Towards Human-in-the-Loop Based Query Rewriting for Exploring Datasets

Genoveva Vargas-Solar, Mehrdad Farokhnejad, Javier A Espinosa-Oviedo

Towards Human-in-the-Loop Based Query Rewriting for Exploring Datasets

Genoveva Vargas-Solar
CNRS, LIRIS-LAFMIA
Lyon, France
genoveva.vargas-solar@liris.cnrs.fr

Mehrdad Farokhnejad
Univ. Grenoble Alpes, Grenoble INP, CNRS, LIG
Grenoble, France
mehrdad.farokhnejad@univ-grenoble-alpes.fr

Javier A. Espinosa-Oviedo
Univ. Lyon 2, ERIC-LAFMIA
Lyon, France
javier.espinosa@acm.org

ABSTRACT

Data exploration promotes a new querying philosophy that gradually converges into queries that can be used to exploit raw data collections according to data explorers (i.e., users) expectations. Data exploration aims to guide the understanding of data collections with different rawness degree and define the type of questions that can be asked on top of them, often through interactive exploration processes. This paper introduces a human-guided data exploration approach defining exploration operations that result in different types of factual and analytic queries. Our first results include a proposal of query morphing and queries as answers strategies. This paper describes an experiment setting used for testing the data exploration techniques.

1 INTRODUCTION

The evolution in querying, information retrieval and human-computer interaction has led to the shift of interest from traditional Query-Response paradigm to actual human intelligence systems. Approaches such as interactive query expansion (IQE) [3, 9, 19] have shown the importance of data consumers in the data exploration process. Users' intention helps to navigate through the unknown data, formulate queries and find the desired information. In most of the occurrences, user feedback acts as vital relevance criteria for next query search iteration. Such novel requirements of modern exploration driven processes call to rethink data querying processes.

Traditional data management systems assume that when users ask a query (i) they have good knowledge of the schema, meaning and contents of the database, and (ii) they are sure that this particular query is the one they wanted to ask. In short, it is assumed that users know what they are looking for. In response, systems like DBMS, always try to produce correct and complete results [7]. These assumptions are becoming less true as the volume and diversity of data grow, and as raw datasets representing phenomena observations, rather than facts, need to be explored by data scientists and other users. First, the structure and content of the database are hard to understand. Second, finding the right question to ask is a long-running and complex task, often requiring a great deal of experimentation with queries, backtracking query results, and revision of results at various points in the process [18]. Existing systems have limited provisions to help the users to reformulate their queries as they evolve with the search progression [10].

This paper proposes a data exploration approach that

- defines a loop where, given a user intention expressed using terms and raw data collections, the exploration strategies propose different types of possible queries that can be asked on top of data, and that potentially correspond to user's expectations;
- interacts with the user for refining intentions based on the proposed queries and starts the loop again until the queries proposal converges with user expectations.

Different data exploration pipelines can be defined combining different exploration techniques for performing specific data exploration tasks.

The rest of the paper is organised as follows. Section 2 summarises the related work done in the field of data exploration proposing a classification of approaches and techniques. Section 3 provides a detailed description of the approach. Section 4 describes our experimental setting including the dataset and its pre-processing steps. Section 5 concludes the paper and discusses future work.

2 RELATED WORK

Data exploration calls for combining different exploration, querying and processing methods and strategies proposed in diverse domains. Therefore we performed a systematic review to classify them (see Figure 1). According to a systematic review we performed, we propose a classification of existing data exploration techniques and methods. The classification consists of facets representing an aspect of data exploration and dimensions that denote the concepts that define each facet. As shown in Figure 1, the facets classify: (F1) the type of queries addressed by existing work; (F2) the type of algorithms used for exploring data collections; (F3) the knowledge domain of data collections and data types; (F4) the exploration processes done with human intervention; and (F5) data exploration techniques and systems conceived for understanding raw datasets content.

Since exploration can put different types of queries in action, facet F1 classifies the types of queries that are defined and used in different works that exploit datasets. The spectrum goes from "classic" keyword and relational queries evaluated on top of more or less curated datasets, to data processing operations on raw datasets (e.g., descriptive statistics). In this spectrum, these types represent families of queries that can include aggregation, clustering operations. We mainly identify "query by example" techniques useful particularly in cases where the knowledge about the datasets' content is too weak (see d1.8). Query by example is an intuitive way to explore data, so many techniques are applying it to data exploration. Examples can either represent approaches like reverse engineering querying and queries like query morphing or queries. We also note that data exploration is a loop that obtains approximated results and the techniques

are specialised according to the type of data models (relational, graph, semi-structured, text, multimedia).

Depending on the domain, works propose algorithms rather than operators (like in relational contexts) to process datasets and to discover and derive a precise statistical understanding of their content (facet F2). Algorithms sometimes depend on the type of data structures used for representing data. For example, there are algorithms for processing graphs (centrality, pathfinding, etc.) or querying tables (selection, projection, etc.). Many works use well-known heuristics, data mining, machine learning, artificial intelligence algorithms for processing datasets, and insight into their content. Finally, other works propose their strategies without adhering to a specific domain.

The vision of data exploration in this work is that it should be a human-guided process. Therefore, we have studied techniques where humans intervene to adjust and guide the process of receiving information (d.4.5). We studied works on group recommendation, consensus functions, group preference and group disagreement. These study address objectives like designing consensus functions that aggregate individual group members' preferences to reflect the overall group's preference for each item [1, 4, 13] or disagreement about an item [16]. Consensus functions can be applied within a data exploration process given where a user can agree and disagree about the proposed queries; the system can recommend queries according to given constraints that can be interpreted as preferences.

According to our classification, facet F5 considers dimensions that represent exploration techniques. Regarding exploration query expression (d5.1), we have identified three types of approaches: multi-scale query processing for gradual exploration; query morphing to adjust for proximity results; queries as answers as query alternatives to cope with lack of providence. Results filtering (d5.2) addresses analysis and visualisation to give insight to data content. Finally, data exploration systems & environments (d5.3) are tailored for exploring data incrementally and adaptively.

Concerning data exploration techniques, M. L. Kersten et al. [13] have compiled five methods to explore data sets querying: one-minute DB kernels, multi-scale queries, result set post-processing and query morphing and queries as answers. These methods revisit fundamental characteristics of existing systems like the notion of results completeness and correctness promoted by traditional databases, splitting queries execution on different fragments of a database, precision of queries, and one-shot computation of query results. These query systems provide a broader (i.e., less precise but with a broader scope) approach, discarding exactness and completeness for speed and a more global vision of the data.

Finally, facet F3 classifies the type of datasets used to test different exploration techniques and approaches. Datasets content is often textual and with different rawness degree (newspapers, micro-texts from social networks) and already processed content using NLP (Natural Language Processing) techniques and represented as graphs or tables. Other datasets are built by collecting observations monitored using, for example, IoT infrastructures. These data sets contain records of measures or even video or images.

We have the following remarks about what we have studied in state of the art. Data exploration pipelines are mostly ad-hoc, implemented in an artisanal manner, and partially human-guided. Machine learning, analytics and querying techniques (e.g., query by example, queries as answers, etc.) are complementary. We observed that no existing system integrates them so that data scientists can develop exploration pipelines that can thoroughly understand data and its analytics potential. Therefore, there is room for proposing approaches for each of them, defining rules on how they can be combined within data exploration pipelines and integrating them to provide a whole data exploration environment.

3 QUERYING PIPELINES FOR EXPLORING DATASETS

Figure 2 shows our general approach based on query rewriting techniques and summarised as follows: "given an initial query, provide sets of queries that can help data consumers better exploit data collections". The approach considers that data collections are textual and indexed (not necessarily cleaned) and the representative vocabulary used in their content has been extracted and classified. For example in a crisis management scenario, the classes are events (e.g., someone looks for shelter, a building has been damaged) and actions (e.g., a hotel provides shelter for victims, people is approaching a damaged building to search victims).

The approach is intended to rewrite initial keyword queries by morphing expressions to produce results that can retrieve representative insights into these collections' content. The rewriting process is gradual and interactive, where the user expresses an initial expression, and the exploration process provides new queries associated with content samples that can give insight into the content of the dataset. The alternative queries are assessed and adjusted by the user. Then, the exploration process is triggered again until a set of queries is chosen to be evaluated to produce results. Results produced by different exploration strategies can also be used as input to others. For example, query morphing's output can be used as an input for the queries as answers.

The next sections describe two rewriting techniques query morphing and queries as answers (expansion) that we have proposed for exploring datasets.

3.1 Query morphing

Query morphing is the process of rewriting conjunctive and disjunctive keyword queries, by adding terms, to increase the possibility of exploring the most number of items in a collection. We proposed and implemented a "query morphing" pipeline that can help the data scientist better precise her query (see Figure 3). Our query morphing pipeline uses a vocabulary and Wordnet to look for associated terms and synonyms that help expand the terms to enhance the chance of matching it with relevant data items in the target collection. The pipeline is described as follows. Given a conjunctive and disjunctive keyword query represented as an expression tree go through the tree in depth-first until finding a leaf representing a term and then:

(1) Use a vocabulary representing the dataset content and wordnet seeking for:
   (a) equivalent terms and generate a node with the operator and then connect the initial term with the equivalent terms in a conjunctive expression subtree.
   (b) more general terms and connect the initial term with these terms in a disjunctive expression subtree.
   (c) assess and adjust the morphed query by the user and eventually restart the expansion process. The assessment process includes pondering the terms and exploring result samples to see potential results.

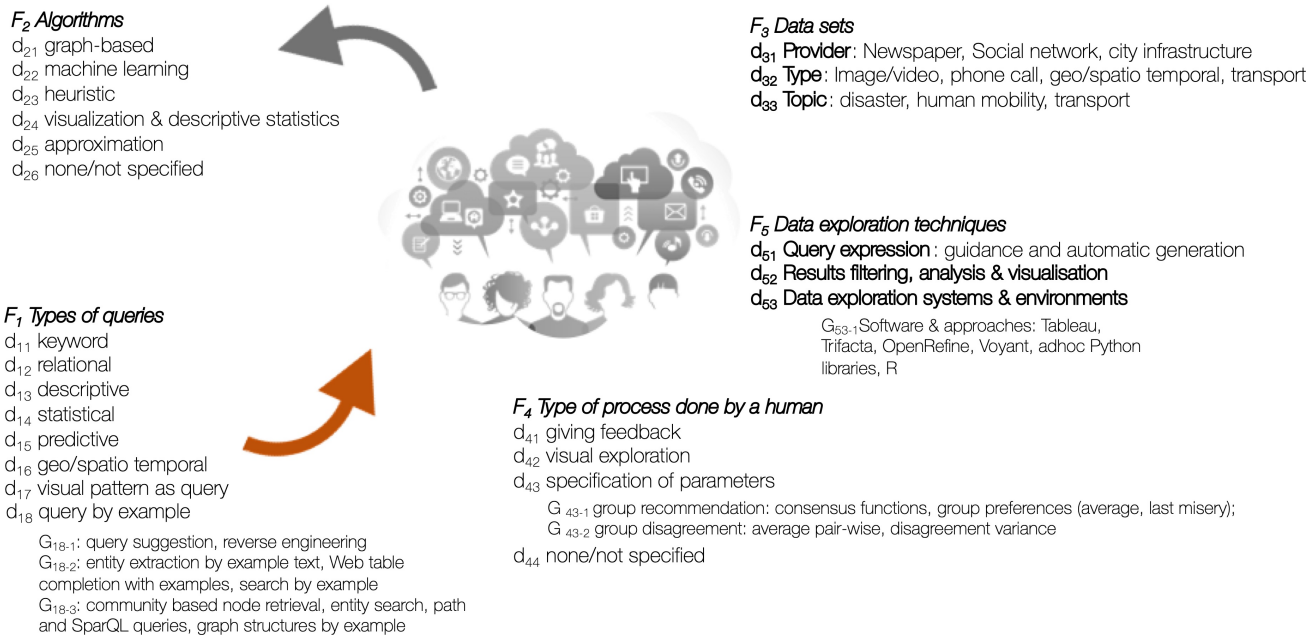

Figure 1: Querying techniques for exploring datasets

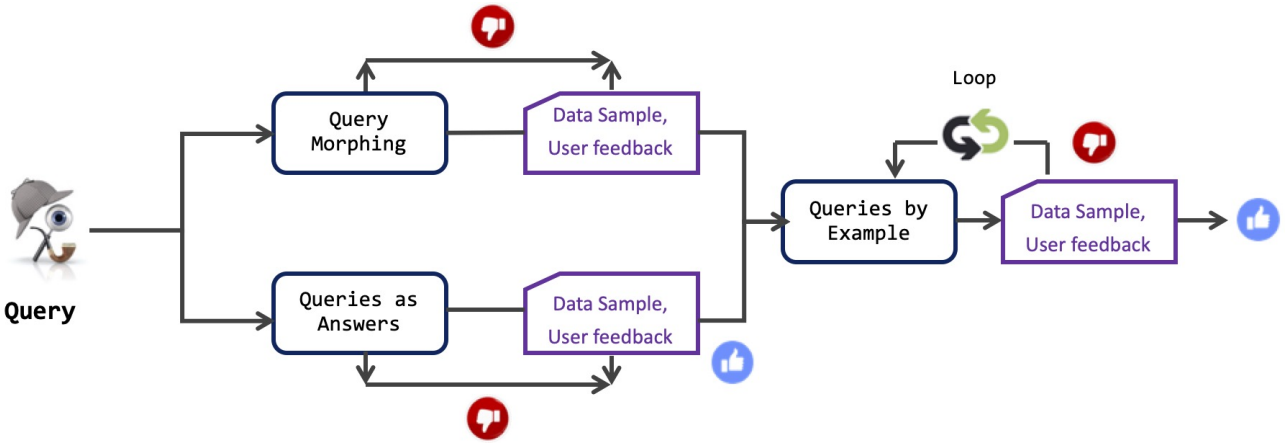

Figure 2: Deriving queries to explore data collections

For example, for a query "victims AND missing AND shelter", using Wordnet[1], the query can be expanded as follows: "(victim OR casualty OR "unfortunate person") AND (missing OR absent) AND (shelter OR protection OR housing)". The key is using a concept ontology or glossary and can find the maximum equivalent and general terms. In this example, we only found equivalent terms. The user can then mark which terms should or should not be included in the expanded query. She can also test different combinations of the query and compare the results to see which morphed query can produce the results that best respond to her expectations.

Once the new query expression has been rewritten, as done in information retrieval techniques, we use the inverted index to find the corresponding documents where the query terms occur. Then, we use the frequency matrix to compute the final result set tagged with precision and recall measures.

3.2 Query as answers

Given an initial conjunctive/disjunctive keyword query, the query is rewritten and transformed into several queries by extending it with general and more specific terms, synonyms, etc., and by exploiting the knowledge domain (see Figure 3). The result is a set of possible alternative queries with associated sample results so that the user can choose which ones to execute.

In our approach, the initial query is represented by an expression tree (intermediate representation) where nodes are conjunction and disjunction operators and leaves are terms. During the rewriting process, the tree is modified by adding new types of nodes and tagged arcs. New nodes represent “and” and “or” nodes that do not belong to the initial query and more general/specific terms associated with an “initial” term. These new nodes are connected with the nodes of the initial query by tagged arcs. A tag can indicate whether it connects a node with a conjunction or disjunction of more general/precise terms.

[1]http://wordnetweb.princeton.edu/perl/

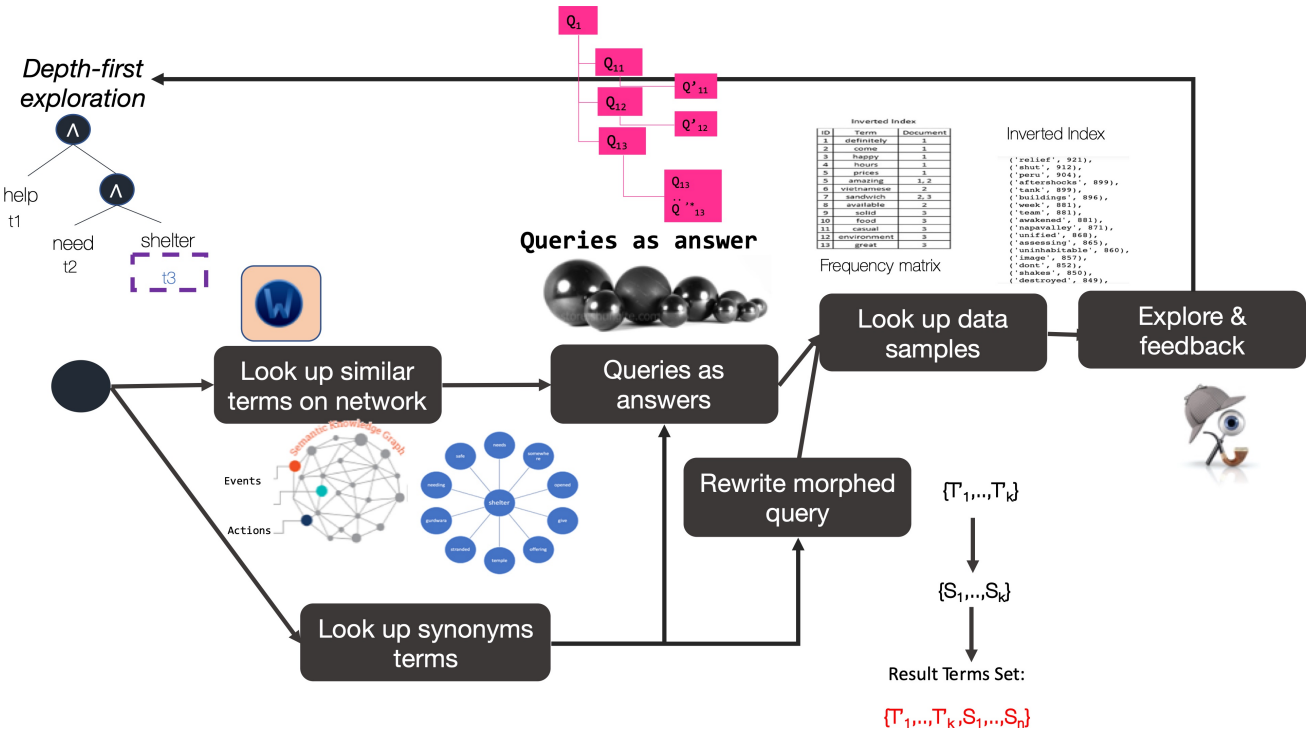

Figure 3: Query morphing as answers pipeline

For example, consider an initial query linking three terms t_1, t_2 and t_3 with a conjunctive and disjunctive initial query (see Figure 3). It is then re-written in a new query represented by a tree that extends the query with terms that can be synonyms or related terms to t_1 and t_2. Three possible queries are derived: Q_1^1 which provides an alternative to t_1 with a new complex query with a synonym/or an associated term t_4, saying that we can look for "t_1 or t_1 and t_4". Similarly, Q_2^1 provides an alternative to t_2 saying that we can either look for t_2 or t_2 and t_6, which could be a synonym or a related term. Finally, Q_3^1 is a complex query that integrated Q_1^1, Q_2^1 with t_3 of the initial query.

The following steps are performed for computing queries alternatives where every step aims at deriving the initial query into queries that add knowledge. For each leaf in the expression tree of the query:

(1) Use a vocabulary (extracted from the dataset content) and Wordnet seeking for:
- equivalent terms and generate a node with the operator and then connect the initial term with the equivalent terms in a conjunctive expression subtree;
- more general terms and connect the initial term with these terms in a disjunctive expression subtree.

(2) Use a frequency matrix for looking for terms that are often associated with the initial term with a specific frequency and getting a sample of documents that can belong to query results.

The user can choose those queries that best target her expectations. A history of queries is maintained that can be reused for suggesting or pre-calculating morphing or query as answers results or for adjusting the chosen query set with new queries as the dataset evolves.

4 EXPERIMENTS

To experiment our general approach, let us consider a disaster management scenario where various data collections are produced during the life cycle of the disaster and must be explored to organise relief, resilience and duty of memory actions. The scenario we use is related to disaster management under a horizontal organisation [2], where civilians take active action when an event happens (e.g., earthquake, flooding, fire) and continue to influence decision making during the other phases of its management. In this context, social media is a fast-paced channel used by affected people to describe their situation and observations, seek information, specify their requests, and offer their voluntary assistance; providing actionable information [17, 21]. Critical data is continuously posted on social media like Twitter, during the disaster life cycle (the event, relief, resilience, duty of memory).

During such life-threatening emergencies, affected and vulnerable people, humanitarian organisations, and other concerned authorities search for information useful to provide help and prevent a crisis. Nobody has control over the type of data exchanged by actors. These data are crucial in making critical decisions like saving lives, searching people, and providing shelter and medical assistance. For this reason, it is required to explore past and present in an agile manner to find hints to make decisions and act individually and collectively. Social network data collections can include reports on architectural and building environment damages and volunteers informing them that they have answered calls for help (see Figure 4).

The question is whether these data collections can help to find (i) causal correlations, for example, is it possible to know given a post asking for help whether actions have already been

[2] The phenomenon of organisation of civilians under horizontal and marginal groups has come up in different countries during rivers flooding, annual landslides in diverse regions particularly in Latin America.

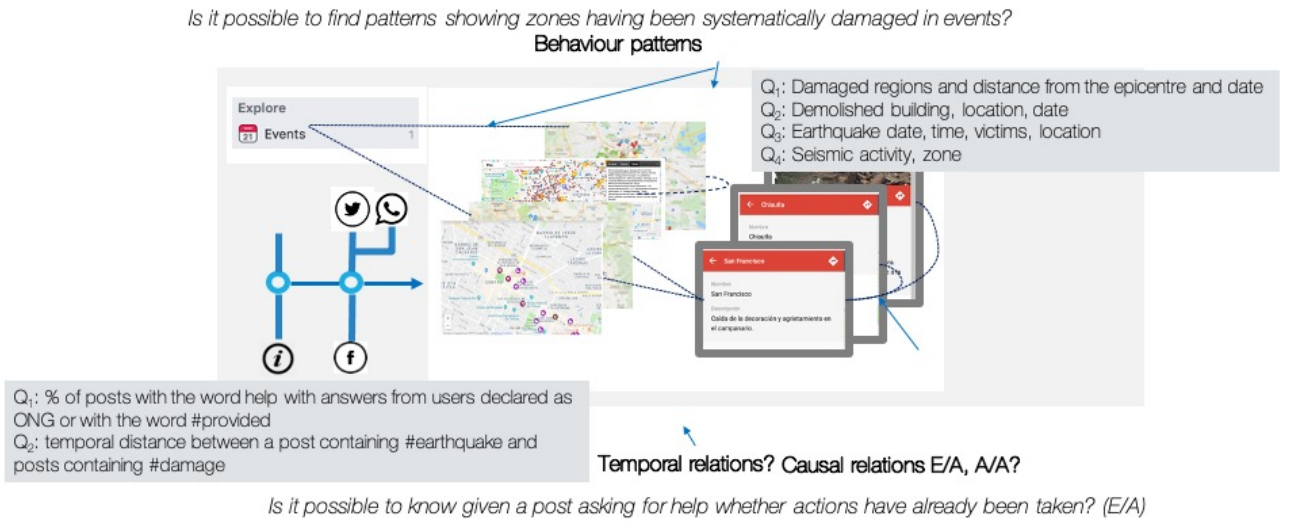

Figure 4: Exploring crisis social network posts during crisis

taken? since when and whether the problem has been solved; (ii) is it possible to find patterns showing which zones have been systematically damaged in other events? Is there more risk and help required in those regions?; (iii) Spatio-temporal relations, is it possible to figure out from the beginning of the event until a given subsequent time? have actors installed camps to provide first aid? Does help come from urban areas?; (iv) how to ask about the type of help still being required after a day of the event?

Note that these questions are not asking for results, they are asking for assistance on how to ask them on top of data to potentially best explore data. How can I express my query to expect to receive the best guidance to act? Is my query pertinent to be asked given the data I can have access to?

Data exploration techniques can help assist in expressing queries that can potentially explore data collections and be pertinent according to their content.

This section describes the experimental setting for the assessment of our approach. Our experiments deal with the crisis scenario introduced previously, and they use micro-texts datasets from Twitter concerning this topic. Given Twitter's 140 characters limit restriction, the frequency matrix cannot be useful, so we used the word2vec model to pre-process the dataset and then find similar terms for rewriting queries. In this work, we consider those words provided by word2vec model, and words are also indexed in the frequency matrix for extending query. With this information, we modify the tree adding "AND" and "OR" nodes, and thereby we create other possible queries that derive from the initial one.

We have experimented with generating the knowledge domain and then using it for validating morphing queries. Our experiment is based on the disaster management use case using Twitter posts as documents collections. The experiment applies text mining techniques to build the vocabulary and classify it into events produced and actions performed during a disaster life-cycle. In this section, we first describe the datasets we used in our experiments and then the experiment setting, including the algorithms used to process data collections and classify the extracted vocabulary.

4.1 Dataset preparation

Among social media studies, most of them focus on Twitter, mainly because of its timeliness and availability of information from a large user base. We use CrisisNLP [11] labelled and unlabelled datasets. The datasets contain approximately 5 million unlabelled and 50k labelled tweets. The size of this dataset is about 7 gigabytes. The datasets consist of various event types such as earthquakes, floods, typhoons, etc. The datasets were collected from the Twitter streaming API using different keywords and hashtags during the disaster. The tweets are labelled into various informative classes (e.g., urgent needs, donation offers, infrastructure damage, dead or injured people) and one not-related or irrelevant class. Table 5 shows a sample of some labelled tweets from data collection.

Data Preprocessing. Since the tweet texts are brief, informal, noisy, unstructured, and often contain misspellings and grammatical mistakes, preprocessing must be done before using them in further analysis. Moreover, due to Twitter's 140 characters limit restriction, Twitter users intentionally shorten words using abbreviations, acronyms, slang, and sometimes words without spaces; hence we need to normalise those OOV terms [11]. Besides, tweets frequently contain duplicates as the same information is often retweeted / re-posted by many users [20]. Presence of duplicates can result in an over-estimation of the performance of retrieval/extraction methodologies. Therefore, we eliminated duplicate tweets using 'remove duplicates toolkit' by Excel. Currently, we use 73562 unlabelled data set related to 2014. We performed the following preprocessing steps to clean the micro-documents:

(1) We removed stop words (e.g. 'a', 'at', 'here'), non-ASCII characters, punctuations (e.g. '.', '!'), URLs (e.g. ' http://t.co/24Db832o4U '), hashtags (e.g. '#Napaquake ') and Twitter reserved words (e.g. 'RT', 'via').
(2) We further tokenize the tweets using nltk.tokenize library [24].
(3) We performed stemming using the WordNet Lemmatizer library [24]: e.g. troubled (trouble).

category	tweet_text
infrastructure_and_utilities_damage	RT @sofiaorden: √Ç¬ª http://t.co/qPBRxnTMmA #Californiaearthquake 117 California earthquake: wine country clears up damage around Napa Califov¢,Ç¨¬¶
infrastructure_and_utilities_damage	RT @svqjournalist: I'm taking over coverage of the #napaquake from downtown. The damage looks striking in person. http://t.co/jHWeYdhHAI
injured_or_dead_people	#BREAKING New Injury Numbers 172 injured, 7 fractures, 1 critical #napaquake http://t.co/nVYLPewtFR
injured_or_dead_people	RT @stormchaser4850: UPDATE: 170+ people treated for injuries in aftermath of strong #California #earthquake http://t.co/hY1I7ToEur http://√¢,Ç¨¬¶
donation_needs_or_offers_or_volunteering_services	Full statement by Napa Valley Vintners on new #earthquake relief find, with link for making donations: http://t.co/KuWzMz5zpV via NapaRegis
donation_needs_or_offers_or_volunteering_services	RT @NapaCoRedCross: The @RedCross shelters in Napa and Vallejo remain open to assist those affected by the earthquake. #NapaQuake
sympathy_and_emotional_support	"ELCA offers prayers for those affected by California earthquake" via @ELCANews http://t.co/80awJvXtvj #UNYS

Figure 5: Examples of some labelled tweets, posted during the 2014 California

used a list of the crisis related OOVs [11] to normalize tweets' terms: e.g. govt (government), 2morrow (tomorrow), missin (missing).

(4) We removed duplicate tweets.

After performing the cleaning 126161 unlabelled data related to the 2014 California Earthquake, we obtained a set of 73562 tweets. This set is used for all experiments reported in this work.

The pipeline implemented to create the knowledge base required for experimenting our data exploration techniques consists of two steps: (i) indexing data collections content using information retrieval techniques; (ii) create a vocabulary using classification techniques.

Indexing the data collection. As a result of indexing the cleaned tweets collection, we created an inverted index and a frequency matrix representing the content of the collection. We implemented an inverted index to provide agile access to a document's position in which a term appears. The inverted index is used as a dictionary that associates each word with a list of document identifiers where the word appears. This structure prevents making the running time of token comparisons quadratically. So, instead of comparing, record by record, each token to every other token to see if they match, the inverted indices are used to look up records that match a particular token.

Currently, we use 73562 unlabelled data set related to the 2014 California Earthquake. We generated an inverted index consisting of 20313 rows. The rows correspond to terms in our raw data collection, and columns correspond to documents where the terms occur. The inverted index allows a fast full-text search. It can help to explore queries' terms to find the documents where the terms occur.

A term frequency matrix is a mathematical matrix that describes the frequency of terms that occur in a collection of documents [14]. The matrix contains 73562 columns, where each column corresponds to a document (tweet) and each row to a term. A cell in the matrix contains the number of times that the term appears in the document. The top 20 most frequent terms in our data collection can help us expand the query using data collection.

Creating a vocabulary. We implemented a classification pipeline to build a vocabulary of events and actions related to disasters, thereby generating a knowledge base describing the tweets' data collections used for our experiment. The pipeline combines machine learning methods reproducing an existing work proposed by [11, 15].

We applied supervised techniques such as Random Forest [5], Support Vector Machines (SVM) and Convolutional Neural Network (CNN) [6, 8] to classify the tweets of our experiment dataset to build the vocabulary of events and actions. As the word "Event" suggests, we considered tweets containing a subject related to any occurrence during or after the crisis. For example- damage happened to a building, or people are trapped in buildings. For an "Action" we considered those tweets that focus on operations and activities during or after the crisis. Such as government or NGOs providing help to the affected people.

We performed a set of experiments on California and Nepal earthquake datasets consisting of approximately 3032 labelled tweets, out of which 2203 tweets of Nepal and 829 tweets of California dataset. The datasets are divided into two sets. As usual in machine learning techniques, we divided the data collection into training and test datasets. The first set comprised of 70% of the messages (i.e. training set) and the second comprised of 30% of the messages (i.e. test set). We trained all three different kinds of classifiers using the preprocessed data.

We used multilayer perceptron with a CNN. We conducted experiments on the same dataset and eventually established that CNN outperformed the task with an adequate margin compared to our previous work.

For the evaluation of the trained models, we compared the results to [11, 15]. The results obtained by CNN model are better than traditional techniques, and we were able to obtain the same results as the original paper [11, 15] (see Table 1).

Table 1: Accuracy, Precision, Recall and f-score of CNN model with respect to California Earthquake and Nepal Earthquake crisis tweet data.

Datasets/SYS	Accuracy	Precision	Recall	f-Score
California Earthquake	**92.72**	**86.53**	**90.00**	**88.23**
Nepal Earthquake[12]	89.31	91.25	91.87	91.85

4.2 Testing query morphing

We implemented the "query morphing" process that we proposed to help the data scientist better precise her query, or define several queries representing what she is looking for. Our query morphing algorithm uses Wordnet to look for associated terms and synonyms that help expand the terms to enhance the chance of matching it with relevant tweets in the target collection.

For assessing expanded term quality, we have compared the performance of our proposed classification based query expanding method against the traditional query expanding method. We calculated the mean average of Cosine Similarity (MACS) between the query and expanded query terms to assess the proposed approach's performance. The experimental results show that the expanded query terms, obtained from the classified query expansion model, are more similar and relevant than the non-classification model.

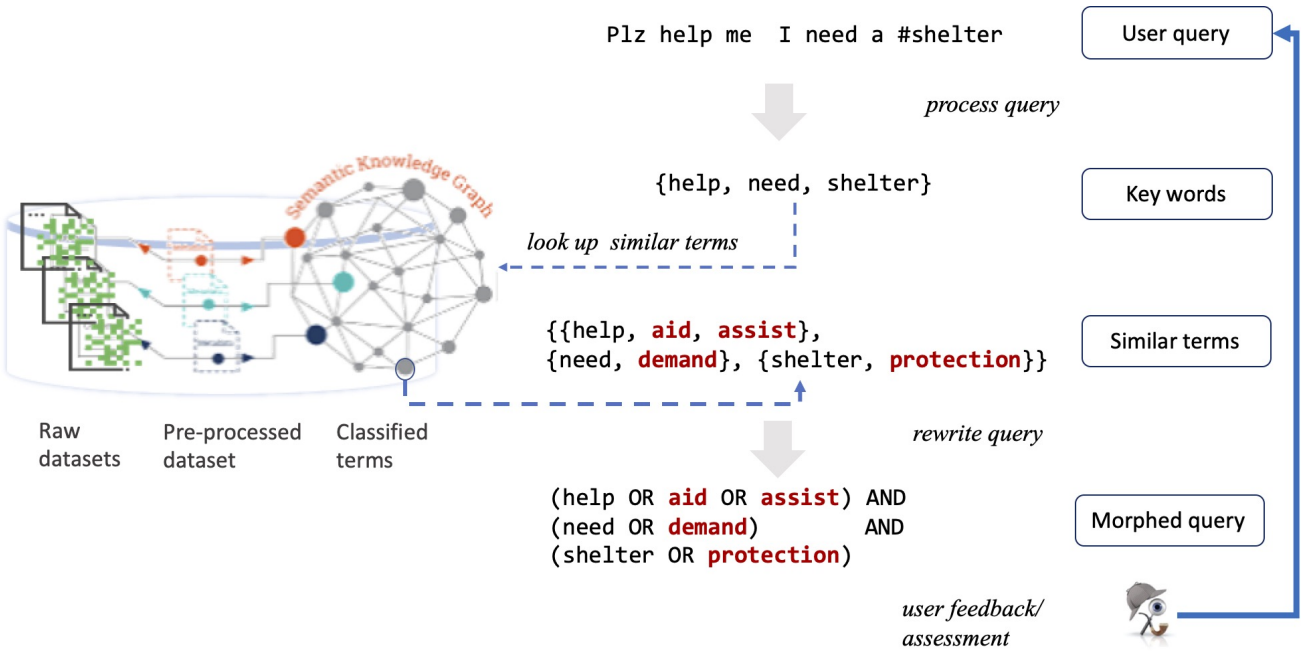

Figure 6: Query morphing example

In this section, we presented an ablation study about the performance of our proposed classification based query morphing method. We use the available crisisNLP pre-trained word embedding via word2vec method [11] to obtain query and expansion terms vectors. In the vector space model, all queries and terms are represented as vectors in dimensional space 300. Documents similarity is determined by computing the similarity of their content vector. To obtain a query vector, we represent keywords in user queries as vectors, and then sum all the keyword vectors followed by averaging them. For our analysis, we calculated the average similarity between the query vector (Q_vector) and 'm' keyword vectors obtained for a given query (T_vector) by using the formula of similarity *Sim* given in equation 1.

$$Sim(CTs, Query) = \frac{\sum_{i=1}^{m}(Cosine(Q\_vector, T\_vector[i]))}{m} \quad (1)$$

where CTs are candidate terms, 'm' is a hyper-parameter in query expansion-based retrieval, which shows the number of expansion terms (ET), using as reference the studies[2, 22]. We set the number of expansion terms to 10, 20 and 30 (ET@10, ET@20, ET@30). We repeat this task for 100 queries and report the mean of average of each **ET@** set in table 2. The experimental results show that the morphed query expanded with new terms obtained from the classified query morphing model are more similar and relevant than the non-classification model. The ET@10, ET@20 and ET@30 scores of our proposed classification model surpassed the transition non-classification based model. Also, we observe that when we set the number of expansion terms to 10, we achieve the best performance.

Currently, we used pseudo relevance feedback. This method automates the manual part of relevance feedback. It is assumed that the user takes top-m ranked morphed query terms returned by the initial query as relevant to expand her query. Results scoring must be completed with user feedback that finally guides the process. We have proposed a solution for exploring scientific papers through an experiment defining a set of exploration queries. Results were assessed by scientists of the National Institute of Genetic Engineering and Biotechnology, Tehran, Iran and Golestan University of Medical Sciences. Scientists provided feedback about exploration operations through questionnaires that are processed for obtaining satisfaction metrics. We are currently defining a crowd-based setting for obtaining feedback in the case of crisis datasets. The idea is to work with different groups of users (victims, volunteers, logistics decision-makers, police, medical staff) and queries to assess exploration results.

Table 2: The mean average of Cosine Similarity (MACS) between query and morphed query terms with and without classification model.

Query Expansion Model	ET@10	ET@20	ET@30
Classification	**0.420**	**0.377**	**0.371**
Non-classification	0.401	0.366	0.369

5 CONCLUSION AND FUTURE WORK

This paper introduced a general datasets exploration approach that includes human in the loop. The current approach includes two exploration techniques (i.e., query morphing and queries as answers) to help define queries that can fully explore and exploit a dataset. They are complementary query rewriting techniques where initially expanding a query can help adjust the terms used for exploring a dataset and then produce possible combinations of terms with possible queries that can lead to different scopes. In both cases, the user finally chooses a set of representative queries to her interests and the produced results that target her expectations. We have tested query morphing in the case of crisis dataset exploration, where people involved in a critical event either as victim or volunteers can define queries for retrieving information to look for or provide help.

Our future work includes modelling query exploration pipelines that can combine different techniques for exploring data collections. We will also propose ways of morphing and giving queries as answers where queries can be analytical or imply quantitative data views.

REFERENCES

[1] Sihem Amer-Yahia, Senjuti Basu Roy, Ashish Chawlat, Gautam Das, and Cong Yu. 2009. Group recommendation: Semantics and efficiency. *Proceedings of the VLDB Endowment* 2, 1 (2009), 754–765.

[2] Hiteshwar Kumar Azad and Akshay Deepak. 2019. Query expansion techniques for information retrieval: a survey. *Information Processing & Management* 56, 5 (2019), 1698–1735.

[3] Nicholas J Belkin. 2008. Some (what) grand challenges for information retrieval. In *ACM SIGIR Forum*, Vol. 42. ACM New York, NY, USA, 47–54.

[4] Ludovico Boratto, Salvatore Carta, Alessandro Chessa, Maurizio Agelli, and M Laura Clemente. 2009. Group recommendation with automatic identification of users communities. In *2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, Vol. 3. IEEE, 547–550.

[5] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.

[6] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20, 3 (1995), 273–297.

[7] Kyriaki Dimitriadou, Olga Papaemmanouil, and Yanlei Diao. 2016. AIDE: an active learning-based approach for interactive data exploration. *IEEE Transactions on Knowledge and Data Engineering* 28, 11 (2016), 2842–2856.

[8] Kunihiko Fukushima and Sei Miyake. 1982. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*. Springer, 267–285.

[9] Parantapa Goswami, Eric Gaussier, and Massih-Reza Amini. 2017. Exploring the space of information retrieval term scoring functions. *Information Processing & Management* 53, 2 (2017), 454–472.

[10] Stratos Idreos, Olga Papaemmanouil, and Surajit Chaudhuri. 2015. Overview of data exploration techniques. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. 277–281.

[11] Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a lifeline: Human-annotated twitter corpora for NLP of crisis-related messages. *arXiv preprint arXiv:1605.05894* (2016).

[12] Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a Lifeline: Human-annotated Twitter Corpora for NLP of Crisis-related Messages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)* (23-28). European Language Resources Association (ELRA), Paris, France.

[13] Martin L Kersten, Stratos Idreos, Stefan Manegold, and Erietta Liarou. 2011. The researcher's guide to the data deluge: Querying a scientific database in just a few seconds. *Proceedings of the VLDB Endowment* 4, 12 (2011), 1474–1477.

[14] Hans Peter Luhn. 1957. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of research and development* 1, 4 (1957), 309–317.

[15] Dat Tien Nguyen, Kamela Ali Al Mannai, Shafiq Joty, Hassan Sajjad, Muhammad Imran, and Prasenjit Mitra. 2016. Rapid classification of crisis-related data on social networks using convolutional neural networks. *arXiv preprint arXiv:1608.03902* (2016).

[16] Mark O'Connor, Dan Cosley, Joseph A Konstan, and John Riedl. 2001. PolyLens: a recommender system for groups of users. In *ECSCW 2001*. Springer, 199–218.

[17] Leysia Palen and Sarah Vieweg. 2008. The emergence of online widescale interaction in unexpected events: assistance, alliance & retreat. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*. 117–126.

[18] Olga Papaemmanouil, Yanlei Diao, Kyriaki Dimitriadou, and Liping Peng. 2016. Interactive Data Exploration via Machine Learning Models. *IEEE Data Eng. Bull.* 39, 4 (2016), 38–49.

[19] Ian Ruthven. 2003. Re-examining the potential effectiveness of interactive query expansion. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. 213–220.

[20] Ke Tao, Fabian Abel, Claudia Hauff, Geert-Jan Houben, and Ujwal Gadiraju. 2013. Groundhog day: near-duplicate detection on twitter. In *Proceedings of the 22nd international conference on World Wide Web*. 1273–1284.

[21] Sarah Vieweg, Amanda L Hughes, Kate Starbird, and Leysia Palen. 2010. Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 1079–1088.

[22] Chengxiang Zhai and John Lafferty. 2001. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the tenth international conference on Information and knowledge management*. 403–410.