Query log analysis with GALATEAS LangLog

Marco Trevisan and **Luca Dini**
CELI
trevisan@celi.it
dini@celi.it

Eduard Barbu
Università di Trento
eduard.barbu@unitn.it

Igor Barsanti
Gonetwork
i.barsanti@gonetwork.it

Nikolaos Lagos
Xerox Research Centre Europe
Nikolaos.Lagos@xrce.xerox.com

Frédérique Segond and **Mathieu Rhulmann**
Objet Direct
fsegond@objetdirect.com
mruhlmann@objetdirect.com

Ed Vald
Bridgeman Art Library
ed.vald@bridgemanart.co.uk

Abstract

This article describes GALATEAS LangLog, a system performing Search Log Analysis. LangLog illustrates how NLP technologies can be a powerful support tool for market research even when the source of information is a collection of queries each one consisting of few words. We push the standard Search Log Analysis forward taking into account the semantics of the queries. The main innovation of LangLog is the implementation of two highly customizable components that cluster and classify the queries in the log.

1 Introduction

Transaction logs become increasingly important for studying the user interaction with systems like Web Searching Engines, Digital Libraries, Intranet Servers and others (Jansen, 2006). Various service providers keep log files recording the user interaction with the searching engines. Transaction logs are useful to understand the user search strategy but also to improve query suggestions (Wen and Zhang, 2003) and to enhance the retrieval quality of search engines (Joachims, 2002). The process of analyzing the transaction logs to understand the user behaviour and to assess the system performance is known as Transaction Log Analysis (TLA). Transaction Log Analysis is concerned with the analysis of both browsing and searching activity inside a website. The analysis of transaction logs that focuses on search activity only is known as Search Log Analysis (SLA). According to Jansen (2008) both TLA and SLA have three stages: data collection, data preparation and data analysis. In the data collection stage one collects data describing the user interaction with the system. Data preparation is the process of loading the collected data in a relational database. The data loaded in the database gives a transaction log representation independent of the particular log syntax. In the final stage the data prepared at the previous step is analyzed. One may notice that the traditional three levels log analyses give a syntactic view of the information in the logs. Counting terms, measuring the logical complexity of queries or the simple procedures that associate queries with the sessions in no way accesses the semantics of queries. LangLog system addreses the semantic problem performing clustering and classification for real query logs. Clustering the queries in the logs allows the identification of meaningful groups of queries. Classifying the queries according to a relevant list of categories permits the assessment of how well the searching engine meets the user needs. In addition the LangLog system address problems like automatic language identification, Name Entity Recognition, and automatic query translation. The rest of the paper is organized as follows: the next section briefly reviews some systems performing SLA. Then we present the data sources the architecture and the analysis process of the LangLog system. The conclusion section concludes the article summarizing the work and presenting some new possible enhancements of the LangLog.

Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pages 87–91,
Avignon, France, April 23 - 27 2012. ©2012 Association for Computational Linguistics

2 Related work

The information in the log files is useful in many ways, but its extraction raises many challenges and issues. Facca and Lanzi (2005) offer a survey of the topic. There are several commercial systems to extract and analyze this information, such as Adobe web analytics[1], SAS Web Analytics[2], Infor Epiphany[3], IBM SPSS[4]. These products are often part of a customer relation management (CRM) system. None of those showcases include any form of linguistic processing. On the other hand, Web queries have been the subject of linguistic analysis, to improve the performance of information retrieval systems. For example, a study (Monz and de Rijke, 2002) experimented with shallow morphological analysis, another (Li et al., 2006) analyzed queries to remove spelling mistakes. These works encourage our belief that linguistic analysis could be beneficial for Web log analysis systems.

3 Data sources

LangLog requires the following information from the Web logs: the time of the interaction, the query, click-through information and possibly more. LangLog processes log files which conform to the W3C extended log format. No other formats are supported. The system prototype is based on query logs spanning one month of interactions recorded at the Bridgeman Art Library[5]. Bridgeman Art library contains a large repository of images coming from 8000 collections and representing more than 29.000 artists.

4 Analyses

LangLog organizes the search log data into units called queries and hits. In a typical searching scenario a user submits a query to the content provider's site-searching engine and clicks on some (or none) of the search results. From now on we will refer to a clicked item as a hit, and we will refer to the text typed by the user as the query. This information alone is valuable to the content provider because it allows to discover which queries were served with results that satisfied the user, and which queries were not.

LangLog extracts queries and hits from the log files, and performs the following analyses on the queries:

- language identification
- tokenization and lemmatization
- named entity recognition
- classification
- cluster analysis

Language information may help the content provider decide whether to translate the content into new languages.

Lemmatization is especially important in languages like German and Italian that have a rich morphology. Frequency statistics of keywords help understand what users want, but they are biased towards items associated with words with lesser ortographic and morpho-syntactic variation. For example, two thousand queries for ”trousers”, one thousand queries for ”handbag” and another thousand queries for ”handbags” means that handbags are twice as popular as trousers, although statistics based on raw words would say otherwise.

Named entities extraction helps the content provider for the same reasons lemmatization does. Named entities are especially important because they identify real-world items that the content provider can relate to, while lemmas less often do so. The name entities and the most important concepts can be linked afterwards with resources like Wikipedia which offer a rich specification of their properties.

Both classification and clustering allow the content provider to understand what kind of the users look for and how this information is targeted by means of queries.

Classification consists of classifying queries into categories drawn from a classification schema. When the schema used to classify is different from the schema used in the content provider's website, classification may provide hints as to what kind of queries are not matched by items in the website. In a similar way, cluster analysis can be used to identify new market segments or new trends in the user's behaviour. Clus-

[1] http://www.omniture.com/en/products/analytics

[2] http://www.sas.com/solutions/webanalytics/index.html

[3] http://www.infor.com

[4] http://www-01.ibm.com/software/analytics/spss/

[5] http://www.bridgemanart.com

ter analysis provide more flexybility than classification, but the information it produces is less precise. Many trials and errors may be necessary before finding interesting results. One hopes that the final clustering solution will give insights into the patterns of users' searches. For example an online book store may discover that one cluster contains many software-related terms, altough none of those terms is popular enough to be noticeable in the statistics.

5 Architecture

LangLog consists of three subsystems: log acquisition, log analysis, log disclosure. Periodically the log acquisition subsystem gathers new data which it passes to the log analyses component. The results of the analyses are then available through the log disclosure subsystem.

Log acquisition deals with the acquisition and normalization and anonymization of the data contained in the content provider's log files. The data flows from the content provider's servers to LangLog's central database. This process is carried out by a series of Pentaho Data Integration[6] procedures.

Log analysis deals with the anaysis of the data. The analyses proper are executed by NLP systems provided by third parties and accessible as Web services. LangLog uses NLP Web services for language identification, morpho-syntactic analysis, named entity recognition, classification and clustering. The analyses are stored in the database along with the original data.

Log disclosure is actually a collection of independent systems that allow the content providers to access their information and the analyses. Log disclosure systems are also concerned with access control and protection of privacy. The content provider can access the output of LangLog using AWStats, QlikView, or JPivot.

- AWStats[7] is a widely used log analysis system for websites. The logs gathered from the websites are parsed by AWStats, which generates a complete report about visitors, visits duration, visitor's countries and other data to disclose useful information about the visitor's behavior.
- QlikView[8] is a business intelligence (BI) platform. A BI platform provides historical, current, and predictive views of business operations. Usually such tools are used by companies to have a clear view of their business over time. In LangLog, QlickView does not display sales or costs evolution over time. Instead, it displays queries on the content provider's website over time. A dashboard with many elements (input selections, tables, charts, etc.) provides a wide range of tools to visualize the data.
- JPivot[9] is a front-end for Mondrian. Mondrian[10] is an Online Analytical Processing (OLAP) engine, a system capable of handling and analyzing large quantities of data. JPivot allows the user to explore the output of LangLog, by slicing the data along many dimensions. JPivot allows the user to display charts, export results to Microsoft Excel or CSV, and use custom OLAP MDX queries.

Log analysis deals with the anaysis of the data. The analyses proper are executed by NLP systems provided by third parties and accessible as Web services. LangLog uses NLP Web services for language identification, morpho-syntactic analysis, named entity recognition, classification and clustering. The analyses are stored in the database along with the original data.

5.1 Language Identification

The system uses a language identification system (Bosca and Dini, 2010) which offers language identification for English, French, Italian, Spanish, Polish and German. The system uses four different strategies:

- N-gram character models: uses the distance between the character based models of the input and of a reference corpus for the language (Wikipedia).
- Word frequency: looks up the frequency of the words in the query with respect to a reference corpus for the language.
- Function words: searches for particles highly connoting a specific language (such as prepositions, conjunctions).

[6] http://kettle.pentaho.com

[7] http://awstats.sourceforge.net

[8] http://www.qlikview.com

[9] http://jpivot.sourceforge.net

[10] http://mondrian.pentaho.com

- Prior knowledge: provides a default guess based on a set of hypothesis and heuristics like region/browser language.

5.2 Lemmatization

To perform lemmatization, Langlog uses general-purpose morpho-syntactic analysers based on the Xerox Incremental Parser (XIP), a deep robust syntactic parser (Ait-Mokhtar et al., 2002). The system has been adapted with domain-specific part of speech disambiguation grammar rules, according to the results a linguistic study of the development corpus.

5.3 Named entity recognition

LangLog uses the Xerox named entity recognition web service (Brun and Ehrmann, 2009) for English and French. XIP includes also a named entity detection component, based on a combination of lexical information and hand-crafted contextual rules. For example, the named entity recognition system was adapted to handle titles of portraits, which were frequent in our dataset. While for other NLP tasks LangLog uses the same system for every content provider, named entity recognition is a task that produces better analyses when it is tailored to the domain of the content. Because LangLog uses a NER Web service, it is easy to replace the default NER system with a different one. So if the content provider is interested in the development of a NER system tailored for a specific domain, LangLog can accomodate this.

5.4 Clustering

We developed two clustering systems: one performs hierarchical clustering, another performs soft clustering.

- CLUTO: the hierarchical clustering system relies on CLUTO4[11], a clustering toolkit. To understand the main ideas CLUTO is based on one might consult Zhao and Karypis (2002). The clustering process proceeds as follows. First, the set of queries to be clustered is partitioned in k groups where k is the number of desired clusters. To do so, the system uses a partitional clustering algorithm which finds the k-way clustering solution making repeated bisections. Then the system arranges the clusters in a hierarchy by successively merging the most similar clusters in a tree.

- MALLET: the soft clustering system we developed relies on MALLET (McCallum, 2002), a Latent Dirichlet Allocation (LDA) toolkit (Steyvers and Griffiths, 2007).

  Our MALLET-based system considers that each query is a document and builds a topic model describing the documents. The resulting topics are the clusters. Each query is associated with each topic according to a certain strenght. Unlike the system based on CLUTO, this system produces soft clusters, i.e. each query may belong to more than one cluster.

5.5 Classification

LangLog allows the same query to be classified many times using different classification schemas and different classification strategies. The result of the classification of an input query is always a map that assigns each category a weight, where the higher the weight, the more likely the query belongs to the category. If NER performs better when tailored to a specific domain, classification is a task that is hardly useful without any customization. We need a different classification schema for each content provider. We developed two classification system: an unsupervised system and a supervised one.

- Unsupervised: this system does not require any training data nor any domain-specific corpus. The output weight of each category is computed as the cosine similarity between the vector models of the most representative Wikipedia article for the category and the collection of Wikipedia articles most relevant to the input query. Our evaluation in the KDD-Cup 2005 dataset results in 19.14 precision and 22.22 F-measure. For comparison, the state of the art in the competition achieved a 46.1 F-measure. Our system could not achieve a similar score because it is unsupervised, and therefore it cannot make use of the KDD-Cup training dataset. In addition, it uses only the query to perform classification, whereas KDD-Cup systems were also able to access the result sets associated to the queries.

[11] http://glaros.dtc.umn.edu/gkhome/views/cluto

- Supervised: this system is based on the Weka framework. Therefore it can use any machine learning algorithm implemented in Weka. It uses features derived from the queries and from Bridgeman metadata. We trained a Naive Bayes classifier on a set of 15.000 queries annotated with 55 categories and hits and obtained a F-measure of 0.26. The results obtained for the classification are encouraging but not yet at the level of the state of the art. The main reason for this is the use of only in-house meta-data in the feature computation. In the future we will improve both components by providing them with features from large resources like Wikipedia or exploiting the results returned by Web Searching engines.

6 Demonstration

Our demonstration presents:

- The setting of our case study: the Bridgeman Art Library website, a typical user search, and what is recorded in the log file.

- The conceptual model of the results of the analyses: search episodes, queries, lemmas, named entities, classification, clustering.

- The data flow across the parts of the system, from content provider's servers to the front-end through databases, NLP Web services and data marts.

- The result of the analyses via QlikView.

7 Conclusion

In this paper we presented the LangLog system, a customizable system for analyzing query logs. The LangLog performs language identification, lemmatization, NER, classification and clustering for query logs. We tested the LangLog system on queries in Bridgeman Library Art. In the future we will test the system on query logs in different domains (e.g. pharmaceutical, hardware and software, etc.) thus increasing the coverage and the significance of the results. Moreover we will incorporate in our system the session information which should increase the precision of both clustering and classification components.

References

Salah Ait-Mokhtar, Jean-Pierre Chanod and Claude Roux 2002. *Robustness Beyond Shallowness: Incremental Deep Parsing. Journal of Natural Language Engineering* 8, 2-3, 121-144.

Alessio Bosca and Luca Dini. 2010. *Language Identification Strategies for Cross Language Information Retrieval. CLEF 2010 Working Notes.*

C. Brun and M. Ehrmann. 2007. *Adaptation of a Named Entity Recognition System for the ESTER 2 Evaluation Campaign.* In proceedings of *the IEEE International Conference on Natural Language Processing and Knowledge Engineering.*

F. M. Facca and P. L. Lanzi. 2005. *Mining interesting knowledge from weblogs: a survey. Data Knowl. Eng.* 53(3):225241.

Jansen, B. J. 2006. *Search log analysis: What is it; what's been done; how to do it. Library and Information Science Research* 28(3):407-432.

Jansen, B. J. 2008. *The methodology of search log analysis.* In B. J. Jansen, A. Spink and I. Taksa (eds) *Handbook of Web log analysis* 100-123. Hershey, PA: IGI.

Joachims T. 2002. *Optimizing search engines using clickthrough data.* In proceedings of *the 8th ACM SIGKDD international conference on Knowledge discovery and data mining* 133-142.

M. Li, Y. Zhang, M. Zhu, and M. Zhou. 2006. *Exploring distributional similarity based models for query spelling correction.* In proceedings of *In ACL 06: the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL* 10251032, 2006.

Andrew Kachites McCallum. 2002. *MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu.*

C. Monz and M. de Rijke. 2002. *Shallow Morphological Analysis in Monolingual Information Retrieval for Dutch, German and Italian. In Proceedings of CLEF 2001.* Springer

M. Steyvers and T. Griffiths. 2007. *Probabilistic Topic Models.* In T. Landauer, D McNamara, S. Dennis and W. Kintsch (eds), *Handbook of Latent Semantic Analysis*, Psychology Press.

J. R. Wen and H.J. Zhang 2003. *Query Clustering in the Web Context.* In Wu, Xiong and Shekhar (eds) *Information Retrieval and Clustering* 195-226. Kluwer Academic Publishers.

Y. Zhao and G. Karypis. 2002. *Evaluation of hierarchical clustering algorithms for document datasets.* In proceedings of *the ACM Conference on Information and Knowledge Management.*