

# Journal of Visualized Experiments

## Cloud based phrase mining and analysis of user-defined phrase-category association in biomedical publications --Manuscript Draft--

Article Type:	Methods Article - JoVE Produced Video
Manuscript Number:	JoVE59108R1
Full Title:	Cloud based phrase mining and analysis of user-defined phrase-category association in biomedical publications
Keywords:	text mining, data science, medical informatics, phrase mining, cloud computing
Corresponding Author:	peipei Ping, PhD University of California Los Angeles Los Angeles, California UNITED STATES
Corresponding Author's Institution:	University of California Los Angeles
Corresponding Author E-Mail:	ppingucla@gmail.com
Order of Authors:	Dibakar Sigdel Vincent Kyi Aiden Zhang Shaun Setty David A. Liem Yu Shi Xuan Wang Jiaming Shen Wei Wang JiaWei Han peipei Ping, PhD
Additional Information:	
Question	Response
Please indicate whether this article will be Standard Access or Open Access.	Open Access (US\$4,200)
Please indicate the <b>city, state/province, and country</b> where this article will be <b>filmed</b> . Please do not use abbreviations.	Los Angeles, California, US



Peipei Ping, PhD, FAHA, FISHR  
 Professor of Physiology, Medicine/Cardiology & Bioinformatics  
 Director of NIH BD2K Center of Excellence for Biomedical Computing at UCLA  
 Director of NHLBI Integrated Data Science Training Program in  
 Cardiovascular Medicine (iDISCOVER) at UCLA  
 David Geffen School of Medicine at UCLA  
 Los Angeles, CA 90095  
 Tel: 310-267-5624 (Lab)  
 Fax: 310-267-5623 (Lab)  
 Email: ppingucla@gmail.com

The Editorial Board of JoVE,  
 1 Alewife Center, Suite 200  
 Cambridge, MA 02140  
 United States

December 3<sup>rd</sup>, 2018

To the Editorial Board of JoVE,

We sincerely thank you and the reviewers for your support; your constructive comments have elevated the overall quality and impact of the manuscript. We have carefully addressed all comments from the editors and reviewers in an itemized fashion (see our responses to reviewer comments, 11 pages total). Specifically, we have created several new subsections in the 'Discussion'; revised our protocols and tables, created a GitHub repository for downloadable codes, and added ten new references. We hope our responses have clarified all elements and satisfied the reviewers' comments. Please see a summary of our response in the list below:

- In response to Reviewer #1:
  - JoVE being a method journal, we clarified that our focus in this manuscript is to provide methods to establish a cloud computing platform for the phrase mining and analyses.
- In response to Reviewer #2:
  - We created a GitHub repository and placed all download program codes and command files as suggested by the reviewer. We also provided all required steps to implement our algorithm in their cloud-based platform.
  - We provided the 'general applicability', 'comparison with other algorithms', 'limitation of the algorithm' and 'future application' with references in the 'Discussion' section.
  - We revised the sample result and presented with statistical error and confidence interval.
  - We provided an interface in 'config' repository for the user to control the process.
  - For debugging, log messages are printed out in the log files under the 'log' directory. We provided the required steps in the protocol for debugging.
- In response to Reviewer #3:
  - We fixed the broken data file.
  - We provided a section in 'Discussion' about the future scope of our algorithm.
- In response to Reviewer #4:

- We clarified user-defined entity-category selection. Our protocol now includes all required instructions to prepare user-defined entities and categories.
- Detail about CaseOLAP score calculation has been provided as the response to the comment. Accordingly, we revised our 'Introduction' section.
- We added a new section in the 'Discussion' section to address the comparison of our algorithm with other similar algorithms in Text Mining with references.

We are deeply grateful for this opportunity to revise our manuscript according to the standards of *JoVE* and contribute on methods to a cloud computing platform for biomedical communities. We hope the manuscript is now ready for publication.

Sincerely yours,

A handwritten signature in blue ink, appearing to be 'PP' or 'Ping', written in a cursive style.

Peipei Ping, PhD, FISHR, FAHA  
Corresponding Author  
UCLA School of Medicine

**TITLE:**

Cloud-Based Phrase Mining and Analysis of User-Defined Phrase-Category Association in Biomedical Publications

**AUTHORS AND AFFILIATIONS:**

Dibakar Sigdel<sup>\*1,2</sup>, Vincent Kyi<sup>\*1,2</sup>, Aiden Zhang<sup>\*1</sup>, Shaun P. Setty<sup>3</sup>, David A. Liem<sup>1,2,4</sup>, Yu Shi<sup>5</sup>, Xuan Wang<sup>5</sup>, Jiaming Shen<sup>5</sup>, Wei Wang<sup>1,6,7</sup>, JiaWei Han<sup>5</sup>, Peipei Ping<sup>1,2,4,6</sup>

<sup>1</sup>The NIH BD2K Center of Excellence in Biomedical Computing, University of California, Los Angeles, Los Angeles, CA, USA

<sup>2</sup>Department of Physiology, University of California, Los Angeles, Los Angeles, CA, USA

<sup>3</sup>Department of Pediatric and Adult Congenital Heart Surgery, Miller Children's and Women's Hospital and Long Beach Memorial Hospital, Long Beach, CA, USA

<sup>4</sup>Department of Medicine/Cardiology, University of California, Los Angeles, Los Angeles, CA, USA

<sup>5</sup>NIH BD2K Program Centers of Excellence for Big Data Computing -- KnowEng Center, Department of Computer Science, University of Illinois at Urbana-Champaign (UIUC), Champaign, IL

<sup>6</sup>Scalable Analytics Institute (ScAi), University of California, Los Angeles, Los Angeles, CA, USA

<sup>7</sup>Department of Computer Science, University of California, Los Angeles, Los Angeles, CA, USA

\* These authors contributed equally.

Corresponding Author:

Peipei Ping  
[ppingucla@gmail.com](mailto:ppingucla@gmail.com)

Email Addresses of Co-authors:

Dibakar Sigdel ([dsigdel@mednet.ucla.edu](mailto:dsigdel@mednet.ucla.edu))

Vincent Kyi ([vincekyi@gmail.com](mailto:vincekyi@gmail.com))

Aiden Zhang ([aidencz8@gmail.com](mailto:aidencz8@gmail.com))

Shaun P. Setty ([ssetty@memorialcare.org](mailto:ssetty@memorialcare.org))

David A. Liem ([DLiem@mednet.ucla.edu](mailto:DLiem@mednet.ucla.edu))

Yu Shi ([yushi2@illinois.edu](mailto:yushi2@illinois.edu))

Jiaming Shen ([js2@illinois.edu](mailto:js2@illinois.edu))

Xuan Wang ([xwang174@illinois.edu](mailto:xwang174@illinois.edu))

Wei Wang ([weiwang@cs.ucla.edu](mailto:weiwang@cs.ucla.edu))

JiaWei Han ([hanj@illinois.edu](mailto:hanj@illinois.edu))

**KEYWORDS:**

text mining, data science, medical informatics, phrase mining, cloud computing

**SUMMARY:**

We present a protocol and associated programming code as well as metadata samples to

support a cloud-based automated identification of phrases-category association representing unique concepts in user selected knowledge domain in biomedical literature. The phrase-category association quantified by this protocol can facilitate in depth analysis in the selected knowledge domain.

#### **ABSTRACT:**

The rapid accumulation of biomedical textual data has far exceeded the human capacity of manual curation and analysis, necessitating novel text-mining tools to extract biological insights from large volumes of scientific reports. The Context-aware Semantic Online Analytical Processing (CaseOLAP) pipeline, developed in 2016, successfully quantifies user-defined phrase-category relationships through the analysis of textual data. CaseOLAP has many biomedical applications.

We have developed a protocol for a cloud-based environment supporting the end-to-end phrase-mining and analyses platform. Our protocol includes data preprocessing (e.g., downloading, extraction, and parsing text documents), indexing and searching with Elasticsearch, creating a functional document structure called Text-Cube, and quantifying phrase-category relationships using the core CaseOLAP algorithm.

Our data preprocessing generates key-value mappings for all documents involved. The preprocessed data is indexed to carry out a search of documents including entities, which further facilitates the Text-Cube creation and CaseOLAP score calculation. The obtained raw CaseOLAP scores are interpreted using a series of integrative analyses, including dimensionality reduction, clustering, temporal, and geographical analyses. Additionally, the CaseOLAP scores are used to create a graphical database, which enables semantic mapping of the documents.

CaseOLAP defines phrase-category relationships in an accurate (identifies relationships), consistent (highly reproducible), and efficient manner (processes 100,000 words/sec). Following this protocol, users can access a cloud-computing environment to support their own configurations and applications of CaseOLAP. This platform offers enhanced accessibility and empowers the biomedical community with phrase-mining tools for widespread biomedical research applications.

#### **INTRODUCTION**

Manual evaluation of millions of text files for the study of phrase-category association (e.g., Age group to protein association) is incomparable with the efficiency provided by an automated computational method. We want to introduce the cloud-based Context-aware Semantic Online Analytical Processing (CaseOLAP) platform as a phrase-mining method for automated computation of phrase-category association in the biomedical context.

The CaseOLAP platform, which was first defined in 2016<sup>1</sup>, is very efficient compared to the traditional methods of data management and computation because of its functional document management called Text-Cube<sup>2,3,4</sup>, which distributes the documents while maintaining underlying hierarchy and neighbourhoods. It has been applied in biomedical research<sup>5</sup> to study

entity-category association. The CaseOLAP platform consists of six major steps including download and extraction of data, parsing, indexing, Text-Cube creation, entity count, and CaseOLAP score calculation; which is the main focus of the protocol (**Figure 1, Figure 2, Table 1**).

To implement the CaseOLAP algorithm, the user sets up categories of interest (e.g., disease, signs and symptoms, age groups, diagnosis) and entities of interest (e.g., proteins, drugs). One example of a category included in this article is the 'Age Groups', which has 'infant', 'child', 'adolescent', and 'adult' subcategories as cells of the Text-Cube and protein names (synonyms) and abbreviations as entities. Medical Subject Headings (MeSH) are implemented to retrieve publications corresponding to the defined categories (**Table 2**). MeSH descriptors are organized in a hierarchical tree structure to permit search for publications at varying levels of specificity (a sample shown in **Figure 3**). The CaseOLAP platform utilizes the data indexing and search functionality for curation of the documents associated with an entity which further facilitate document to entity count mapping and CaseOLAP score calculation.

The details of the CaseOLAP score calculation is available in previous publications<sup>1-5</sup>. This score is computed using specific ranking criteria based on underlying Text-Cube document structure. The final score is the product of *Integrity*, *Popularity*, and *Distinctiveness*. *Integrity* describes whether a representative entity is an integral semantic unit that collectively refers to a meaningful concept. The *integrity* of the user-defined phrase is taken to be 1.0 because it stands as a standard phrase in the literature. *Distinctiveness* represents the relative relevance of a phrase in one subset of documents compared to the rest of the other cells. It first calculates the relevance of an entity to a specific cell by comparing the occurrence of the protein name in the target data set and provides a normalized *Distinctiveness* score. *Popularity* represents the fact that phrase with a higher *popularity* score appears more frequently in one subset of documents. Rare protein names in a cell are ranked low, while an increase in their frequency of mention has a diminishing return due to the implementation of the logarithmic function of frequency. Quantitatively measuring these three concepts depends on the (1) term frequency of the entity over a cell and across the cells and (2) number of documents having that entity (document frequency) within the cell and across the cells.

We have studied two representative scenarios using a PubMed dataset and our algorithm. We are interested in how mitochondrial proteins are associated with two unique categories of MeSH descriptors; "Age Groups" and "Nutritional and Metabolic Diseases". Specifically, we retrieved 15,728,250 publications from 20 years publications collected by PubMed (1998 to 2018), among them, 8,123,458 unique abstracts have had full MeSH descriptors. Accordingly, 1,842 human mitochondrial protein names (including abbreviations and synonyms), acquired from UniProt (uniprot.org) as well as from MitoCarta2.0 (www.broadinstitute.org) and MitoMiner4.0 (<http://mitominer.mrc-mbu.cam.ac.uk/release-4.0/begin.do>), are systematically examined. Their associations with these 8,899,019 publications and entities were studied using our protocol; we constructed a Text-Cube and calculated the respective CaseOLAP scores.

## PROTOCOL:

NOTE: We have developed this protocol based on the Python programming language. To run this program, have Anaconda Python and Git pre-installed on the device. The commands provided in this protocol are based on Unix environment. This protocol provides the detail of downloading data from PubMed (MEDLINE) database, parsing the data, and setting up a cloud computing platform for the phrase mining and quantification of user-defined entity-category association.

## 1. Getting code and python environment setup

1.1. Download or clone the code repository from Github (<https://github.com/CaseOLAP/caseolap>) or by typing '*git clone https://github.com/CaseOLAP/caseolap.git*' in the terminal window.

1.2. Navigate to the 'caseolap' directory. This is the root directory of the project. Within this directory, the 'data' directory will be populated with multiple data sets as you progress through these steps in the protocol. The 'input' directory is for user-provided data. The 'log' directory has log files for troubleshooting purposes. The 'result' directory is where the final results will be stored.

1.3. Using the terminal window, go to the directory where you cloned our GitHub repository. Create the CaseOLAP environment using the '*environment.yml*' file by typing '*conda env create -f environment.yaml*' in the terminal. Then activate the environment by typing '*source activate caseolap*' in the terminal.

## 2. Downloading documents

2.1. Make sure that the FTP address in '*ftp\_configuration.json*' in the config directory is the same as the Annual Baseline or Daily Update Files link address, found in the link ([https://www.nlm.nih.gov/databases/download/pubmed\\_medline.html](https://www.nlm.nih.gov/databases/download/pubmed_medline.html)).

2.2. To download baseline only or update files only, set 'true' in the '*download\_config.json*' file in the 'config' directory. By default, it downloads and extracts both baseline and update files. A sample of extracted XML data can be viewed at (<https://github.com/CaseOLAP/caseolap-pipelines/blob/master/data/extracted-data-sample.xml>)

2.3. Type '*python run\_download.py*' in the terminal window to download abstracts from the Pubmed database. This will create a directory called 'ftp.ncbi.nlm.nih.gov' in the current directory. This process checks the integrity of the downloaded data and extracts it to the target directory.

2.4. Go to the 'log' directory to read the log messages in '*download\_log.txt*' in case the download process fails. If the process is completed successfully, the debugging messages of the

download process will be printed out in this log file.

2.5. When the download is complete, navigate through 'ftp.ncbi.nlm.nih.gov' to make sure that there is 'updatefiles' or 'basefiles' or both directories based on download configuration in 'download\_config.json'. The file statistics become available at 'filestat.txt' in the 'data' directory.

### 3. Parsing documents

3.1. Make sure that downloaded and extracted data is available at 'ftp.ncbi.nlm.nih.gov' directory from step 2. This directory is the input data directory in this step.

3.2. To modify the data-parsing schema, select parameters in 'parsing\_config.json' file in the 'config' directory by setting their value to 'true'. By default, it parses the *PMID*, *authors*, *abstract*, *MeSH*, *location*, *journal*, *publication date*.

3.3. Type 'python run\_parsing.py' in the terminal to parse the documents from downloaded (or extracted) files. This step parses all downloaded XML files and creates a python dictionary for each document with keys (e.g., *PMID*, *authors*, *abstract*, *MeSH* of the file based on parsing schema setup at step 3.2).

3.4. Once data parsing is completed, make sure that parsed data is saved in the file called 'pubmed.json' in the data directory. A sample of parsed data is available at **Figure 3**.

3.5. Go to the 'log' directory to read the log messages in 'parsing\_log.txt' in case the parsing process fails. If the process is completed successfully, the debugging messages will be printed out in the log file.

### 4. Mesh to PMID mapping

4.1. Make sure that parsed data ('pubmed.json') is available at the 'data' directory.

4.2. Type 'python run\_mesh2pmid.py' in the terminal to perform MeSH to PMID mapping. This creates a mapping table where each of the MeSH collects associated PMIDs. A Single PMID may fall under the multiple MeSH terms.

4.3. Once the mapping is completed, make sure that there is 'mesh2pmid.json' in the data directory. A sample of the top 20 mapping statistics is available in **Table-2**, **Figures 4** and **5**.

4.4. Go to the 'log' directory to read the log messages in 'mesh2pmid\_mapping\_log.txt' in case this process fails. If the process is completed successfully, the debugging messages of the mapping will be printed out in this log file.



## 5. Document indexing

- 5.1. Download the Elasticsearch application from <https://www.elastic.co>. Currently, the download is available at (<https://www.elastic.co/downloads/elasticsearch>). To download the software in the remote cloud, type `wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-x.x.x.tar.gz` in the terminal. Make sure that 'x.x.x' in the above command is replaced by proper version number.
- 5.2. Make sure that downloaded '*elasticsearch-x.x.x.tar.gz*' file appears in the root directory then extract the files by typing '*tar xvzf elasticsearch-x.x.x.tar.gz*' in the terminal window.
- 5.3. Open a new terminal and go to the ElasticSearch bin directory by typing '*cd Elasticsearch/bin*' in the terminal from the root directory.
- 5.4. Start the Elasticsearch server by typing '*./Elasticsearch*' in the terminal window. Make sure that the server is started without error messages. In case of error on starting Elasticsearch server, follow the instructions at (<https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html>).
- 5.5. Modify the contents in the '*index\_init\_config.json*' in 'config' directory to set index initiation. By default, it will select all items present.
- 5.6. Type '*python run\_index\_init.py*' in the terminal to initiate an index-database in the Elasticsearch server. This initializes the index with a set of criteria known as index information (e.g., *index name*, *type name*, *number of shards*, *number of replicas*). You will see the message mentioning index is successfully created.
- 5.7. Select the items in the '*index\_populate\_config.json*' in the 'config' directory by setting their value to 'true'. By default, it will select all items present.
- 5.8. Make sure that parsed data ('pubmed.json') is present in the 'data' directory.
- 5.9. Type '*python run\_index\_populate.py*' in the terminal to populate the index by creating bulk data with two components. A first component is a dictionary with metadata information on the *index name*, *type name*, and *bulk id* (e.g., '*PMID*'). A second component is a data dictionary containing all the information on the tags (e.g., '*title*', '*abstract*', '*MeSH*').
- 5.10. Go to the 'log' directory to read the log messages in '*indexing\_log.txt*' in case this process fails. If the process is completed successfully, the debugging messages of the indexing will be printed out in the log file.

## 6. Text-cube creation

- 6.1. Download the latest MeSH Tree available at

(<https://www.nlm.nih.gov/mesh/filelist.html>). The current version of the code is using MeSH Tree 2018 as 'meshtree2018.bin' in the input directory.

6.2. Define the categories of interest (e.g., Disease names, Age groups, Gender). A category may include one or more MeSH descriptors (<https://meshb-prev.nlm.nih.gov/treeView>). Collect MeSH IDs for a category. Save the names of the categories in the file 'textcube\_config.json' in the config directory (see a sample of the category in 'Age Group' in the downloaded version of 'textcube\_config.json' file).

6.3. Put the collected categories of MeSH IDs in a line separated by a space. Save the category file as 'categories.txt' in the 'input' directory (see a sample of 'Age Group' MeSH IDs in the downloaded version of 'categories.txt' file). This algorithm automatically selects all descendent MeSH descriptors. An example of root nodes and descendants are presented in **Figure 4**.

6.4. Make sure that 'mesh2pmid.json' is in the 'data' directory. If the MeSH Tree has been updated with a different name (e.g., 'meshtree2019.bin') in 'input' directory, make sure that this is properly represented in the input data path in the 'run\_textcube.py' file.

6.5. Type 'python run\_textcube.py' in the terminal to create a document data structure called Text-Cube. This creates a collection of documents (PMIDs) for each category. A single document (PMID) may fall under multiple categories, (see **Table 3A**, **Table 3B**, **Figure 6A** and **Figure 7A**).

6.6. Once Text-Cube creation step is completed, make sure that following data files are saved in the 'data' directory: (1) a cell to PMID table as "textcube\_cell2pmid.json", (2) a PMID to cell mapping table as "textcube\_pmid2cell.json", (3) a collection of all descendant MeSH terms for a cell as "meshterms\_per\_cat.json" (4) Text-Cube data statistics as "textcube\_stat.txt".

6.7. Go to the 'log' directory to read the log messages in 'textcube\_log.txt' in case this process fails. If the process is completed successfully, the debugging messages of the Text-Cube creation will be printed out in the log file.

## 7. Entity count

7.1. Create user-defined entities (e.g., protein names, genes, chemicals). Put one entity and its abbreviations in a single line separated by "|". Save the entity file as 'entities.txt' in the 'input' directory. A sample of entities can be found in **Table 4**.

7.2. Make sure that Elasticsearch server is running. Otherwise, go to step 5.2 and 5.3 to restart the Elasticsearch server. It is expected to have an indexed database called 'pubmed' in your Elasticsearch server which was established in step 5.

309 7.3. Make sure that 'textcube\_pmid2cell.json' is in the 'data' directory.

310  
311 7.4. Type 'python run\_entitycount.py' in the terminal to perform Entity Count operation. This  
312 searches the documents from the indexed database and counts the entity in each document as  
313 well as collects the PMIDs in which entities were found.

314  
315 7.5. Once the Entity Count is completed, make sure that the final results are saved as  
316 'entitycount.txt' and 'entityfound\_pmid2cell.json' in the 'data' directory.

317  
318 7.6. Go to the 'log' directory to read the log messages in 'entitycount\_log.txt' in case this  
319 process fails. If the process is completed successfully, the debugging messages of the Entity  
320 Count will be printed out in the log file.

## 321 322 8. Metadata update

323  
324 8.1. Make sure that all input data ('entitycount.txt', 'textcube\_pmid2cell.json',  
325 'entityfound\_pmid2cell.txt') are in the 'data' directory. These are the input data for Metadata  
326 Update.

327  
328 8.2. Type 'python run\_metadata\_update.py' in the terminal to update the metadata. This  
329 prepares a collection of metadata (e.g., cell name, associated MeSH, PMIDs) representing each  
330 text document in the cell. A sample of Text-Cube metadata is presented in **Table 3A** and **Table**  
331 **3B**.

332  
333 8.3. Once the Metadata Update is completed, make sure that 'metadata\_pmid2pcount.json'  
334 and 'metadata\_cell2pmid.json' files are saved in 'data' directory.

335  
336 8.4. Go to the 'log' directory to read the log messages in 'metadata\_update\_log.txt' in case  
337 this process fails. If the process is completed successfully, the debugging messages of the  
338 metadata update will be printed out in the log file.

## 339 340 9. CaseOLAP score calculation

341  
342 9.1. Make sure that 'metadata\_pmid2pcount.json' and 'metadata\_cell2pmid.json' files are  
343 present in the 'data' directory. These are the input data for score calculation.

344  
345 9.2. Type 'python run\_caseolap\_score.py' in the terminal to perform CaseOLAP score  
346 calculation. This calculates the CaseOLAP score of the entities based on user-defined categories.  
347 The CaseOLAP score is the product of *Integrity*, *Popularity*, and *Distinctiveness*.

348  
349 9.3. Once the score computation is completed, make sure that this saves the results in  
350 multiple files (e.g., popularity as 'pop.csv', distinctiveness as 'dist.csv', CaseOLAP score as  
351 'caseolap.csv'), in the 'result' directory. The summary of the CaseOLAP score calculation is also  
352 presented in **Table 5**.

9.4. Go to the 'log' directory to read the log messages in '*caseolap\_score\_log.txt*' in case this process fails. If the process is completed successfully, the debugging messages of the indexing will be printed out in the log file.

## REPRESENTATIVE RESULTS:

To produce sample results, we implemented the CaseOLAP algorithm in two subject headings/descriptors: "Age Groups" and "Nutritional and Metabolic Diseases" as use cases.

**Age Groups.** We selected all 4 subcategories of "Age Groups" (infant, child, adolescent, and adult) as cells in a Text-Cube. The obtained metadata and statistics are shown in **Table 3A**. The comparison of the number of documents among the Text-Cube cells is displayed in **Figure 6A**. Adult contains 172,394 documents which is the highest number across all cells. The adult and adolescent subcategories have the highest number of shared documents (26,858 documents). Notably, these documents included the entity of our interest only (i.e., mitochondrial proteins). The Venn diagram in **Figure 6B** represents the number of entities (i.e., mitochondrial proteins) found within each cell, and within multiple overlaps among the cells. The number of proteins shared within all Age Groups subcategories is 162. The adult subcategory depicts the highest number of unique proteins (151) followed by child (16), infant (8) and adolescent (1). We calculated the protein-age group association as a CaseOLAP score. The top 10 proteins (based on their average CaseOLAP score) associated with infant, child, adolescent and adult subcategories are Sterol 26-hydroxylase, Alpha-crystallin B chain, 25-hydroxyvitamin D-1 alpha-hydroxylase, Serotransferrin, Citrate synthase, L-seryl-tRNA, Sodium/potassium-transporting ATPase subunit alpha-3, Glutathione S-transferase omega-1, NADPH:adrenodoxin oxidoreductase, and Mitochondrial peptide methionine sulfoxide reductase (shown in **Figure 6C**). The adult subcategory displays 10 heatmap cells with a higher intensity compared to the heatmap cells of the adolescent, child and infant subcategory, indicating that the top 10 mitochondrial proteins exhibit the strongest associations to the adult subcategory. The mitochondrial protein Sterol 26-hydroxylase has high associations in all age subcategories which is demonstrated by heatmap cells with higher intensities compared to the heatmap cells of the other 9 mitochondrial proteins. The statistical distribution of the absolute difference in the score between two groups shows the following range for mean difference with a 99% confidence interval: (1) the mean difference between 'ADLT' and 'INFT' lies in the range (0.029 to 0.042), (2) the mean difference between 'ADLT' and 'CHLD' lies in the range (0.021 to 0.030), (3) the mean difference between 'ADLT' and 'ADOL' lies in the range (0.020 to 0.029), (4) the mean difference between 'ADOL' and 'INFT' lies in the range (0.015 to 0.022), (5) the mean difference between 'ADOL' and 'CHLD' lies in the range (0.007 to 0.010), (6) the mean difference between 'CHLD' and 'INFT' lies in the range (0.011 to 0.016).

**Nutritional and Metabolic Diseases.** We selected 2 subcategories of "Nutritional and Metabolic Diseases" (i.e., metabolic disease and nutritional disorders) to create 2 cells in a Text-Cube. The obtained metadata and statistics are shown in **Table 3B**. The comparison of the number of documents among the Text-Cube cells is displayed in **Figure 7A**. The subcategory metabolic disease contains 54,762 documents followed by 19,181 documents in nutritional disorders. The

subcategories metabolic disease and nutritional disorders have 7,101 shared documents. Notably, these documents included the entity of our interest only (i.e., mitochondrial proteins). The Venn diagram in **Figure 7B** represents the number of entities found within each cell, and within multiple overlaps between the cells. We calculated the protein-“Nutritional and Metabolic Diseases” association as a CaseOLAP score. The top 10 proteins (based on their average CaseOLAP score) associated with this use case are Sterol 26-hydroxylase, Alpha-crystallin B chain, L-seryl-tRNA, Citrate synthase, tRNA pseudouridine synthase A, 25-hydroxyvitamin D-1 alpha-hydroxylase, Glutathione S-transferase omega-1, NADPH: adrenodoxin oxidoreductase, Mitochondrial peptide methionine sulfoxide reductase, Plasminogen activator inhibitor 1 (shown in **Figure 7C**). More than half (54%) of all proteins are shared between the subcategories metabolic diseases and nutritional disorders (397 proteins). Interestingly, almost half (43%) of all associated proteins in the metabolic disease subcategory are unique (300 proteins), whereas nutritional disorders exhibit only a few unique proteins (35). Alpha-crystallin B chain displays the strongest association to the subcategory metabolic diseases. Sterol 26-hydroxylase, mitochondrial displays the strongest association in the nutritional disorders subcategory, indicating that this mitochondrial protein is highly relevant in studies describing nutritional disorders. The statistical distribution of the absolute difference in the score between two groups ‘MBD’ and ‘NTD’ shows the range (0.046 to 0.061) for the mean difference as a 99% confidence interval.

## FIGURE AND TABLE LEGENDS

**Figure 1. Dynamic view of the CaseOLAP Workflow.** This figure represents the 5 major steps in the CaseOLAP workflow. In step 1, the workflow begins by downloading and extracting textual documents (e.g., from PubMed). In step 2, extracted data are parsed to create a data dictionary for each document as well as a MeSH to PMID mapping. In step 3, data indexing is conducted to facilitate fast and efficient entity search. In step 4, implementation of user-provided category information (e.g., root MeSH for each cell) is carried out to construct a Text-Cube. In step 5, the entity count operation is implemented over index data to calculate the CaseOLAP scores. These steps are repeated in an iterative manner to update the system with the latest information available in a public database (e.g., PubMed).

**Figure 2. Technical Architecture of the CaseOLAP Workflow.** This figure represents the technical details of the CaseOLAP workflow. Data from the PubMed repository are obtained from the PubMed FTP server. The user connects to the cloud server (e.g., AWS connectivity) via their device and creates a Download Pipeline which downloads and extracts the data to a local repository in the cloud. Extracted data are structured, verified, and brought to a proper format with a Data Parsing Pipeline. Simultaneously, a MeSH to PMID mapping table is created during the parsing step, which is used for Text-Cube construction. Parsed data are stored as a JSON like key-value dictionary format with document metadata (e.g., PMID, MeSH, publishing year). The Indexing step further improves the data by implementing Elasticsearch to handle bulk data. Next, the Text-Cube is created with user-defined categories by implementing MeSH to PMID mapping. When the Text-Cube formation and Indexing steps are completed, an entity count is conducted. Entity count data are implemented to the Text-Cube metadata. Finally, the

CaseOLAP score is calculated based on the underlying Text-Cube structure.

**Figure 3. A sample of a parsed document.** A sample of parsed data is presented in this figure. The parsed data are arranged as a key-value pair which is compatible with indexing and document metadata creation. In this figure, a PMID (e.g., “25896987”) is serving as a key and collection of associated information (e.g., Title, Journal, Publishing date, Abstract, MeSH, Substances, Department and Location) are as value. The very first application of such document metadata is the construction of MeSH to PMID mapping (**Figure 5** and **Table 2**), which is later implemented to create the Text-Cube and to calculate the CaseOLAP score with user-provided entities and categories.

**Figure 4. A sample of a MeSH tree.** The ‘Age Groups’ MeSH tree is adapted from the tree data structure available in the NIH database (MeSH Tree 2018, <https://meshb.nlm.nih.gov/treeView>). MeSH descriptors are implemented with their node IDs (e.g., Persons [M01], Age Groups [M01.060], Adolescent [M01.060.057], Adult [M01.060.116], Child [M01.060.406], Infant [M01.060.703]) to collect the documents relevant to a specific MeSH descriptor (**Table 3A**).

**Figure 5. MeSH to PMID mapping in Age Groups.** This figure presents the number of text documents (each linked with a PMID) collected under the MeSH descriptors in “Age Groups” as a bubble plot. The MeSH to PMID mapping is generated to provide the exact number of documents collected under the MeSH descriptors. A total number of 3,062,143 unique documents were collected under the 18 descendent MeSH descriptors (see **Table 2**). The higher the number of PMIDs selected under a specific MeSH descriptor, the larger the radius of the bubble representing the MeSH descriptor. For instance, the highest number of documents were collected under the MeSH descriptor “Adult” (1,786,371 documents), whereas the fewest number of text documents were collected under the MeSH descriptor “Infant, Postmature” (62 documents).

An additional example of MeSH to PMID mapping is given for “Nutritional and Metabolic Diseases” (<https://caseolap.github.io/mesh2pmid-mapping/bubble/meta.html>). A total number of 422,039 unique documents were collected under the 361 descendent MeSH descriptors in “Nutritional and Metabolic Diseases”. The highest number of documents were collected under the MeSH descriptor “Obesity” (77,881 documents) followed by “Diabetes Mellitus, Type 2” (61,901 documents), whereas “Glycogen Storage Disease, Type VIII” exhibited the fewest number of documents (1 document). A related table is also available online at (<https://github.com/CaseOLAP/mesh2pmid-mapping/blob/master/data/diseaseall.csv>).

**Figure 6. “Age Groups” as a use case.** This figure presents the results from a use case of the CaseOLAP platform. In this instance, protein names and their abbreviations (see sample in **Table 4**) are implemented as entities and “Age Groups” including the cells: infant (INFT), child (CHLD), adolescent (ADOL), and adult (ADLT), are implemented as subcategories (see **Table 3A**). **(A) Number of documents in “Age Groups”:** This heat map shows the number of documents distributed across the cells of “Aged Groups” (for details on the Text-Cube creation see Protocol

4 and **Table 3A**). A higher number of documents is presented with a darker intensity of the heatmap cell (see the scale). A single document may be included in more than one cell. The heatmap presents the number of documents within a cell along the diagonal position (e.g., ADLT contains 172,394 documents which is the highest number across all cells). The nondiagonal position represents the number of documents falling under two cells (e.g., ADLT and ADOL have 26,858 shared documents). **(B). Entity count in “Age Groups”**: The Venn diagram represents the number of proteins found in the four cells representing “Age Groups” (INFT, CHLD, ADOL, and ADLT). The number of proteins shared within all cells is 162. The age group ADLT depicts the highest number of unique proteins (151) followed by CHLD (16), INFT (8) and ADOL (1). **(C) CaseOLAP score presentation in “Age Groups”**: The top 10 proteins with the highest average CaseOLAP scores in each group are presented in a heat map. A higher CaseOLAP score is presented with a darker intensity of the heatmap cell (see the scale). The protein names are displayed on the left column and the cells (INFT, CHLD, ADOL, ADLT) are displayed along the x-axis. Some proteins show a strong association to a specific age group (e.g., Sterol 26-hydroxylase, alpha-crystallin B chain and L-seryl-tRNA have strong associations with ADLT, whereas Sodium/potassium-transporting ATPase subunit alpha-3 has a strong association with INFT).

**Figure 7. “Nutritional and Metabolic Diseases” as a use case**: This figure presents the results from another use case of the CaseOLAP platform. In this instance, protein names and their abbreviations (see sample at **Table 4**) are implemented as entities and “Nutritional and Metabolic Disease” including the two cells: metabolic disease (MBD) and nutritional disorders (NTD) are implemented as subcategories (see **Table 3B**). **(A). Number of documents in “Nutritional and Metabolic Diseases”**: This heatmap depicts the number of text documents in the cells of “Nutritional and Metabolic Diseases” (for details on the Text-Cube creation see Protocol 4 and **Table 3B**). A higher number of documents is presented with a darker intensity of the heatmap cell (see scale). A single document may be included in more than one cell. The heatmap presents the total number of documents within a cell along the diagonal position (e.g., MBD contains 54,762 documents which is the highest number across the two cells). The nondiagonal position represents the number of documents shared by the two cells (e.g., MBD and NTD have 7,101 shared documents). **(B). Entity Count in “Nutritional and Metabolic Diseases”**: The Venn diagram represents the number of proteins found in the two cells representing “Nutritional and Metabolic Diseases” (MBD and NTD). The number of proteins shared within the two cells is 397. The MBD cell depicts 300 unique proteins, and the NTD cell depicts 35 unique proteins. **(C). CaseOLAP score presentation in “Nutritional and Metabolic Diseases”**: The top 10 proteins with the highest average CaseOLAP scores in “Nutritional and Metabolic Diseases” are presented in a heat map. A higher CaseOLAP score is presented with a darker intensity of the heatmap cell (see scale). The protein names are displayed on the left column and cells (MBD and NTD) are displayed along the x-axis. Some proteins show a strong association to a specific disease category (e.g., alpha-crystallin B chain has a high association with metabolic disease and sterol 26-hydroxylase has a high association with nutritional disorders).

**Table 1. Algorithms and Complexities.** This table presents information on the time spent



(percentage of total time spent) on the procedures (e.g., downloading, parsing), data structure and details about the implemented algorithms in the CaseOLAP platform. CaseOLAP implements the professional indexing and search application called Elasticsearch. Additional information on complexities related to Elasticsearch and internal algorithms can be found at (<https://www.elastic.co>).

**Table 2. MeSH to PMID mapping statistics.** This table presents all descendant MeSH descriptors from “Age Groups” and their number of collected PMIDs (text documents). The visualization of these statistics is presented in **Figure 5**.

**Table 3. Text-Cube Metadata.** A tabular view of Text-Cube metadata is presented. The tables provide information about the categories and MeSH descriptor roots and descendants, which are implemented to collect the documents in each cell. The table also provides the statistics of the collected documents and entities. **(A) “Age Groups”:** This is a tabular display of “Age Groups” including infant (INFT), child (CHLD), adolescent (ADOL), and adult (ADLT) and their MeSH root IDs, number of descendant MeSH descriptors, number of selected PMIDs and number of found entities. **(B) “Nutritional and Metabolic Diseases”:** This is a tabular display of “Nutritional and Metabolic Diseases” including metabolic disease (MBD) and nutritional disorders (NTD) with their MeSH root IDs, number of descendant MeSH descriptors, number of selected PMIDs and the number of found entities.

**Table 4. Sample Entity Table.** This table presents the sample of entities implemented in our two use cases: “Age Groups” and “Nutritional and Metabolic Diseases” (**Figure 6** and **Figure 7, Table 3A,B**). The entities include protein names, synonyms, and abbreviations. Each entity (with its synonyms and abbreviations) is selected one by one and is passed through the entity search operation over indexed data (see protocol 3 and 5). The search produces a list of documents which further facilitate the entity count operation.

**Table 5. CaseOLAP equations:** The CaseOLAP algorithm was developed by Fangbo Tao and Jiawei Han et al. in 2016<sup>1</sup>. Briefly, this table presents the CaseOLAP score calculation consisting of three components: integrity, popularity, and distinctiveness, and their associated mathematical meaning. In our use cases, the integrity score for proteins is 1.0 (the maximum score) because they stand as established entity names. The CaseOLAP scores in our use cases can be seen in **Figure 6C** and **Figure 7C**.

## DISCUSSION:

We have demonstrated that the CaseOLAP algorithm can create a phrase based quantitative association to a knowledge-based category over large volumes of textual data for extraction of meaningful insights. Following our protocol, one can build the CaseOLAP framework to create a desired Text-Cube and quantify entity-category associations through CaseOLAP score calculation. The obtained raw CaseOLAP scores can be taken to integrative analyses including dimensionality reduction, clustering, temporal and geographical analysis, as well as the creation of a graphical database which enables semantic mapping of the documents.



**Applicability of the algorithm.** Examples of user-defined entities, other than proteins, could be a list of gene names, drugs, specific signs and symptoms including their abbreviations and synonyms. Furthermore, there are many choices for category selection to facilitate specific user-defined biomedical analyses (e.g., Anatomy [A], Discipline and Occupation [H], Phenomena and Processes [G]). In our two use cases, all scientific publications and their textual data are retrieved from the MEDLINE database using PubMed as the search engine, both managed by the National Library of Medicine. However, the CaseOLAP platform may be applied to other databases of interest containing biomedical documents with textual data such as the FDA Adverse Event Reporting System (FAERS). This is an open database containing information on medical adverse events and medication error reports submitted to FDA. In contrast to MEDLINE and FAERS, databases in hospitals containing electronic health records from patients are not open to the public and are restricted by the Health Insurance Portability and Accountability Act known as HIPAA.

CaseOLAP algorithm has been successfully applied to the different types of data (e.g., news articles)<sup>1</sup>. The implementation of this algorithm in biomedical documents has been made in 2018<sup>5</sup>. The requirements for applicability of CaseOLAP algorithm is that each of the documents should be assigned with keywords associated with the concepts (e.g., MeSH descriptors in biomedical publications, keywords in news articles). If keywords are not found, one can apply Autophrase<sup>6,7</sup> to collect top representative phrases and build the entity list before implementing our protocol. Our protocol does not provide the step to perform Autophrase.

**Comparison with other algorithms.** The concept of using a Data-Cube<sup>8,9,10</sup> and a Text-Cube<sup>2,3,4</sup> has been evolving since 2005 with new advancements to make data mining more applicable. The concept of Online Analytical Processing (OLAP)<sup>11,12,13,14,15</sup> in data mining and business intelligence goes back to 1993. OLAP, in general, aggregates the information from multiple systems, and stores it in a multi-dimensional format. There are different types of OLAP systems implemented in data mining. For example (1) Hybrid Transaction/Analytical Processing (HTAP)<sup>16,17</sup>, (2) Multidimensional OLAP (MOLAP)<sup>18,19</sup>—Cube based, and (3) Relational OLAP (ROLAP)<sup>20</sup>.

Specifically, the CaseOLAP algorithm has been compared with numerous existing algorithms, specifically, with their phrase segmentation enhancements, including TF-IDF+Seg, MCX+Seg, MCX, and SegPhrase. Moreover, RepPhrase (RP, also known as SegPhrase+) has been compared with its own ablation variations, including (1) RP without the Integrity measure incorporated (RP No INT), (2) RP without the Popularity measure incorporated (RP No POP), and (3) RP without the Distinctiveness measure incorporated (RP No DIS). The benchmark results are shown in the study by Fangbo Tao et al.<sup>1</sup>.

There are still challenges on data mining which can add additional functionality over saving and retrieving the data from the database. Context-aware semantic Analytical Processing (CaseOLAP) systematically implements the Elasticsearch to build an indexing database of millions of documents (Protocol 5). The Text-Cube is a document structure built over the indexed data with user-provided categories (Protocol 6). This enhances the functionality to the

documents within and across the cell of the Text-Cube and allow us to calculate term frequency of the entities over a document and document frequency over a specific cell (Protocol 8). The final CaseOLAP score utilizes these frequency calculations to output a final score (Protocol 9). In 2018, we implemented this algorithm to study ECM proteins and six heart diseases to analyze protein-disease associations. The details of this study can be found in the study by Liem, D.A. et al.<sup>5</sup>. indicating that CaseOLAP could be widely used in the biomedical community exploring a variety of diseases and mechanisms.

**Limitations of the algorithm.** Phrase mining itself is a technique to manage and retrieve important concepts from textual data. While discovering entity-category association as a mathematical quantity (vector), this technique is unable to figure out the polarity (e.g., positive or negative inclination) of the association. One can build the quantitative summarization of the data utilizing the Text-Cube document structure with assigned entities and categories, but a qualitative concept with microscopic granularities cannot be reached. Some concepts are continuously evolving from past till now. The summarization presented for a specific entity-category association includes all incidences throughout the literature. This may lack the temporal propagation of the innovation. In the future, we plan to address these limitations.

**Future Applications.** About 90% of the accumulated data in the world is in the unstructured text data. Finding a representative phrase and relation to the entities embedded in the text is a very important task for the implementation of new technologies (e.g., Machine Learning, Information Extraction, Artificial Intelligence). To make the text-data machine readable, data need to be organized in the database over which the next layer of tools could be implemented. In the future, this algorithm can be a crucial step in making data mining more functional for the retrieval of information and the quantification of the entity-category associations.

#### **ACKNOWLEDGMENTS:**

This work was supported in part by National Heart, Lung, and Blood Institute: R35 HL135772 (to P. Ping); National Institute of General Medical Sciences: U54 GM114833 (to P. Ping, K. Watson, and W. Wang); U54 GM114838 (to J. Han); a gift from the Hellen & Larry Hoag Foundation and Dr. S. Setty; and the T.C. Laubisch endowment at UCLA (to P. Ping).

#### **DISCLOSURES:**

The authors have nothing to disclose.

#### **REFERENCES**

1. Tao HZ, F. et al. Multidimensional, Phrase-Based Summarization in Text Cubes. *IEEE Data Engineering Bulletin*. **39**, 74-84 (2016).
2. Ding, B., Zhao, B., Lin, C.X., Han, J., Zhai, C. TopCells: Keyword-based search of top-k aggregated documents in text cube. *IEEE 26th International Conference on Data Engineering (ICDE)*. 381-384 (2010).
3. Ding, B., et al. Efficient Keyword-Based Search for Top-K Cells in Text Cube. *IEEE Transactions on Knowledge and Data Engineering*. **23** (12), 1795-1810 (2011).
4. Liu, X. et al. A Text Cube Approach to Human, Social and Cultural Behavior in the Twitter

- Stream.Social Computing, Behavioral-Cultural Modeling and Prediction. *Lecture Notes in Computer Science*. **7812** (2013).
5. Liem, D.A. et al. Phrase Mining of Textual Data to analyze extracellular matrix protein patterns across cardiovascular disease. *American Journal of Physiology-Heart and Circulatory Physiology* (2018)
  6. Shang, J. et al. Automated Phrase Mining from Massive Text Corpora. *IEEE Transactions on Knowledge and Data Engineering. IEEE Transactions on Knowledge & Data Engineering*. **30** (10), 1825-1837 (2018).
  7. Liu J., Shang J., Wang C., Ren X., Han J. Mining Quality Phrases from Massive Text Corpora. *Proceedings ACM-Sigmod International Conference on Management of Data*. 1729-1744 (2015).
  8. Lee, S., Kim, N., Kim, J. A Multi-dimensional Analysis and Data Cube for Unstructured Text and Social Media. *IEEE Fourth International Conference on Big Data and Cloud Computing*. 761-764 (2014).
  9. Lin C.X., Ding B., Han J., Zhu F., Zhao, B. Text Cube: Computing IR Measures for Multidimensional Text Database Analysis. *IEEE Data Mining*, 905-910 (2008).
  10. Hsu, W.J., Lu, Y., Lee, Z. Q. Accelerating Topic Exploration of Multi-Dimensional Documents. *Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, IEEE International, 1520-1527 (2017).
  11. Chaudhuri, S., Dayal,U. An overview of data warehousing and OLAP technology. *SIGMOD Record*. **26** (1), 65-74 (1997).
  12. Ravat, F., Teste, O., Tournier, R. Olap aggregation function for textual data warehouse. ICEIS - 9th International Conference on Enterprise Information Systems, Proceedings, 151-156, (2007).
  13. Ho, C.T., Agrawal, R., Megiddo, N., & Srikant, R. Range Queries in OLAP Data Cubes. *SIGMOD Conference* (1997).
  14. Saxena, V., Pratap, A. Olap Cube Representation for Object- Oriented Database *International Journal of Software Engineering & Applications*. **3** (2) (2012).
  15. Maniatis, A.S., Vassiliadis, P., Skiadopoulos, S., Vassiliou, Y. Advanced visualization for OLAP. *DOLAP* (2003).
  16. Bog, A., Benchmarking Transaction and Analytical Processing Systems: The Creation of a Mixed Workload Benchmark and its Application. *Springer Science & Business Media*. 7-13 (2013).
  17. Özcan, F., Tian, Y., Tözün, P. Hybrid Transactional/Analytical Processing: A Survey. In Proceedings of the ACM International Conference on Management of Data (SIGMOD '17), 1771-1775 (2017).
  18. Hasan K.M.A., Tsuji T., Higuchi K. An Efficient Implementation for MOLAP Basic Data Structure and Its Evaluation. *International Conference on Database Systems for Advanced Applications*. 288-299 (2007).
  19. Nantajeewarawat E. (eds) Advances in Databases: Concepts, Systems and Applications. DASFAA 2007. *Lecture Notes in Computer Science*. **4443** (2007).
  20. Shimada, T., Tsuji, T., Higuchi, K., A storage scheme for multidimensional data alleviating dimension dependency, *Third International Conference on Digital Information Management*. 662-668 (2007).

Figure 1

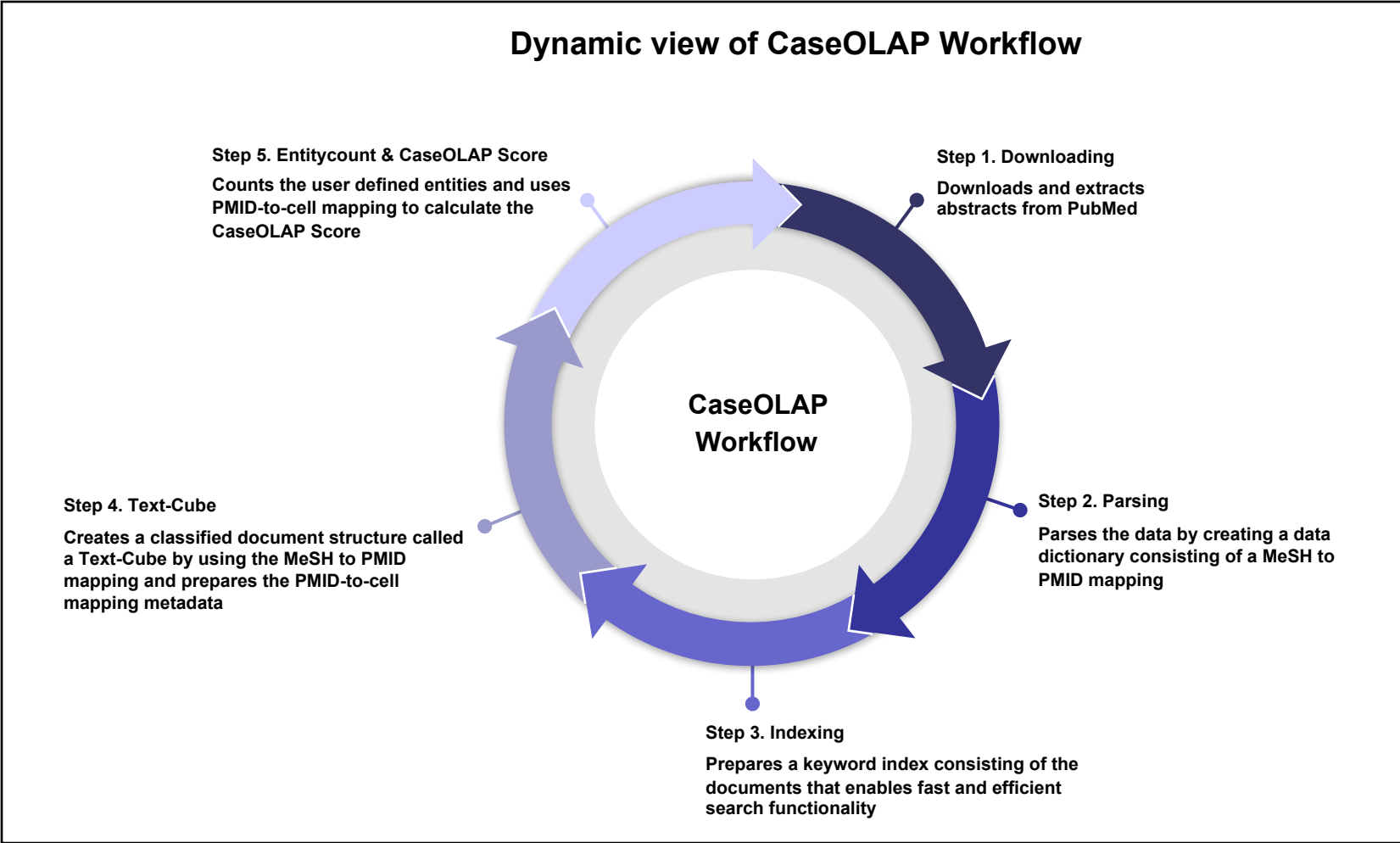
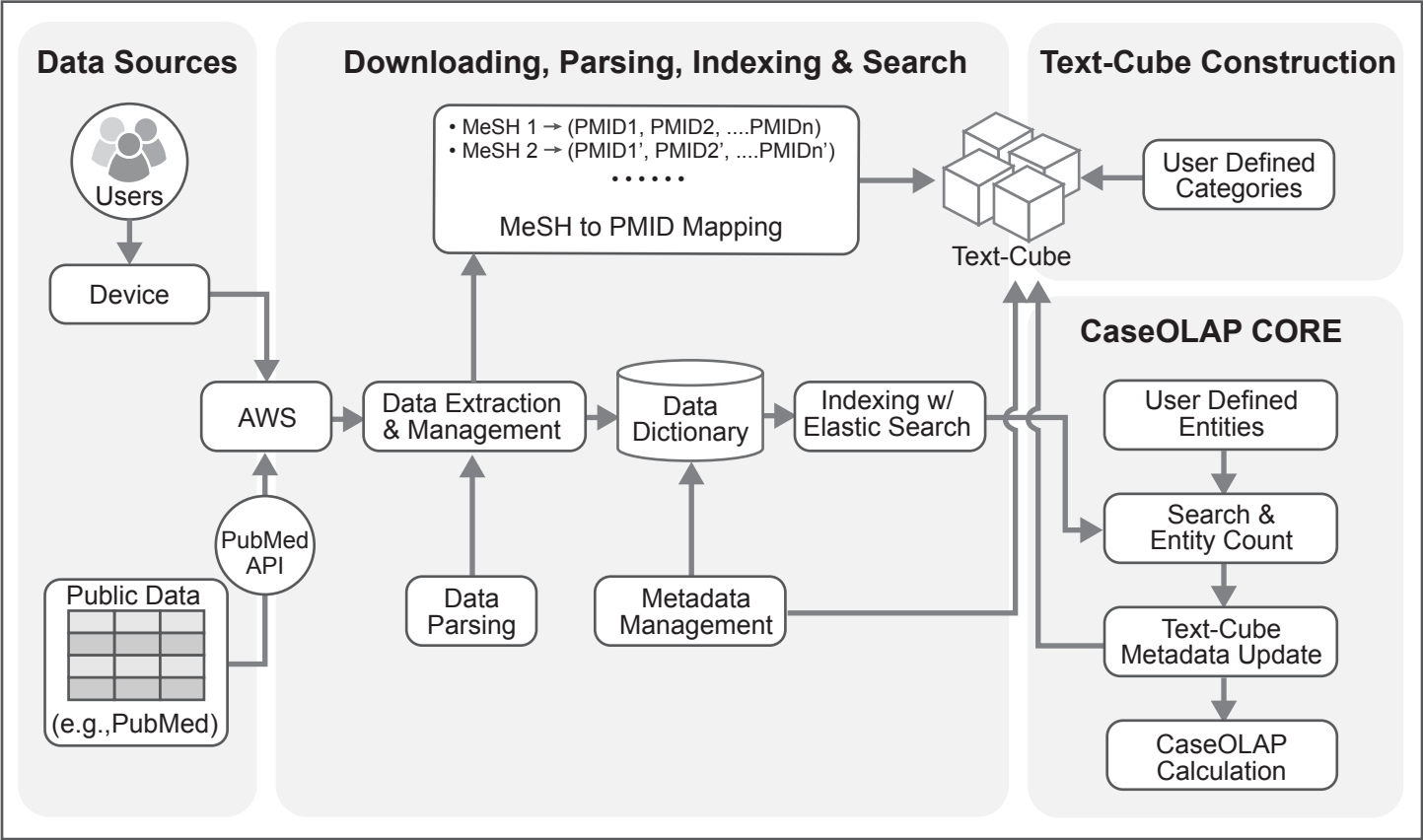




Figure 2.






Click here to access/download  
**Video or Animated Figure**  
Figure3.svg




Click here to access/download  
**Video or Animated Figure**  
Figure4.svg




Click here to access/download  
**Video or Animated Figure**  
Figure5.svg





Click here to access/download  
**Video or Animated Figure**  
Figure6.svg



Click here to access/download  
**Video or Animated Figure**  
Figure7.svg

Time spent (percentage of total time)	Steps in the CaseOLAP platform	Algorithm and Data Structure of the CaseOLAP platform	Complexity of Algorithm and Data Structure
40%	Downloading and Parsing	Iteration and tree parsing algorithms	Iteration with nested loop and constant multiplication: $O(n^2)$ , $O(\log n)$ . Where 'n' is no
30%	Indexing, Searching and Text Cube Creation	Iteration, Search algorithms by Elasticsearch (sorting, Lucene index, priority queues, finite state	<u>Complexity related to Elasticsearch</u> ( <a href="https://www.elastic.co/">https://www.elastic.co/</a> )
30%	Entity Counting and CaseOLAP Calculation	Iteration in Integrity, Popularity, Distinctiveness calculation	complexities related to caseOLAP Score calculation based on iteration types.

<b>Details of the Steps</b>
The Downloading pipeline iterates each procedure over multiple files. Parsing of a single document runs each
<del>Documents are indexed by</del> implementing the iteration process over the data dictionary. The Text-Cube creation implements
documents and make an count operation over the list. The entity count data is used to calculate CaseOLAP score.

MeSH descriptors	Number of of PMIDs collected
Adult	1,786,371
Middle Aged	1,661,882
Aged	1,198,778
Adolescent	706,429
Young Adult	486,259
Child	480,218
Aged, 80 and over	453,348
Child, Preschool	285,183
Infant	218,242
Infant, Newborn	160,702
Infant, Premature	17,701
Infant, Low Birth Weight	5,707
Frail Elderly	4,811
Infant, Very Low Birth Weight	4,458
Infant, Small for Gestational Age	3,168
Infant, Extremely Premature	1,171
Infant, Extremely Low Birth Weight	1,003
Infant, Postmature	62

	Infant (INFT)	Child (CHLD)	Adolescent (ADOL)	Adult (ADLT)
MeSH root ID	M01.060.703	M01.060.406	M01.060.057	M01.060.116
Number of descendant MeSH descriptors	9	2	1	6
Number of PMIDs selected	16,466	26,907	35,158	172,394
Number of entities found	233	297	257	443

	Metabolic Diseases (MBD)	Nutritional Disorders (NTD)
MeSH root ID	C18.452	C18.654
Number of descendant MeSH descriptors	308	53
Number of PMIDs collected	54,762	19,181
Number of entities found	697	432

Protein names and Synonyms	Abbreviations
N-acetylglutamate synthase, mitochondrial, Amino-acid acetyltransferase, N-acetylglutamate synthase long form; N-acetylglutamate synthase short form; N-acetylglutamate synthase conserved domain form]	(EC 2.3.1.1)
Protein/nucleic acid deglycase DJ-1 (Maillard deglycase) (Oncogene DJ1) (Parkinson disease protein 7) (Parkinsonism-associated deglycase) (Protein DJ-1)	(EC 3.1.2.-) (EC 3.5.1.-) (EC 3.5.1.124)(DJ-1)
Pyruvate carboxylase, mitochondrial (Pyruvic carboxylase)	(EC 6.4.1.1)(PCB)
Bcl-2-binding component 3 (p53 up-regulated modulator of apoptosis)	(JFY-1)
BH3-interacting domain death agonist [BH3-interacting domain death agonist p15 (p15 BID); BH3-interacting domain death agonist p13 ; BH3-interacting domain death agonist p11 ]	(p22 BID) (BID) (p13 BID)(p11 BID)
ATP synthase subunit alpha, mitochondrial (ATP synthase F1 subunit alpha)	
Cytochrome P450 11B2, mitochondrial (Aldosterone synthase) (Aldosterone-synthesizing enzyme) (CYPXIB2) (Cytochrome P-450Aldo) (Cytochrome P-450C18) (Steroid 18-hydroxylase)	(ALDOS) (EC 1.14.15.4) (EC 1.14.15.5)
60 kDa heat shock protein, mitochondrial (60 kDa chaperonin) (Chaperonin 60) (CPN60) (Heat shock protein 60) (Mitochondrial matrix protein P1) (P60 lymphocyte protein)	(HSP-60) (Hsp60) (HuCHA60)(EC 3.6.4.9)
Caspase-4 (ICE and Ced-3 homolog 2) (Protease TX) [Cleaved into: Caspase-4 subunit 1; Caspase-4 subunit 2]	(CASP-4) (EC 3.4.22.57)(ICH-2) (ICE(rel)-II) (Mih1)



<b>Quantities</b>	<b>User Defined</b>	<b>Calculated</b>	<b>Equation of the quantity</b>
<b>Integrity</b>	Yes	No	Integrity of user defined entities considered to be 1.0.
<b>Popularity</b>	No	Yes	Popularity equation in Figure 1 (Workflow and Algorithm) from reference 5, 'Materials and Methods' section.
<b>Distinctiveness</b>	No	Yes	Distinctiveness equation in Figure 1 (Workflow and Algorithm) from reference 5, 'Materials and Methods' section.
<b>CaseOLAP score</b>	No	Yes	CaseOLAP score equation in Figure 1 (Workflow and Algorithm) from reference 5, 'Materials and Methods' section.

<b>Meaning of the quantity</b>
Represents a meaningful phrase. Numerical value is 1.0 when it is already an established phrase.
Based on term frequency of the phrase within a cell. Normalized by total term frequency of the cell. Increase in term frequency has diminishing result.
Based on term frequency and document frequency within a cell and across the neighbouring cells. Normalized by total term frequency and document frequency. Quantitatively, it is the probability that a phrase is unique in a specific cell.
Based on Integrity, Popularity, and Distinctiveness. Numerical value always falls within 0 to 1. Quantitatively the CaseOLAP score represents the phrase-category association

None



1 Alewife Center #200  
Cambridge, MA 02140  
tel. 617.945.9051  
[www.jove.com](http://www.jove.com)

## ARTICLE AND VIDEO LICENSE AGREEMENT

Title of Article:	Cloud-based phrase mining and analysis of user-defined phrase-category association in biomedical publications
Author(s):	Dibakar Sigdel, Vincent Kyi, Aiden Zhang, Shaun P. Setty, David A. Liem, Yu Shi, Xuan Wang, Jiaming Shen, Wei Wang, JiaWei Han, Peipei Ping

Item 1: The Author elects to have the Materials be made available (as described at <http://www.jove.com/publish>) via:

☐ Standard Access ☒ Open Access

Item 2: Please select one of the following items:

- ☒ The Author is **NOT** a United States government employee.
- ☐ The Author is a United States government employee and the Materials were prepared in the course of his or her duties as a United States government employee.
- ☐ The Author is a United States government employee but the Materials were NOT prepared in the course of his or her duties as a United States government employee.

### ARTICLE AND VIDEO LICENSE AGREEMENT

1. **Defined Terms.** As used in this Article and Video License Agreement, the following terms shall have the following meanings: “**Agreement**” means this Article and Video License Agreement; “**Article**” means the article specified on the last page of this Agreement, including any associated materials such as texts, figures, tables, artwork, abstracts, or summaries contained therein; “**Author**” means the author who is a signatory to this Agreement; “**Collective Work**” means a work, such as a periodical issue, anthology or encyclopedia, in which the Materials in their entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole; “**CRC License**” means the Creative Commons Attribution-Non Commercial-No Derivs 3.0 Unported Agreement, the terms and conditions of which can be found at: <http://creativecommons.org/licenses/by-nc-nd/3.0/legalcode>; “**Derivative Work**” means a work based upon the Materials or upon the Materials and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Materials may be recast, transformed, or adapted; “**Institution**” means the institution, listed on the last page of this Agreement, by which the Author was employed at the time of the creation of the Materials; “**JoVE**” means MyJoVE Corporation, a Massachusetts corporation and the publisher of The Journal of Visualized Experiments; “**Materials**” means the Article and / or the Video; “**Parties**” means the Author and JoVE; “**Video**” means any video(s) made by the Author, alone or in conjunction with any other parties, or by JoVE or its affiliates or agents, individually or in collaboration with the Author or any other parties, incorporating all or any portion

of the Article, and in which the Author may or may not appear.

2. **Background.** The Author, who is the author of the Article, in order to ensure the dissemination and protection of the Article, desires to have the JoVE publish the Article and create and transmit videos based on the Article. In furtherance of such goals, the Parties desire to memorialize in this Agreement the respective rights of each Party in and to the Article and the Video.

3. **Grant of Rights in Article.** In consideration of JoVE agreeing to publish the Article, the Author hereby grants to JoVE, subject to **Sections 4** and **7** below, the exclusive, royalty-free, perpetual (for the full term of copyright in the Article, including any extensions thereto) license (a) to publish, reproduce, distribute, display and store the Article in all forms, formats and media whether now known or hereafter developed (including without limitation in print, digital and electronic form) throughout the world, (b) to translate the Article into other languages, create adaptations, summaries or extracts of the Article or other Derivative Works (including, without limitation, the Video) or Collective Works based on all or any portion of the Article and exercise all of the rights set forth in (a) above in such translations, adaptations, summaries, extracts, Derivative Works or Collective Works and (c) to license others to do any or all of the above. The foregoing rights may be exercised in all media and formats, whether now known or hereafter devised, and include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. If the “Open Access” box has been checked in **Item 1** above, JoVE and the Author hereby grant to the public all such rights in the Article as provided in, but subject to all limitations and requirements set forth in, the CRC License.

## ARTICLE AND VIDEO LICENSE AGREEMENT

4. **Retention of Rights in Article.** Notwithstanding the exclusive license granted to JoVE in **Section 3** above, the Author shall, with respect to the Article, retain the non-exclusive right to use all or part of the Article for the non-commercial purpose of giving lectures, presentations or teaching classes, and to post a copy of the Article on the Institution's website or the Author's personal website, in each case provided that a link to the Article on the JoVE website is provided and notice of JoVE's copyright in the Article is included. All non-copyright intellectual property rights in and to the Article, such as patent rights, shall remain with the Author.

5. **Grant of Rights in Video – Standard Access.** This **Section 5** applies if the "Standard Access" box has been checked in **Item 1** above or if no box has been checked in **Item 1** above. In consideration of JoVE agreeing to produce, display or otherwise assist with the Video, the Author hereby acknowledges and agrees that, Subject to **Section 7** below, JoVE is and shall be the sole and exclusive owner of all rights of any nature, including, without limitation, all copyrights, in and to the Video. To the extent that, by law, the Author is deemed, now or at any time in the future, to have any rights of any nature in or to the Video, the Author hereby disclaims all such rights and transfers all such rights to JoVE.

6. **Grant of Rights in Video – Open Access.** This **Section 6** applies only if the "Open Access" box has been checked in **Item 1** above. In consideration of JoVE agreeing to produce, display or otherwise assist with the Video, the Author hereby grants to JoVE, subject to **Section 7** below, the exclusive, royalty-free, perpetual (for the full term of copyright in the Article, including any extensions thereto) license (a) to publish, reproduce, distribute, display and store the Video in all forms, formats and media whether now known or hereafter developed (including without limitation in print, digital and electronic form) throughout the world, (b) to translate the Video into other languages, create adaptations, summaries or extracts of the Video or other Derivative Works or Collective Works based on all or any portion of the Video and exercise all of the rights set forth in (a) above in such translations, adaptations, summaries, extracts, Derivative Works or Collective Works and (c) to license others to do any or all of the above. The foregoing rights may be exercised in all media and formats, whether now known or hereafter devised, and include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. For any Video to which this **Section 6** is applicable, JoVE and the Author hereby grant to the public all such rights in the Video as provided in, but subject to all limitations and requirements set forth in, the CRC License.

7. **Government Employees.** If the Author is a United States government employee and the Article was prepared in the course of his or her duties as a United States government employee, as indicated in **Item 2** above, and any of the licenses or grants granted by the Author hereunder exceed the scope of the 17 U.S.C. 403, then the rights granted hereunder shall be limited to the maximum

rights permitted under such statute. In such case, all provisions contained herein that are not in conflict with such statute shall remain in full force and effect, and all provisions contained herein that do so conflict shall be deemed to be amended so as to provide to JoVE the maximum rights permissible within such statute.

8. **Protection of the Work.** The Author(s) authorize JoVE to take steps in the Author(s) name and on their behalf if JoVE believes some third party could be infringing or might infringe the copyright of either the Author's Article and/or Video.

9. **Likeness, Privacy, Personality.** The Author hereby grants JoVE the right to use the Author's name, voice, likeness, picture, photograph, image, biography and performance in any way, commercial or otherwise, in connection with the Materials and the sale, promotion and distribution thereof. The Author hereby waives any and all rights he or she may have, relating to his or her appearance in the Video or otherwise relating to the Materials, under all applicable privacy, likeness, personality or similar laws.

10. **Author Warranties.** The Author represents and warrants that the Article is original, that it has not been published, that the copyright interest is owned by the Author (or, if more than one author is listed at the beginning of this Agreement, by such authors collectively) and has not been assigned, licensed, or otherwise transferred to any other party. The Author represents and warrants that the author(s) listed at the top of this Agreement are the only authors of the Materials. If more than one author is listed at the top of this Agreement and if any such author has not entered into a separate Article and Video License Agreement with JoVE relating to the Materials, the Author represents and warrants that the Author has been authorized by each of the other such authors to execute this Agreement on his or her behalf and to bind him or her with respect to the terms of this Agreement as if each of them had been a party hereto as an Author. The Author warrants that the use, reproduction, distribution, public or private performance or display, and/or modification of all or any portion of the Materials does not and will not violate, infringe and/or misappropriate the patent, trademark, intellectual property or other rights of any third party. The Author represents and warrants that it has and will continue to comply with all government, institutional and other regulations, including, without limitation all institutional, laboratory, hospital, ethical, human and animal treatment, privacy, and all other rules, regulations, laws, procedures or guidelines, applicable to the Materials, and that all research involving human and animal subjects has been approved by the Author's relevant institutional review board.

11. **JoVE Discretion.** If the Author requests the assistance of JoVE in producing the Video in the Author's facility, the Author shall ensure that the presence of JoVE employees, agents or independent contractors is in accordance with the relevant regulations of the Author's institution. If more than one author is listed at the beginning of this Agreement, JoVE may, in its sole

## ARTICLE AND VIDEO LICENSE AGREEMENT

discretion, elect not take any action with respect to the Article until such time as it has received complete, executed Article and Video License Agreements from each such author. JoVE reserves the right, in its absolute and sole discretion and without giving any reason therefore, to accept or decline any work submitted to JoVE. JoVE and its employees, agents and independent contractors shall have full, unfettered access to the facilities of the Author or of the Author's institution as necessary to make the Video, whether actually published or not. JoVE has sole discretion as to the method of making and publishing the Materials, including, without limitation, to all decisions regarding editing, lighting, filming, timing of publication, if any, length, quality, content and the like.

12. **Indemnification.** The Author agrees to indemnify JoVE and/or its successors and assigns from and against any and all claims, costs, and expenses, including attorney's fees, arising out of any breach of any warranty or other representations contained herein. The Author further agrees to indemnify and hold harmless JoVE from and against any and all claims, costs, and expenses, including attorney's fees, resulting from the breach by the Author of any representation or warranty contained herein or from allegations or instances of violation of intellectual property rights, damage to the Author's or the Author's institution's facilities, fraud, libel, defamation, research, equipment, experiments, property damage, personal injury, violations of institutional, laboratory, hospital, ethical, human and animal treatment, privacy or other rules, regulations, laws, procedures or guidelines, liabilities and other losses or damages related in any way to the submission of work to JoVE, making of videos by JoVE, or publication in JoVE or elsewhere by JoVE. The Author shall be responsible for, and shall hold JoVE harmless from, damages caused by lack of sterilization, lack of cleanliness or by contamination due to


the making of a video by JoVE its employees, agents or independent contractors. All sterilization, cleanliness or decontamination procedures shall be solely the responsibility of the Author and shall be undertaken at the Author's expense. All indemnifications provided herein shall include JoVE's attorney's fees and costs related to said losses or damages. Such indemnification and holding harmless shall include such losses or damages incurred by, or in connection with, acts or omissions of JoVE, its employees, agents or independent contractors.

13. **Fees.** To cover the cost incurred for publication, JoVE must receive payment before production and publication of the Materials. Payment is due in 21 days of invoice. Should the Materials not be published due to an editorial or production decision, these funds will be returned to the Author. Withdrawal by the Author of any submitted Materials after final peer review approval will result in a US\$1,200 fee to cover pre-production expenses incurred by JoVE. If payment is not received by the completion of filming, production and publication of the Materials will be suspended until payment is received.

14. **Transfer, Governing Law.** This Agreement may be assigned by JoVE and shall inure to the benefits of any of JoVE's successors and assignees. This Agreement shall be governed and construed by the internal laws of the Commonwealth of Massachusetts without giving effect to any conflict of law provision thereunder. This Agreement may be executed in counterparts, each of which shall be deemed an original, but all of which together shall be deemed to be one and the same agreement. A signed copy of this Agreement delivered by facsimile, e-mail or other means of electronic transmission shall be deemed to have the same legal effect as delivery of an original signed copy of this Agreement.

A signed copy of this document must be sent with all new submissions. Only one Agreement is required per submission.

### CORRESPONDING AUTHOR

Name:	Peipei Ping		
Department:	Department of Physiology		
Institution:	University of California, Los Angeles		
Title:	Professor of Physiology, Medicine/Cardiology, & Bioinformatics		
Signature:		Date:	09/14/2018

Please submit a **signed** and **dated** copy of this license by one of the following three methods:

1. Upload an electronic version on the JoVE submission site
2. Fax the document to +1.866.381.2236
3. Mail the document to JoVE / Attn: JoVE Editorial / 1 Alewife Center #200 / Cambridge, MA 02140

## Editorial comments:

Changes to be made by the Author(s):

*1. Please take this opportunity to thoroughly proofread the manuscript to ensure that there are no spelling or grammar issues. The JoVE editor will not copy-edit your manuscript and any errors in the submitted revision may be present in the published version.*

Response: We thank the journal for the opportunity to elevate the manuscript according to the comments and suggestions. As recommended, we have thoroughly proofread the manuscript to remove spelling or grammar issues.

*2. Please upload Table 5 as an xls/xlsx. Tables 1-4 are just Tables, and not Materials Tables.*

Response: As requested, we have created a new table 5 in xls format.

*3. Please revise the table of the essential supplies, reagents, and equipment. The table should include the name, company, and catalog number of all relevant materials in separate columns in an xls/xlsx file.*

Response: Our manuscript highlights a cloud computing platform for Text Mining. In this manuscript, we do not describe any supplies, reagents, and equipment. Our downloadable code is in the GitHub repository. Users can follow our instructions in the protocol to establish a cloud computing framework for phrase mining and data analysis.

*4. Please remove references from the Abstract.*

Response: As requested by the editors, we have removed all references in the abstract.

*5. JoVE policy states that the video narrative is objective and not biased towards a particular product featured in the video. The goal of this policy is to focus on the science rather than to present a technique as an advertisement for a specific item. To this end, we ask that you please reduce the number of instances of “CaseOLAP” within your text. The term may be introduced but please use it infrequently and when directly relevant. Otherwise, please refer to the term using generic language.*

Response: In the updated version of the manuscript, we have replaced the term “CaseOLAP” with “phrase mining” a general term in Text Mining. Accordingly, the number of instances of “CaseOLAP” has been reduced to a minimum in the revised manuscript.

*6. Please ensure that all text in the protocol section is written in the imperative tense as if telling someone how to do the technique (e.g., “Do this,” “Ensure that,” etc.). The actions should be described in the imperative tense in complete sentences wherever possible. Avoid usage of phrases such as “could be,” “should be,” and “would be” throughout the Protocol. Any text that cannot be written in the imperative tense may be added as a “Note.” However, notes should be concise and used sparingly. Please include all safety procedures and use of hoods, etc.*

Response: We have thoroughly revised our protocol keeping our original framework of procedural steps. As recommended by the editors, the protocol now guides the reader step by

step with all commands to run the algorithm and manage intermediate data in creating a cloud computing framework for phrase mining and data analysis.

*7. The Protocol should contain only action items that direct the reader to do something. Please move the discussion about the protocol to the Discussion.*

Response: We have relocated the discussion part of the protocol to the 'Introduction' and 'Discussion' section. The revised protocol now includes only action items to perform in a sequence.

*8. Please add more details to your protocol steps. Please ensure you answer the "how" question, i.e., how is the step performed? Alternatively, add references to published material specifying how to perform the protocol action.*

Response: We have significantly revised our protocol to guide the user on how to conduct the designated task in a step by step fashion. Samples of results in intermediate steps are provided as figures, tables, and data links to GitHub. Log files are generated at each steps allowing the user to understand how the algorithm is performing the task. If an error occurs, the error messages are printed out in the log files.

*9. Please revise the text to avoid the use of any personal pronouns (e.g., "we", "you", "our" etc.).*

Response: As requested by the editors, we have removed any personal pronouns from the manuscript.

*10. 1.31: How is the checksum retrieved?*

Response: We have moved internal details of the md5-checksum from the protocol to the core of the CaseOLAP package. This allows the user to perform checksum automatically during the download process. Checksum file is retrieved from an FTP address and compared with one calculated from downloaded document to confirm the integrity of the downloaded data.

*11. Please provide all user input commands: File | Save | etc.*

Response: In the revised protocol, we have provided all required steps and commands to create, save, and run programs and data files systematically in allocated directories.

*12. 2.2.2: How is this done?*

Response: Section 2.2.2 in the previous protocol is now section 3.2 in our revised protocol. This parsing step is built on a tree data structure of extracted XML files. For this specific type of data, we have implemented the code to create a key-value pair dictionary for each document. By performing step 3.2, the user automatically gets the parsed data saved as '*pubmed.json*' in the 'data' directory.

*13. 2.3: How is this done explicitly?*

Response: Section 2.3 in the previous protocol is now section 4 in the revised protocol. We have explicitly provided the steps to run MeSH to PMID mapping using parsed data. In the first step, the algorithm reads the MeSH Tree as input data, and in the second step performs the



MeSH to PMID mapping and finally saves the mapped table as a *'mesh2pmid.json'* file in the *'data'* directory.

*14. 3.2: Initialize how?*

Response: We have provided all the steps to initiate an 'indexing' server using Elasticsearch. Section 3.2 in the previous protocol is now section 5.5 in the revised protocol. With *'run\_index\_init.py'* and the command provided to run this file, the user can initiate indexing of the database with the addressed schema from the *'index\_init\_config.json'* in the *'config'* directory.

*15. 3.3: Create how?*

Response: In the revised manuscript, we have provided all the steps to populate indexing server using Elasticsearch. Accordingly, section 3.3 in the previous protocol is now section 5.6 in the revised protocol. With *'run\_index\_populate.py'* and the command to run this file, the user can populate the indexing database with the addressed schema and parsed data.

*16. 4-6: How are these steps explicitly done?*

Response: Sections 4-6 in the previous protocol is now step 6-9 In the revised manuscript. We have provided a step by step breakdown of the process through 4-6 with required code files and commands to run. Based on the user's entity and category information, the user can perform an entity-category analysis similar to the sample results presented.

*17. Please provide a specific protocol with specific values instead of a generalized one. It helps to have specific search terms for a specific example.*

Response: In the revised manuscript, we have provided all required specific packages with specific versions in *'environment.yml'*. Our presented protocol uses the data downloaded from PubMed to produce the output. This protocol is capable of producing quantified phrase based summarizations of phrase-category associations from about 26 million abstracts.

*18. Please highlight 2.75 pages or less of the Protocol (including headings and spacing) that identifies the essential steps of the protocol for the video, i.e., the steps that should be visualized to tell the most cohesive story of the Protocol. Remember that non-highlighted Protocol steps will remain in the manuscript, and therefore will still be available to the reader.*

Response: As requested by the editors, we have limited the highlighted sections of the protocol to 2.75 pages including headings and spacing.

*19. Please ensure that the highlighted steps form a cohesive narrative with a logical flow from one highlighted step to the next. Please highlight complete sentences (not parts of sentences). Please ensure that the highlighted part of the step includes at least one action that is written in imperative tense.*

Response: We have highlighted the portion of the protocol as recommended by the editor.

*20. As we are a methods journal, please revise the Discussion to explicitly cover the following in detail in 3-6 paragraphs with citations:*

*a) Critical steps within the protocol*

Response: The critical steps in selecting user-defined entities and categories in the protocol are discussed in the 'Discussion' section [**Line 609-613**].

*b) Any modifications and troubleshooting of the technique*

Response: For modification of codes, we have provided a configuration interface in a 'configuration' directory, where the user can change the schema or configuration of the process. When this process is successfully completed, a log file provides the detail of internal processing. Or otherwise, if an error occurs, it will show where the error was produced.

*c) Any limitations of the technique*

Response: The limitation of the current phrase mining technique is discussed in the 'Discussion' section [**Line 645-653**].

*d) Any future applications of the technique*

Response: We have discussed the future application of the current phrase mining technique in the 'Discussion' section [**Line 655-661**]

*21. Please ensure that the references appear as the following: [Lastname, F.I., LastName, F.I., LastName, F.I. Article Title. Source. Volume (Issue), FirstPage – LastPage (YEAR).] For more than 6 authors, list only the first author then et al.*

Response: We have modified the reference section according to the editor's recommendations.

## **Reviewers' comments:**

Please note that novelty is not a requirement for publication and reviewer comments questioning the novelty of the article can be disregarded.

Please note that the reviewers raised some significant concerns regarding your method and your manuscript. Please thoroughly address each concern by revising the manuscript or addressing the comment in your rebuttal letter.

### **Reviewer #1:**

#### **Major Concerns:**

I am unable to find novelty

Response: JOVE is a journal with the main focus on visualization of experimental protocols. Accordingly, we aimed to highlight the methodology of CaseOLAP which could be beneficial for biomedical communities.

### **Reviewer #2:**

### Manuscript Summary:

This manuscript describes a protocol that has been developed by the authors, the CaseOLAP process, and a data structure called the Text-cube (from prior work). They describe the process of implementing the protocol for an example dataset. A set of figures and tables are included to show the example in detail.

### Major Concerns:

*1. JOVE is a journal for visualization of experimental protocols. This is most useful when the protocol has many physical steps. This protocol does not. Working with software, especially software that is NOT point and click, is better served by providing script examples and downloadable code for researchers to use directly, which these authors do in their Github repository. If this software has been shown to generally be too hard to setup and use by other researchers, they would be better served using tools like Singularity or Docker and providing images for users to download and use. A screen capture video of this protocol, which seems the most likely visual to be made, is not particularly useful for the most likely users who would be computing researchers.*

Response: We thank our reviewers for the valuable suggestion. We have elevated our protocol to address the above concerns. We have provided a GitHub repository with downloadable code and a sample data directory. Users can download or clone the project directory and set up a python environment on their device and build a cloud-based text mining platform by following the steps in our protocol. The visualization video of this manuscript will demonstrate how to perform the steps in the protocol with our sample results from intermediate steps as well as the final output. We have clearly defined what users need to modify in order to implement this protocol with their set of entities and categories. The video presentation will demonstrate all interfaces in which the user can remodel the whole process based on their entities and categories.

*2. The level of detail in the protocol is not sufficient. Although the authors do provide links to example python code in Jupyter notebooks, the protocol included in the text itself does not cover many of the minor details that need to be covered to make all of this work. Compare it, for instance, with other software based protocols published by JOVE previously, for instance: [www.jove.com/video/51639/high-throughput-image-analysis-tumor-spheroids-user-friendly-software](http://www.jove.com/video/51639/high-throughput-image-analysis-tumor-spheroids-user-friendly-software) which gives a much deeper level of detail of what needs to be done to set things up and use them. Note that if the difference between the protocol here and in the example depends on the nature of this software, then see my comment 1, above. Either there is not enough detail or the more scriptable nature of this software argues for a non-visual presentation.*

*Much of the protocol (lines 115 to 302) varies from being what look like draft notes (incomplete) to text describing things that do not seem to be part of the actual protocol implementing the system in hardware and software (280 on integrity is an example), so it is hard to evaluate this as a protocol per se.*

Response: We appreciate the reviewer bringing this issue to our attention, and it does require clarification. We have benefited from the method presented in the link provided by the reviewer. We have elevated our protocol to address the above concerns and have

systematically presented the commands to run program files and have provided steps to manage the intermediate data. To allow users to modify the input data and output results, we have provided a configuration directory ('config') with data configuration files. We hope, with commands to run programs, data handling instructions, and user based control to data input and output, this protocol sufficiently provides enough scriptability nature to this software.

In the 'Introduction' section of the revised manuscript, we have clearly explained the CaseOLAP score calculation with a link to 'Materials and Methods' section of the previous publication **(Ref-5) [Line 108-122]**. We have also provided a summary of the three sub-scores: Integrity, Popularity, and Distinctiveness in **Table 5**.

The integrity score denotes the quality of the phrases mined from the documents. In our analysis, the phrases (entities) applied to extract information were the mitochondrial protein names (including abbreviations and synonyms), acquired from UniProt (uniprot.org). Thus, the CaseOLAP algorithm was not used to determine the integrity of these phrases, and the integrity score is the same across all proteins. The integrity score for each protein name is 1.0 (the maximum score), because the UniProt protein naming system has been well established and broadly respected and applied. We have clarified about the 'integrity' in our 'Introduction' section **[Line 110-113]**.

*3. The paper's organization makes little sense to me. After "representative results" (lines 304 - 352) there are what appear to be figure captions (lines 353 to 506). If this is meant to be the "representative results" narrative, then it fails to read like a narrative (and, in this case, the figures appear to have no captions!). If these are meant as captions, then the "representative results" are incomplete. But either way this is not a coherently written section.*

Response: We thank the reviewer for pointing out missing "FIGURE AND TABLE LEGENDS". We have placed this title at the appropriate location in the manuscript.

*4. Neither the introduction nor the discussion at the end cite sufficient references to convince me of the general applicability of this method. While there is some reference in the general literature, my own searching was unable to show that this protocol is something that has wide enough interest for the community at large. This seems to be something of a niche technique. Perhaps if more works can be done by the developers to show general applicability, then this would change.*

Response: We thank the reviewer for providing us an opportunity to elaborate general applicability of this method. As suggested by the reviewer, we have added the comparison of CaseOLAP with other Text Mining techniques in the 'Discussion' section with corresponding references **[Line 615-643]**.

CaseOLAP was developed in 2016 **[Ref 1]**. This algorithm is novel in the Text Mining field. The concept of using a Data-Cube **[Ref 8,9,10]** and a Text-Cube **[Ref 2,3,4]** has been evolving since 2005 with new advancements to make data mining more applicable. The concept of Online Analytical Processing (OLAP) **[Ref 11,12,13,14,15]** in data mining and business intelligence goes back to 1993. There are different types of OLAP systems implemented in data mining. For

example: (1) Hybrid Transaction/Analytical Processing (HTAP) [Ref 16, 17], (2) Multidimensional OLAP (MOLAP) [Ref 18, 19] – Cube based, (3) Relational OLAP (ROLAP) [Ref 20].

Specifically, the CaseOLAP algorithm has been compared with numerous existing algorithms, specifically, with their phrase segmentation enhancements, including TF-IDF+Seg, MCX+Seg, MCX, and SegPhrase. Moreover, RepPhrase (RP, also known as SegPhrase+) has been compared with its own ablation variations, including (1) RP without the Integrity measure incorporated (RP No INT), (2) RP without the Popularity measure incorporated (RP No POP), and (3) RP without the Distinctiveness measure incorporated (RP No DIS). The benchmark results are shown in the study by Fangbo Tao, et. al [Ref 1].

There are still challenges on data mining which can add additional functionality over saving and retrieving the data from the database. Context-aware semantic Analytical Processing (CaseOLAP) systematically implements the Elasticsearch to build an indexing database of millions of documents (Protocol 5). The Text-Cube is a document structure built over the indexed data with user provided categories (Protocol 6). This enhances the functionality to the documents within and across the cell of the Text-Cube and allows us to calculate term frequency of the entities over a document and document frequency over a specific cell (Protocol 8). The final CaseOLAP score utilizes these frequency calculations to output a final score (Protocol 9). In 2018 we implemented this algorithm to study ECM proteins and six heart diseases to analyze protein-disease associations. The details of this study can be found at [Ref 5] indicating that CaseOLAP could be widely used in the biomedical community exploring a variety of diseases and mechanisms.

*5. Additionally, the properties claimed for the system (especially lines 509-511) have not been shown statistically, nor has sufficient supporting literature is cited in the text. Therefore, I do not know if there is any reason to expect this method to work when applied to other data sets. While in machine learning and optimization there are the "no free lunch" theorems that suggest that we cannot prove general applicability for any method like this, there are two problems specific to this paper:*

Response: We thank the reviewer for this comment. The CaseOLAP algorithm has been successfully applied to different types of data (e.g., news articles) and the results have been published [Ref 1]. We applied this algorithm to biomedical documents [Ref 5]. The CaseOLAP algorithm can be implemented on any documents that are associated with keywords (e.g., MeSH terms in biomedical publications, keywords in news articles).

Our protocol provides the method to establish a cloud computing platform including the general steps: downloading, parsing, indexing, mapping, counting and score calculation. It is the user's decision to select proper machine learning approaches to analyze entity-category associations represented by the caseOLAP scores.

*5a. There is no discussion and no apparent citation of another paper that discusses the*

*statistical nature of the results. For example, the authors claim that in figure 6 the difference between 0.14 and 0.17 is a significant difference in their heatmap. Why? What is the basis for this claim? Should I really on the basis of this result claim that Sodium/Potassium-transporting ATPase-s-alpha-3 is significantly more referred to in INFT vs ADLT literature? What are the "meaningful insights" the authors claim (lines 509-511)? How did they show meaningfulness here?*

Response: We thank the reviewer for providing us an opportunity to clarify our results. The results presented in **Figure 5** and **Figure 6** are the top 10 proteins based on their scores. Our result consists of nearly 3000 proteins and their scores in 4 different age groups. We appreciate the reviewer's suggestion to statistically compare scores between groups. We want to emphasise that the CaseOLAP scores are constructed from three different sub-scores that are normalized within the cell and across the cell. This allows us to compare the score of the entities within the cell and across the cells. In the revised manuscript we have included the significance of the scores and statistical error. Our sample result now reads as *Sodium/Potassium-transporting ATPase-s-alpha-3* is significantly more referred to in *INFT* vs *ADLT* group with '0.03' difference is significant according to a range of mean difference (0.029 to 0.042) with '99%' confidence level [**Line 399-406**].

*5b. Failing to provide some theory supporting the idea that the differences are "meaningful" in some sense, then the authors need to cite a sufficient number of previous use examples where the system has been used and shown to produce actionable insight/useful summarization or some other positive metric of performance. My reading of the citations does not show that this has been done. If there are other papers I was unable to find them, but ultimately it is the responsibility of the authors to make this point.*

*While I do think the technique may be useful in some sense and is worthy of additional study, it cannot be described at present as an established protocol of general interest to the community.*

Response: We thank the reviewer for this comment. The previous publication [**Ref 1**] by Fangbo Tao, et. al. presents sufficient results demonstrating the performance metric of the algorithm. The CaseOLAP algorithm has been compared with numerous existing algorithms, with their phrase segmentation enhancements, including TF-IDF+Seg, MCX+Seg, MCX, and SegPhrase. Please also see our response to comment 4.

We want to provide the biomedical community with a step by step procedure to establish the cloud computing framework. In our previous publications [**Ref 5**], we have compared the results with manually searched results from the reactome database and presented the results discussion.

#### **Minor Concerns:**

- 1. While undoubtedly a typesetting issue, the tables are horrendous in appearance and not properly labeled. The bad formatting definitely contributed to the delay in returning the*

review.

*Not enough detail is given on the steps described in lines 523 to 527, which are not detailed in the protocol, either, and appear to be critical to the whole example (collecting protein names). This was an important detail of the protocol example not addressed earlier. Perhaps this should be a major issue.*

Response: We thank the reviewer for the comment. Accordingly, we have upgraded our tables regarding label and formatting. We have discussed the user provided entities in the discussion. The user may use a set of entities and categories according to his/her own choice. In our protocol, we have provided steps to set up user-defined entities and categories. The details of protein name selection are highlighted in the ‘introduction’ section [Line 124-133].

*2. Much more detail needs to be given on how the caseOLAP scores are computed and interpreted, or a citation given that does this. This section is not detailed enough to show the general usefulness of these scores.*

Response: We thank the reviewer for this comment. We have revised the ‘Introduction’ section to elaborate the CaseOLAP score calculation. The provided references present in depth descriptions of the score calculation [Line 108-122].

The CaseOLAP score computation is based on user-provided entities and categories. The algorithm compares the entity count data within a cell and across the cells of the Text-Cube. There is not a standard method in the Text Mining literature capable of quantifying entity-category associations based on millions of text documents. We have demonstrated that the CaseOLAP platform fulfills this need.

### **Reviewer #3:**

#### **Manuscript Summary:**

A complete protocol is provided for automatic identification of phrases-category associations using CaseOLAP on the cloud.

#### **Major Concerns:**

None

#### **Minor Concerns:**

*1. The JSON file for the parsed sample on the git repo is broken.*

Response: We apologize for this issue. This issue has now been resolved in the revised manuscript.

*2. It will be better to have discussions on the limitation of the system and future development/improvement plans.*

Response: We thank the reviewer for this suggestion. We have included a paragraph addressing the limitations of the platform and future developments in our 'Discussion' section.

## **Reviewer #4:**

### **Manuscript Summary:**

The authors propose platform, namely cloud-based Context-aware Semantic Online Analytical Processing (CaseOLAP), for automatically computing a score that represents how strongly phrase (e.g. portions names, gene names) is associated with a category (e.g. Age group "adult, child") in the biomedical context. The proposed platform includes six steps: Download and extracting of data, Parsing data, Indexing, Text-cube creation, Entity count, CaseOLAP score computation. Authors carry out two case studies using PubMed dataset and CaseOLAP to demonstrate the efficiency of their proposed protocol.

### **Major Concerns:**

*1- The authors need to show illustrative example in how entity including its abbreviation and synonym is counting. Also, they need to clarify how final term frequency is computed in the paper. It is only shown in the code that the final term frequency is computed by summing up the count of all its abbreviation and synonym.*

Response: We appreciate the reviewer's suggestion to improve the manuscript. Protocol 7.1 provides instructions to specify entities (with abbreviation and synonyms). During the entity count step (Protocol 7.3.), the algorithm reads entity names with all abbreviations and synonyms and creates a key-value dictionary with a key as an entity name and values as a collection of all abbreviations and synonyms.

This entity dictionary is used to search a specific entity in the indexed database and the counting operation computes the sum of all occurrences, including entity the name, synonyms, and abbreviations. The entity count becomes available as a 'PMID to entity count' table in the data repository.

In the revised manuscript ('Introduction' section), we have provided the references to detail term frequency, document frequency, Text-Cube document structure, and score calculation [Line 108-122].

*2- The authors in step 5.1 "Selection of Entities" mentions that if user doesn't define entities, adopted AutoPhrase approach will be used. Will the adopted method automatically group abbreviations and synonyms of phrase in one entity? The authors need to clarify that.*

Response: We thank the reviewer for this important comment. To avoid the confusion in the



protocol, we have moved the discussion about 'Autophrase' into the 'Discussion' section [Line 608-610]. Our protocol is designed to provide the method for user-defined entities and categories. 'Autophrase' in the current version is not capable of distinguishing synonyms and abbreviations for a representative entity. 'Autophrase' can be implemented to find top phrases in the textual data. The user can manually separate top-phrases as representative entities, its abbreviations, and synonyms. This prepared entity list can be implemented while following our protocol.

*3- CaseOLAP Algorithm relies on statistical information "term frequency" to compute CaseOLAP score. For infrequent but important entity, CaseOLAP score will be small. How will the proposed protocol handle these entities?*

Response: We thank the reviewer for this comment. The final CaseOLAP score is based on three ranking criteria. We have provided a brief introduction and references providing details about these concepts in the 'Introduction' section [Line 108-122]. The algorithm calculates the term frequency as well as the document frequency to compute the 'Popularity' and 'Distinctiveness' scores. For infrequent but important concepts, the 'Popularity' could be small but the 'Distinctiveness' score could be high based on document frequency, and vice versa.

#### **Minor Concerns:**

*1- The authors do not provide enough details information about how their approach is more efficient compared to others approaches. They need to show how text-cube technique make their approach is favorable over others approaches. Also, the authors did not name competitive approaches. It would be useful if the authors include some references to those approaches.*

Response: We thank the reviewer for this recommendation. We have included more references to address previous Text Mining methods in comparison to the Text Mining algorithm that is described in this manuscript [Line 612-640]. We have added the comparison of this technique with other existing OLAP techniques in the discussion section. Specifically, the CaseOLAP algorithm has been compared with numerous existing algorithms, with their phrase segmentation enhancements, including TF-IDF+Seg, MCX+Seg, MCX, and SegPhrase and RepPhrase. The benchmark results are shown in Figures 5A and 5B in the study by Fangbo Tao et. al. [Ref 1].