NoSQL vs Relational Database: A Comparative Study About the Generation of the Most Frequent N-Grams

Jardel Ribeiro, Jonas Henrique, Rodrigo Ribeiro and Rosalvo Neto
Department of Engineering Computer
Federal University of Sao Francisco (Univasf)
Juazeiro-BA, Brazil

***Abstract*—This study intends to help data mining developers to get better performance when obtaining the most frequent N-grams in Text Mining projects. The process of building new variables is one of the oldest and still challenging problems in Data Mining projects. The most frequent N-grams are commonly used as input variables in Text Mining projects. The N-grams represent the occurrence of N items in sequence in a given text. The items can be letters or words. This paper presents a performance comparison between the two main approaches of data storage, relational and NoSQL databases in the task of obtaining the most frequent N-grams. Validation of the study was executed using a database from a known benchmark from an international competition organized by PAN@CLEF 2013. The one-tailed paired t-test showed that NoSQL approach is statistically superior to the relational approach with a confidence level of 95%.**

***Index Terms*—Big Data; Text Mining; N-gram; Relational Database; NoSQL.**

I. Introduction

The term WEB 2.0 appeared in the middle of the 2000's [1] and refers to the current version of the internet, where users have more interaction and collaboration among themselves, differently from the first version of the internet when users were limited to the passive visualization of content. As examples of sites created after the WEB 2.0 that may be mentioned are: social networks, blogs and video sharing sites.

The technology of relational databases did not meet the storage needs of this new demand. In order to deal with this demand the Big Data technology emerged. [2] describe Big Data as a technology that has three characteristics: Volume, Variety and Velocity, also known as the 3 Vs:

- **Volume**: Storing and analyzing a huge volume of data;
- **Variety**: Handling different types of data, from structured to semi-structured ones or non-structured;
- **Velocity**: Handling with the demand by high speed that data are accessed and stored.

The NoSQL databases (Not only SQL) came to fill the 3Vs demands of the Big Data. [3] defines NoSQL as a new generation of databases allowing a high performance and fast processing of information in large scales.

With the consolidation of WEB 2.0, a sub-area of Data Mining became popular: Text Mining. It involves the extraction of previously unknown knowledge from texts. Among its main applications we may highlight: sentiment analysis [4] and inferring the gender and age of a user from text of social networks [5]. As a subarea of Data Mining one of the main challenges of Text Mining is the building of its input variables. Among the main variables used by his area the highlight are the N-grams, which consider the number of times than n terms appear together in a specific order in a text [5], [6].

Although NoSQL is frequently used to storage WEB 2.0 data, the developers of Text Mining solutions have no access to NoSQL databases when they build their solutions. For instance, the PAN@CLEF competition [7], one of the main ones in this area, releases data for building solutions using XML (Extensible Markup Language) files containing the posts of users. A question that arises in this scene is: "which is the most effective way of obtaining N-grams from XML files?". The objective of this paper was to answer this research question. In order to do that, an experimental study was executed comparing the two main technologies of data storage, NoSQL and Relational databases. The study used the database of the international competition PAN@CLEF 2013. The metric used for performance evaluation was the time in seconds that each approach needs to obtain the most frequent N-grams. Results obtained show that the performance of NoSQL databases is superior that the one of the relational databases.

The remainder of this paper is organized as follows. Section II presents the problem definition. Section III presents related works. Section IV details the compared approaches. Section V describes the data used as a test bed for the comparison executed in this paper. Section VI shows the experimental methodology. Section VII presents the experimental results and their interpretation. Finally, Section VIII concludes this paper and proposes future works.

II. Problem Definition

The process of building variables is one of the oldest and still challenging problems of the Data Mining area [8]. [6] emphasizes that variables in Text Mining may be represented as a vector of features that captures potentially relevant characteristics from the text. Among the most common ones extracted from text it is possible to highlight the N-grams. They represent the occurrence of N items in sequence in a given text [9]. The items can be letters or words. For example,

978-1-5386-1107-4/17/$31.00 ©2017 IEEE

for the sentence “Attitude is a little thing that makes a big difference”, if N=3 (words), then the N-grams would be:

1) ”Attitude is a”
2) ”is a little”
3) ”a little thing”
4) ”little thing that”
5) ”thing that makes”
6) ”that makes a”
7) ”makes a big”
8) ”a big difference”

The quantity of N-grams built in a database, generally, is very large, because of that it becomes unfeasible to use all N-grams for applying machine learning algorithms, because they can create a phenomenon known in literature as the “curse of dimensionality” [10]. In order to overcome this problem it is common to select the N-grams that will be used as variables. This selection should use some criterion, such as: the frequency of N-grams [11]. In this way, first all N-grams of each text are obtained and then the X most frequent N-grams are selected. Figure 1 illustrates this process. A practical problem that developers of Data Mining solutions meet is how to obtain the most frequent N-grams in an effective way regarding time. Generally, data available to developers are in files representing a sample from all texts. In order to help developers in this task, this work performed a comparative study between the two most frequent approaches for obtaining N-grams.

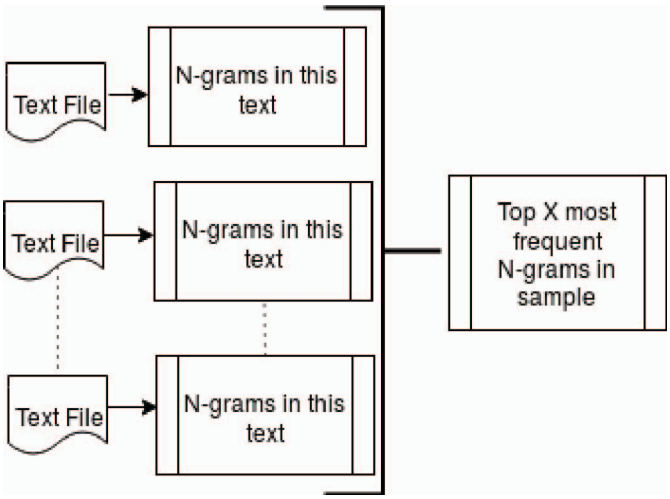

Fig. 1. Generation process of the most frequent N-grams

III. Related Works

Several works performed performance comparisons between relational and NoSQL databases. However, according to the literature research performed in this study, no one compares the performance of those two databases for obtaining the most frequent N-grams. Following will be described works that did some type of comparison between the two database approaches.

In [12], the authors did a comparative study of the operations of inserting, excluding and querying between NoSQL and relational databases. The MongoDB, RavenDB, CouchDB, Casandra, Hypertable and Couchbase databases were used as representatives of the NoSQL approach. Microsoft SQL Server Express was used as representative of the relational approach. The study found that not all NoSQL databases had a better performance regarding the same operations implemented in relational database. Besides, the study found that there is a considerable difference of performance between NoSQL databases.

In [13], the authors compare the NoSQL database based in documents MongoDB with the relational database PostgreSQL regarding execution time of select and insert operations. The study indicates superiority of the MongoDB in the insertion text. The PostgreSQL time was 1106 seconds while in MongoDB it was 17.78 seconds for inserting 100000 records. In the text of complex search, the PostgreSQL took 278438 seconds while MongoDB needed only 51.71 seconds to perform the same task.

In [14], the authors compared the NoSQL database based in documents MongoDB with the relational database MySQL in an application of information management. Results showed that MongoDB is faster than MySQL in insert and query expressions. However, the authors inform that the lack of references about MongoDB slowed down the process of developing the solution.

In [15], the authors compared the NoSQL database based in documents MongoDB with the relational database Oracle. The authors conclude that if the objective is speed and scalability, NoSQL is the most indicated one. On the other hand, if speed is not the priority and the objective is to better structure the data, the relational approach is indicated.

In [16], the authors compared the NoSQL HBase with the relational database MySQl. Times of reading and writing large amounts of data were evaluated. The study pointed that the reading and writing times were smaller using HBase. The authors highlight scalability, availability, performance and fault tolerance as advantages of HBase regarding MySQL.

In [17], the authors compared the NoSQL and relational approaches in the distributed management of RDF (Resource Description Framework) data. HBase and MySQL Cluster were used as representatives of the NoSQL and relational approaches respectively. The study showed HBase as superior for querying large sets of RDF data.

IV. Approaches Compared

The two main technologies of data storage were selected, aiming to help developers of Text Mining solutions to decide the most effective way of obtaining the most frequent N-grams. The methodologies for obtaining the most frequent N-grams using NoSQL and relational databases will be presented next.

A. NoSQL

The NoSQL database selected for this study was HBase. It is the no relational database based in the BigTable technology

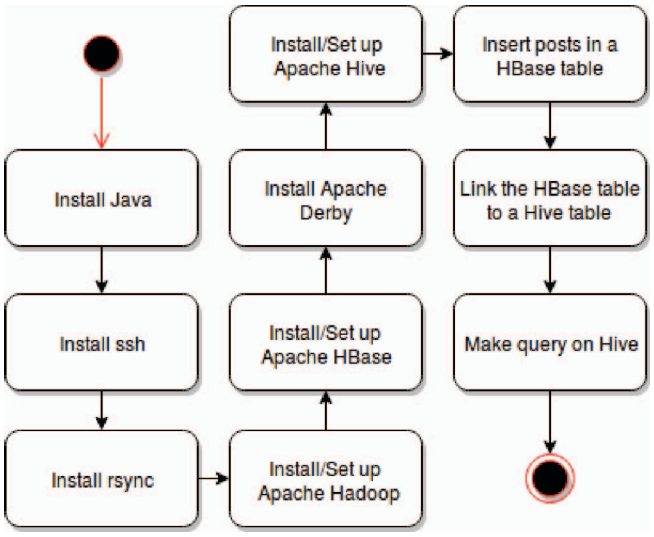

Fig. 2. Activity diagram for obtaining the most frequent N-grams using the NoSQL database approach

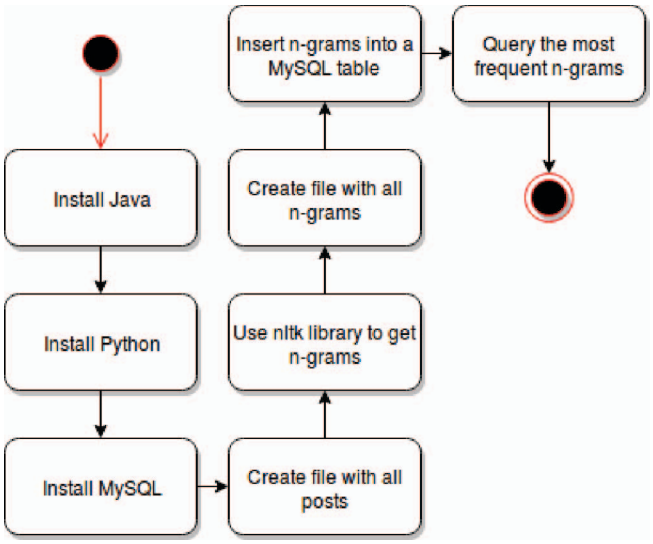

Fig. 3. Activity diagram for obtaining the most frequent N-grams using the Relational database approach

of Google [18]. HBase is a component from the Apache Hadoop framework that uses the Hadoop files system, the HDFS (Hadoop Distributed File System), for data storage [19]. Hadoop allows the distribution and processing of large data amounts in computer *clusters* using a simple programming model and offering high fault tolerance.

Hadoop offers tools enabling an easier extraction of relevant information from a database. One of those tools is Hive. Hive is a *warehouse* software facilitating the reading, writing and handling of large data sets stored in a distributed way using SQL commands. It allows doing a query of the most frequent N-grams in a table using a simples command, similar to a SQL command.

For obtaining N-grams using HBase, this study followed the steps illustrated in Figure 2. Initially it was necessary to install the tools used by Hadoop: Java, ssh (secure shell) and rsync (a tool for synchronizing files in computer systems). Then, it was necessary to install and configure Hadoop. Next, to install and configure HBase for using the HDFS of the installed Hadoop. Derby was needed, because it is used to save the metadata stored by Hive.

After all Hadoop tools were installed and configured, a Java program for extracting the posts that were in several XML files was used for inserting the data in the HBase database. A table was created in Hive and connected with the table where the posts where inserted in HBase. Finally, a query was executed in Hive in order to get the most frequent N-grams.

B. Relational database

The relational database selected in this study was MySQL. Figure 3 illustrated the step followed in order to get the most frequent N-grams using a relational database in this study. First, the following tools were installed: Java, Python and the MySQL database. Java was installed for using the program that creates a file with all posts. This program was adapted from the program used in the NoSQL approach for inserting data in HBase. Python was installed for allowing the use of the NLTK (Natural Language Toolkit) library, in order to obtain the N-grams. NLTK is a leading platform for building Python programs that work with human language data [20].

After installing the tools, the sequence of activities is: 1) executing the Java program for creating a file with all posts; 2) running a program written in Python which obtains the N-grams and inserts them in a new file; 3) inserting the file with all N-grams in a MySQL table and; 4) running a SQL command in order to get the most frequent N-grams in the MySQL table.

V. Data Set Used

The database used in this paper was made available by PAN@CLEF2013 [7]. PAN is a series of scientific events and tasks shared in forensic digital media that is part of CLEF (Conference and Lab of the Evaluation Forum). The task for this competition was to determinate the age and sex of the author according to its post. Posts were extracted from the Netlog site and made available in 233,600 files in English and Spanish languages, in XML format. Figure 4 illustrates the content of files format. Tag conversation contains the text of the post.

VI. Experimental Methodology

In order to verify if there is difference of performance to obtain the most frequent N-grams using the NoSQL and Relational database approaches, the selected database was split in 34 disjoint sets with equal number of posts (33 with 7,000 and the last one with 5,600). The goal of this partitioning was to made samples with different amounts of posts available. Thus, 34 experiments were executed and in each experiment the sets used in the previous experiment were grouped. The first experiment was executed with 7,000 files, the second

```
<author
  lang="lang_code"
  gender="gender_code"
  age_group="age_group">
    <conversations
      count="number_of_conversations_in_file">
      <conversation id="UUID">
        [Original HTML Content of the conversation]
      </conversation>
      <conversation id="UUID">
        [Original HTML Content of the conversation]
      </conversation>

      ....

    </conversations>
</author>
```

Fig. 4. Example of content of XML file from PAN@CLEF2013

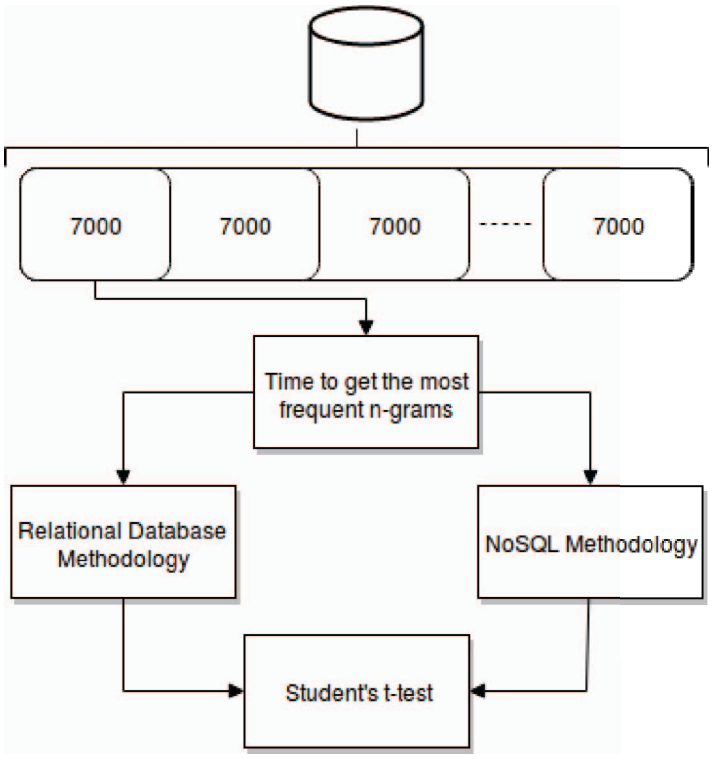

Fig. 5. Resume of the adopted experimental methodology

experiment with 14,000 files, until the final experiment which used the 34 sets (236,600 files). Performance was measured by means of time in seconds demanded by each technology for generating the N-grams. It is important to highlight that the study is not taking into consideration the insertion times, only the times to obtain the N-grams. For this study the one-hundred most frequent 2-grams were obtained. t-student's test was applied in order to verify if the difference of performance was statistically significant. Figure 5 illustrates the methodology applied.

A. Test t-student

The paired t-Student test is a special case of the hypothesis test applied with observations when two populations of interest are collected in pairs, each pair of observations being collected under homogeneous conditions [21]. For this work, the performance metric desired is the difference in time in seconds obtained by each one of the approaches. The test configuration used in this study is detailed next:

- **Null hypothesis**: $\mu 1 - \mu = 0$;
- **Alternate hypothesis**: $\mu 1 > \mu 2$.

Where

- $\mu 1$ represents the mean of time obtained by the approach using the Relational database.
- $\mu 2$ represents the mean of time obtained by the approach using the NoSQL database;;

B. Experiment setup

In order to avoid possible problems caused by programs already installed or differences of hardware, all experiments were performed on virtual machines with 8Gb memory, 8 CPU cores and Ubuntu 16.04 operating system. Each experiment was performed in different machines with the same configuration.

VII. Results

Simulations were performed according to the experimental configuration previously described for each one of the two approaches. Time difference is seen by Figure 6, which shows the results obtained in the experiments. It is possible to see that the time for obtaining N-grams using the NoSQL approach is smaller than using the Relational approach. The NoSQL approach was 18 times faster, in average (104s NoSQL x 1943s Relational). Time difference increases linearly between the two approaches as the number of posts increases. Figure 7 exhibits the graphic with the time difference between the two approaches, since the shape of the graphic is a straight line, it is possible to infer that this is a linear relationship.

Table I shows the resume of results obtained in paired t-test. Since p-value is smaller than 0.05, it is concluded that the two approaches provide different results. Specifically, data indicate that the NoSQL approach obtains N-grams in a smaller time than the relational database approach, with a confidence level of 95%.

TABLE I
Resume of the Result of the Hypothesis testing

Lower Limit	$\mu_1 - \mu_2$	**Upper Limit**	p-$value$
1521.018	1838.882	∞	1.372e-11

VIII. Concluding Remarks

This work presented a comparison between relational and NoSQL database approaches for obtaining the most frequent N-grams. The comparison was performed using a database from an important international competition, considered as a benchmark in the area. As experimental methodology, t-Student's paired one-tailed test was applied in the performance

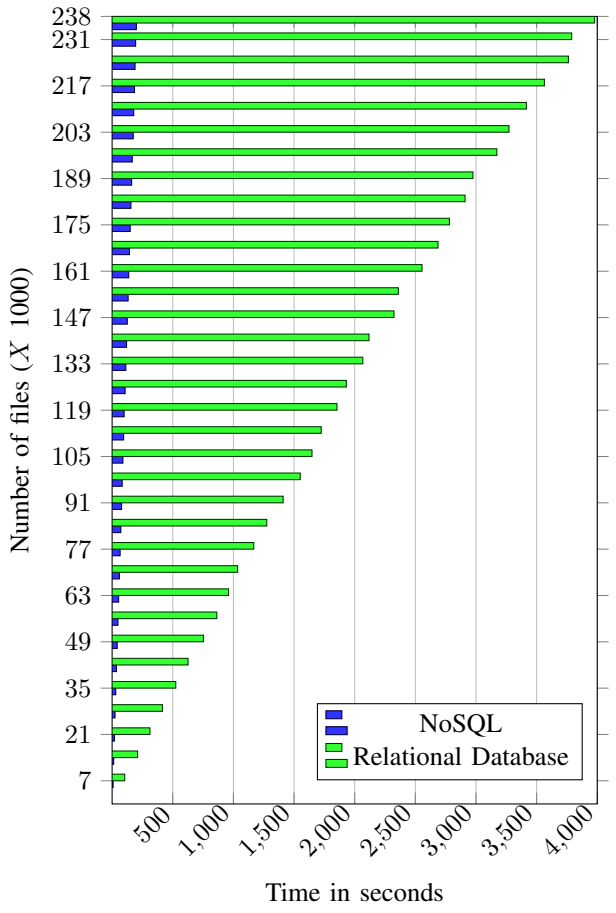

Fig. 6. Obtention time of N-grams

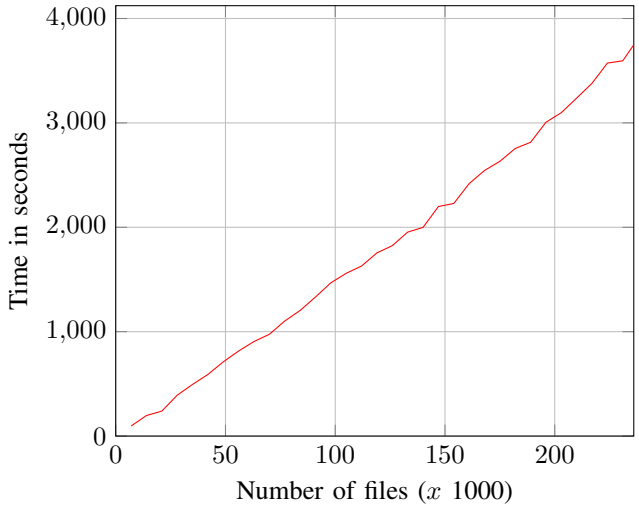

Fig. 7. Time difference for obtaining N-grams between the two approaches

measured by the time needed for each approach for obtaining the N-grams. The study showed that the NoSQL approach overcomes, in a statistically significant way, the relational database with a confidence level of 95%. The main contribution of this work is to indicate for the developers of Text Mining solutions the use of NoSQL technology for obtaining the most frequent N-grams. The main limitation of this work was to use only one representative for each technology. Thus, as a future work the authors pretend to expand this work using other representatives of the NoSQL and Relational database approaches.

REFERENCES

[1] A. Gandomi and M. Haider, "The hype: big data concepts, methods, and analytics," *International Journal of Information Management*, vol. 35, no. 2, pp. 137–144, 2015.

[2] C.-W. Tsai, C.-F. Lai, H.-C. Chao, and A. Vasilakos, "Big data analytics: a survey," *Journal of Big Data*, vol. 2, no. 1, pp. 1–32, 2015.

[3] R. Cattell, "Scalable sql and nosql data stores," *SIGMOD Rec.*, vol. 39, no. 4, pp. 12–27, 2011.

[4] X. Fan, X. Li, F. Du, X. Li, and M. Wei, "Apply word vectors for sentiment analysis of app reviews," in *2016 3rd International Conference on Systems and Informatics (ICSAI)*. IEEE, 2016, pp. 1062–1066.

[5] K. Santosh, R. Bansal, M. Shekhar, and V. Varma, "Author profiling: Predicting age and gender from blogs," *Notebook for PAN at CLEF*, pp. 119–124, 2013.

[6] M. Koppel, J. Schler, and S. Argamon, "Computational methods in authorship attribution," *Journal of the American Society for information Science and Technology*, vol. 60, no. 1, pp. 9–26, 2009.

[7] PAN@CLEF, "A series of scientific events and shared tasks on digital text forensics," 2017, uRL: http://pan.webis.de/.

[8] R. Neto, P. J. Adeodato, and A. C. Salgado, "A framework for data transformation in credit behavioral scoring applications based on model driven development," *Expert Systems with Applications*, vol. 72, no. 1, pp. 293–305, 2017.

[9] T. Jauhiainen, H. Jauhiainen, K. Lindén *et al.*, "Discriminating similar languages with token-based backoff," in *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, 2015.

[10] J. Fan, Y. Fan, and Y. Wu, *High-Dimensional Classification*. World Scientific, 2011, ch. 1, pp. 3–37.

[11] L. Havrlant and V. Kreinovich, "A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation)," *International Journal of General Systems*, vol. 46, no. 1, pp. 27–36, 2017.

[12] Y. Li and S. Manoharan, "A performance comparison of sql and nosql databases," in *Communications, Computers and Signal Processing (PACRIM), 2013 IEEE Pacific Rim Conference on*. IEEE, 2013, pp. 15–19.

[13] C. Politowski and V. Maran, "Comparaç ao de performance entre postgresql e mongodb," in *X Escola Regional de Banco de Dados*. SBC, 2014, pp. 1–10.

[14] Z. Wei-ping, L. Ming-Xin, and C. Huan, "Using mongodb to implement textbook management system instead of mysql," in *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*. IEEE, 2011, pp. 303–305.

[15] A. Boicea, F. Radulescu, and L. I. Agapin, "Mongodb vs oracle-database comparison." in *EIDWT*, 2012, pp. 330–335.

[16] H. Ding, Y. Jin, Y. Cui, and T. Yang, "Distributed storage of network measurement data on hbase," in *Cloud Computing and Intelligent Systems (CCIS), 2012 IEEE 2nd International Conference on*, vol. 2. IEEE, 2012, pp. 716–720.

[17] C. Franke, S. Morin, A. Chebotko, J. Abraham, and P. Brazier, "Efficient processing of semantic web queries in hbase and mysql cluster," *It Professional*, vol. 15, no. 3, pp. 36–43, 2013.

[18] D. Carstoiu, E. Lepadatu, and M. Gaspar, "Hbase-non sql database, performances evaluation," in *in Computer Science (1986), Master in Computer Science (1990), and PhD in Computer Science*. Citeseer, 2010.

[19] A. Hadoop, "Hadoop," 2009.

[20] E. Loper and S. Bird, "Nltk: The natural language toolkit," in *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ser. ETMTNLP '02. Association for Computational Linguistics, 2002, pp. 63–70.

[21] D. Montgomery and G. Runger, *Applied Statistics and Probability for Engineers*. John Wiley & Sons, 2010.