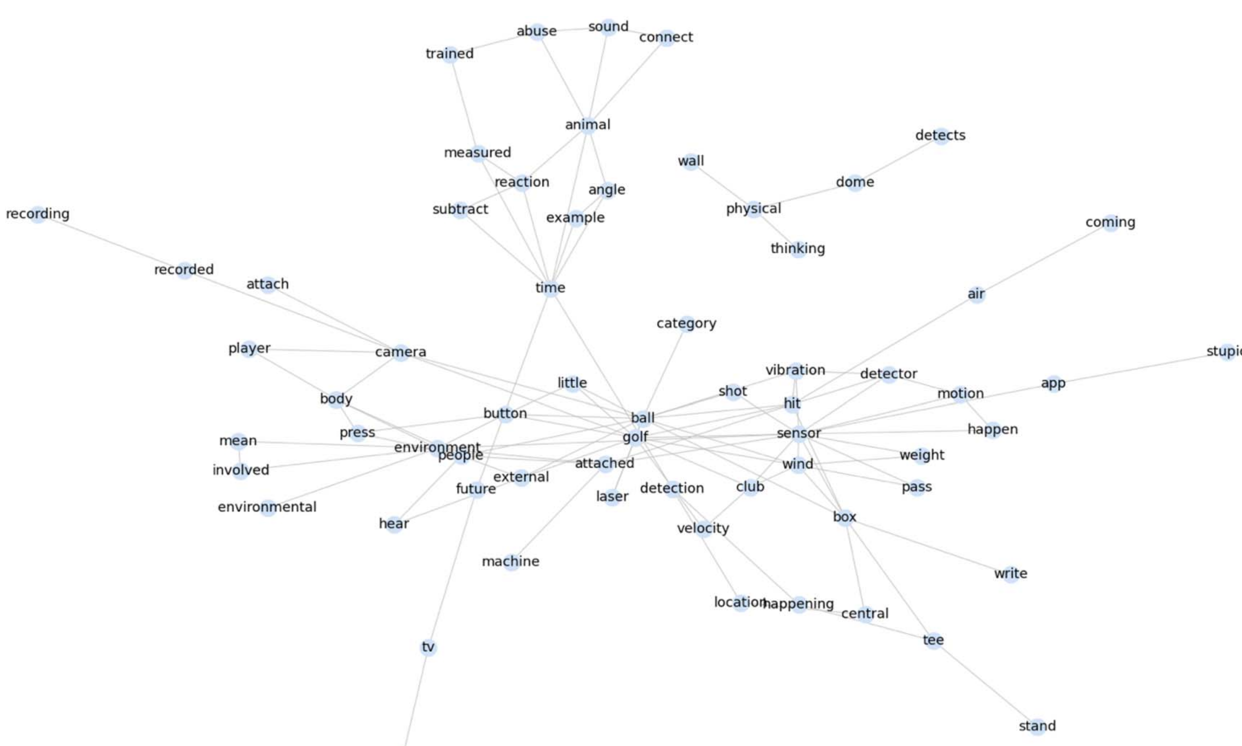

Fig. 8 An example mindmap created by Speech2Mindmap

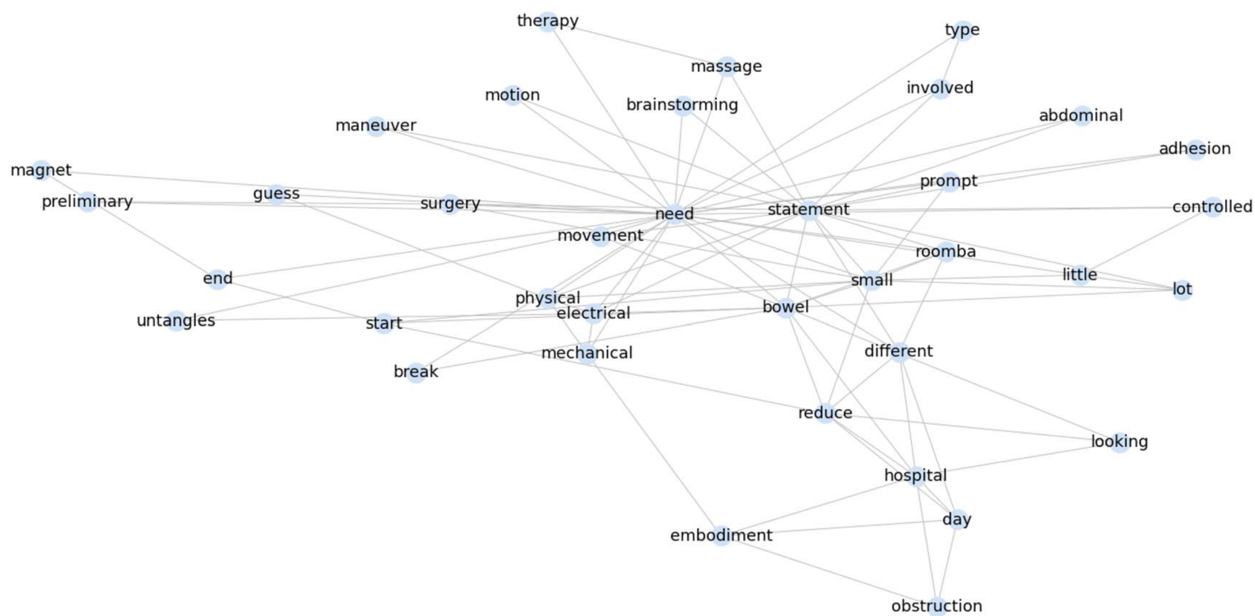

Fig. 9 An example computer-generated mindmap created by the Speech2Mindmap algorithm

According to the linear regression analysis summarized in Table 1, the linear regression model showed a high degree of correlation with the R value of 0.885, and the 78.4% of variations of the dependent variable of this model, percentage of accordance with computer-generated mindmap, can be explained by the independent variable, degree of agreement among human mindmaps. Also, according to the ANOVA test, the model is statistically significant, $p < 0.001$. In this particular comparison, the nodes that more than four participants had were all included in the computer-generated mindmap. Similar to the nodes, the edges that were in all five participants' mindmaps were all included in the computer-generated mindmap. This analysis explains that as the degree of agreement among human-generated mindmaps increased, the chance of computer-generated mindmap agreeing increased as well. As a result, the Speech2Mindmap algorithm is reliable in representing manually generated human mindmapping output.

4.1.2 Granular Similarity Results. Next, tests to examine what factors improve the similarity between human and computer-generated mindmaps were conducted to find the best conditions

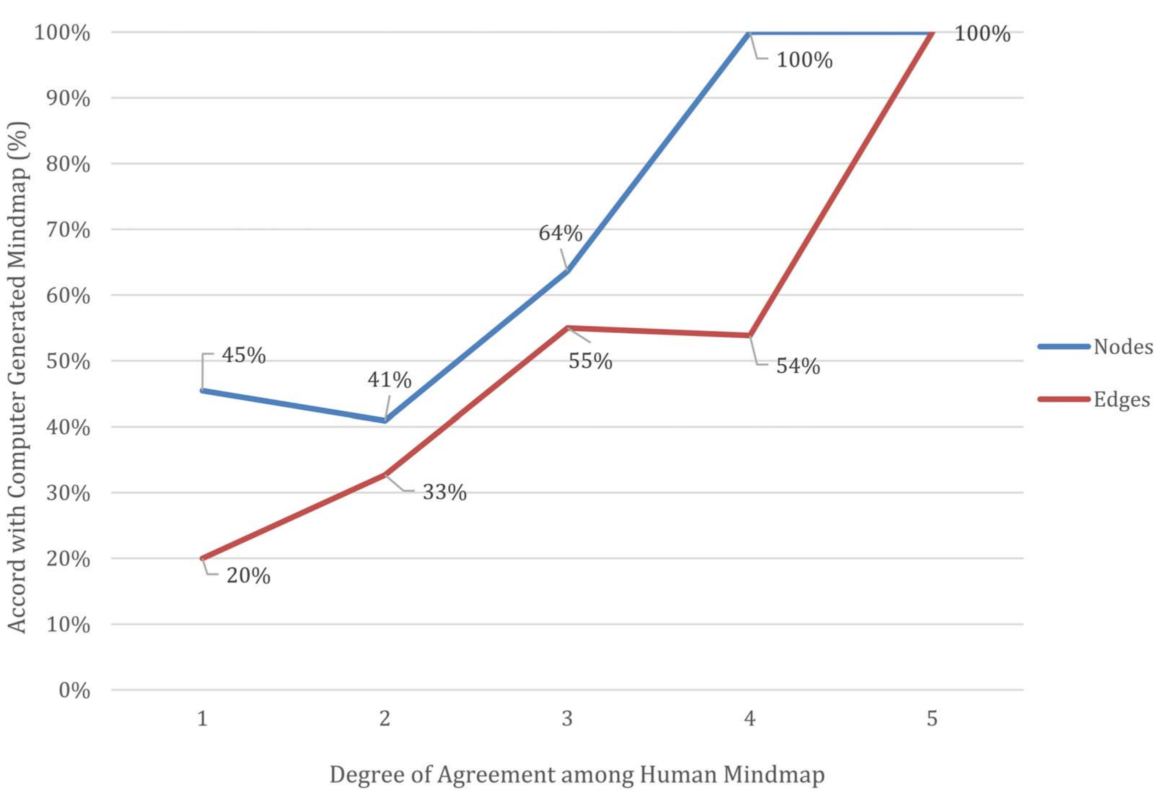

Fig. 10 Match percentage between human and computer-generated mindmaps depending on the degree of agreement among five human-generated mindmaps

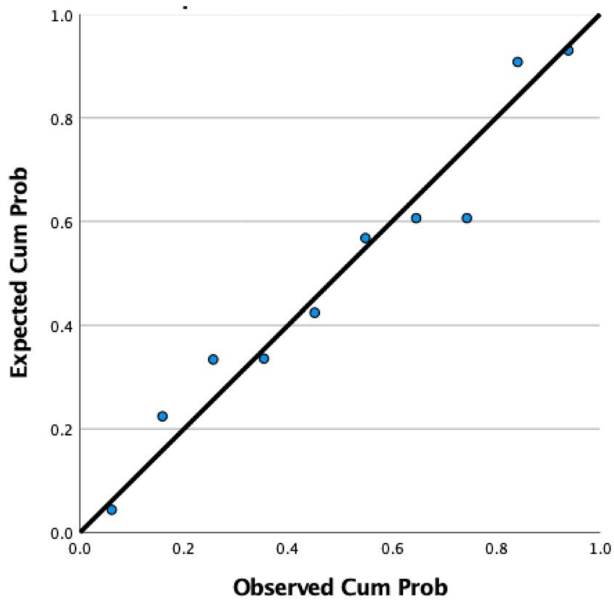

Fig. 11 Normal P–P plot of regression standardized residual

Table 1 Linear regression model summary of match percentage between human and computer

Model	R	R Square	Adjusted R square	Std. error of the estimate	Sig. F change	Durbin–Watson
Linear regression	0.885	0.784	0.757	0.145	<0.001	1.5

Table 2 Initial Granular Similarity value between human- and computer-generated mindmaps

Human mindmap	Granular Similarity value
Human mindmap 1	0.656
Human mindmap 2	0.667
Human mindmap 3	0.655
Human mindmap 4	0.500
Human mindmap 5	0.719
Average Granular Similarity value	0.639

for each data processing stage. The initial Granular Similarity value between each human-generated mindmap and the computer-generated mindmap is shown in Table 2. The average Granular Similarity value among the five comparisons was 0.639, with the lowest being 0.50 and the highest being 0.719.

The Granular Similarity values were tracked while examining different conditions for each data processing stage. While varying conditions for each data processing stage were applied, the conditions of other data processing stages remained constant. The STT data processing stages were examined first.

4.1.2.1 Comparison of Speech-To-Text technologies. In order to test which STT product performs better, four Granular Similarity values were obtained for each mindmap using different products, along with an average Granular Similarity value among five mindmaps for each product. As shown in Fig. 12, using Google Cloud and Microsoft Azure as the STT products resulted in the highest Granular Similarity value (0.663). However, there was no statistically significant difference among STT products as determined by one-way ANOVA ($F(3,20) = 1.506$, $p = 0.243$), as the data set had a normal distribution and homogeneity of variances.

4.1.2.2 Comparison of keyword extraction damping factor values. With improved text input, extracting the right keyword is the next important stage. As the TextRank Eq. (1) shows, the damping factor d is the only value that can vary beside the input variables, ranging from 0 to 1. Therefore, three different damping factor values, 0.80, 0.90, and 1.00, were explored, as shown in Fig. 13. Among all comparisons, the results showed a trend: as

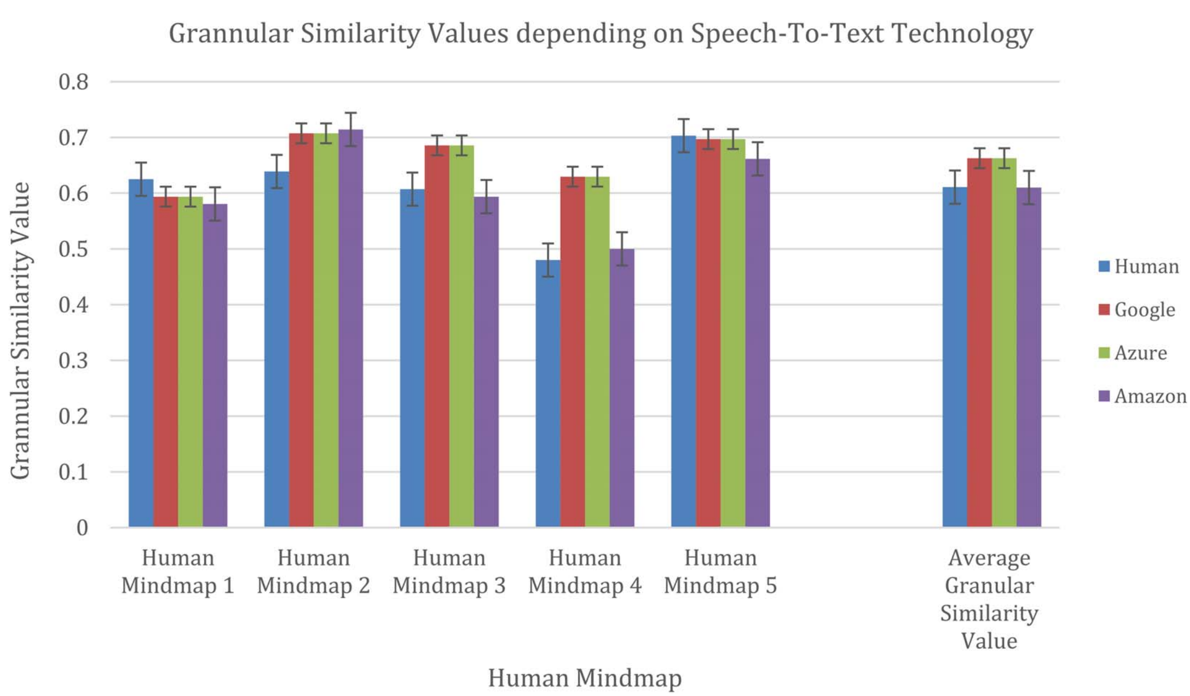

Fig. 12 Granular Similarity value comparison with different STT products, error bars show ± One Standard Error

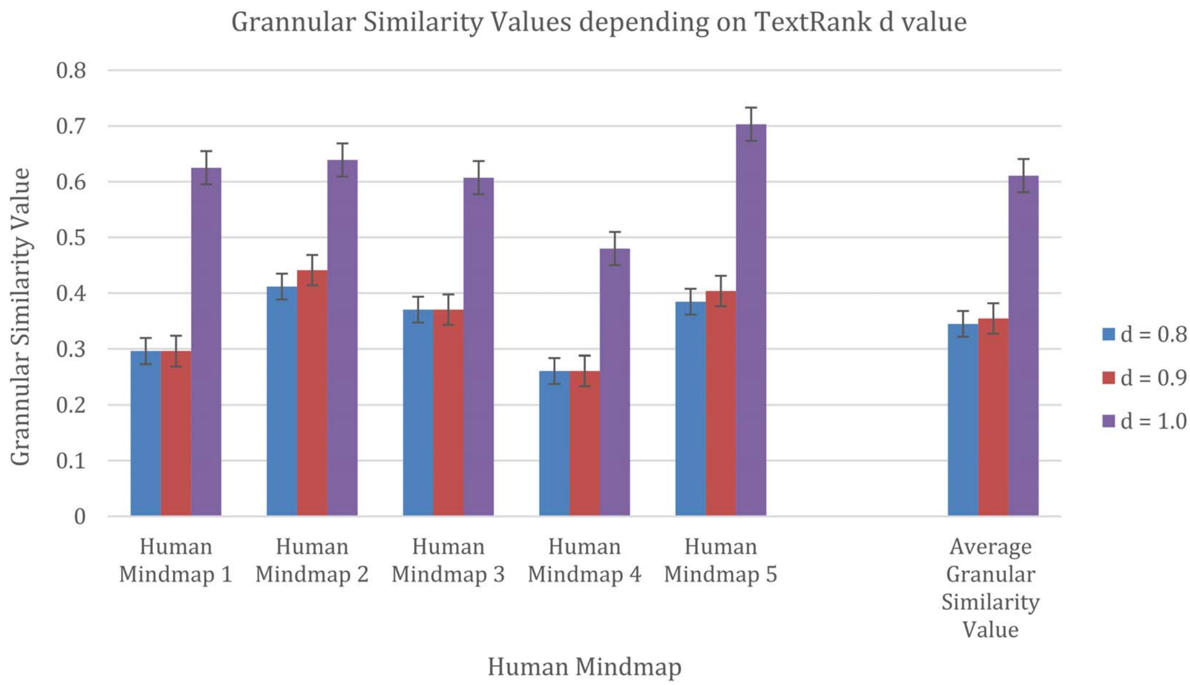

Fig. 13 Granular Similarity value comparison with TextRank Damping Factor variation, error bars show ± One Standard Error

the damping factor value *d* increased, the Granular Similarity value between human- and computer-generated mindmaps increased. There was a statistically significant difference as the damping factor increased, according to the one-way ANOVA ($F(2,15) = 31.46$, $p < 0.001$). A Tukey post hoc test revealed that the Granular Similarity value significantly increased when the damping factor was set to 1.0, (Granular Similarity Mean ± Std Dev.), (0.6108 ± 0.07299), compared to 0.80, (Granular Similarity Mean ± Std Dev., Sig.), (0.345 ± 0.057, $p < 0.001$), and 0.90, (0.355 ± 0. 0669, $p < 0.$ 001). However, there was no statistically significant difference between the damping factor being 0.80 and 0.90 ($p = 0.965$). In addition, TextRank extracted more keywords as the damping facto increased. Therefore, in this context, when the TextRank damping factor is set to 1.00, it performed the best.

4.1.2.3 Comparison of keyword relation extraction techniques. To examine the relation extraction method, four different combinations of keyword relation extraction methods were explored in this study. Those combinations are co-occurrence, co-occurrence + FastText, co-occurrence + WordNet, and

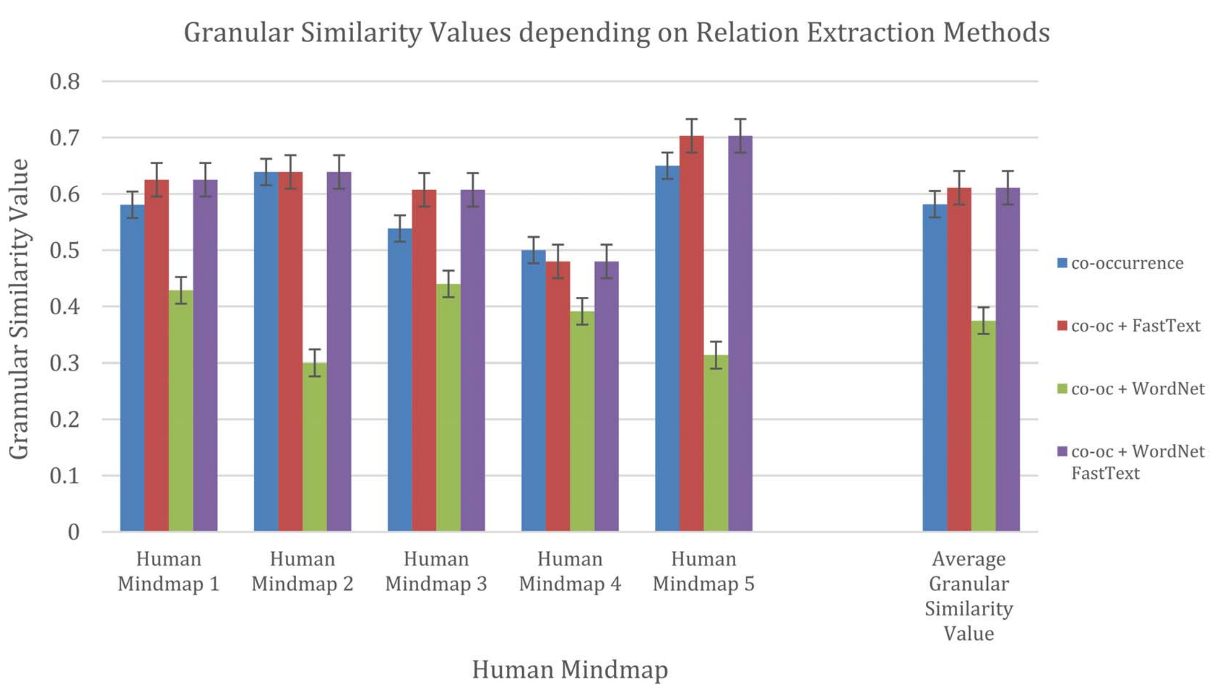

Fig. 14 Granular Similarity value comparison with keyword relation extraction method combination variations, error bars show ± One Standard Error

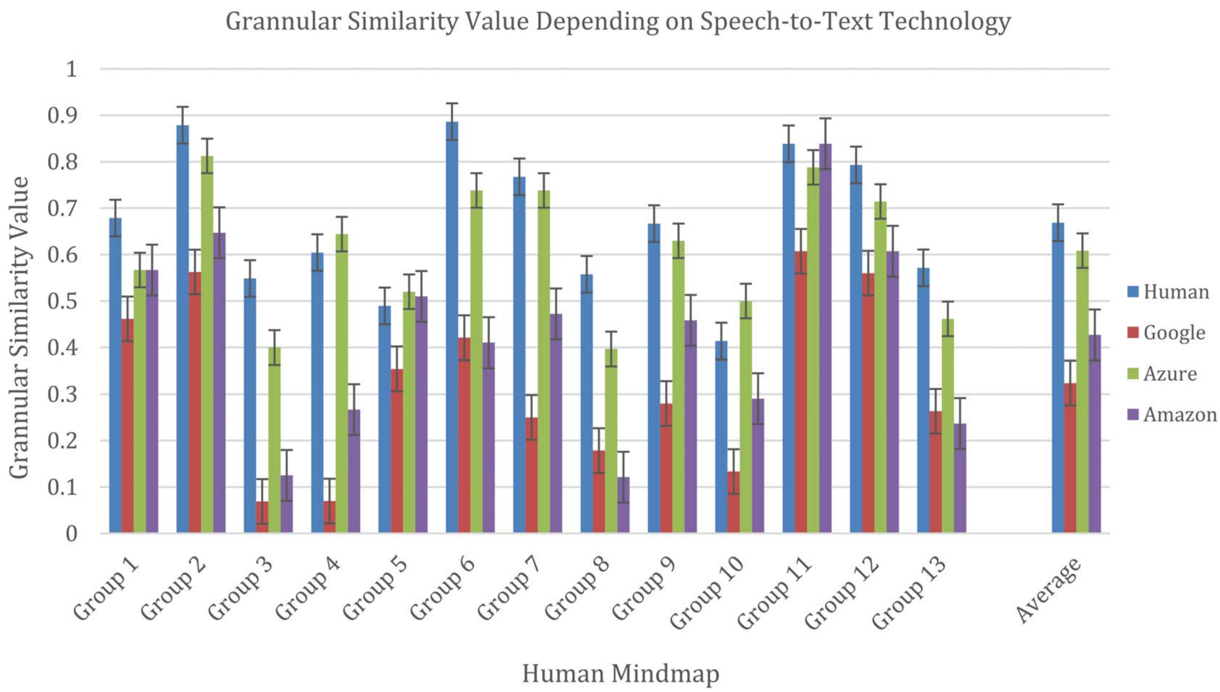

Fig. 15 Granular Similarity value comparison with different STT Products, error bars show ± One Standard Error

co-occurrence + FastText + WordNet. The Granular Similarity values for each human-generated mindmap and the average Granular Similarity values were calculated for each extraction method. As shown in Fig. 14, using the co-occurrence + FastText method and the co-occurrence + FastText + WordNet method for the keyword relation extraction method had the highest average Granular Similarity values. According to a one-way ANOVA, there was a statistically significant difference ($F(2,15) = 24.91$, $p < 0.001$) between relation extraction methods. In the Tukey post hoc test, it was found that there were statistically significant differences between the co-occurrence + WordNet method and the other three methods, but no statistically significant differences between those three extraction methods. Since the co-occurrence + FastText method had the same Granular Similarity value but performed faster than the co-occurrence + FastText + WordNet method, in this context, the best performing combination of keyword relation extraction methods is the co-occurrence + FastText method.

After testing the performance of the Speech2Mindmap algorithm and exploring different variations within each data processing stage, the Speech2Mindmap algorithm was able to generate a mindmap that represents manually generated human mindmapping output. In the context of the first case study, the Speech2Mindmap algorithm performed best using the Google Cloud or Microsoft Azure STT product, setting the TextRank damping factor d equal to 1.0, and using the combination of co-occurrence and FastText as the keyword relation extraction method. To validate the algorithm further, a second case study was conducted to test the usability of the Speech2Mindmap algorithm in a real brainstorming session.

4.2 Case Study 2: Reliability and Consistency of the Speech2Mindmap Algorithm. In order to test if the Speech2-Mindmap algorithm can generate mindmaps that are similar enough to human-generated mindmaps consistently, 13 group brainstorming sessions were conducted, obtaining 13 human-generated mindmaps and 13 computer-generated mindmaps using the Speech2Mindmap algorithm. The Granular Similarity values between human and computer-generated mindmaps were obtained to compare different STT technologies, variations of TextRank damping factor d value, and keyword relation extraction method combinations, similar to the first case study.

4.2.1 Granular Similarity Results

4.2.1.1 Comparison of Speech-to-Text technologies. Similar to the first case study, three different STT off-the-shelf products, Google Cloud, Microsoft Azure, and Amazon AWS, were compared along with a human transcription. As shown in Fig. 15, mindmaps produced with human transcription had the highest average Granular Similarity value, and Microsoft Azure followed as the second best. Different from the previous case study, Google Cloud performed very poorly during this experiment. As determined by a one-way ANOVA, there was a statistically significant difference between different STT products ($F(3,52) = 12.357$, $p < 0.001$). According to the Tukey post hoc test, the Granular Similarity value significantly increased when using Azure compared to using Google, (Granular Similarity Mean ± Std Dev., Sig.), (0.324 ± 0.180, $p < 0.001$), or Amazon (0.427 ± 0.205, $p = 0.033$). Surprisingly, there was no significant difference between using Azure STT and using human transcription ($p = 0.784$).

4.2.1.2 Comparison of keyword extraction damping factor values. For the experiment to find the best performing TextRank damping factor d value, three variations, 0.8, 0.9, and 1.0 were examined. The Granular Similarity values were obtained for each damping factor, as shown in Fig. 16. The results indicate the same finding as that found in the first case study: when the TextRank damping factor is set to 1.0, the highest Granular Similarity values were achieved among all 13 data points. Similar to the first case study, there was a statistically significant difference as the damping factor increased, according to a one-way ANOVA ($F(2,39) = 4.955$, $p = 0.012$). A Tukey post hoc test revealed that the Granular Similarity value significantly increased when the damping factor was set to 1.0, (Granular Similarity Mean ± Std Dev.), (0.807 ± 0.089), compared to setting it to 0.80 (Granular Similarity Mean ± Std Dev., Sig.), (0.686 ± 0.125, $p = 0.019$), or to 0.90 (0.697 ± 0.119, $p = 0.035$). However, there was no statistically significant difference between the damping factor being set to 0.80 and 0.90 ($p = 0.966$).

4.2.1.3 Comparison of keyword relation extraction techniques. Lastly, four different keyword relation extraction method combinations were compared as follows: co-occurrence, co-occurrence + FastText, co-occurrence + WordNet, and co-occurrence + FastText

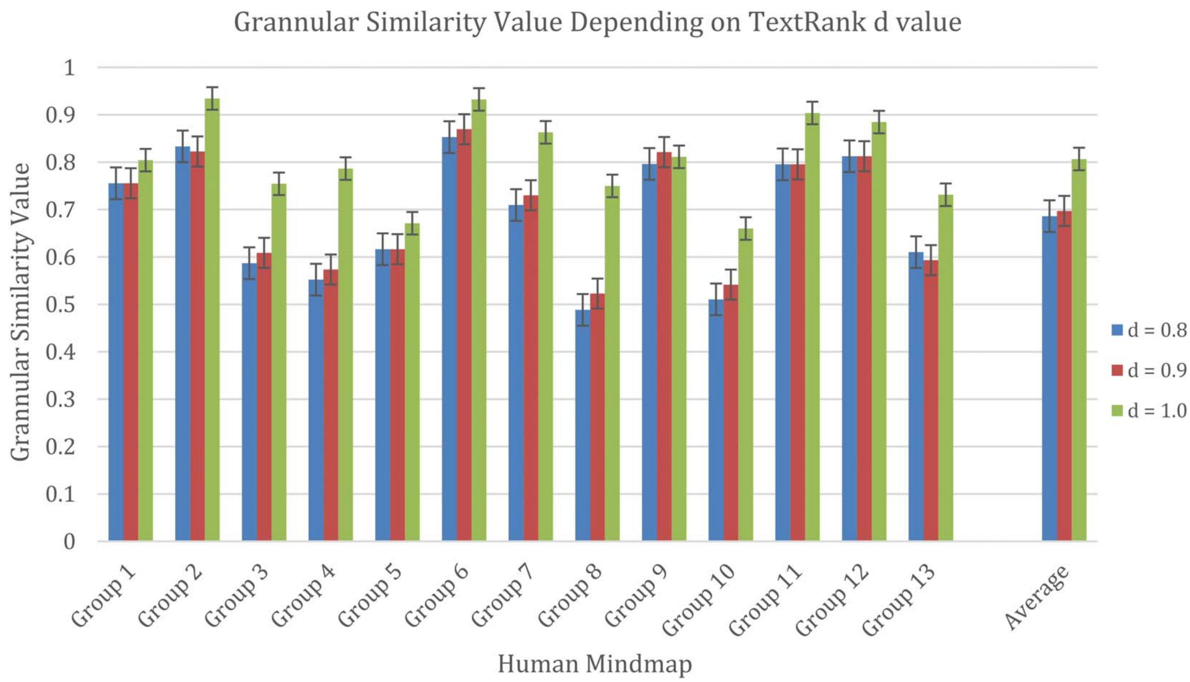

Fig. 16 Granular Similarity value comparison with different damping factor values, error bars show ± One Standard Error

+ WordNet. The Granular Similarity value of each group and the overall average Granular Similarity value across all groups were obtained for each extraction method combination. As shown in Fig. 17, the performance of the extraction methods was similar across all conditions. However, similar to the previous comparison, using the combination of the co-occurrence and FastText, and the combination of the co-occurrence, FastText, and WordNet performed better than the other two combinations. Also, there were no statistically significant differences among relation extraction methods as determined by one-way ANOVA ($p = 0.821$). Even though the Granular Similarity value between two combinations were the same, the combination of the co-occurrence and FastText methods performed faster. Since the Speech2Mindmap will be performing in real-time in the future, the processing time is very important. Therefore, in the context of this study, the combination of co-occurrence and FastText for the keyword relation extraction methods is the best performing method for the Speech2Mindmap algorithm.

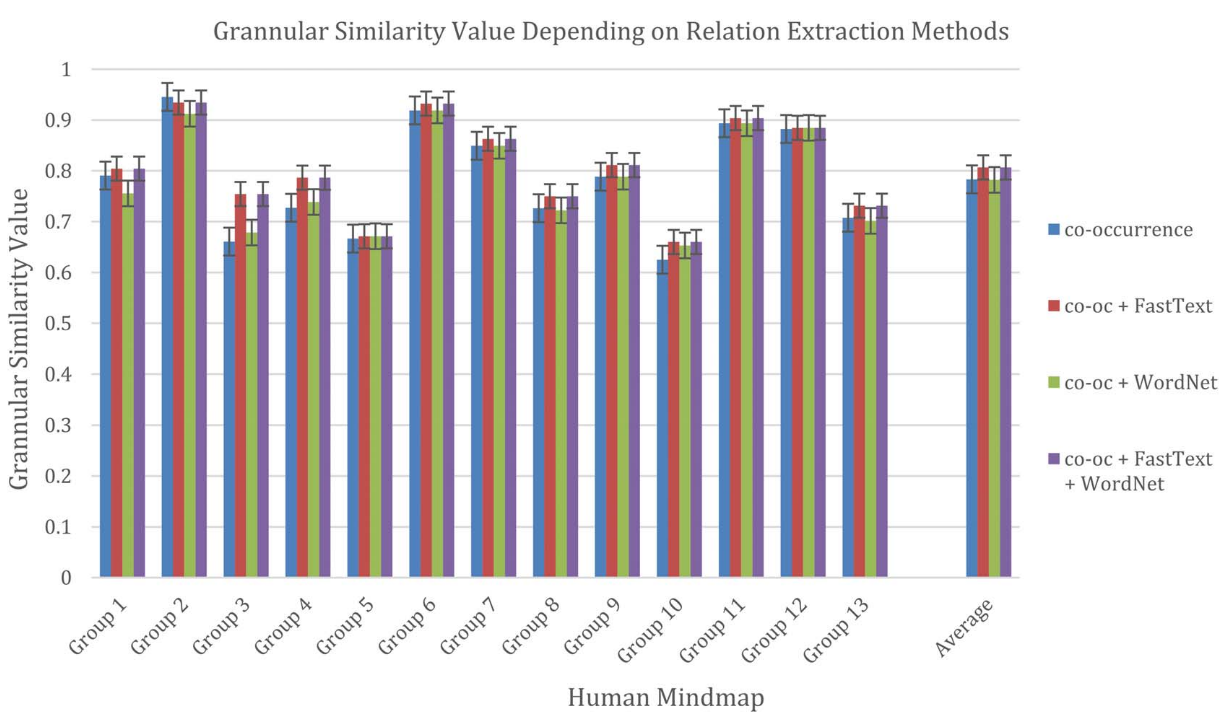

Fig. 17 Granular Similarity value comparison with keyword relation extraction method combination variations, error bars show ± One Standard Error

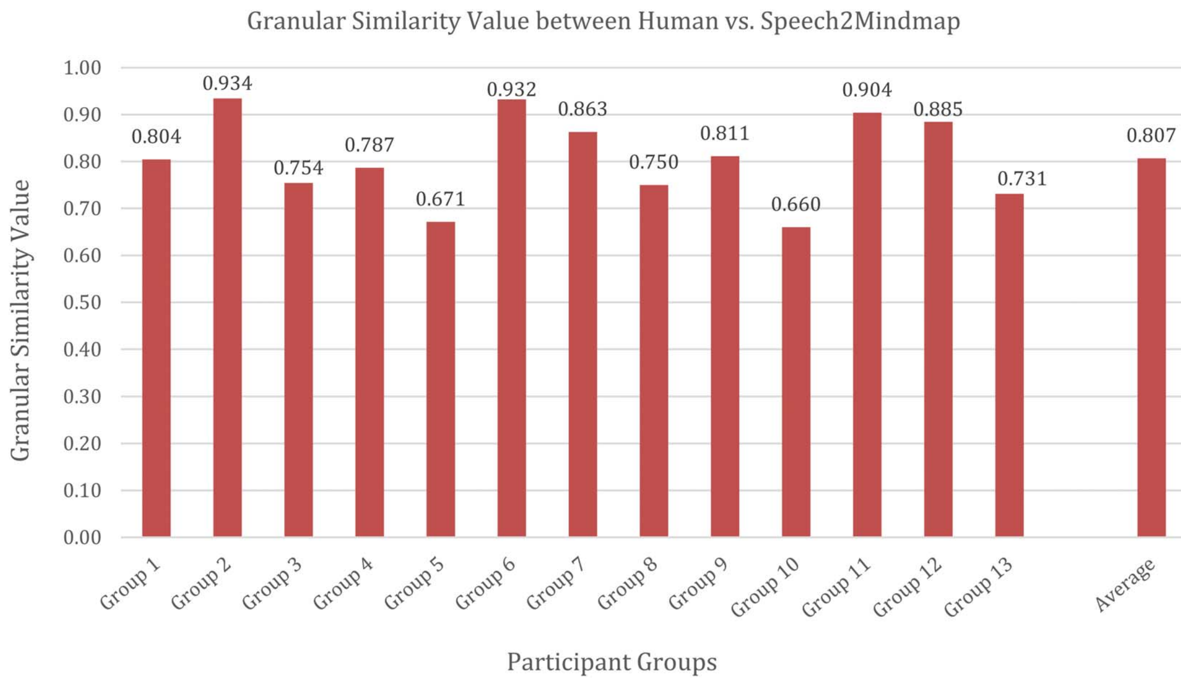

Fig. 18 Granular similarity value between human- and computer-generated mindmaps

5 Discussion

In this research, the Speech2Mindmap algorithm was developed to attempt to reduce designers' cognitive load during ideation and brainstorming sessions. The Speech2Mindmap algorithm was tested to see if it could reliably generate a mindmap that was similar to a human-generated mindmap. A few different conditions for each data processing stage were also explored to examine their impact on the similarity between human- and computer-generated mindmaps. The Speech2Mindmap algorithm is shown to be reliable to represent the manually generated human mindmapping output by comparing the degree of agreement among the five human-generated mindmaps and a computer-generated mindmap, based on the same audio file input source. With further analysis to improve the performance of the Speech2Mindmap algorithm, when Microsoft Azure is used for STT transcription, the TextRank damping factor d is set to 1.0, and the combination of co-occurrence and FastText methods is used for the keyword relations extraction method, the highest similarity was achieved between the human-generated and computer-generated mindmaps. When all the methods and conditions were set to their best performing option, as mentioned earlier, the average Granular Similarity value rose to 0.807, with the highest and lowest individual group value being 0.934 and 0.660, respectively, as shown in Fig. 18.

According to the results, the Speech2Mindmap algorithm is able to produce a mindmap that is approximately 80% similar to human-generated mindmaps on average, using speech data as input. Analyzing the recorded audio and video data of each group, the Granular Similarity value very much depended on the quality of the audio data. The group that had a higher Granular Similarity value had consistent audio clarity and relatively few instances of one person talking over another person. With better quality audio data input, it is expected that the similarity of the automatically generated mindmap to human-generated mindmap could improve further.

Ultimately, this work has contributed to the techniques available for computational design support in the area of automated design documentation. First, we have developed a Speech2Mindmap algorithm that will automatically generate a mindmap based on a speech data input, providing a new tool that could later support the generation of creative and innovative ideas while brainstorming. Generating a mindmap automatically, unsupervised, and from speech data is particularly important since it may provide opportunities to expand human creativity by allowing designers to save time and to lower their cognitive load. This might result in an increase in focus on ideation and brainstorming while providing all the benefits that traditional manual mindmapping provides.

Also, in an online environment and from a virtual collaboration perspective, this study could open up a new solution for online group brainstorming. Until now, there are limitations to using brainstorming tools in an online environment due to difficulties in interaction, participation, and visualization.

6 Limitations

There were some expected challenges and limitations in this research. Since the proposed work is very dependent on developments in STT technology, the accuracy of the STT systems is paramount and will directly determine the accuracy of the Speech2Mindmap algorithm. Further analyzing the recordings, because participants moved around while brainstorming, the audio of the participants faded in and out, causing the STT technology to misunderstand or omit some words. Adjustments and optimization procedures were conducted to achieve the best results. However, since the research is not about improving the accuracy of the STT technology, but rather about identifying ideas or keywords to expand creativity by using these technologies, the implementation focus was not on comparative performance.

When people explain personal opinions or ideas, they tend to use hand gestures or facial expressions to support the interpretation of their speech. Since the speech data were audio data recorded from group conversations, the data did not capture the use of non-verbal communication in conveying thoughts or ideas. Further analyzing the video recordings along with the audio showed that there were some cases where participants included keywords or associations without actually mentioning the word, which was not detectable by the Speech2Mindmap algorithm. As such, some implicit and subtext information was lost.

In addition, as most audio data in this research was conversational in nature, it contained some conversational speech that did not directly relate to the main subject or design problem, such as small talk. Some keywords that were not related to the topic were included in computer-generated mindmaps due to this extra dialogue. However, those keywords did not greatly impact the results of the Granular

Similarity value between the human-generated mindmaps and the computer-generated mindmap. When this additional small talk dialogue is longer than three minutes or is included in the middle of the brainstorming session, it might cause a significant impact on the computer-generated mindmap results. Recognizing and filtering out this non-topical information is another challenge in this research.

Finally, it is acknowledged that the traditional representation of mindmaps is a hierarchical structure. In this work, the mindmaps generated by both humans and Speech2Mindmap are not represented as hierarchical structures, but rather more interconnect graphs. In analyzing the human-generated mindmaps after translating them into a computer visualization, the research team found that human-created mindmaps did not necessarily form a hierarchical structure, as might be expected, but rather a more interconnected graph; this is because participants used the same words in different places in their mindmaps. As such, this constraint on the type of graph was relaxed in the algorithmic approach to allow more freedom for connections among nodes, with the idea that these may reveal an area that participants have not yet explored or provide opportunities to think outside of the box. The effect of this difference on design ideation outcomes will be tested in future work.

7 Conclusions

Since the framework of the Speech2Mindmap algorithm using the recorded audio files was introduced and analyzed, making the Speech2Mindmap algorithm a real-time responsive algorithm will be the next step. The usability and effectiveness of the Speech2Mindmap algorithm while brainstorming will be able to be evaluated when it is able to process live streams speech data and present a mindmap in real-time. Along with the development of a real-time Speech2-Mindmap algorithm, the ways to improve keyword extraction and keyword relations will continue to be explored.

This research has the potential to be highly impactful in design and virtual collaboration. With cognitive assistance to better visualize, connect, and uncover ideas, designers will be more efficient, effective, and innovative in their problem-solving and design processes, leading to better solutions developed in virtual environments. The geographic and temporal flexibility of virtual design teams and those who lack face-to-face interactions present challenges related to design team cognition and communication. The goal of this research is to pioneer advanced digital mindmapping technologies to overcome these limitations, which will empower skilled designers around the world. Furthermore, increasing the effectiveness of design team cognition and communication will improve team productivity, ultimately increasing business profitability and economic growth.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The data sets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request. The authors attest that all data for this study are included in the paper.

Appendix

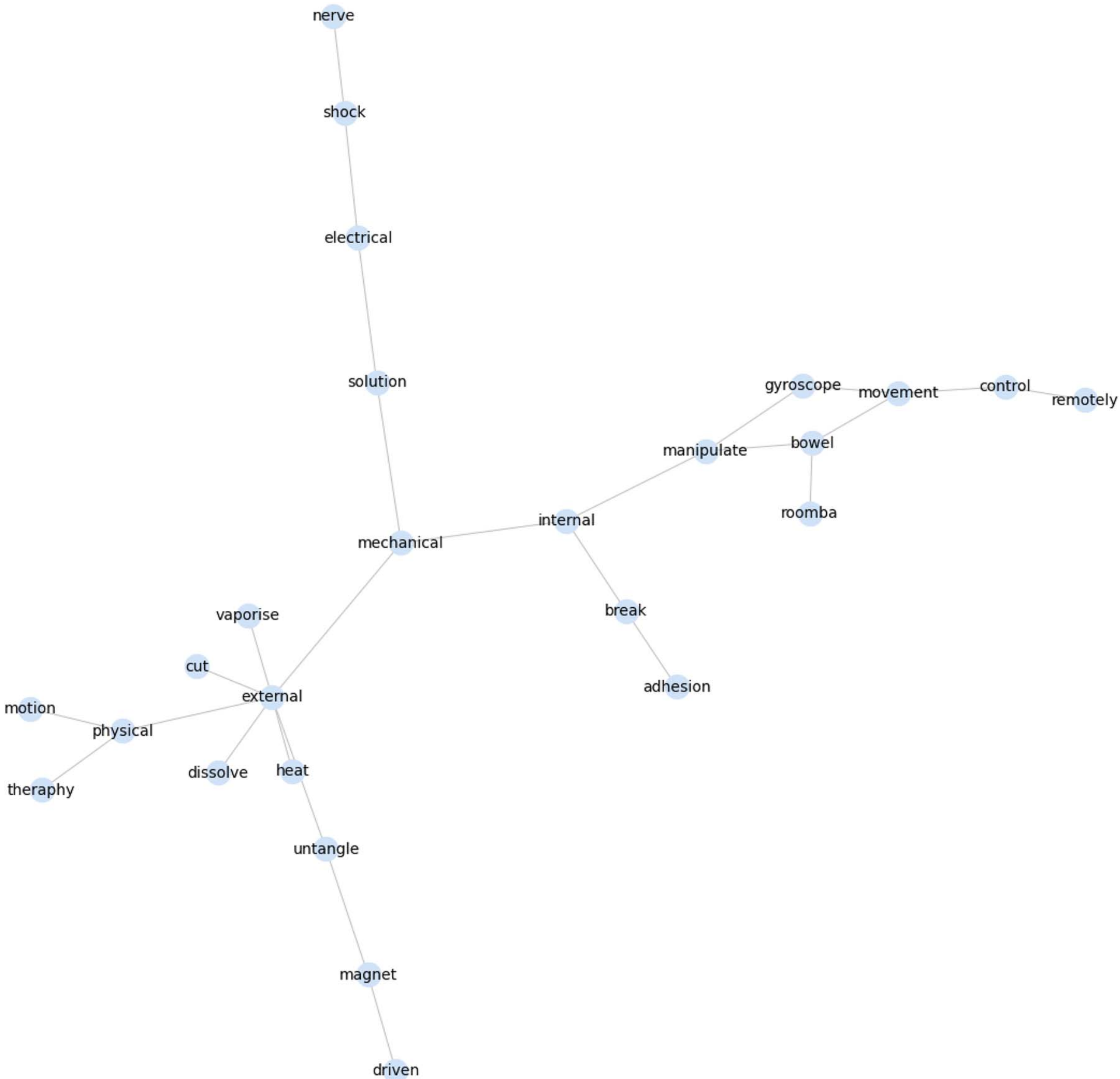

Fig. 19 Case study 1: human-generated mindmap 1

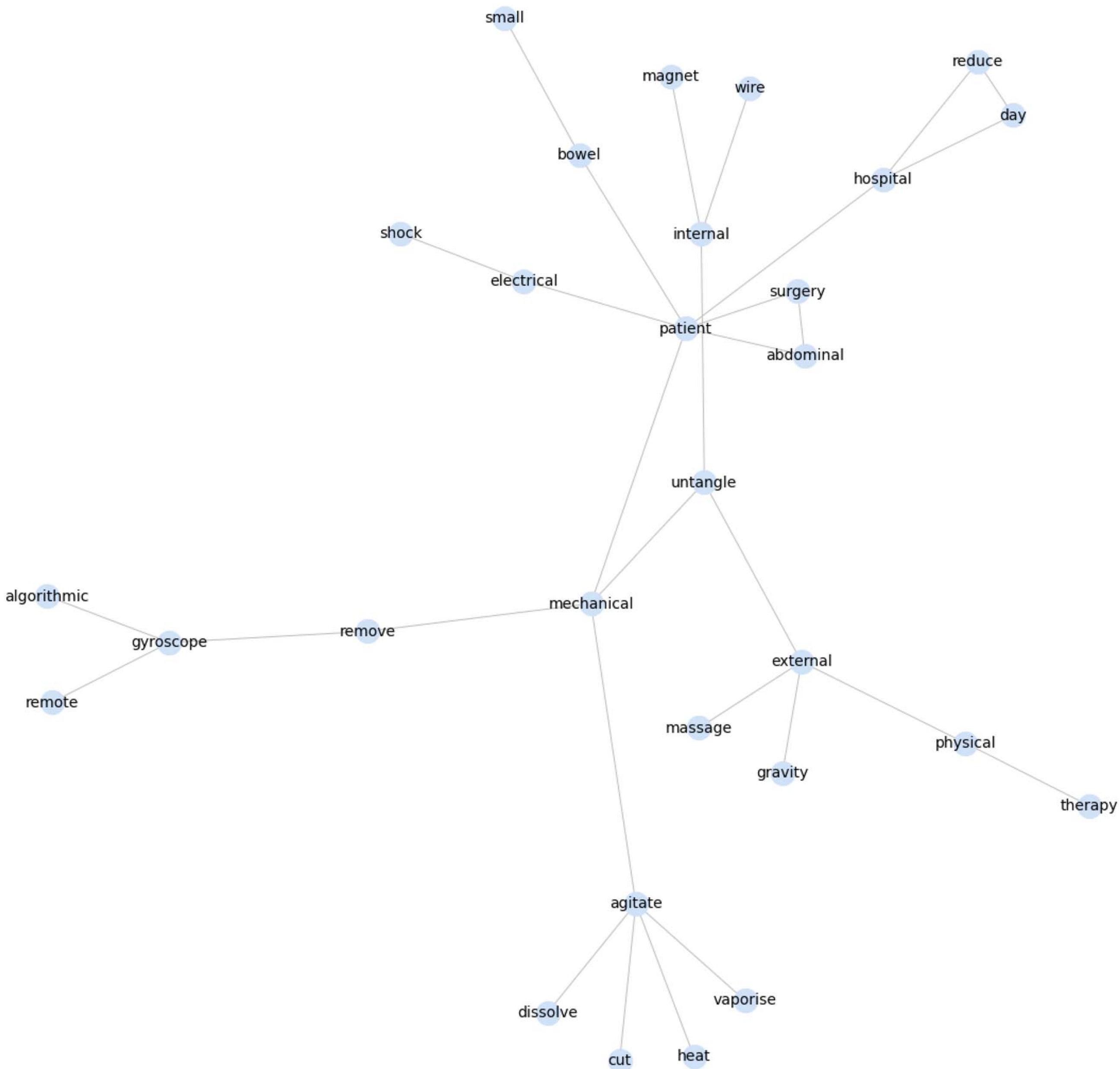

Fig. 20 Case study 1: human-generated mindmap 2

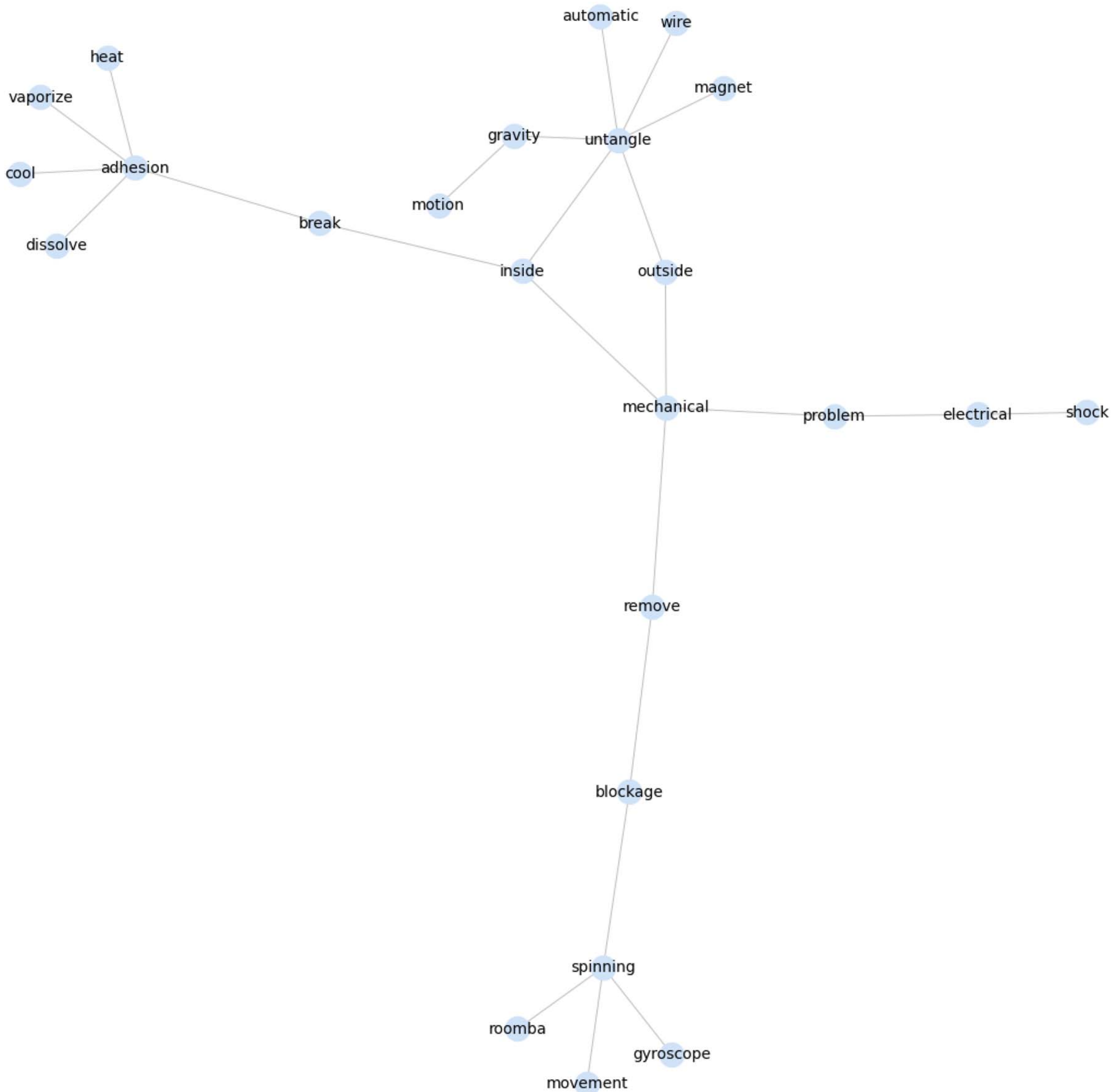

Fig. 21 Case study 1: human-generated mindmap 3

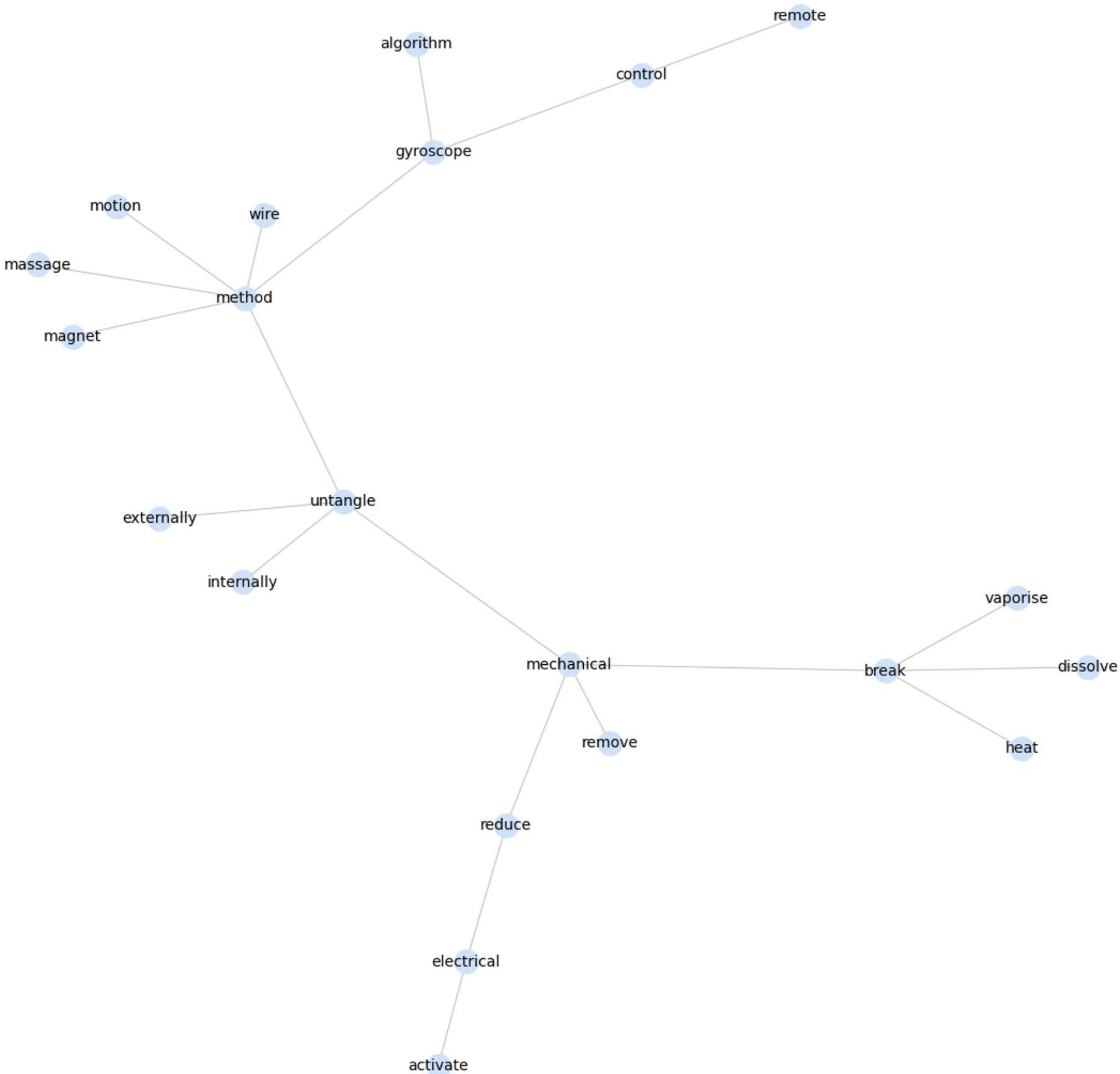

Fig. 22 Case study 1: human-generated mindmap 4

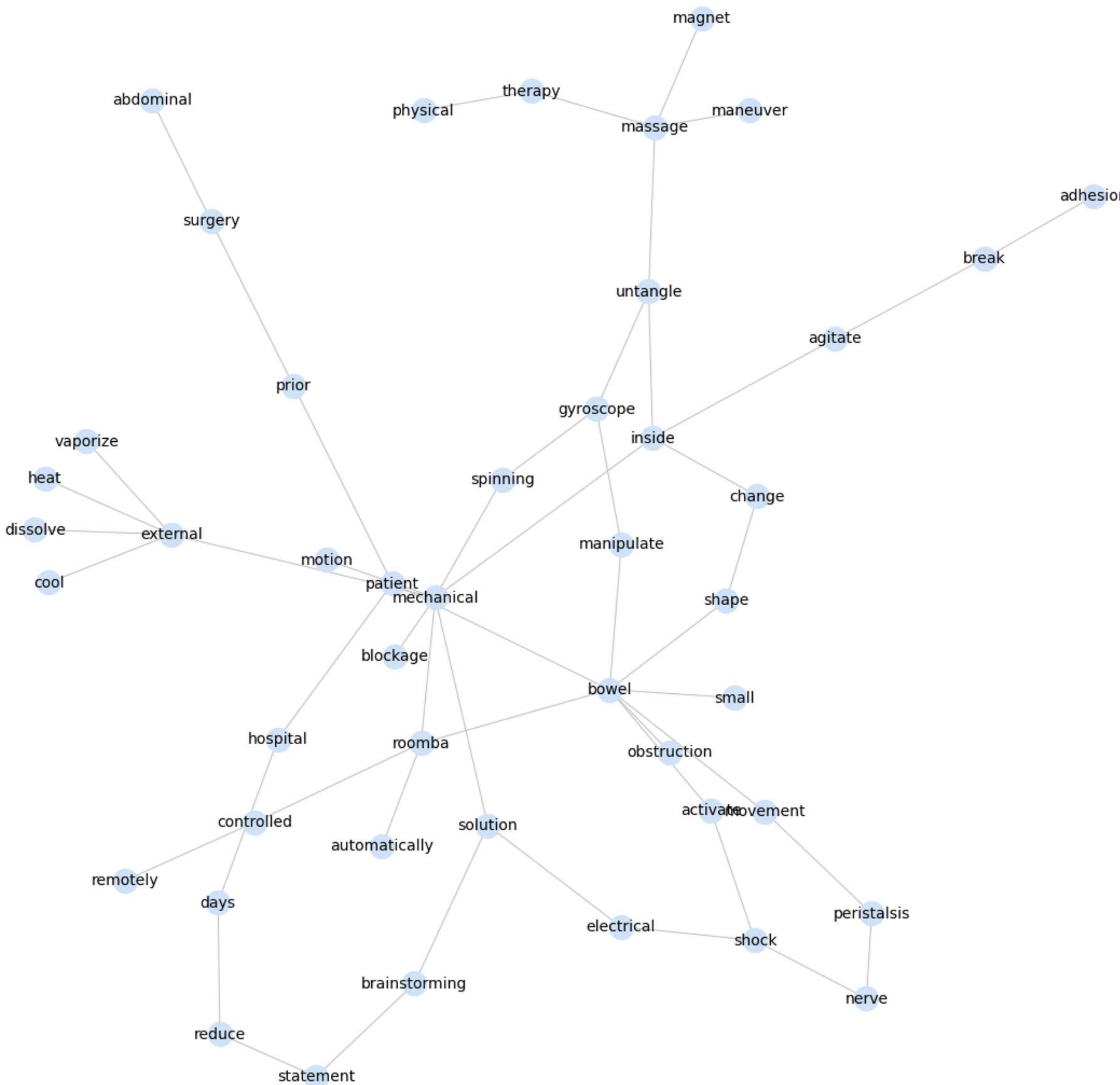

Fig. 23 Case study 1: human-generated mindmap 5

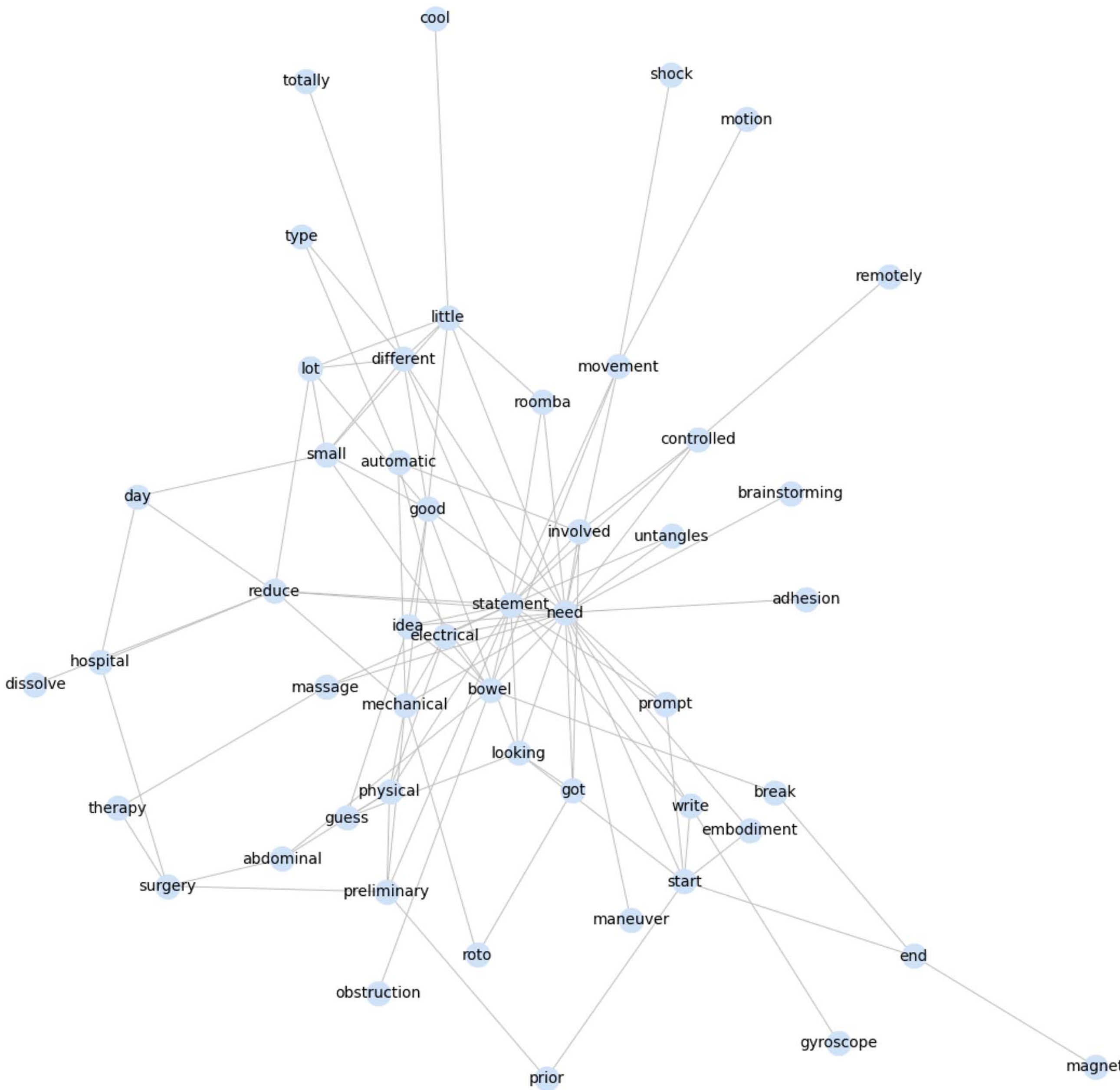

Fig. 24 Case study 1: computer-generated mindmap

References

[1] Cross, N., and Roy, R., 1989, *Engineering Design Methods*, Wiley, New York.
[2] Thompson, G., and Lordan, M., 1999, "A Review of Creativity Principles Applied to Engineering Design," Proc. Inst. Mech. Eng., Part E, **213**(1), pp. 17–31.
[3] Pugh, S., 1991, *Total Design: Integrated Methods for Successful Product Engineering*, Addison-Wesley, Reading, MA.
[4] Buzan, T., and Buzan, B., 1993, *The Mind Map Book*, BBC Books, London.
[5] Shneiderman, B., 2003, "The Eyes Have it: A Task by Data Type Taxonomy for Information Visualizations," *The Craft of Information Visualization*, Elsevier, New York, pp. 364–371.
[6] Chen, J., 2008, "The Using of Mind Map in Concept Design," 9th International Conference on Computer-Aided Industrial Design and Conceptual Design (CAID/CD), Kunming, China, Nov. 22–25, IEEE, pp. 1034–1037.
[7] Elhoseiny, M., and Elgammal, A., 2016, "Text to Multi-Level MindMaps," Multimed. Tools. Appl., **75**(8), pp. 4217–4244.
[8] Lin, H., and Faste, H., 2011, "Digital Mind Mapping: Innovations for Real-Time Collaborative Thinking," CHI'11 Extended Abstracts on Human Factors in Computing Systems, Vancouver, BC, Canada, May 7–12, ACM, pp. 2137–2142.
[9] Otto, K. N., 2003, *Product Design: Techniques in Reverse Engineering and New Product Development*, Prentice Hall, Upper Saddle River, NJ.
[10] Osborn, A., 1957, *Applied Imagination*, Scribner, New York, NY.
[11] Elhoseiny, M., and Elgammal, A., 2012, "English2mindmap: An Automated System for Mindmap Generation From English Text," 2012 IEEE International Symposium on Multimedia (ISM), Irvine, CA, Dec. 10–12, IEEE, pp. 326–331.
[12] Kudelić, R., Konecki, M., and Maleković, M., 2011, "Mind Map Generator Software Model With Text Mining Algorithm," Proceedings of the ITI 2011 33rd International Conference on Information Technology Interfaces (ITI), Cavtat, Croatia, June 27–30, IEEE, pp. 487–494.
[13] Kudelić, R., Maleković, M., and Lovrenčić, A., 2012, "Mind Map Generator Software," 2012 IEEE International Conference on Computer Science and Automation Engineering (CSAE), Zhangjiajie, China, May 25–27, Vol. 3, IEEE, pp. 123–127.
[14] Gonçalves, M., Cardoso, C., and Badke-Schaub, P., 2014, "What Inspires Designers? Preferences on Inspirational Approaches During Idea Generation," Des. Stud., **35**(1), pp. 29–53.
[15] Beel, J., and Langer, S., 2011, "An Exploratory Analysis of Mind Maps," Proceedings of the 11th ACM Symposium on Document Engineering, Mountain View, CA, Sept. 19–22, ACM, pp. 81–84.
[16] Tergan, S.-O., 2005, "Digital Concept Maps for Managing Knowledge and Information," *Knowledge and Information Visualization*, Springer, New York, pp. 185–204.
[17] Redifer, J. L., Bae, C. L., and Zhao, Q., "Self-Efficacy and Performance Feedback: Impacts on Cognitive Load During Creative Thinking," Learn. Instr., **71**, p. 101395.
[18] Thinkmap Inc. "ThinkMap." Thinkmap Inc. http://thinkmap.com/
[19] Speer, R., Chin, J., and Havasi, C., 2017, "Conceptnet 5.5: An Open Multilingual Graph of General Knowledge," 31st AAAI Conference on Artificial Intelligence, San Francisco, CA, Feb. 4–10.
[20] Chen, T.-J., and Krishnamurthy, V. R., 2020, "Investigating a Mixed-Initiative Workflow for Digital Mind-Mapping," ASME J. Mech. Des., **142**(10), p. 101404.
[21] Ang, S., Rzadca, K., and Datta, A., 2010, "SharedMind: A Tool for Collaborative Mind-Mapping," 2010 IEEE International Conference on Multimedia and Expo (ICME), Singapore, July 19–23, IEEE, pp. 1154–1155.
[22] Shih, P. C., Nguyen, D. H., Hirano, S. H., Redmiles, D. F., and Hayes, G. R., 2009, "GroupMind: Supporting Idea Generation Through a Collaborative Mind-Mapping Tool," Proceedings of the ACM 2009 International Conference on Supporting Group Work, Sanibel Island, FL, May 10–13, ACM, pp. 139–148.
[23] Hollingshead, A. B., McGrath, J. E., and O'Connor, K. M., 1993, "Group Task Performance and Communication Technology: A Longitudinal Study of Computer-Mediated Versus Face-to-Face Work Groups," Small Group Res., **24**(3), pp. 307–333.
[24] McLeod, P. L., 1992, "An Assessment of the Experimental Literature on Electronic Support of Group Work: Results of a Meta-Analysis," Hum. Comput, Int., **7**(3), pp. 257–280.
[25] Abdeen, M., El-Sahan, R., Ismaeil, A., El-Harouny, S., Shalaby, M., and Yagoub, M., 2009, "Direct Automatic Generation of Mind Maps From Text With M 2 Gen," 2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH), Toronto, ON, Canada, Sept. 26–27, IEEE, pp. 95–99.
[26] Hartmann, K. D., 2013, "The Power of Speech: Using Voice-to-Text Apps," OT Pract., **18**(14), pp. 18–19.
[27] Zhang, C., 2008, "Automatic Keyword Extraction From Documents Using Conditional Random Fields," J. Comput. Inf. Syst., **4**(3), pp. 1169–1180.
[28] Chen, P. I., and Lin, S. J., 2010, "Automatic Keyword Prediction Using Google Similarity Distance," Expert Syst. Appl., **37**(3), pp. 1928–1938.
[29] Beliga, S., Meštrović, A., and Martinčić-Ipšić, S., 2015, "An Overview of Graph-Based Keyword Extraction Methods and Approaches," J. Inf. Organ. Sci., **39**(1), pp. 1–20.
[30] Cheong, H., Li, W., Cheung, A., Nogueira, A., and Iorio, F., 2017, "Automated Extraction of Function Knowledge From Text," ASME J. Mech. Des., **139**(11), p. 111407.
[31] Shi, F., Chen, L., Han, J., and Childs, P., 2017, "A Data-Driven Text Mining and Semantic Network Analysis for Design Information Retrieval," ASME J. Mech. Des., **139**(11), p. 111402.
[32] Song, B., Luo, J., and Wood, K., 2019, "Data-Driven Platform Design: Patent Data and Function Network Analysis," ASME J. Mech. Des., **141**(2), p. 021101.
[33] Luo, J., Yan, B., and Wood, K., 2017, "InnoGPS for Data-Driven Exploration of Design Opportunities and Directions: The Case of Google Driverless car Project," ASME J. Mech. Des., **139**(11), p. 111416.
[34] Fu, K., Cagan, J., Kotovsky, K., and Wood, K., 2013, "Discovering Structure in Design Databases Through Functional and Surface Based Mapping," ASME J. Mech. Des., **135**(3), p. 031006.
[35] Fu, K., Chan, J., Cagan, J., Kotovsky, K., Schunn, C., and Wood, K., 2013, "The Meaning of "Near" and "Far": the Impact of Structuring Design Databases and the Effect of Distance of Analogy on Design Output," ASME J. Mech. Des., **135**(2), p. 021007.
[36] Steyvers, M., and Tenenbaum, J. B., 2005, "The Large-Scale Structure of Semantic Networks: Statistical Analyses and a Model of Semantic Growth," Cogn. Sci., **29**(1), pp. 41–78.
[37] Mihalcea, R., and Radev, D., 2011, *Graph-Based Natural Language Processing and Information Retrieval*, Cambridge University Press, Cambridge, UK.
[38] Solé, R. V., Corominas-Murtra, B., Valverde, S., and Steels, L. J. C., 2010, "Language Networks: Their Structure, Function, and Evolution," Complexity, **15**(6), pp. 20–26.
[39] Sonawane, S. S., and Kulkarni, P. A., 2014, "Graph Based Representation and Analysis of Text Document: A Survey of Techniques," Int. J. Comput. Appl., **96**(19), pp. 1–8.
[40] Li, X., Chen, C.-H., Zheng, P., Wang, Z., Jiang, Z., and Jiang, Z., 2020, "A Knowledge Graph-Aided Concept–Knowledge Approach for Evolutionary Smart Product–Service System Development," ASME J. Mech. Des., **142**(10), p. 101403.
[41] Lim, S. Y. C., Camburn, B. A., Moreno, D., Huang, Z., and Wood, K., 2016, "Design Concept Structures in Massive Group Ideation," International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Charlotte, NC, Aug. 21–24, Vol. 50190, American Society of Mechanical Engineers, p. V007T06A006.
[42] He, Y., Camburn, B., Liu, H., Luo, J., Yang, M., and Wood, K., 2019, "Mining and Representing the Concept Space of Existing Ideas for Directed Ideation," ASME J. Mech. Des., **141**(12), p. 121101.
[43] Mihalcea, R., and Tarau, P., 2004, "Textrank: Bringing Order Into Text," Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain, July 25–26.
[44] Brin, S., and Page, L., 1998, "The Anatomy of a Large-Scale Hypertextual web Search Engine," J. Comput. Networks ISDN Syst., **30**(1–7), pp. 107–117.
[45] Jamieson, P., 2012, "Using Modern Graph Analysis Techniques on Mind Maps to Help Quantify Learning," IEEE 2012 Frontiers in Education (FIE) Conference, Seattle, WA, Oct. 3–6, IEEE, pp. 1–6.
[46] Franklin, A., Li, T., Jamieson, P., Semlak, J., and Vanderbush, W., 2015, "Evaluating Metrics for Automatic Mind Map Assessment in Various Classes," 2015 IEEE Frontiers in Education (FIE) Conference, Camino Real El Paso, El Paso, TX, Oct. 21–24, IEEE, pp. 1–8.