https://doi.org/10.48009/4_iis_2020_67-72

Issues in Information Systems
Volume 21, Issue 4, pp. 67-72, 2020

TOWARDS THE DEVELOPMENT OF A CLASSIFICATION MODEL FOR TECHNICAL DOCUMENTS IN KNOWLEDGE DISCOVERY SYSTEMS

Roman Melnyk, Nova Southeastern University, rm1731@mynsu.nova.edu
Martha M. Snyder, Nova Southeastern University, smithmt@nova.edu
Alexander Verner, Nova Southeastern University, av836@mynsu.nova.edu

ABSTRACT

An important component of knowledge management is the organization of documents for quick and easy access. One effective way of organizing these documents is to group them by a fixed set of specific knowledge categories. For large-scale technical teams, the number of categories can reach thousands or even tens of thousands, which makes this type of cataloging especially useful. Text classification is a sophisticated process that involves data pre-processing, transformation, dimensionality reduction, application of classification techniques, classifier evaluation, and classifier validation. This paper describes the preliminary results from phase one of a design-science research study for the development of a model that can be used for classification of financial software development documentation in knowledge discovery systems using machine learning. Specifically, testing with a small dataset of 64 documents from the Natural Language Technical Project Documentation dataset was conducted to assess the effectiveness of traditional text classification methods. Results indicate limitations to these traditional methods for classifying technical documents. The next steps include evaluating the performance on a bigger dataset of technical documents and testing new deep learning techniques.

Keywords: Knowledge Management, Knowledge Discovery, Text Classification, Machine Learning

INTRODUCTION

Text Classification in Knowledge Management

Organization of documents for quick and easy access is an essential component of knowledge management in technical teams. For large-scale technical companies, the number of categories can reach thousands or even tens of thousands. The goal of this study is to investigate how machine learning can help to organize technical documents by comparing the effectiveness of traditional text classification methods with more modern deep neural network pre-trained models.

Text classification has been extensively investigated in the past years; however, it has several challenges and still depends on manual labor rather than on automated processes, such as machine learning (Fromm, et al., 2019). In a large organization, the abundance of large disordered technical texts is likely to cause an increase in development cost and time-to-market of the product (Liu, Lu, & Loh, 2007). Due to the amount of work, it is often impractical to assign document classification tasks to company employees (Rossi, Lopes, Faleiros, & Rezende, 2014), which leaves two options: either leave the documentation as is or find a way to automate the process. The latter option leads to a shorter development cycle and the production of better products to the market (Kobayashi et al., 2018a). According to Kao, Quach, Poteet, and Woods (2003) and Sabbah et al. (2017), the automated organization of the documents into categories of interest also accelerates the knowledge discovery processes of the company. Text classification techniques have become essential to support knowledge discovery as the volume of digital documents has expended in recent years (Sinoara, Camacho-Collados, Rossi, Navigli, & Rezende, 2019).

Text Classification Methods

Starting with the work of Maron (1961) in the early 1960s, text classification has been used in various fields, including political and social sciences, business and marketing, law, finance, health-care and personality research (Kowsari et. al., 2019; Silwattananusarn & Tuamsuk, 2012). Text classification methods are largely based on machine learning and data mining, which makes them especially effective for applications such as information retrieval and filtering, sentiment analysis, recommender systems, etc. (Kobayashi, Mol, Berkers, Kismihók, & Hartog, 2018a). However, the

applications of text mining in knowledge management are limited (Fromm, Wambsganss, Söllner, 2019; Kowsari et al., 2019; Usai, Pironti, Mital, & Aouina Mejri, 2018).

Kowsari et al. (2019) conducted a review of the literature on different text classification methods including various feature extraction and dimensionality reduction techniques. The authors pointed out that textual data represent a significant source of knowledge and nearly 80% of corporate information exists in unstructured textual data formats and document categorization is one of the most common methods for mining document-based data. Kowsari et al. (2019) discussed the limitations of various text classification techniques and their applications. They highlighted that it is a challenge to find suitable techniques for text classification and it is one of the most indispensable problems in machine learning.

Usai et al. (2018) defined text mining as the "most natural process of knowledge discovery by adopting textual databases" (p. 1472). The authors conducted a review of 85 academic publications focused on knowledge discovery derived from text mining techniques. They observed that knowledge discovery and text mining can be categorized into two periods. The first period is from 1998 to 2009, where academic research was focused on technical aspects of knowledge discovery through text mining. In the second period starting from 2010, the interest in data analytics increased exponentially with the heightened interest in customer behavior research. The authors noted that companies have a large amount of information in the form of textual data, which needs to be embedded and categorized in the knowledge management system.

Fromm, Wambsganss and Söllner (2019) developed a natural language processing feature taxonomy to help researchers and practitioners with text mining studies. Wambsganss and Söllner conducted a literature review and observed that most of the text mining applications solely rely on the basic bag-of-words feature generation technique in which the order and co-occurrence of words are not taken into account. They pointed out that since text mining requires deep domain knowledge, it still depends on human work and techniques that evolved in one discipline are rarely used in the other disciplines. Fromm et al., (2019) presented the initial development results of a taxonomy of features used in text mining. However, they suggested that further research across different domains is needed to capture all relevant features into the final taxonomy.

Purpose

In this paper we present the preliminary results from phase one of a design-science research study (Hevner, et al., 2004; Peffers, et al. 2007) for the development of a model that can be used for classification of financial software development documentation in knowledge discovery systems using machine learning. This research is carried out in the following four phases: (1) identify a proper data representation technique, (2) develop a prototype text classification model, (3) collect stakeholder feedback, and (4) validate the model internally based in stakeholder input. The first two phases embody the construction of a schematic TC model that will serve for capturing and codifying explicit knowledge that resides in technical documents such as business requirements, project plans, architecture blueprints, design specifications, and release plans. Kobayashi, et al.'s (2018b) text classification process guides these two phases and includes the following six steps: (1) text preprocessing, (2) text representation or transformation, (3) dimensionality reduction, (4) selection and application of classification techniques, (5) classifier evaluation, and (6) classifier validation. Research methods and results of steps 1, 2 and 3 (i.e., text preprocessing, text representation, and dimensionality reduction) are presented in this paper.

RESEARCH METHODOLOGY

Phase one is guided by the following research question: What text classification tools for knowledge discovery are available and what are their benefits and limitations? To address this question, it is important to identify a proper data representation technique by investigating the effectiveness of traditional models for technical documents classification. This phase aims to carry out the automated cataloging of technical documents by means of text classification. Following the work of Allahyari et al. (2017) we define the problem of text classification as follows: Given a training set of documents $D = \{d\_1, d\_2, \dots, d\_n\}$, where each document d_i is labeled with a label l_i from the set $L = \{l\_1, l\_2, \dots, l\_k\}$. The goal is to find a classification model f s.t. $f: D \rightarrow L, f(d) = l$, which can assign the

correct class label to new document d. As noted above, the text classification process will be used as described by Kobayashi, et al. (2018b).

The goal of the first three steps of this process is to find a good data representation technique. As noted by Sinoara, Camacho-Collados, Rossi, Navigli, and Rezende (2019), the performance of a text classification model is directly related to the quality of the data representation. The dimensionality reduction phase can be seen as a sort of data compression and the goal of this step is to improve the computational time by reducing the difficulty of the classification problem (Mironczuk, & Protasiewicz, 2018). The last three steps of the process will be focused on the model training and determining the most effective classification algorithms. Several classifiers will be evaluated including random forest and neural networks as suggested by Kobayashi et al. (2018b).

RESULTS

To ensure credible results, the chosen dataset should be representative of real-world data. The collection of documents was created from several publicly available documents and datasets (Natural Language Technical Project Documentation). The dataset consists of 64 technical documents that are labeled to 9 different classes. The following classes and respective number of documents in the dataset are presented in Table 1.

Table 1. Classes and Technical Documents

Document Class	Number of Documents
Architecture and Design	17
UI Design	10
Legal and Regulatory	10
Requirements	8
Quality Assurance	7
Security	7
Software and Development Process	2
Accessibility	2
Strategy	1
Total	64

Based on the study of Dobbin and Simon (2011) the optimal proportion of cases for the testing set tended to be in the range of 20% to 60%. To evaluate different approaches to the discussed classification task, we took 20% of the initial dataset and calculated the accuracy of predictive models on it.

The original text of technical documents went through a few stages of text preprocessing:

- Normalize: lowercasing and removal of non-alphabet characters.
- Tokenize: splitting text to a list of tokens. In our case each token represents a word.
- Lemmatize: reducing inflected (or sometimes derived) words to their word stem, base or root form (e.g. running -> run). Lemmatization reduces the inflected words properly ensuring that the root word belongs to the language.
- Removal of stop words: These are words, which do not add much meaning to a sentence.

Step 2 is text representation or transformation. All words from documents need to be represented in a meaningful and convenient format for processing. For this purpose, we used a technique called word embedding. It translates each word in a given dictionary into a vector of numbers of a specified length. Following the work of Kobayashi, et al. (2018b) we used the term frequency / inverse document frequency (TF/IDF) approach for text transformation. For a specific document, it determines how important a word is by looking at how frequently it appears in the document. IDF was used to calculate the weight of rare words across all documents.

Step 3 is dimensionality reduction. The perceived set of embeddings for each document is large. In average, there were 124525 words per document. After applying TF/IDF the vectors size was reduced to 3432. In addition, truncated singular value decomposition (SVD) was applied to reduce the dimensionality of feature vectors even more, to 49.

First, traditional models were evaluated by four commonly used metrics in information retrieval (Combarro, Montanes, Diaz, Ranilla & Mones, 2005): accuracy, precision, recall, and F1-measure. Tang, Kay and He (2016) defined precision as the percentage of documents that are correctly classified as positive out of all the documents that are classified as positive. The authors defined the recall as the percentage of documents that are correctly classified as positive out of all the documents that are positive. Precision can be computed as $Precision = \frac{TP}{TP+FP}$ and recall as $Recall = \frac{TP}{TP+FN}$, where TP denotes the number of true positive, FP denotes the number of false positive, and FN denotes the number of false negative. Precision and recall have inverse correlation between each other and F1-measure is one of the most popular that combine them as one single measure. F-measure can be computed as $F1 = \frac{2\ x\ Presicion\ x\ Recall}{Presision+Recall}$. The results for simple models are presented in Table 2.

Table 2. Traditional Models Results

	model_name	accuracy_score	precision_score	recall_score	f1_score
1	Stochastic Gradient Descent	0.692308	0.638889	0.694444	0.661905
5	Gaussian Naive Bayes	0.615385	0.622222	0.638889	0.595238
3	Decsision Tree	0.538462	0.5	0.527778	0.483333
6	K Nearest Neighbor	0.384615	0.397959	0.333333	0.311111
2	Random Forest	0.230769	0.144444	0.222222	0.169841
4	AdaBoost	0.307692	0.103175	0.277778	0.148148
0	Dummy	0.153846	0.163265	0.119048	0.12381

As shown in Table 2, most traditional machine learning models demonstrated relatively low results, which are insufficient for practical usage. These low results stem from the fact that our dataset was very small. Even though, stochastic gradient descent and gaussian naive bayes models achieved reasonable results. These results are sufficient to determine that machine-learning-based methods can be used to develop a model to simplify the task of classification of financial software development documentation in knowledge discovery systems. Furthermore, we believe, that with a significantly larger dataset these methods would have shown significantly better results. Having a larger dataset would allow to use more advanced machine learning algorithms based on deep neural networks, such as Bidirectional Encoder Representations from Transformers (BERT) (Devlin, Chang, Lee, & Toutanova, 2018)

SUMMARY

In this paper, preliminary results from phase one of a design-science research study to develop a model for classification of financial software development documentation in knowledge discovery systems are presented. In this phase, we investigated the effectiveness of traditional models for technical documents classification. The results of our experiments demonstrated that this approach is inefficient in the case of a smaller dataset of technical documents. Our next steps are to evaluate the performance on a bigger dataset of technical documents and evaluate new deep learning techniques such as Word2Vec (Mikolov, Chen, Corrado, & Dean, 2016), GloVe (Pennington, Socher, & Manning, 2014), BERT, convolution neural networks (Kim, 2014) and recurrent neural networks (Lai, Xu, Liu, & Zhao, 2015).

REFERENCES

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Gutierrez, J. B., & Kochut, K. (2017). A brief survey of text mining: Classification, clustering and extraction techniques. *ArXiv preprint arXiv:*1707.02919.

Combarro, E. F., Montanes, E., Diaz, I., Ranilla, J., & Mones, R. (2005). Introducing a family of linear measures for feature selection in text categorization. *IEEE transactions on Knowledge and Data Engineering, 17*(9), 1223-1232.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv*:1810.04805.

Dobbin, K. K., & Simon, R. M. (2011). Optimally splitting cases for training and testing high dimensional classifiers. *BMC medical genomics, 4*(1), 31.

Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996, August). Knowledge Discovery and Data Mining: Towards a Unifying Framework. *In KDD* (Vol. 96, pp. 82-88).

Fromm, H., Wambsganss, T., & Söllner, M. (2019). Towards a taxonomy of text mining features. Proceedings of the Twenty-Seventh European Conference on Information Systems (ECIS2019), Stockholm-Uppsala, Sweden, 1-12.

Hevner, A., March, S., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1), 75-105.

Kao, A., Quach, L., Poteet, S., & Woods, S. (2003, November). User assisted text classification and knowledge management. *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, 524-527.

Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2018a). Text mining in organizational research. *Organizational Research Methods, 21*(3), 733-765.

Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2018b). Text classification for organizational researchers: A tutorial. *Organizational Research Methods, 21*(3), 766-799.

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. *Information, 10*(4), 150.

Lai, S., Xu, L., Liu, K., & Zhao, J. (2015). Recurrent convolutional neural networks for text classification. *In Twenty-ninth AAAI conference on artificial intelligence*

Liu, Y., Lu, W. F., & Loh, H. T. (2007). Knowledge discovery and management for product design through text mining-a case study of online information integration for designers. In DS 42: *Proceedings of ICED 2007, the 16th International Conference on Engineering Design, Paris, France*, 28-31.

Maron, M. E. (1961). Automatic indexing: An experimental inquiry. *Journal of the ACM (JACM), 8*(3), 404-417.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2016). Efficient estimation of word representations in vector space. Cornell University Library. *ArXiv preprint arXiv*:1301.3781.

Mirończuk, M. M., & Protasiewicz, J. (2018). A recent overview of the state-of-the-art elements of text classification. *Expert Systems with Applications, 106*, 36-54.

Natural Language Technical Project Documentation. (n.d.). Retrieved from https://www.kaggle.com/sophiamelnyk/natural-language-technical-project-documentation

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. *In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*,1532-1543.

Rossi, R. G., Lopes, A. A., Faleiros, T. P., & Rezende, S. O. (2014). Inductive model generation for text classification using a bipartite heterogeneous network. *Journal of Computer Science and Technology*, *3*(29), 361-375.

Sabbah, T., Selamat, A., Selamat, M. H., Al-Anzi, F. S., Viedma, E. H., Krejcar, O., & Fujita, H. (2017). Modified frequency-based term weighting schemes for text classification. *Applied Soft Computing*, *58*, 193-206.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys (CSUR)*, *34*(1), 1-47.

Silwattananusarn, T., & Tuamsuk, K. (2012). Data mining and its applications for knowledge management: A literature review from 2007 to 2012. *International Journal of Data Mining & Knowledge Management Process*, *2*(5), 13.

Sinoara, R. A., Camacho-Collados, J., Rossi, R. G., Navigli, R., & Rezende, S. O. (2019). *Knowledge-enhanced document embeddings for text classification. Knowledge-Based Systems*, *163*, 955-971.

Tang, B., Kay, S., & He, H. (2016). Toward optimal feature selection in naive Bayes for text categorization. *IEEE transactions on knowledge and data engineering*, *28*(9), 2508-2521.

Usai, A., Pironti, M., Mital, M., & Aouina Mejri, C. (2018). Knowledge discovery out of text data: *A systematic review via text mining. Journal of Knowledge Management*, *22*(7), 1471-1488.