Earth Mover’s Web Service Searcher

Dr. Ngu, Scott Julian, Xiaojie Jiang
Computer Science Dept. Texas State University

Introduction

WHAT IS A WEB SERVICE?

- Web services are modular, self-describing, and loosely coupled software components that can be located and used over the Internet
- A web service is defined by a WSDL file
  - The WSDL (Web Service Description Language) standard specifies the interface of a service in terms of operations and messages
- Because of their reusability and platform independence, using web services in applications is becoming a widely popular and successful way of creating cloud based applications
- Finding relevant Web services for creating useful and robust applications is becoming an emergent and challenging research problem

Problem

PROBLEM – FINDING WEB SERVICES

- Increasing number of available web services
  - (e.g. programmableweb.com)
- Finding relevant web services is becoming more important and challenging because of the increase of web services
- Searching for a web service has moved from repositories, like UDDI, to web based search engines
  - (e.g. seekda.com)

PROBLEM – SEARCH ENGINES

- Current web based search engines are designed for searching through webpages that is unstructured or semi-structure like html page
- Current web based search engines are not designed to take advantage of the structure of WSDL file for finding web services
- Typically Vector Space Model is used to compute the similarity between keywords in the query against keywords at fixed position in the service
- Exact keyword matching is used (e.g. Seekda) which does not take account the impact on the similarity of the neighborhood keywords.

Goal

OUR GOAL

- To introduce EMD as a partial keyword matching to existing web service search engine.
- To experimentally evaluate the performance of EMD against Vector Space Model for Web Service Retrieval
- To deploy EMD service search engine as a web application that can be used by WSDL providers or WSDL consumers

$$\mathrm{EMD}(P,Q) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}d_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}}\ .$$

EMD Explanation

EMD – EARTH MOVER’S DISTANCE

- EMD is an evaluation method of dissimilarity between two multi-dimensional distributions
- Given two distributions, one can be seen as a mass of earth properly spread in space, the other as a collection of holes in that same space. Then, the EMD measures the least amount of work needed to fill the holes with earth
- Computing the EMD is based on the solution to the well-known transportation problem

EMD – EARTH MOVER’S DISTANCE

- EMD describes the normalized amount of work required to transform one distribution to the other
- The subtask of our project is to find the EMD between a keyword and an attributed word sequence from a query and a record, which describes their similarity
- This process will be performed for all records, and the one with the lowest EMD will be returned as the highest ranked result

Computing EMD

COMPUTING EMD

- Let any subtask to involve two word sequence *Kw* and *Aw*(keyword and attributed word), where $|Kw| = n_1$ and $|Aw| = n_2$
- Finding the minimum amount of work to transfer *Kw* to *Aw* we use the following linear program(LP) with variables f_{ij}(flow) and d_{ij} (ground distance matrix between *Kw* and *Aw*)

$$\begin{aligned}
&\text{minimize} : \textstyle\sum_{i=1}^{n_1}\sum_{j=1}^{n_2} f_{ij}d_{ij} \\
&\text{subject to} : \\
&\forall 1 \le i \le n_1 : \textstyle\sum_{j=1}^{n_2} f_{ij} \le w_{kw_i} && (1.1)\\
&\forall 1 \le j \le n_2 : \textstyle\sum_{i=1}^{n_1} f_{ij} \le w_{aw_j} && (1.2)\\
&\forall 1 \le i \le n_1, 1 \le j \le n_2 : f_{ij} \ge 0 && (1.3)\\
&\textstyle\sum_{i=1}^{n_1}\sum_{j=1}^{n_2} f_{ij} = \min\left(\sum_{i=1}^{n_1} w_{kw_i}, \sum_{j=1}^{n_2} w_{aw_j}\right) && (1.4)
\end{aligned}$$

COMPUTING EMD

- From the linear program, if we assume that the optimal flow *f** is found, we obtain the following EMD equation
- To compute the EMD between two distributions we use the following formula:

$$EMD\,(\mathbf{kw}, \mathbf{aw}) = \frac{\sum_{i=1}^{n_1}\sum_{j=1}^{n_2} f^*_{ij}d_{ij}}{\sum_{i=1}^{n_1}\sum_{j=1}^{n_2} f^*_{ij}}$$

SED & LD

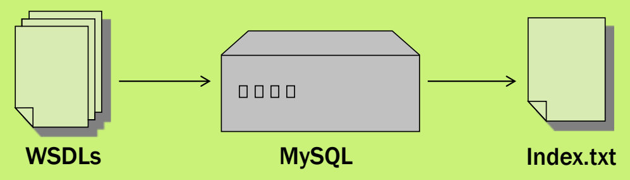
S.E.D. & L.D.

- However, *SED* is not robust enough to capture words with similar structures
  - *SED*(‘Sale’, ‘Wholesale’) = 9
- For words that have a common prefix of suffix, then EMD will use the LD technique to compute the weight
- Edit distance(*LD*) between a *Kw* and an *Aw* is the least number of character insertions, deletions, and substitutions required to transform one to the other
  - *LD*(‘Sale’, ‘Wholesale’) = 5 since a minimum of 5 insertions are required

Methods

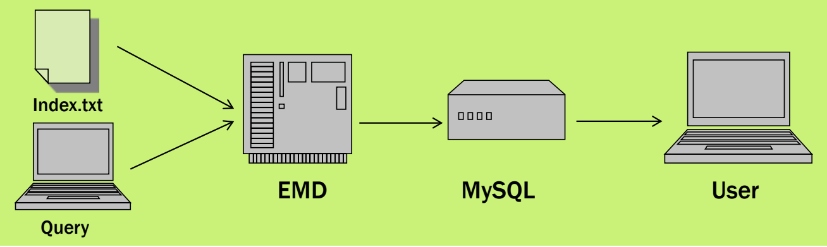
OUR APPROACH

- 1) Parse WSDL files
  - Name, documentation, locations, IO names
- 2) Store WSDL data in a MySQL server
  - Assign each web service a unique ID
- 3) Index the data into index.txt
  - Create an index file for EMD. Each line in the index is a web service. Each web service is treated as a bag of words



OUR APPROACH

- 4) Get user’s query and compute EMD
  - for each web service in the index file
- 5) Get ranked results from EMD
  - EMD will return the top 25 closest matched web services
- 6) Fetch web services from database
  - Return the matched web services to the user in ranked order



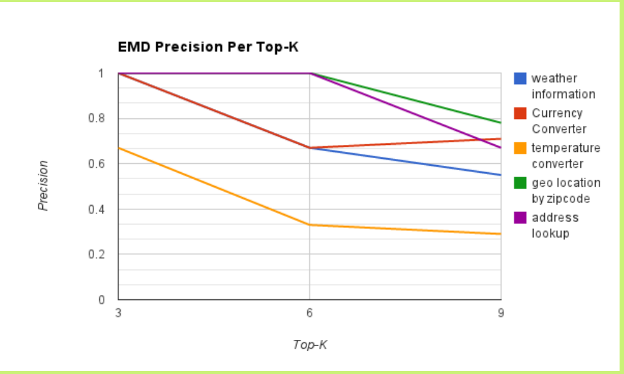
VECTOR SPACE MODEL

- To test our approach, we compared precision and recall to the current exact keyword matching standard which is the Vector Space Model (VSM)
- The weight of the keyword in the VSM is defined:
  - w = tf * idf
- We utilize MySQL’s normalized factor form of the VSM
  - w = (log(dtf)+1)/sumdtf * U/(1+0.0115*U) * log((N-nf)/nf)

Result

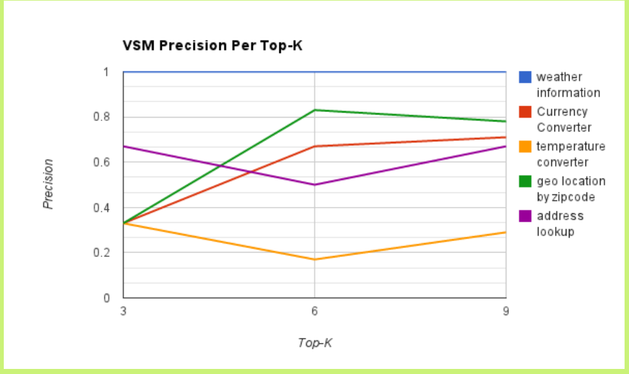
EMD PRECISION

- Database Size: 454



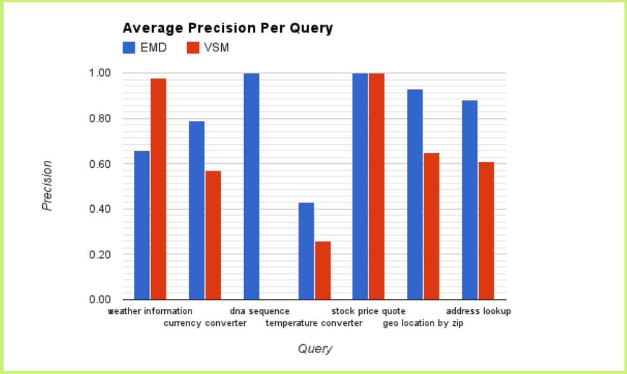
VSM PRECISION

- Database Size: 454



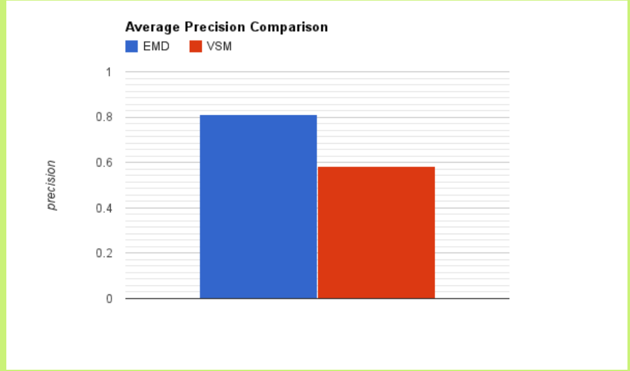
AVG PRECISION PER QUERY

- Database Size: 454
- EMD vs VSM



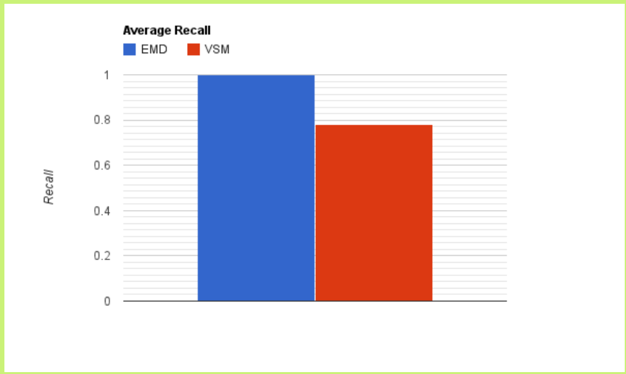
AVERAGE PRECISION

- Database Size: 454
- EMD vs VSM



AVERAGE RECALL

- Database Size: 454
- EMD vs VSM



TEXAS STATE UNIVERSITY
SAN MARCOS
The rising STAR of Texas

NSF