Multidimensional Text Warehousing for Automated Text Classification

Jiyun Kim, University of Seoul, Seoul, Korea
Han-joon Kim, University of Seoul, Seoul, Korea

ABSTRACT

This article describes how, in the era of big data, a data warehouse is an integrated multidimensional database that provides the basis for the decision making required to establish crucial business strategies. Efficient, effective analysis requires a data organization system that integrates and manages data of various dimensions. However, conventional data warehousing techniques do not consider the various data manipulation operations required for data-mining activities. With the current explosion of text data, much research has examined text (or document) repositories to support text mining and document retrieval. Therefore, this article presents a method of developing a text warehouse that provides a machine-learning-based text classification service. The document is represented as a term-by-concept matrix using a 3rd-order tensor-based textual representation model, which emphasizes the meaning of words occurring in the document. As a result, the proposed text warehouse makes it possible to develop a semantic Naïve Bayes text classifier only by executing appropriate SQL statements.

KEYWORDS

Data warehouse, Han-joon Kim, Jiyun Kim, Naïve Bayes, SQL, Text mining, Text warehouse, University of Seoul

INTRODUCTION

In the era of big data, a data warehouse with multidimensional on-line analytical processing (OLAP) capabilities would be an important information repository for business intelligence (Elliott et al., 2013; Hira & Deshpande, 2014; Miranda, 2015; Roelofs et al., 2013); it could provide significant multidimensional views of fact measures by integrating huge amounts of structured data retrieved from various sources in relational databases. With the explosion of textual data, much research has examined text mining and document retrieval from text (or document) repositories. One study proposed a concept of multidimensional text warehousing that contains text-specific dimensions, document tuples, and document cubes (Feki, Ben, & Zurfluh, 2013; Linxiu, 2016; Mendoza et al., 2015; Messaoud, Feki, & Zurfluh, 2012; Sudhakaran et al., 2013).

Simply put, a text warehouse is a data warehouse that organizes textual data. The difference from data warehouses in general is that a text warehouse must manage diverse concepts (or senses) involved in documents and their weights. Moreover, we expect that the goal of text warehousing is not only to enable a multidimensional view of large amounts of textual data but also to offer intelligent data mining functions for business intelligence (Mahapatra, Jagadev, & Naik, 2011; Wu et al., 2011). Conventional data warehousing techniques do not consider the various data manipulation operations required for data mining activities.

Here, we propose a new way of building a text warehouse to rapidly implement a machine learning-based text classification system. In our work, we use a 3rd-order tensor-based textual representation...
model in which the document is represented by a term-by-concept matrix (Kim & Chang, 2014; Hong, Kim, & Lee, 2015). The proposed method emphasizes the sense of the words that occur in the document, so all related probabilistic weights are stored in a relational table named the ‘bridge’ table (Kimball & Ross, 2011). As a result, we show that the proposed multidimensional text warehouse can perform semantic Naïve Bayes text classification only by executing the appropriate Structured Query Language (SQL) statements (Kim & Kim, 2016).

PRELIMINARIES FOR SEMANTIC TEXT WAREHOUSE

Text Representation Model

In the vector space model, documents are represented as vectors in which each element has a weighting. By contrast, our semantic tensor space model represents documents as 2nd-order tensors (i.e., matrices) \( \mathcal{R}^{S} \otimes \mathcal{R}^{V} \), where \(|S|\) is the number of concepts (or semantics), \(|V|\) is the number of terms indexed, and \( \mathcal{R}^{S} \) and \( \mathcal{R}^{V} \) are the vector spaces for the concepts and terms, respectively. We regard the ‘concept space’ as an independent space equated to the ‘term’ and ‘document’ spaces used in the vector space model (Hong et al., 2015; Kim & Chang, 2014).

According to the formal concept analysis principle, a concept is defined by a pair of ‘intent’ and ‘extent’. Here, the extent means the set of instances that are included in the concept and the intent means the set of all common attributes of instances included from the extent. In our work, the extent that represents a concept consists of a set of documents related with the concept, whereas the intent consists of a set of keywords extracted from the set of documents. Figure 1 illustrates a term-by-document matrix and a term-by-document-by-concept tensor representations for a given corpus. To represent a document corpus, rather than a term-by-document matrix, we can generate a 3rd-order tensor with distinct document, term, and concept spaces. As a result, we can represent terms or concepts as matrices; given a 3rd-order tensor of a document corpus, we can represent a component of each space using the other two vector spaces. That is, we can represent a document as a concept-by-term matrix, a term as a concept-by-document matrix, and a concept as a term-by-document matrix.

Semantic Naïve Bayes Text Classification

Basically, our work uses the Naïve Bayes (NB) learning algorithm to classify text. This algorithm assumes that the terms in a document are mutually independent and the probability of term occurrence is independent of position within the document. This assumption simplifies the classification function, and in spite of such a wrong independence assumption, its performance is reasonably accurate. Besides, the Naïve Bayes algorithm has a number of superior advantages compared with other learning algorithms. Basically, machine learning algorithms should effectively deal with the curse-of-dimensionality problem since text data have a huge number of term features. In terms of overcoming the problem, the Naïve Bayes algorithm is less sensitive than other learning algorithms such as k-nearest neighbors, decision trees, neural networks and support vector machine. Moreover, the Naïve Bayes algorithm is suitable for operational text classification system since it is very easy to incrementally update the classification model due to its simplicity; that is, when new documents are given as training data, the current word feature statistics are updated and additional feature evaluation is immediately carried out without re-processing the past training data. This characteristic is essential in the case where the document collection is highly evolutionary. More importantly, the Naïve Bayes algorithm does not require a complex generalization process unlike decision tree, neural networks and support vector machine; it has only to calculate the feature statistics per class as seen in Equation 1.

The Naïve Bayes text classifier \( F_{\theta_{a}} \) predicts category \( c \) with the largest posterior probability corresponding to document \( d \) by using the following equation when given the set of words \( (t_{1}, t_{2}, \ldots, t_{m}) \) that make up a set of class \( C \) and document \( d \).

\[
\text{arg} \max_{c} \frac{p(c|d) \cdot p(d)}{p(c)} = \text{arg} \max_{c} \frac{p(d|c) \cdot p(c)}{p(d)}
\]
The Rationale Behind Strategic Alliances in Application Service Provision
www.igi-global.com/article/rationale-behind-strategic-alliances-application/3230?camid=4v1a

Bridging the Knowledge Gap in Management and Operations of Transfusion Medicine: Planning, Policy and Leadership Issues
Cees Th. Smit Sibinga and Maruff Akinwale Oladejo (2013). *Teaching Cases Collection* (pp. 69-82).
www.igi-global.com/article/bridging-knowledge-gap-management-operations/78358?camid=4v1a