BubbleNet: An Innovative Exploratory Search and Summarization Interface with Applicability in Health Social Media

Saeed Mohajeri
Dept. of Computing Science,
University of Alberta, Canada
Email: smohajer@ualberta.ca

Hamman W. Samuel
Dept. of Computing Science,
University of Alberta, Canada
Email: hwsamuel@ualberta.ca

Osmar R. Zaïane
Dept. of Computing Science,
University of Alberta, Canada
Email: zaiane@ualberta.ca

Davood Rafiei
Dept. of Computing Science,
University of Alberta, Canada
Email: drafiei@ualberta.ca

***Abstract*—We analyse the application of various interfaces to facilitate exploratory search and summarization of documents, especially BubbleNet, an innovative interface for summarizing corpus that also allows discovery of new knowledge that the user may not have previously been looking for. BubbleNet is a visual force-directed graph that displays an interactive and dynamic network of topics, semantic relationships, and related documents based on a corpus. Our experimental results show that BubbleNet gives a better user experience and faster performance in comparison with other exploratory search and summarization interfaces such as query-based search, word clouds, hierarchical directories, and topic graphs. We also explore the applicability of BubbleNet to Cardea, a health portal under development for patients and medics.**

I. Introduction

The World Wide Web has become the de facto source of knowledge to find, publish and share information. Websites such as Wikipedia and Google are now synonymous with knowledge, search and information. In parallel, the tools and methods to store, maintain, and retrieve large volumes of information have also been evolving. Two major user-centric challenges with the online information explosion are searching for the right information and summarization of large sets of information. Search methods have been moving away from matching just words to understanding word semantics and relationships in order to help users locate the most appropriate information based on their intent. In addition, summarization has been developing towards visual methodologies.

A majority of the users searching for technical information online are unfamiliar with the domain they are searching in, and are exploring the knowledge available. For instance, users may not fully know the domain-specific terminology to use, which limits their keywords for searching. This leads to exploratory search behaviours, where users attempt to use different search keywords, and also evaluate the results by trial and error to find the best matches [1], [2].

Moreover, when looking at results that contain thousands of documents, the user may want to get a high-level overview of the contents of all the indexed documents without the need to read all of them. Also, the user may want to find a particular document or topic within all the topics contained in the results easily and quickly.

There are several methods to help users perform exploratory search. Query-based search tools allow entry of keywords that are matched with desired documents. Alternatively, hierarchies of categories and links within websites can be used to reach a particular document. For summarization, text summaries and word clouds are frequently used. Word clouds can help users find documents based on frequent words and tags appearing in them.

In this paper, we introduce another approach, BubbleNet, that addresses both challenges of information overload on the Internet, with focus on online health information in health discussion forums. BubbleNet presents an abstract and high-level representation of major concepts discussed in a set of documents which can be explored without domain knowledge in health. These could be a set of individual documents, such as news articles, or parts of a long document, such as individual comments made within the same news article.

Previously, research has been presented on using BubbleNet for navigating health discussion forums [3]. This paper generalizes BubbleNet as an exploratory search and summarization interface for online health information, and also provides further details on the algorithms used. Moreover, a new evaluation of BubbleNet is presented by comparing with other methods for exploratory search and summarization. Also, the applicability of BubbleNet to health social media is appraised, particularly for a new health portal called Cardea, that allows patients and medics to communicate online.

In Section II, we present an overview of the proposed system, while in Section III, related research work in this area is investigated, including other interfaces for exploratory search. In Section IV, an evaluation of the proposed system is given, and Section V gives a high-level overview of the Cardea health portal, as well as an appraisal of the applicability of BubbleNet within Cardea. Section VI concludes with possible future directions for this research.

II. Methodology

Given a set of indexed and searchable documents, $D = \{d_1, d_2, \ldots\}$, where each document d_i is composed of a set of keyword entities from a universe $E = \{e_1, e_2, \ldots\}$, the search task is to identify documents d_i that contain search query keywords, $Q = \{q_1, q_2, \ldots\}$, i.e. $\exists q_j \in Q$ such that q_j is mentioned in d_i. We can also identify a network of relationships between entities in E where there is a relationship between entities e_p and e_q if there is a document where both entities are mentioned. This network of entities is also representative of the topics in a given corpus, and therefore serves the purpose of summarizing the corpus. An exploratory search task can then be described as a combination of querying via Q and browsing strategies based on the relationships between entities, thereby enabling learning and investigation [1]. The rest of this section gives a formulation of BubbleNet and its implementation details.

A. Overview of BubbleNet

Human experts, such as librarians, can typically identify the most pertinent keywords for a document that represent a high-level representation of the major topics. This can also be achieved computationally: given a document, there are several algorithms for extracting important terms that represent the major topics. Within a corpus of documents, such as a news website with thousands of articles, or a discussion forum with thousands of discussion threads, there are often many related documents talking about the same concepts. For example, a search for any given keyword would return hundreds of documents in the results. However, even though the documents contain certain shared keywords, they are also likely to contain different aspects of the topics represented by the keywords.

When users are trying to find a document, they have a topic in mind. Using query-based search engines, they have to convert that topic to one or more keywords and find their desired document within a set of documents returned by the search engine. More advanced search engines help users enhance their queries by suggesting similar words or automatically refining the query to match more related documents.

BubbleNet, on the other hand, provides a high-level representation of topics appearing in the corpus in the form of a network, showing the topics as well as their relationships. This network is built using an estimation of semantic relationships between topics. Having such a network, a user can see the big picture of all major concepts within a searchable corpus. The user then can navigate through this network by either refining or expansion. The user can drill down from a given topic to see other related concepts in a lower and more detailed level. The user can also navigate to other related topics and finally find a set of documents talking about their desired topics.

B. Searchable Documents Indexing

BubbleNet is able to provide exploratory functionality by building a searchable database of indexed documents. An overview of the system architecture for indexing documents to add to the BubbleNet documents database is given in Figure 1. As new documents arrive, they are loaded by Document Loader and stored in document objects. Document objects are then passed to the Entity Extractor component, which produces

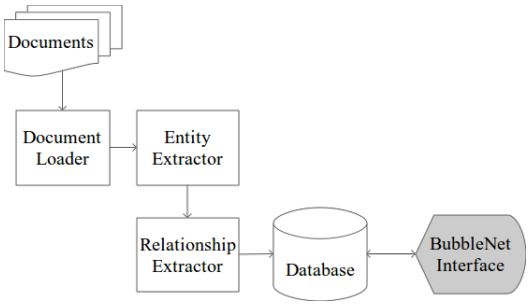


Fig. 1: BubbleNet System Architecture

entity objects. The document and entity objects are then passed to the Relationship Extractor component to create a set of relationship objects. Ultimately, the document, entity and relationship objects are stored in a database. The User Interface component helps the user to visually retrieve these stored objects from the database and perform exploratory search.

Formally, given a set of documents, $D = \{d_1, \ldots, d_n\}$ and a set E of entities that are mentioned in those documents, the task is to find a set of relationships $R = \{r_1, r_2, \ldots\}$ where for every $r_i \in R$ there is a pair of entities $\{e_p, e_q\}$ in E such that both entities are mentioned in the same document.

1) Topic Extraction: For each document, the representative keywords that model the main topics of that document need to be extracted. There are several algorithms for extracting keywords, such as use statistical methods, linguistic methods, machine learning approaches, or hybrid approaches. Statistical methods use the statistics of words, their relative frequency, and position in a document to estimate importance and representativeness of keywords. On the other hand, linguistic methods use linguistic features such as sentence structure and parts of speech to determine representative keywords. In machine learning methods, training data is used to learn models for recognizing important keywords. Moreover, in addition to extracting keywords, named entities can also be useful, which are terms and phrases that refer to special entities, such as persons, organizations, locations, facilities, and others.

BubbleNet implements a hybrid method by combining keyword extraction and Named Entity Recognition (NER). There are various tools that implement algorithms for keyword extraction and NER, such as Yahoo Content Analysis, Keyphrase Extraction Algorithm (KEA), Thomson Reuters Open Calais, Stanford Named Entity Recognizer, Apache OpenNLP, and AlchemyAPI. BubbleNet leverages AlchemyAPI, a commercial online service for text analysis, including keyphrase extraction, NER and several other tasks. Users can use a free version of this system that has a limitation of 1,000 transactions per day. Alchemy granted permission to use their service for free for this research on BubbleNet. For each document to be indexed, we send the contents to the Alchemy server and get a list of scored keyphrases and a list of scored named entities. Combining those two lists, we will have a complete list of extracted entities with their estimated scores.

2) Relation Extraction: Given a set of representative keywords for modelling topics in each indexed document, the next step is to extract and estimate relationships between those

topics. We use statistical information based on the appearance of keywords in sentences within the indexed documents, i.e. co-occurrence. The relatedness of entities is estimated based on how frequent they co-occur in documents, as well as how close they appear in a particular document. In other words, every co-occurrence of two entities in a single document causes an increase in the total score of the relationship between those two entities, and the amount of this increase depends on their distance. To take the distance between the occurrences of two entities, a and b, in a document d_i, we define a distance-based score $dbs(a, b)$. The following algorithm is used to compute a normalized score, $\mathcal{C}^*$, for the co-occurrence relationship between a and b. In the algorithm, A and B are sets of offsets of occurrences of a and b respectively, measured in terms of the characters from the beginning of the document.

1. Let A and B be the set of offsets of respectively a and b in d_i

2. Compute

$$\mathcal{C} = \sum_{(o_i,o_j)\in A\times B}^{i\times j} dbs(A_{oi}, B_{oj}) \tag{1}$$

3. Normalize

$$\mathcal{C}^*_{a,b} = \frac{C}{\sqrt{|A||B|}} \tag{2}$$

In addition to the co-occurrence of entities, additional relationships can be derived between entities by using ontologies to expand the links between entities. For this purpose, the WordNet lexical database is used to extract additional relationships. WordNet provides a distance measure w between two entities a and b, i.e. $w(a, b)$, which is between 0 and 1, or -1 if the entities do not exist in the ontology. Consequently, a WordNet ontology-based score $\mathcal{W}$ can be defined using the following heuristics, where $\tau = 0.001$.

$$\mathcal{W}_{a,b} = \begin{cases} \tau & \text{if } w(a,b) = -1 \\ (w(a,b)+\tau)^{-1} & \text{if } w(a,b) = [0,1] \end{cases} \tag{3}$$

The overall effective score, $\mathcal{S}$ for relationships between entities a and b is calculated as a sum of all scores across all indexed documents, i.e. $\mathcal{S}_{a,b} = \mathcal{C}^*_{a,b} + \mathcal{W}_{a,b}$. In this manner, BubbleNet is able to identify and link additional related keywords to a given set of search query keywords for exploration.

C. Temporal Context

BubbleNet also supports contextualizing entities and their relationships to particular time spans. This is done by taking into account the time stamp of the documents being indexed. Therefore, the definition of the effective score $\mathcal{S}$ can be extended to $\mathcal{S}(a, b, t)$, where t is a given time span. Consequently, $\mathcal{S}$ can be different for two given entities for different temporal contexts. This allows observation of relationship strengths changing over time.

D. Visualization as Force-Directed Graph

Given an indexed database of documents, entities and relationships, the last step in constructing BubbleNet is to provide a graphical interface that visualizes this network of entities and relationships. Our goal in this visualization is to represent the bubbles in a way that the user can understand the retrieved information, including entities and their importance, as well as relationships between entities and their strength. To this end, the concept of simulating physical masses, springs and forces is used as a visualization layout for the BubbleNet network. Using the physics of masses and springs is an effective way to visualize graphs and has been widely used in other visualization systems. In addition to representing a network of interconnected entities, a force-directed graph can provide a dynamic and interactive representation of the relative importance between entities and relationships based on the physical interaction between the modelled masses and springs.

1) Entities as Bubble Masses: In its interface, BubbleNet models an entity as a bubble. A bubble is a simple circle representing an entity, with a label on it, representing the entity caption, i.e. the topic or keyword. To represent the importance of the entities, we use different radii and fill colours. The radius of a bubble i is specified as $r_i = 55 \times s^*_i + 15 + 1.5 \times len_i$, where s^*_i is the normalized score of bubble i among other retrieved bubbles and len_i is the length of the phrase of entity i in characters. The parameters in this formula are chosen by experiments in a way that entity captions fit in the circles.

The colour of a bubble is then selected according to its calculated radius based on a scale that maps radii to colours. We used 5 different colours for drawing the bubbles. The font sizes for the bubbles texts are also chosen based on both the bubbles radii and scores, so that the more important an entity is, the larger its font size.

In addition, a bubble is modelled as having a physical mass, which is proportional to its size. Considering bubbles as discs with equal thicknesses, the mass of a bubble i is proportional to r_i^2. The mass of the bubble also models the relative importance of the associated entity compared to other entities.

2) Relationships as Springs Between Bubbles: In BubbleNet, a link is modelled as a spring between two entities and is shown as a line between the circles. The strength of a relationship is represented using both line thickness and length. Stronger relationships are thicker. They are also shorter, resulting in the two bubbles standing closer to each other following the intuition of their relatedness. The intial length of a spring is specified as $len_{i,j} = 50 \times s^*_{i,j} + 20$, where $len_{i,j}$ is the length of the spring between bubbles i and j and $s^*_{i,j}$ is the normalized relative score of the relationship. The initial positions of bubbles are random. The springs always connect the closest points of the two bubbles, thus, their lengths are not necessarily equal to their initial lengths until they reach an equilibrium. An equilibrium is achieved by balancing the spring tensions against the bubble masses.

3) Masses, Springs and Forces Simulation: We simulate physical laws governing masses and springs by calculating forces exerted on the bubbles, their accelerations, velocities and position updates as the time goes by. There are four forces exerted on bubbles.

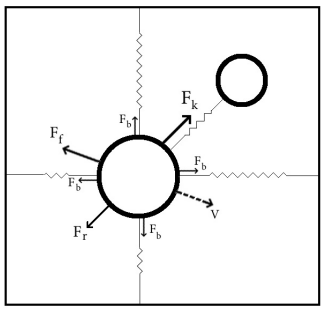


Fig. 2: Forces Exerted on a Bubble

- **Spring Forces**: Every spring that is connected to a bubble exerts a force on it. According to Hooke's law, the magnitude and direction of this force depends on the distance x the spring is extended or compressed. The direction of this force is specified by the relative position of the other tail of the spring, which is the other bubble, i.e. $F_k \propto \triangle x$.
- **Repulsion Between Bubbles**: To avoid bubbles having overlap, we consider a repulsion force between two bubbles i and j. This force is proportional to the masses of the bubbles, m_i, m_j, and is inversely proportional to their distance $d_{i,j}$, i.e. $F_r \propto \frac{m_i \times m_j}{d_{i,j}}$.
- **Boundary Repulsion**: To keep the bubbles inside a bounding box, we consider four springs connected to a bubble i and the borders where these springs can freely slide on the borders so that they always exert forces in horizontal or vertical direction on the bubble. Boundary forces are considered to be proportional to bubble's mass m_i and its distance from the borders $d_{i,bor}$, i.e. $F_b \propto -m_i \times d_{i,bor}$.
- **Frictional Forces**: If two bubbles are connected by a spring that is not in its initial length, they will fluctuate around an equilibrium point forever. To avoid this, we simulate a friction point that causes the bubbles to gradually lose their velocities. The friction force is proportional to the mass m_i of a bubble i, as well as its velocity v_i, and is always in reverse direction of its velocity, i.e. $F_f \propto -m_i \times v_i$.

Summing up all the forces in a two dimensional space, the BubbleNet interface repeatedly calculates the bubbles' accelerations and updates their velocities and positions to simulate their movements. Figure 2 summarizes the forces exerted on bubbles.

E. Prototype Walkthrough

Figure 3a shows the BubbleNet interface, demonstrating an overview of the controls, entities and their relationships within indexed documents. This interface can be used to look up keywords and maintain the visual cues that allows exploratory search using other related entities. Filtering is available through the text box provided. In addition, there is a time interval slide bar control provided that can be used to visually define a time span, as shown in Figure 3b. To do this, the user is able to slide handles to indicate the desired span.

When the user points to a bubble, that bubble gets a thicker stroke to tell the user that some actions are available if the user hovers or clicks on the bubble. Also, when the user hovers on a bubble for a short while, the script expands that bubble as a preview, and highlights the rest of the directly related bubbles by giving unrelated bubbles more opacity. This is demonstrated in Figure 3c.

Also, when the user hovers on a link or on an expanded bubble, the script retrieves a list of most relevant documents and shows it to the user, as depicted in Figure 3d. By clicking an item in that list, the document will be shown to the user with the keywords highlighted.

When the user clicks on a bubble, other bubbles are removed, a new request is sent to the server to retrieve related entities to the clicked one, and once the results are received, new bubbles are generated and connected to the originally clicked bubble. Physics laws then move this expanded bubble to the centre as other bubbles tend to stabilize around it due to the springs between them.

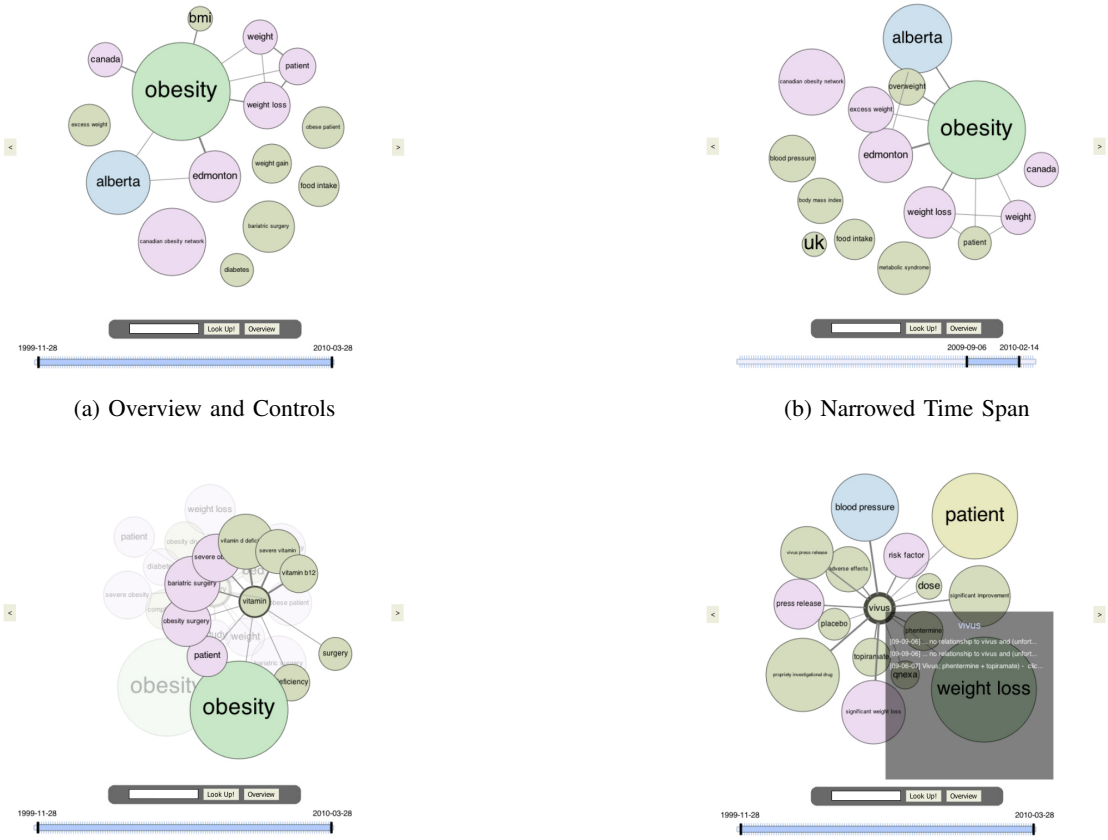
III. Related Literature

In this section, other approaches and interfaces to facilitate exploratory search are investigated and compared with BubbleNet: document clustering, query-based search, hierarchical directories, word clouds, and topic graphs. In addition, a comparison is done of BubbleNet and SKIMMR, a tool for visually summarizing documents.

A. Document Clustering

Document clustering refers to grouping documents in categories based on their content. Using document clustering, documents of a corpus can be organized for access via topical categories. By looking at clusters, a user can infer the main categories that documents of a corpus fall into. In addition, it helps users find their desired documents because documents that share similar topics are placed in proximity of each other. This also helps users explore documents related to their intended topic. Two possible approaches include hierarchical clustering, and *k*-means clustering [4]. The outcome of these clustering methods is either in a flat or hierarchical grouping. In a flat cluster, each document belongs to only one category, while in a hierarchical structure, documents can belong to a set of categories, from the most general to the most specific.

A common drawback of clustering approaches is the limitation on groupings. A document normally cannot belong to more than one category, or in the case of a hierarchy, a document cannot have more than one parent. This limits the freedom to model the topics appearing in documents, especially when considering both general and detailed topics in documents. This is a common feature of hard clustering. On the other hand, fuzzy or soft clustering methods allow an element to belong to more than one cluster. Recently, [5] have covered detection of non-disjoint groups where a document mentions several topics and ought to belong to several topical groups. Their strategy is to use an overlapping clustering method, Kernel Overlapping K-Means-based Word Sequence Kernel (KOKM-based WSK), where text is modelled as an ordered sequences of n-grams and WSK is used as a similarity metric between documents.



(a) Overview and Controls

(b) Narrowed Time Span

(c) Highlighted Related Entities

(d) Related Documents

Fig. 3: BubbleNet Interface

B. Automatic Summarization

Automatic summarization generates a summary of a given document so that a user can understand the main points discussed in a document by reading the summary instead of the entire document. Summaries can be generated by selecting important clauses and sentences from text using pre-defined heuristics. For example, the first sentence a paragraph often carries significant information about the main points of that paragraph. Important clauses and sentences can be recognized based on such rules [6].

C. Query-Based Search

In most search engines, users have to enter a query as a set of words or phrases, optionally combined with some operators, to express what they are looking for. This is not always an easy task for users to do, as they may not be aware of the correct domain terminology, or they may have difficulty finding the words that precisely explain their desired topic. In general, search algorithms attempt to retrieve documents that contain words mentioned in user's query. This includes refinements to the query, such as handling misspelt words, abbreviations, alternate forms such as plurals, and query expansion by finding synonyms and assigning weights to keywords [7].

In addition, search engines try to help users refine queries by suggesting additional terms that are related to the query. The suggested terms can be determined by using query logs, user feedback, semantic relations, and personalization. By analyzing a large number of queries within query logs from other users, a search engine can find the words that are likely to appear together in queries. In addition to co-occurrences of words in queries, query logs can be analyzed to find how users change their queries by adding, removing or reordering words in their queries after seeing search results [8]. Also, words that are semantically related to query words can be suggested [9], [10]. Moreover, by profiling recent queries of a user, and providing context based on the user's location and other features, search engines can suggest more personalized terms to a specific user. This can improve user satisfaction, especially when the query is ambiguous [11].

One drawback of term suggestion is the need to have a large amount of query logs and statistics. Another limitation is that this method does not give users an overview of the main topics mentioned in the corpus, so they have to spend a lot of time exploring documents to get a summary.

D. Hierarchical Directories

In this approach, documents are clustered hierarchically and a directory of topics is shown on a given website. Users

can browse this hierarchy to reach a particular document. Many websites partially provide this features. This method provides an overview of topics in the corpus in an abstract form. Users can also find documents without the need of expressing their meaning in terms of a query, as they can follow the hierarchy to find what they want [12]. To build a hierarchy of documents, document clustering is used. Thus, hierarchical directories have the same limitations as document clustering. In addition, relations between topics discussed in documents are not easy to navigate, except through the generalization-specialization links of the hierarchy. This prevents users from freely exploring relevant topics regardless of the structure of the hierarchy.

E. Word Clouds

Word clouds have been increasing used by websites to provide a means for summarizing and navigating through website contents. A word cloud visualizes statistics of term usage in a text or corpus. The more frequently a word is used in a corpus, the larger that word appears in the cloud. Word clouds can also be constructed using folksonomies: tags that users assign to documents. By using tags instead of words, a tag cloud can represent the most important topics discussed in a corpus [13]. An advantage of word cloud is that it gives the user a broad perspective of important topics discussed in a document or a corpus of documents in just one glance, as a way of summarizing its contents. Another advantage is that users can decide which word to click by looking at the list of words, not by having query words beforehand.

On the other hand, a word cloud is not able to represent relationships between important topics. Words are ordered alphabetically and there is no clue about possible relationships between entities. There are variations of word clouds that tried to solve this problem by arranging relevant words together. Another limitation is that a word cloud generates a flat view of topics and does not provide any means for interacting with users. Thus, a user can only look at the list of words and choose a topic to retrieve relevant documents. Furthermore, it is important to select appropriate terms, and there could be several unimportant word listed in the word cloud interface.

F. Topic Graphs

Topic graphs use named entities and keyphrases from content, represented graphically according to frequency, with links between entities representing their co-occurrence rate [14]. However, topic graphs do not capture all topics in documents. Also, they do not allow users to retrieve documents based on combinations of topics, since the only way of accessing documents is by clicking on entities.

G. SKIMMR: Machine-Aided Skim-Reading

SKIMMR is a tool for summarizing and retrieving documents that automatically extracts entities from a set of documents using natural language tools and provides a web-based interface to an interconnected graph of the extracted entities [15]. BubbleNet improves various aspects of SKIMMR. For instance, SKIMMR only extracts named entities using domain-specific tools in medicine and life sciences, while BubbleNet allows indexing of mixed-domain corpora. Also, SKIMMR requires an entire corpus to evaluate corpus-wide relationships, while BubbleNet allows additional documents to be indexed over time. In addition, an evolutionary view of relationships between entities over time is not possible in SKIMMR, while BubbleNet defines relationships within the context of time spans. The SKIMMR interface requires an initial query to begin searching, while BubbleNet provides a truly graphical navigation of a given corpus, where search terms are mainly for filtering and drilling down. Ultimately, BubbleNet enables exploratory search without the need for text-based queries.

IV. Evaluation and Discussion

Tasks were given to users to perform, and their responses used to evaluate BubbleNet against word clouds and query-based search. In addition, an exit questionnaire was provided. The tasks were created based on heterogeneous datasets that were indexed in BubbleNet.

A. Datasets

Three datasets were used from varying document types: Reuters news articles, health forums, and a medical blog by a health professional.

- **Reuters News Articles**: This dataset consisted of 15,000 news articles. The original dataset consists of 21,578 categorized news articles from Reuters and is widely used for text categorization tasks. Only the first 15,000 articles, ignoring all categorization metadata.
- **Health Forums**: Discussions from three health forums were used: eHealthForum, HealthBoards, and MedHelp. A total of 5,000 discussions from each of the three discussion forums were retrieved, resulting in a dataset with a total of 15,000 discussions.
- **Obesity Blog**: A set of 900 blog posts from a professional blog about obesity were also retrieved [16].

B. Experiment Setup

For evaluation, users were invited to complete an online survey, consisting of two information retrieval tasks and a questionnaire.

1) Task 1 - Summarization: In this task, BubbleNet was compared against the word cloud summarization method. Users were provided with a random document from the experiment datasets and asked to skim the document to get an idea of the topics mentioned in it. To help users organize and summarize the contents of the document, they were asked to choose a set of 3 to 5 keywords that best represented the document. After selecting these keywords, the survey displayed two different summaries of the document using a word cloud and BubbleNet. Tag clouds were created using the same set of entities extracted by the Entity Extractor module to avoid bias towards BubbleNet. Also, the tag cloud was configured to show the top 50 most frequent entities, while BubbleNet displayed 20 entities initially, and 15 sub-entities when expanding a given entity.

Ultimately, users were asked to choose between the word cloud and BubbleNet based on which gave a better overview

of the document. Open text comments were also allowed. The users were allowed to repeat the experiment for up to 15 different documents randomly selected from the dataset.

2) Task 2 - Exploratory Search: In this task, BubbleNet was evaluated against query-based search and specifically for exploration of information. The task was to find the names of 3 to 5 symptoms, medications, diagnosis or treatments, given the name of a disease. The users were given a set of documents to search for this information. For this task, the datasets were limited to the health forums.

The following disease names were used, based on their availability within the health forums datasets: cancer, flu, migraine, asthma, diabetes, anemia, lupus, mumps, hepatitis, tumor. For each round of this task, a pair of different diseases were assigned to the user from the diseases list. Users could repeat this task up to 10 times.

Firstly, users were asked to find related entities using query-based search for a given disease name. A conventional search interface was developed so that users could enter queries and retrieve documents that match their query. Secondly, users were given another disease to search using BubbleNet. Finally, the users were asked to choose the method that they thought was more appropriate for this task. As an optional question, they were asked why they made that choice. The time spent on the tasks was also compared for the two interfaces.

3) Exit Questionnaire: After performing the two tasks, users were given an exit questionnaire. The questionnaire asked users to choose one of the three provided interfaces (tag cloud, query-based search and BubbleNet) based on two criteria: usefulness and easiness. Open text comments and feedback was also made available.

C. Results and Discussion

A total of 56 users participated in Task 1, 43 participants completed Task 2, and 27 answered the exit questionnaire.

Figure 4a shows the survey results for Task 1, comparing BubbleNet with a tag cloud interface. For all dataset categories, BubbleNet was perceived to be a better interface for understanding document topics, by a significant majority of the surveyed users.

Figure 4b shows the results for Task 2 which compared the use of query search against BubbleNet for exploratory search. Two parameters are used for evaluation, and in both situations, BubbleNet out-performs query-based searching. In terms of the average time users took to locate information, BubbleNet was faster by an average of approximately 67 seconds. Also, over 60% of the users found BubbleNet better suited for exploratory search.

Figure 4c shows the feedback from the exit questionnaire, where BubbleNet was perceived to be better in terms of usefulness and ease of use by a significant number of users, as compared with both the word cloud and query-based methods.

Following are paraphrased listings of select open text comments that were submitted by users regarding BubbleNet.

- The relationships between topics are useful
- Topics are simple to understand and remember
- It is dynamic, interactive, and intuitive
- It looks nicer and more attractive
- It enhances focusability
- It enables fast lookup of relevant information
- The additional keywords that are suggested are useful to discover new information
- It cannot combine several keywords
- It sometimes contains misleading topics and irrelevant keywords
- Not all documents shown were relevant to the topic

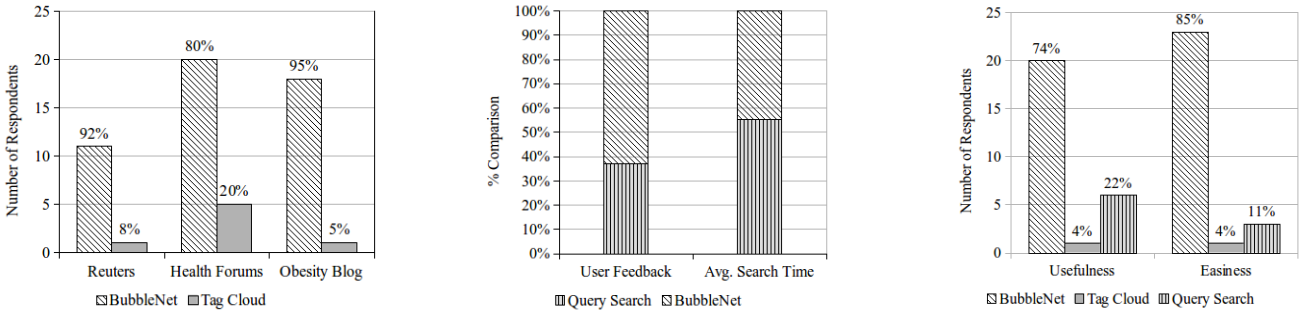
V. Cardea Health Portal and BubbleNet

Cardea is a new health portal currently under development for medical professionals or medics, and laypersons or patients. Cardea aims to include features from existing health social networks, forums, blogs, and other social media to empower patients to consume credible health information. Patients can also interact with medics, and can share experiences, write blogs, ask questions, chat real-time, or get answers in three streamlined environments: patient to patient, patient to medic, and medic to medic. Medics can interact with patients or other medics to create articles, answer questions and collaborate via wikis. Each of these areas has threads associated with specific health topics, such as “obesity”, “cancer”, and others. Registered users can subscribe to threads, and also connect with other users. Content in Cardea is associated with various trust metrics to inform users about its credibility. Cardea also provides advanced privacy options for users to control content visibility.

Previous research has looked at using BubbleNet for navigating health discussion forums [3]. In Cardea, BubbleNet is an ideal interface for navigating threads, blogs, articles, answers, and wikis. The information being discussed within multiple threads under a given topic can be summarized using the BubbleNet interface. This would allow users to know the issues being addressed for a given topic without reading many threads, and also enable exploration of new information. In addition, a thread itself can tend to become long, and BubbleNet can provide a granular-level overview of individual threads. Lengthy articles, blogs, answers, and wikis can also be summarized individually to facilitate faster skimming and exploration for users. Our future work will look into incorporating Cardea’s content trust metrics within BubbleNet, so that credible content can be promoted within summarizations and exploratory search pathways.

VI. Conclusion

There have been several interfaces developed for the information retrieval systems tasks of exploring and summarizing large volumes of indexed information, such as traditional query-based search and word clouds. Traditional search is very common and effective in retrieving documents, but does not help users get an overview of the documents and requires them to have the right keywords in mind to explain their information need. Word clouds, on the other hand, provide a summary of contents of documents based on frequency,



(a) BubbleNet and Word Cloud (b) BubbleNet and Query Search (c) Exit Questionnaire

Fig. 4: Survey Feedback for Evaluation of BubbleNet

but are not explorable. An interactive user interface, called BubbleNet, is presented that facilitates users to get an overview of the contents of a corpus at a glance in the form of an interconnected network of topics represented as bubbles. It also enables exploratory searching of related concepts and topics the user may not have been originally searching for. BubbleNet was evaluated by a survey methodology, where users were asked to complete tasks and compare BubbleNet with other interfaces suitable for exploratory search. The results show that users expressed more satisfaction when using BubbleNet compared with query-based search or word clouds. Moreover, BubbleNet provided a faster user experience compared with other methods. The applicability of BubbleNet to Cardea, a new health portal was investigated. The current BubbleNet prototype has limitations that can be improved in future work. For instance, BubbleNet currently does not distinguish between word senses, and does not incorporate spelling corrections. Another important direction of improvement to amalgamate BubbleNet with other search methods, such as query-based search. Also, semantic relationships between topics are currently being inferred using WordNet, but other methods can be incorporated and compared, such as Wikipedia and DBpedia [17]. Finally, a precise evaluation of BubbleNet needs more experimentation, consisting of more information retrieval tasks and a larger sample population of users. Additional future work includes incorporating document trust metrics into BubbleNet.

ACKNOWLEDGEMENT

The authors wish to thank the Alberta Innovates Centre for Machine Learning (AICML) and the Natural Sciences and Engineering Research Council (NSERC) for funding this research project.

REFERENCES

[1] G. Marchionini, "Exploratory Search: From Finding to Understanding," *Communications of the ACM*, vol. 49, no. 4, pp. 41–46, 2006.

[2] C. Janiszewski, "The Influence of Display Characteristics on Visual Exploratory Search Behavior," *Journal of Consumer Research*, vol. 25, no. 3, pp. 290–301, 1998.

[3] S. Mohajeri, A. Esteki, O. R. Zaïane, and D. Rafiei, "Innovative Navigation of Health Discussion Forums based on Relationship Extraction and Medical Ontologies," in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2013, pp. 13–14.

[4] M. Steinbach, G. Karypis, V. Kumar *et al.*, "A Comparison of Document Clustering Techniques," in *KDD Workshop on Text Mining*, vol. 400, no. 1, 2000, pp. 525–526.

[5] C.-E. Ben N'Cir and N. Essoussi, "Using Sequences of Words for Non-Disjoint Grouping of Documents," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 29, no. 03, p. 1550013, 2015.

[6] M. Hu, A. Sun, and E.-P. Lim, "Comments-Oriented Blog Summarization by Sentence Extraction," in *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM)*. ACM, 2007, pp. 901–904.

[7] Y.-C. Chang and S.-M. Chen, "A New Query Reweighting Method for Document Retrieval based on Genetic Algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 617–622, 2006.

[8] M. Agosti, F. Crivellari, and G. M. Di Nunzio, "Web Log Analysis: A Review of a Decade of Studies about Information Acquisition, Inspection and Interpretation of User Interaction," *Data Mining and Knowledge Discovery*, vol. 24, no. 3, pp. 663–696, 2012.

[9] R. Navigli and P. Velardi, "An Analysis of Ontology-Based Query Expansion Strategies," in *Proceedings of the European Conference on Machine Learning, Workshop on Adaptive Text Extraction and Mining*, 2003, pp. 42–49.

[10] Y. Song, D. Zhou, and L.-w. He, "Query Suggestion by Constructing Term-Transition Graphs," in *Proceedings of the ACM International Conference on Web Search and Data Mining*. ACM, 2012, pp. 353–362.

[11] P.-A. Chirita, C. S. Firan, and W. Nejdl, "Personalized Query Expansion for the Web," in *Proceedings of the ACM International SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2007, pp. 7–14.

[12] H.-L. Lee and H. A. Olson, "Hierarchical Navigation: An Exploration of Yahoo! Directories," *Knowledge Organization*, vol. 32, no. 1, pp. 10–24, 2005.

[13] Y. Hassan-Montero and V. Herrero-Solana, "Improving Tag-Clouds as Visual Information Retrieval Interfaces," in *International Conference on Multidisciplinary Information Sciences and Technologies*, 2006, pp. 25–28.

[14] B. L. Grand and M. Soto, "Visualisation of the Semantic Web: Topic Maps Visualisation," in *Information Visualisation, 2002. Proceedings. Sixth International Conference on*. IEEE, 2002, pp. 344–349.

[15] V. Nováček and G. A. Burns, "SKIMMR: Facilitating Knowledge Discovery in Life Sciences by Machine-Aided Skim Reading," *PeerJ*, vol. 2, p. e483, 2014.

[16] A. Sharma, "Dr. Sharma's Obesity Notes," http://www.drsharma.ca, March 2013.

[17] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, *DBpedia: A Nucleus for a Web of Open Data*. Springer, 2007.