COLORADO STATE UNIVERSITY GLOBAL

CSC460: ADVANCED APPLICATIONS OF INFORMATION RETRIEVAL AND WEB SEARCHING

Credit Hours: 3

Contact Hours: This is a three-credit course, offered in accelerated format. This means that 16 weeks of material is covered in eight weeks. The exact number of hours per week that you can expect to spend on each course will vary based upon the weekly coursework, as well as your study style and preferences. You should plan to spend 14-20 hours per week in each course reading material, interacting on the discussion boards, writing papers, completing projects, and doing research.

Faculty Information: Faculty contact information and office hours can be found on the faculty profile page.

COURSE DESCRIPTION AND OUTCOMES

Course Description:

The course provides introduction to advanced techniques associated with the retrieval of information and searching of documents. Students will explore problems in natural language processing related to the retrieval of information using advanced computing constructs and search system algorithms. The course will focus on the application of efficient scoring, ranking, information retrieval evaluation, and efficient information retrieval models.

Course Overview:

CSC460 Advanced Applications of Information Retrieval and Web Searching is a course that deals with methods for searching in (very large) text collections. It covers advanced algorithms and procedures for text preprocessing, query languages, relevance models, and specific problems with search engine efficiencies. Information retrieval algorithmic concepts including Relevant Feedback/Query Expansion, XML Retrieval, Probabilistic IR, Language Models for IR, Text Classification and Naïve Bayes, Vector Space Classification, Support Vector Machine, and Flat Clustering are treated. Each of the eight course modules will be accompanied by a Critical Thinking Assignment based on either Java or Python. In addition, each module requires active participation in a discussion forum that explores more depth of the module topic and completion of a Mastery Exercise. This deepens the learned methods through practical implementation.

Course Learning Outcomes:

1. Apply Boolean retrieval techniques to an information retrieval problem.
2. Identify data structures to use for efficient information retrieval.
3. Discuss applications of vector space models for information retrieval scoring.
4. Demonstrate the computation of scores in an information retrieval system.
5. Determine factors that affect information retrieval systems.

Participation and Attendance

Prompt and consistent attendance in your online courses is essential for your success at CSU-Global Campus. Failure to verify your attendance within the first seven days of this course may result in your withdrawal. If for some reason you would like to drop a course, please contact your advisor.

Online classes have deadlines, assignments, and participation requirements just like on-campus classes. Budget your time carefully and keep an open line of communication with your instructor. If you are having technical problems, problems with your assignments, or other problems that are impeding your progress, let your instructor know as soon as possible.

Course Materials

Required:

Manning, C., Raghavan, P., & Schütze, H. (2008). An introduction to information retrieval. Cambridge, England: Cambridge University Press.

Software Applications:

Python 3, Java, Eclipse/Netbeans

NOTE: *All non-textbook required readings and materials necessary to complete assignments, discussions, and/or supplemental or required exercises are provided within the course itself. Please read through each course module carefully.*

Course Schedule

Due Dates

The Academic Week at CSU-Global begins on Monday and ends the following Sunday.

- **Discussion Boards:** The original post must be completed by Thursday at 11:59 p.m. MT and peer responses posted by Sunday at 11:59 p.m. MT. Late posts may not be awarded points.
- **Opening Exercises:** Take the Opening Exercise before reading each week's content to see which areas you will need to focus on. You may take these exercises as many times as you need. The Opening Exercises will not affect your final grade.
- **Mastery Exercises:** Students may access and retake Mastery Exercises through the last day of class until they achieve the scores they desire.
- **Critical Thinking:** Assignments are due Sunday at 11:59 p.m. MT.

WEEKLY READING AND ASSIGNMENT DETAILS

Module 1

Readings

- Brondwine, E., Shtok, A., & Kurland, O. (2016). Utilizing focused relevance feedback. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR '16)*. ACM, New York, NY, USA, 1061-1064. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/2911451.2914695
- Limsopatham, N., Macdonald, C., & Ounis, I. (2015). Modelling the usefulness of document collections for 1uery expansion in patient search. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM '15).* ACM, New York, NY, USA, 1739-1742. DOI: https://doi-org.csuglobal.idm.oclc.org/10.1145/2806416.2806614
- Williams, K., & Giles, C. L. (2016). Improving similar document retrieval using a recursive pseudo relevance feedback strategy. In *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries (JCDL '16)*. ACM, New York, NY, USA, 275-276. doi:https://doi-org.csuglobal.idm.oclc.org/10.1145/2910896.2925468

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (30 points)

Option #1: **Rocchio Relevance Feedback**

In judging relevancy, a user searches for the following phrases: blanket, thick blanket, expensive blankets, and used blankets. She examines document 1 (d1) and document 2 (d2), finding document 1 relevant because it contains blanket, thick blanket but not document 2, which contains expensive blanket. Suppose we are using direct term frequency (here, no scaling and no document frequency). There is no need to length-normalize vectors. Using Rocchio relevance feedback as in Equation 9.3 in your textbook, determine what the revised query vector should be after relevance feedback. Assume $\alpha = 1$, $\beta = 0.75$, $\gamma = 0.25$.

Submit your response for instructor grading. If you have questions about assignment or file formatting, contact your instructor.

Option #2: Rocchio Relevance Feedback: WE Search Systems

We Search Systems are meant to ease searching tasks for users. For example, Wani has implemented a relevance feedback web search system with intention to do relevance feedback based only on words in the title text returned for a page. The user is going to rank three results. The first user, Dingding, queries for:

banana slug

and the top three titles returned are:

banana slug Ariolimax columbianus

Santa Cruz mountains banana slug

Santa Cruz Campus Mascot

Dingding judges the first two documents as relevant and the third as nonrelevant. Suppose Wani's search engine uses term frequency but neither length normalization nor IDF. Further, suppose that he is using the Rocchio relevance feedback mechanism, with $\alpha = \beta = \gamma = 1$. Show the final revised query that would be run. (Please list the vector elements in alphabetical order.)

Submit your response for instructor grading. If you have questions about assignment or file formatting, contact your instructor.

Mastery Exercise (10 points)

Module 2

Readings

- Chapter 10 in *An Introduction to Information Retrieval*
- Abdelmadjid, L., Mimoun, M., Bachir, S., & Ismail, L. (2017). A semi-automatic solution for XML query response enrichment using a terminological domain ontology. In *Proceedings of the International Conference on Computing for Engineering and Sciences (ICCES '17)*. ACM, New York, NY, USA, 76-81. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3129186.3131941
- Jones, C. A., & Drake, F. L. (n.d.). Python and XML. Retrieved from https://www.oreilly.com/library/view/python-xml/0596001282/ch04s04.html
- Python XML Processing. (n.d.). Retrieved from https://www.tutorialspoint.com/python/python_xml_processing.htm

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (60 points)

Option #1: Computing Document Frequency

It's common to comput df for a structural term as the number of times that the structural term occurs under a particular parent node.

Suppose the structural term (c, t) = author#"Malik" occurs one time as the child of the node squib and there are 10 squib nodes in the collection; (c, t) occurs 1,000 times as the child of article; there are 1,000,000 article nodes in the collection. The idf weight of (c, t) then is $\log_2 10/1 \approx 3.3$ when occurring as the child of squib and $\log_2 1{,}000{,}000/1000 \approx 10.0$ when occurring as the child of article. This does not appear to be an appropriate weighting for (c, t). Why?

Also, why should (c, t) not receive a weight that is three times higher in articles than in squibs? Can you suggest a better way to compute idf? Write and submit an appropriate Python Script (.py) with related files (if any) demonstrating your answer.

Your paper should be 2-3 pages in length and conform to CSU-Global Guide to Writing and APA. Include at least two scholarly references in addition to the course textbook. The CSU-Global Library is a good place to find these references.

Option #2: Identifying Structural Terms

How many structural terms does the following XML document yield? Write and submit an appropriate Python Script (.py) with related files (if any) demonstrating your answer.

```
<play>
<author>Shakespeare</author>
<title>Macbeth</title>
<act number="I">
<scene number="vii">
<title>Macbeth's castle</title>
<verse>Will I with wine and wassail ...</verse>
</scene>
</act>
</play>
```

Submit your response for instructor grading. If you have questions about assignment or file formatting, contact your instructor.

Mastery Exercise (10 points)

Module 3

Readings

- Chapter 11 in *An Introduction to Information Retrieval*
- Sanz-Cruzado, J., Pepa, S. M., & Castells, P. (2018). Recommending contacts in social networks using information retrieval models. In *Proceedings of the 5th Spanish Conference on Information Retrieval (CERI '18).* ACM, New York, NY, USA, Article 19, 8 pages. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3230599.3230619

- Statistics – probability. (n.d.). Retrieved from https://www.tutorialspoint.com/statistics/probability.htm
- Yang P., & Fang, H. (2016). A reproducibility study of information retrieval models. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval (ICTIR '16).* ACM, New York, NY, USA, 77-86. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/2970398.2970415

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (70 points)

Option #1: Vector Space Model

Relevance feedback and probabilistic relevance feedback come in handy in many situations. Describe the differences between the two and implement a Python solution for Vector Space Model for Documents. Submit all related files in zipped folder. Consult the following resources.

Your paper should be 2-3 pages in length and conform to *CSU-Global Guide to Writing and APA*. Include at least two scholarly references in addition to the course textbook. The CSU-Global Library is a good place to find these references.

Option #2: Probabilistic Relevance Feedback

Relevance feedback and probabilistic relevance feedback come in handy in many situations. Describe the differences between the two and implement a Python solution for Probabilistic Relevance Feedback for Documents. Submit all related files in zipped folder. Consult the resources linked in assignment.

Submit your response for instructor grading. If you have questions about assignment or file formatting, contact your instructor.

Mastery Exercise (10 points)

Module 4

Readings

- Chapter 12 in *An Introduction to Information Retrieval*
- Bhattacharya, P., Goyal, P., & Sarkar, S. (2018). Using communities of words derived from multilingual word vectors for cross-language information retrieval in Indian languages. *ACM Transactions on Asian and Low-Resource Language Information Processing, 18*(1). doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3208358
- Deveaud, R., Mothe, J., Ullah, M-Z., & Nie, J-Y. (2018, October). Learning to adaptively rank document retrieval system configurations.
- *ACM Transactions on Information Systems, 37*(1), 1-41. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3231937
- Fan, Y., Ai, Q., Ren, Z., Hong, L., Yin, D., & Guo, J. (2019). DAPA: The WSDM 2019 Workshop on deep matching in practical applications. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19).* ACM, New York, NY, USA, 844-845. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3289600.3291375

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (70 points)

Option #1: Documents Ranking

Assuming we have a collection that consists of the four documents in the following table:

docID	Document text
1	click go the shears boys click click click
2	click click
3	metal here
4	metal shears click here

Observe the collection with the intention to build a query likelihood language model for it. Also suppose a mixture model between the documents and the collection and both weighted at 0.5 is given. MLE (maximum likelihood estimation) can be used to estimate both as unigram models. Now, work out the model probabilities of the queries click, shears, and hence click shears for each document, and then use those probabilities to rank the documents returned by each query. Fill in these probabilities in the following table:

Query	Doc 1	Doc 2	Doc 3	Doc 4
click				
shears				
click shears				

Give the final ranking of the documents for the query click shears.

Now implement your solution in Python and submit all related files in a zipped folder. Consult the resources linked in assignment for additional insights.

Submit your response for instructor grading. If you have questions about assignment or file formatting, contact your instructor.

Option #2: Document Ranking, Exercise 12.7

For this assignment, consult Exercise 12.7 in your textbook and then write one sentence describing the following quantities as done in Equation 12.10. Include whether the observed treatment is present in the model or not and whether the effect is raw or scaled.

a. term frequency in a document
b. collection frequency of a term
c. document frequency of a term
d. length normalization of a term

Now implement your solution in Python and submit all related files in a zipped folder. Consult the resources linked in the assignment for additional insights.

Submit your response for instructor grading. If you have questions about assignment or file formatting, contact your instructor.

Mastery Exercise (10 points)

Portfolio Milestone (25 points)

Options #1 and 2

In your first three Critical Thinking Assignments, chances are you missed a concept and did not earn all the possible points. This is an opportunity to correct any mistakes you may have made in the previous assignments. For this Milestone, make appropriate corrections to the code you submitted in Modules 1-3 (Critical Thinking Assignments). Corrections should reflect feedback from your instructor and improvements in execution, organization, and style. Resubmit your programs from Modules 1-3 with all outlined corrections. If no corrections are necessary, resubmit your original assignments.

Module 5

Readings

- Chapter 13 in *An Introduction to Information Retrieval*
- Agnihotri, d., Verma, K., & Tripathi, P. (2016). Computing correlative association of terms for automatic classification of text documents. In *Proceedings of the Third International Symposium on Computer Vision and the Internet (VisionNet'16).* ACM, New York, NY, USA, 71-80. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/2983402.2983424
- Kaur, R., & Ginige, J. A. (2019). Analysing effectiveness of multi-label classification in clinical coding. In *Proceedings of the Australasian Computer Science Week Multiconference (ACSW 2019)*. ACM, New York, NY, USA. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3290688.3290728
- Yang, E., Grossman, D., Frieder, O., & Yurchak, R. (2017). Effectiveness results for popular e-discovery algorithms. In *Proceedings of the 16th edition of the International Conference on Artificial Intelligence and Law (ICAIL '17)*. ACM, New York, NY, USA, 261-264. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3086512.3086540

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (70 points)

Option #1: Classifiers Estimate

Using data in Table 3.10, fing the following four values:

1. Estimate the Multinomial Naive Bayes classifier
2. Apply the classifier to the test document
3. Estimate a Bernoulli NB classifier
4. Apply the classifier to the test document. For this data, you don't have to estimate parameters that you don't need for classifying the test document.

► **Table 13.10** Data for parameter estimation exercise.

	docID	words in document	in $c = China$?
training set	1	Taipei Taiwan	yes
	2	Macao Taiwan Shanghai	yes
	3	Japan Sapporo	no
	4	Sapporo Osaka Taiwan	no
test set	5	Taiwan Taiwan Sapporo	?

Now implement your solution in Python and submit all related files in a zipped folder. Consult the resources linked in assignment for additional insights.

Your paper should be 2-3 pages in length and conform to *CSU-Global Guide to Writing and APA*. Include at least two scholarly references in addition to the course textbook. The CSU-Global Library is a good place to find these references.

Option #2: Documents Fraction Computation

Figure 13.2 contains a class of priors that are computed as the fraction of documents in the class as opposed to the fraction of tokens in the class. Explain the reason behind this.

Figure 13.2: Naïve Bayes Algorithm (multinomial model): Training and Testing.

```
TRAINMULTINOMIALNB(C, D)
 1  V ← EXTRACTVOCABULARY(D)
 2  N ← COUNTDOCS(D)
 3  for each c ∈ C
 4  do N_c ← COUNTDOCSINCLASS(D, c)
 5     prior[c] ← N_c/N
 6     text_c ← CONCATENATETEXTOFALLDOCSINCLASS(D, c)
 7     for each t ∈ V
 8     do T_ct ← COUNTTOKENSOFTERM(text_c, t)
 9     for each t ∈ V
10     do condprob[t][c] ← (T_ct+1) / (Σ_t'(T_ct'+1))
11  return V, prior, condprob

APPLYMULTINOMIALNB(C, V, prior, condprob, d)
 1  W ← EXTRACTTOKENSFROMDOC(V, d)
 2  for each c ∈ C
 3  do score[c] ← log prior[c]
 4     for each t ∈ W
 5     do score[c] += log condprob[t][c]
 6  return arg max_c∈C score[c]
```

To justify your answer, implement your solution in Python and submit all related files in a zipped folder. Consult the resources linked in the assignment for additional insights.

Your paper should be 2-3 pages in length and conform to *CSU-Global Guide to Writing and APA*. Include at least two scholarly references in addition to the course textbook. The CSU-Global Library is a good place to find these references.

Mastery Exercise (10 points)

Module 6

Readings

- Chapter 14 in *An Introduction to Information Retrieval*
- Manning, C. D., Raghaven, P., & Schutze, H. (2008). Vector space model for scoring. In *Introduction to information retrieval.* Retrieved from https://nlp.stanford.edu/IR-book/html/htmledition/the-vector-space-model-for-scoring-1.html
- Mills, C., & Haiduc, S. (2017). The impact of retrieval direction on IR-based traceability link recovery. In *Proceedings of the 39th International Conference on Software Engineering: New Ideas and Emerging Results Track (ICSE-NIER '17).* IEEE Press, Piscataway, NJ, USA, 51-54. doi: https://doi-org.csuglobal.idm.oclc.org/10.1109/ICSE-NIER.2017.14
- Park, Y., Hwang, H., & Lee, S. (2015). A fast k-nearest neighbor search using query-specific signature selection. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM '15).* ACM, New York, NY, USA, 1883-1886. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/2806416.2806632

Opening Exercise (0 points)

Discussion (25 points)

Critical Thinking (70 points)

OPTION #1: K Nearest Neighbor Implementation

Describe why the K Nearest Neighbor handles multimodal classes better than the Rocchio method.

To justify your answer, implement KNN solution in Python and submit all related files in a zipped folder. Consult the resources linked in the assignment for additional insights.

Your paper should be 2-3 pages in length and conform to *CSU-Global Guide to Writing and APA*. Include at least two scholarly references in addition to the course textbook. The CSU-Global Library is a good place to find these references.

OPTION #2: Creating a Training Set

Create a training set of 150 documents, 50 each from three different languages (e.g., English, French, Spanish). Create a test set by the same procedure, but also add 50 documents from a fourth language. Train (i) a one-of classifier (ii) an any-of classifier on this training set and evaluate it on the test set. (iii) Are there any interesting differences in how the two classifiers behave on this task?

To justify your answer, implement your solution in Python and submit all related files in a zipped folder. Consult the resources linked in the assignment for additional insights.

Your paper should be 2-3 pages in length and conform to *CSU-Global Guide to Writing and APA*. Include at least two scholarly references in addition to the course textbook. The CSU-Global Library is a good place to find these references.

Mastery Exercise (10 points)

Module 7

Readings

- Chapter 15 in *An Introduction to Information Retrieval*
- Gorro, K.D., Sabellano, M. J. G., Maderazo, C. V., Ceniza, A. M., & Gorro, K. (2017). Exploring Facebook for sharing crime experiences using selenium and support vector machine. In *Proceedings of the 2017 International Conference on Information Technology (ICIT 2017).* ACM, New York, NY, USA, 218-222. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3176653.3176692
- Levi, O., Raiber, F., Kurland, O., & Guy, I. (2016). Selective cluster-based document retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16).* ACM, New York, NY, USA, 1473-1482. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/2983323.2983737
- Manning, C. D., Raghaven, P., & Schutze, H. (2008b). Support vector machine (SVM). In *Introduction to information retrieval.* Retrieved from https://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-and-machine-learning-on-documents-1.html

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Portfolio Milestone (25 points)

Options #1 and 2:

In your first three Critical Thinking Assignments, chances are you missed a concept and did not earn all the possible points. This is an opportunity to correct any mistakes you may have made in the previous assignments. So make appropriate corrections to the code you submitted in Modules 4-6. Corrections should reflect feedback from your instructor and improvements in execution, organization, and style. Resubmit your programs from Modules 4-6 with all outlined corrections. If no corrections are necessary, resubmit your original Critical Thinking Assignments.

Module 8

Readings

- Chapter 16 in *An Introduction to Information Retrieval*
- Barton, T., Bruna, T., & Kordik, P. (2019). Chameleon 2: An improved graph-based clustering algorithm. *ACM Transactions on Knowledge Discovery from Data (TKDD), 13*(1). doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3299876
- Hu, Q., Wu, J., Bai, L., Zhang, Y., & Cheng, J. (2017). Fast K-means for large scale clustering. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM '17).* ACM, New York, NY, USA, 2099-2102. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/3132847.3133091
- PhridviRaj, M. S. B., & Guru Rao, C. V.(2015). An approach for clustering text data streams using K-means and ternary feature vector based similarity measure. In *Proceedings of the International Conference on Engineering & MIS 2015 (ICEMIS '15).* ACM, New York, NY, USA. doi: https://doi-org.csuglobal.idm.oclc.org/10.1145/2832987.2833081

Opening Exercise (0 points)

Discussion (25 points)

Mastery Exercise (10 points)

Portfolio Project (300 points)

OPTION #1: Search Result Clustering

Your Portfolio Project for CSC460 consists of the following:

- Module 4 Milestone (due in Module 4)
- Module 7 Milestone (due in Module 7)
- Lessons Learned Reflection (from all modules)
- Final Project

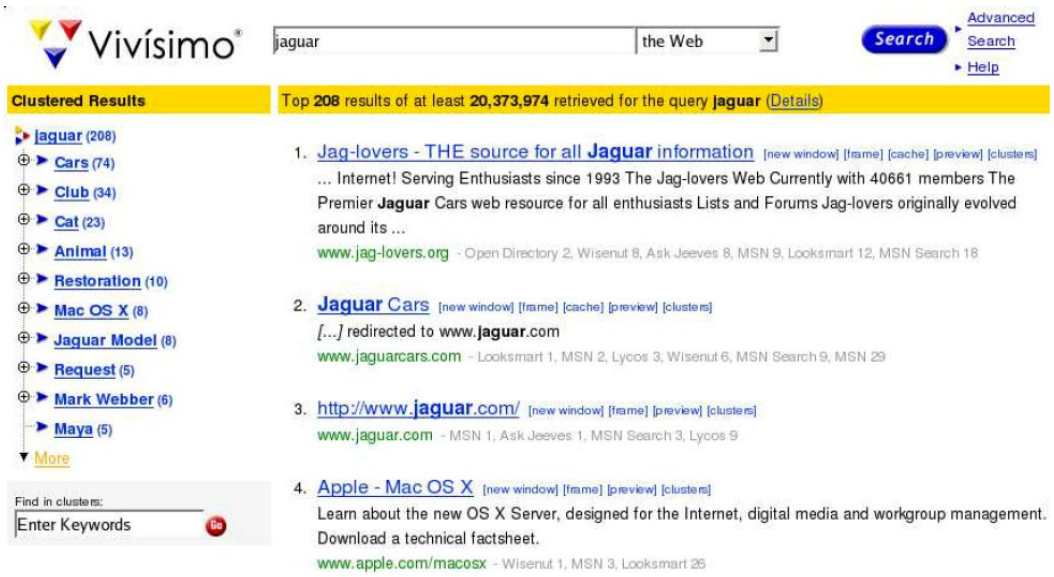
In Week 8, the components left to complete for your Portfolio Project are the **Lessons Learned Reflection** and the **Final Project**. Carefully review the requirements below:

Lessons Learned Reflection:

Write a 2- to 3-page summary that outlines the lessons you have learned in this course. Reflect on how these lessons can be applied to real-world problems or to a specific real world application. How have they impacted (or how could they impact) your life?

Final Project:

SEARCH RESULT CLUSTERING: The figure below shows a Search Result Clustering whereby the "search results" are defined as documents that were returned in response to a query. The default presentation of search results in information retrieval is a simple list. Users scan the list from top to bottom until they have found the information they are looking for. Search result clustering, on the other hand, clusters the search results so that similar documents appear together. It is often easier to scan a few coherent groups than many individual documents. This is particularly useful if a search term has different word senses. The example in Figure 16.2 uses the search term "jaguar." Three frequent senses on the web refer to the car, the animal and an Apple operating system. The Clustered Results panel returned by the Vivísimo search engine (http://www.searchtools.com/tools/vivisimo.html) can be a more effective user interface for understanding what is in the search results than can a simple list of documents.



For this assignment, you will develop Search Result Clustering Application using Python. Submit all related files in a zipped folder. Consult the resources linked in the assignment for additional insights.

Include your reflection in the zipped folder of files for your final submission.

OPTION #2: Scatter-Gather

Your Portfolio Project for CSC460 consists of the following:

- Module 4 Milestone (due in Module 4)
- Module 7 Milestone (due in Module 7)
- Lessons Learned Reflection (from all modules)
- Final Project

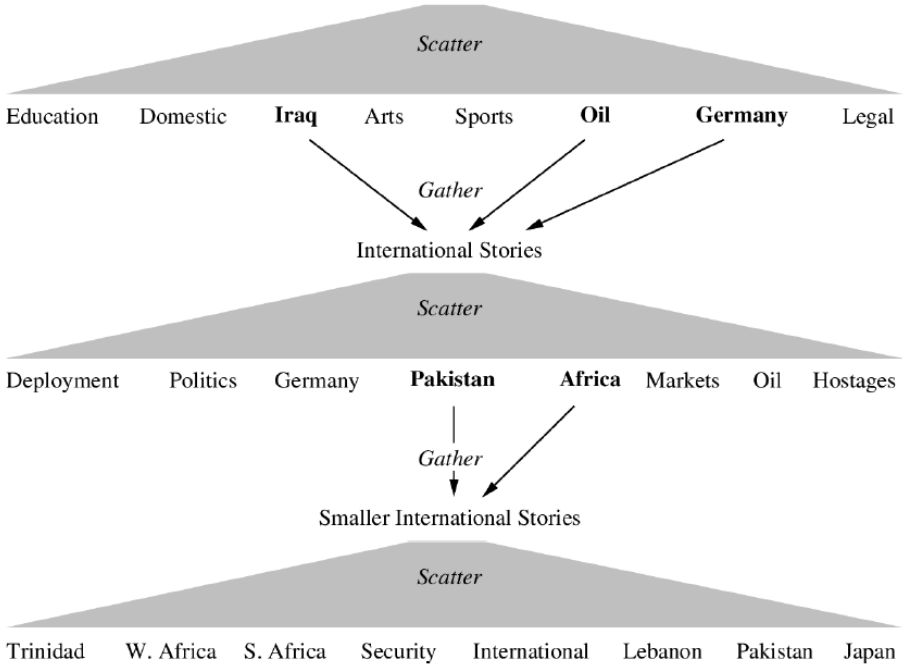
In Week 8, the components left to complete for your Portfolio Project are the **Lessons Learned Reflection** and the **Final Project**. Carefully review the requirements below:

Lessons Learned Reflection:

Write a 2- to 3-page summary that outlines the lessons you have learned in this course. Reflect on how these lessons can be applied to real-world problems or to a specific real world application. How have they impacted (or how could they impact) your life?

Final Project:

SCATTER-GATHER: A better user interface is the goal of Scatter-Gather, the second alternative to Search Result Clustering. Scatter-Gather clusters the whole collection to get groups of documents that the user can select or gather. The selected groups are merged and the resulting set is again clustered. This process is repeated until a cluster of interest is found. An example is shown in the below figure. In the below figure, A collection of *New York Times* news stories is clustered ("scattered") into eight clusters (top row). The user manually gathers three of these into a smaller collection (International Stories) and performs another scattering operation. This process repeats until a small cluster with relevant documents is found (e.g., Trinidad).



For this assignment, you will develop a Scatter-Gather Application using Python. Submit all related files in a zipped folder. Consult the resources linked in the assignment for additional insights.

Include your reflection in the zipped folder of files for your final submission.

COURSE POLICIES

Grading Scale	
A	95.0 – 100
A-	90.0 – 94.9
B+	86.7 – 89.9
B	83.3 – 86.6
B-	80.0 – 83.2
C+	75.0 – 79.9
C	70.0 – 74.9
D	60.0 – 69.9
F	59.9 or below

Course Grading

20% Discussion Participation
0% Opening Exercises
0% Live Classroom
8% Mastery Exercises
37% Critical Thinking Assignments
35% Final Portfolio Project

IN-CLASSROOM POLICIES

For information on late work and incomplete grade policies, please refer to our **In-Classroom Student Policies and Guidelines** or the Academic Catalog for comprehensive documentation of CSU-Global institutional policies.

Academic Integrity
Students must assume responsibility for maintaining honesty in all work submitted for credit and in any other work designated by the instructor of the course. Academic dishonesty includes cheating, fabrication, facilitating academic dishonesty, plagiarism, reusing /repurposing your own work (see *CSU-Global Guide to Writing & APA* for percentage of repurposed work that can be used in an assignment), unauthorized possession of academic materials, and unauthorized collaboration. The CSU-Global Library provides information on how students can avoid plagiarism by understanding what it is and how to use the Library and internet resources.

Citing Sources with APA Style
All students are expected to follow the *CSU-Global Guide to Writing & APA* when citing in APA (based on the most recent APA style manual) for all assignments. A link to this guide should also be provided within most assignment descriptions in your course.

Disability Services Statement
CSU-Global is committed to providing reasonable accommodations for all persons with disabilities. Any student with a documented disability requesting academic accommodations should contact the Disability Resource Coordinator at 720-279-0650 and/or email ada@CSUGlobal.edu for additional information to coordinate reasonable accommodations for students with documented disabilities.

Netiquette
Respect the diversity of opinions among the instructor and classmates and engage with them in a courteous, respectful, and professional manner. All posts and classroom communication must be conducted in accordance with the student code of conduct. Think before you push the Send button. Did you say just what you meant? How will the person on the other end read the words?

Maintain an environment free of harassment, stalking, threats, abuse, insults, or humiliation toward the instructor and classmates. This includes, but is not limited to, demeaning written or oral comments of an ethnic, religious, age, disability, sexist (or sexual orientation), or racist nature; and the unwanted sexual advances or intimidations by email, or on discussion boards and other postings within or connected to the online classroom. If you have concerns about something that has been said, please let your instructor know.