XSEDE Tutorial Session

Proposal submitted: XSEDE16TutorialProposal_v1.docx

Abstract

When: July 18th

Where: XSEDE16, www.xsede.org/xsede16

Tutorial Session Design

Action Plan

June 6th - June 24th - 3 weeks for Development, Environment setup preparation, and reviews of tutorial materials as they are developed.

June 27th - 1st July -   Fine tuning/deployment

2nd July- 8th July - Testing and Demo to BD students

11th July, 13th July - Final Touch up

Tutorial Presenter registration:

All tutorial authors (including teaching assistants) can register as tutorial presenter. The password for doing so is TUTXSEDE16. The conference registration fee is waived for the day of the tutorial.

Introduction to Brown Dog (30m)

This is a presentation and demo of Brown Dog project and service.

Highlights some XSEDE projects that are use cases of Brown Dog projects - VAT, DEBOD, IARP

How to use Brown Dog Services (1h 30m)

This is a session to teach how to use Brown Dog Services

Time break up:

- 20-30 mins environment setup
- 1 hr for examples hands-on

Hands-on outline

- 2 problems + 1 optional problem
- 1 Extraction example - Given a collection of images with text embedded in it and audio files, extract all metadata using (face , ocr, speech2text) extractors, index and do content-based retrieval
- 1 Conversion example - Given a collection of old file formats of images and audios, convert to a format which participants can open
- 1 Combined example - Given an audio format, convert it to a format *.mp3 and then uses speech2text extractor to obtain the text.
- 1 Optional combined example

Environment Set-up Details (20 mins)

- Participant will use his/her own laptop for this part

TODOS

- We will provide a VM with everything pre-installed in it **through Nebula.**
  - ☑ **Rob Kooper will talk to Doug for this if we can spawn 50 VMs on Nebula for the tutorial session. (DONE) We will get 50 VMs on Nebula.**
  - ☑ **Smruti Padhy Make a list of all softwares required and the directory structure for the tutorial**

- ☑ **Smruti PadhyCreate a VM with everything installed in it and take a snapshot which will then be deployed within Nebula. Approx. time required - 2 days**
- ☑ **Rui Liu Write a script for deployment of 50 VMs from the VM snapshot that we created.**
- □ **Luigi Marini Testing BD service with 50 concurrent users to perform conversion/extractions tasks**
- ☑ **Luigi Marini Maximum size of file that can be uploaded to Brown Dog needs to be controlled. This is require to ensure no one uploads any large files.**
- ☑ ~~**Not sure of Jetstream yet.**~~
  - Provide clear instructions as how to access VMs in Nebula with proper credentials.
    - ☑ **Smruti Padhy, Marcus Slavenas, Jong Lee Create powerpoint slides with clear Instructions of how to access the VMs (e.g., through ssh), from different OSes (Linux, Mac, Windows).**
    - ☑ **Rui Liu Need training accounts on Nebula. Provide SSH key-pairs to each participant.**
  - (Before tutorial - wiki pages with clear instructions) Installs Python/R/MATLAB/cURL to use BD Service along with the library required in case any one interested in using the BD services in future.
    - □ **Eugene Roeder Create wiki pages with clear instructions**

- **Backup -** Provide VM through USB sticks in case of network interruption
  In addition to the installation required for VMs in Nebula, following are extra steps required for Backups VMs
  - **TODO**
    - □ **Smruti Padhy Convert the VM created (using Openstack image) to VirtualBox format (*vdi) and test the configurations.**
    - ☑ **Smruti Padhy Order 50+ flash drive for back up that will contain the VMs (Received)**
    - ☑ **Luigi Marini Test local installation of Fence with local authentication instead of Crowd. This is for backup to be provided in the preinstalled VM.**
    - ☑ **Smruti Padhy bdfiddle installation**
    - □ **Copy the VM and virtualbox to the flash drives**

Hands-on Details

- Demonstration of use of BD Fiddle (**20 mins**)
  - Sign up for Brown Dog Service
  - Obtain a key/token using curl on VM
    - Local VM and Nebula VM need to have curl available
  - Use token and bd fiddle interface to obtain to see BD in action.
  - Copy paste the python code snippet and use it the application to be explained next.
  - □ **Kenton McHenry Figure out a good way to accept requests for new accounts for BD service (note this is different than Nebula account)**
    - □ **Jay Alameda Aquire list of attendees**
  - □ **Smruti Padhy, Marcus Slavenas, Jong Lee Create a document for the demo with step-by-step screenshots for all above steps.**
  - ☑ **Eugene Roeder Fix the CORS error for file url option (I think it is a known issue). Please add the JIRA issue number here.**
  - □ **Christopher Navarro, Eugene Roeder Fix the delay experienced when file is uploaded from local directory to the bdfiddle ui**

- Create applications using BD services (**50 mins)**
  - **<u>Conversion Example (20mins):</u>** To convert a collection of images/ps/odp/audio/video files to png/pdf/ppt/mp3. This will demonstrates that if you have a directory with files in old file formats, just use BD to get it all converted without requiring to install any software. (Emphasizes on conversion)
    - Make sure imagemagick and ffmpeg converters are running before the demo
    - Obtain the BD token/key - ask participant to refer to previous bdfiddle step or use the python library
    - Ask the participant to check for the available output formats for specific input formats
    - Ask the participant to use python library to use BD service
    - **TODO:**
      - ☑ **Inna Zharnitsky Provide a Python script for this and let participants use python library to use the BD service**
      - □ **Smruti PadhyProvide a Step-by-step instructions with screenshot to do this**
      - □ **(Optional) Provide R script for this problem.**
  - **<u>Extraction/Indexing/Retrieval Example (20 mins):</u>** Given a collection of images with text embedded in it, and audio files, search images/audio files based on its content. (**Emphasizes on extraction on unstructured data, indexing and content-based retrieval**)
    - Make sure OCR, face, and speech2text extractors are running before starting the demo
    - One can upload images from local directory to obtain images or use external web service.
    - Let the participant use the python library of BD to obtain key/token and submit extraction request to BD-API gateway

- Once technical metadata is obtained from BD, write the tags and technical metadata to a local file /python dictionary.
- Search the file based on the tags/technical metadata by linear search on the index file

- **TODO**
  - ☑ **Make sure the metadata to be posted as json-ld for the extractors**
    - ☑ **OCR**
    - ☑ **Face**
    - ☑ **Eyes**
    - ☑ **Closeup**
    - ☑ **Smruti Padhy speech2text**
  - ☐ **Create an example dataset with images and audios to which we can make interesting query**
  - ☐ **(Optional) Provide a code snippet of using externel service to obtain images. e.g. Flicker API.**
    - This will only be provided as an example and will not be used for the rest of the code.
  - ☐ **Provide the link to the current BD REST API and create a document/wiki page showing step-by-step screenshots of obtaining a key/token using python library.**
  - ☐ **Marcus SlavenasWrite a Python script that will serve as a stub for the BD client**
    - The participant will fill in the code to use python library to call BD REST API and submit their requests.
    - The python script should write the tags and technical metadata to a local file. (Probably can use python library's index method that writes it as feature vectors.)
  - ☐ **Marcus SlavenasWrite a Python script to make interesting search/query to the index file. Again probably use the python library's find method or just read the local file.**
  - ☐ **(Optional) provide R script for this problem**

- **Conversion & Extraction Example (10 min):** Given an image file, convert it to a different format and obtains the face detection and OCR.
  - **TODO**
    - ☑ **Write a Python script that does the conversion and then sends the converted file to the BD service.**
    - ☑ **Create a step-by-step instructions document with screenshots.**
    - ☐ **(Optional) Provide a R script for this problem**

- **(Optional) Combination of Conversion & Extraction Example:** Obtaining Ameriflux data and converting into *.clim format (similar to csv format but tab separated) for SNIPET model. Calculate average air temperature and its standard deviation. (This will emphasize both conversion and analysis)
  - ☐ **Write a R/Python script to call BD conversion API and get data in *clim format and calculate average air temperature. Also plot a graph of the data.**
    - ☐ **Installation of Rstudio server version.**
  - ☐ **(Optional) to calculate average temperature, call BD extraction service. For this write an extractor that accepts *.clim file and outputs average temperature.**

How to add Your Tool to Brown Dog Services (1h 30m)

This is a session to teach how to add user's tool to Brown Dog Services.

- **Part 1: Teach to write an extractor (35 mins)**
  - Start with the bd-template extractor, which is the word count extractor.
  - Ask participant to modify the extractor, which would use '*grep'* to find a specific pattern within the file.
  - Ask to change the name of the extractor from ncsa.wordcount to ncsa.grep.
  - Include yes/no in the metadata if the pattern is found or not found.
  - Briefly describe Json-ld support. Provide intuition behind the idea json-ld with a simple example. No need to go into details of RDF.
  - **TODO**
    - ☑ **Smruti Padhy, Marcus Slavenas, Jong Lee Provide Step-by-step instructions/screenshots of updating the extractor and the output as seen at the Clowder GUI. Also provide link to json-ld for further readings. Provide minimum software requirements for the development such as Clowder, Rabbimq, MongoDB, pyclowder, python libraries, etc.**
    - ☑ **Inna Zharnitsky Write an extractor that does *grep* along with the wordcount for demonstration purpose and include json-ld**
    - ☑ **Sandeep Puthanveetil Satheesan Write an extractor that accepts csv file with say 3 columns (probably with values from weather or bacterial growth model (see Problem 2.2 below)) , calculate the average of a specific column**

        - ☑ Provide step-by-step screenshots for writing such an extractor.

- **Part 2: Teach to write a converter (35 mins)**
    - Start with the bd-template for converter- imagemagick
    - Ask the participant to modify the converter input/output formats in the comment section. And see the result using the polyglot web UI for post and get
    - Another example - FFmpeg converter for audio and video
    - **TODO**
        - ☐ **Smruti Padhy, Marcus Slavenas, Jong Lee Provide step-by-step instructions/screenshots of modifying imagemagick and usage of polyglot GUI. Provide a default username/password**
        - ☑ Marcus Slavenas, Kenton McHenry Write a converter using FFmpeg

- **Part 3: Teach to upload a converter or an extractor to the locally installed Tools Catalog. (20mins)**
    - ☑ Inna Zharnitsky Step-by-step procedure to upload an extractor or a converter, an input file and an output file without a docker file.

- **Part 4 (Optional - For advanced user): Dockerize the tool**

Use Contributors Landing Page for this part of the session.

Participant will be provided with a VM with all required setup so that they can create their own tool.

Wrap up

- **TODO :** Jay Alameda **Design Tutorial feedback forms (SEA workshop form for Eclipse PTP included as a samlple):**

- **TODO :** Jay Alameda **Print the feedback forms**
- Feed back form - updated

- Announcement of next user workshop

-------- OPTIONAL EXAMPLES --------

- **<u>Problem 1 :</u>** Given a collection of text files from a survey or reviews for a book/movie, use sentiment analysis extractor to calculate the sentiment value for each file and group similar values together. (**Emphasizes on extraction on unstructured data and useful analysis**)
    - A collection of text files with reviews
        - □ **Obtain an examples dataset from the web**.
    - Let the participant use the python library of BD to obtain key/token and submit request to BD-API gateway
        - □ **Provide the link to the current BD REST API and create a document/wiki page showing step-by-step screenshots of obtaining a key/token using python library.**
        - □ **Write a Python script that will serve as a stub for the BD client**
            - The participant will fill in the code to BD REST API call to submit their requests.
    - Make sure the Sentiment Analysis extractor is running
    - Saves the results for each text file in a single file with corresponding values
        - □ **Provide code for this in stub script**
    - Create separate folders and move the file based on the sentiment value
        - □ **Provide a code that will do the above action in the stub**
    - **(Optional) Index text files along with the sentiment values and use ES visualization tool to search for documents with sentiment value less than some number.**
- **<u>Problem 2:</u>** Given a collection of *.xlsx files, obtain some results based on some columns value. (**Emphasizes on extraction and analysis on scientific data**)

An example could be - Given a *.xlsx file with max and min temperature for each day of a month. Calculate average temperature max/min/standard deviation for each month.

- ■ Convert *.xlsx file to *.csv using conversion API so that you can see the content of the file on the VM. We are not installing any office software on the VM.
- ■ use extraction API to extract columns from the file and
- ■ Perform some analysis and add to the technical metadata
- □ **Write an extractor/converter** for this problem
    - This should be an enticing yet simple problem that can handle many spreadsheets and get a result.
    - Ideas
        1. An algebra 101, traveling trains problem. 2 trains leave 2 different stations on tracks heading toward a junction. Given a spreadsheet with departure times, distances, velocities, etc., upload all the spreadsheets and determine if they will crash.
            a. This problem is simple and would provide the user an easily understood problem that can clearly be scaled to much more involved traffic problems.
            b. However, it doesn't really present a cool new idea. It may be preferable to think of something more cutting edge technology
        2. A bacterial growth model. Given a culture with varied conditions, eg. pH, stored in multiple spreadsheets, determine the growth rate. Might be able to base this on http://mathinsight.org/bacteria_growth_initial_model
            a. This would require a few minutes of explanation of the model and would require some learning by the developer of the extractor.
            b. Still maybe not that enticing.
        - Better Ideas?

□ **Provide a Python script for obtaining the input files and use BD REST API for to obtain the result.**