BILINGUAL PUBLISHING CO.
Pioneer of Global Academics Since 1984

Journal of Computer Science Research
http://ojs.bilpublishing.com/index.php/jcsr

ARTICLE

Researching the Research: Applying Machine Learning Techniques to Dissertation Classification

Suzanna Schmeelk[1*] Tonya L. Fields[2] Lisa R. Ellrodt[2] Ion C. Freeman[2] Ashley J. Haigler[2]

1. St. John's University, United States
2. Pace University, United States

ARTICLE INFO

Article history
Received: 24 July 2020
Accepted: 10 August 2020
Published Online: 30 September 2020

Keywords:
Machine learning
Natural language processing (NLP)
Abstract vs fulltext dissertation analysis
Industry-based
Dissertation research classification
GraphLab Create library
IBM Watson Discovery

ABSTRACT

This research examines industry-based dissertation research in a doctoral computing program through the lens of machine learning algorithms to determine if natural language processing-based categorization on abstracts alone is adequate for classification. This research categorizes dissertation by both their abstracts and by their full-text using the GraphLab Create library from Apple's Turi to identify if abstract analysis is an adequate measure of content categorization, which we found was not. We also compare the dissertation categorizations using IBM's Watson Discovery deep machine learning tool. Our research provides perspectives on the practicality of the manual classification of technical documents; and, it provides insights into the: (1) categories of academic work created by experienced fulltime working professionals in a Computing doctoral program, (2) viability and performance of automated categorization of the abstract analysis against the fulltext dissertation analysis, and (3) natual language processing versus human manual text classification abstraction.

1. Introduction

This research classifies industry-based doctoral research through the lens of machine learning algorithms to examine what Pace University's industry-geared doctoral students are researching. The Pace University Doctor of Professional Studies (DPS) in Computing began in the year 2000 [1,2]. The doctoral program is designed for experienced full-time working professionals (EFWPs) to study on campus with their faculty and advisers during the weekends. This research examines the first 114 dissertations that were successfully defended in the DPS program to understand industry trends and research needs through the dissertation topics. We employed the IBM Watson Discovery deep learning tool as well as Apple Turi's Graphlab Create in a Jupyter notebook running on an Amazon Web Services (AWS) Elastic Cloud (EC2) instance to classify the full-text of the DPS dissertations. This work extends the TF-IDF classification work of Ellrodt et al. [3,4], Freeman et al. [5], and Haigler et al. [6]; and, this research extends the EFWP research of Haigler et al. [7]. As the aforementioned publications were mutually exclusive studies of dissertation abstracts and fulltext dissertations, this research extends the work to compare the analysis methodologies give the performance differences from analyzing a page of text (dissertation abstract) versus analyzing over a hundred pages of text (dissertation full-text).

**Corresponding Author:*
Suzanna Schmeelk,
St. John's University, United States;
Email: schmeels@stjohns.edu

 DOI: https://doi.org/10.30564/jcsr.v2i4.2230

1.1 Problem Statement

Semantically processing and deriving meaning from text are open research problems. Within the subset of natural language processing problems, there remains open questions about categorizing text. There has been very little meta-research on research text itself, specifically dissertations and theses. In this paper, we analyze the differences between manual and machine classifications of doctoral abstracts and full-text dissertations to understand what topics are being researched by senior-level and experienced fulltime working professionals.

1.2 Review of Literature

Employing machine learning to examine text has been evolving since the early 2000s. Textual-based machine learning has been successfully deployed in many computing fields such as computer security, networking, human computer interaction, medicine, and law.

Fautsch and Savoy [10] showed that adapting term frequency inverse document frequency (TF-IDF) is useful for domain specific information retrieval.

1.2.1 Building Recommender Systems via NLP

A thrust of literature which highly employs TF-IDF is text-based recommender systems. Duan, Gui, Wei, and Wu [11] proposed a personalized resume TF-IDF based recommendation algorithm to help job seekers find relevant jobs and enterprises find relevant talent. Yuan, and Zhang [12] employed TF-IDF classification within a recommendation system for seasonal events based on marketplace inventory.

1.2.2 Machine Learning for Domain Specific Information Retrieval

Yao, Mao Luo [13] proposed a new convolutional networks for text classification. They build a single text graph for a corpus based on word co-occurrence and document word relations They, then, learn a Text Graph Convolutional Network for the corpus. Their work showed promise of less training data in text classification.

Kumar, Alshehri, AlGhamdi, Sharma, and Deep [14] built and trained an artificial neural network (ANN) to detect skin cancer. Their work suggests that DE-ANN is best compared among other traditional classifiers in terms of detection accuracy of 97.4%.

Sinoara, Camacho-Collados, Rossi, Navigli, and Rezende [15] present a natural language processing approach based on embedded representations of words and word sense. Their approach results in semantically enhanced and low-dimensional representations.

Aggarwal, Rani, and Kumar [16] employ machine learning to authenticate license plates. Their method correctly captures the license plates with good performance metrics of 93.34% accuracy (e.g. detection rate and false positive rate).

1.2.3 Recent TF-IDF/K-Means Text Research

Text classification has become an effective means to discover trends in text. Yung [17] employed k-means with TF-IDF to explore all the Queens Memory program`s 400+ oral history interviews collected in Queens, New York. Frymire [18] employed k-means with TF-IDF to explore the Twitter feeds of the Social Movement #me-too.

1.2.4 Dissertation Text Classification

Ellrodt et al. [3,4], Freeman et al. [5] and Haigler et al. [6] examined text classification of these doctoral dissertations. Ellrodt et al. [3,4] examines the abstracts from these 114 dissertations through the lens of machine learning with natural language processing techniques. The goal was to learn about topic categories to understand what the student dissertation topics were and to cluster them to recognize different patterns. Freeman et al. [5] examined the same 114 dissertation abstracts through IBM Watson and additional machine learning algorithms. Haigler et al. [7] examined and reported on the clustering for the full text of 98 (of the 114) dissertations; however, they focused on a smaller cluster count than this research.

1.2.5 EFWP Dissertation Research

Haigler et al. [6,7] explored research topics selected for EFWPs to help understand the research categories and trends. Haigler et al. [6] reported on educational needs for EFWPs obtained from IRB-approved surveys of Pace University's DPS program participants, which was further discussed in Haigler et al. [7]

2. Methodology/Methods

This research performs meta-research on research through both manual and machine classifications. Specifically, we examine all the dissertations defended in Pace University's Doctor of Professional Studies (DPS) from the program inception in the early 2000s until 2018. We analyzed each defended dissertation both through the abstract (e.g. approximately one page) and a full-text analysis (e.g. approximately 150 pages) to gain insights if automated NLP processing on abstracts is an adequate categorical measure for dissertation content over fulltext analysis. As fulltext analysis of approximately 150 pages requires large quantities of memory and storage, we compared the results of

DOI: https://doi.org/10.30564/jcsr.v2i4.2230

the two distinct analysis methodologies. Additionally, we discuss both the natural language processing (NLP) performed on the dissertations as well methods we used to classify the texts.

2.1 Natural Language Processing (NLP)

One of the goals of artificial intelligence is to develop semantic context for human language; the machine learning field of natural language processing pursues this goal for text documents. The seminal textbook on NLP is written by Jurafsky and Martin [19]. At a high-level, the text book describes almost every use case of NLP.

In keeping with the techniques described by Jurafsky and Martin [19], we examined both clustering the dissertation and abstract text with and without cleaning. To clean the data, we used the Python Natural Language Toolkit (*NLTK*) learn library. The tasks included in the data cleaning were for text standardization. First, we made everything in the text lower case. Then, we removed markup symbols, section formatting markers, special characters, stop words, and punctuation from the text. Lastly, we ran classifications on both stemmed and non-stemmed words to see clustering differences. Stemming processes words by reducing inflected and derived words to their root or base language form. It removes different word variations so that the actual word usage is standardized throughout the text.

2.2 Dissertation Classification

One of the goals of artificial intelligence is to develop semantic context for human language; the machine learning field of natural language processing pursues this goal for text documents.

2.2.1 Manual Classification

To manually classify the dissertation abstracts, we divided them amongst 5 people. Each person was tasked with reading some assigned subset of abstracts and determining the correct category for each work. This first pass of the abstracts produced 176 categories and much debate, as discussed in Ellrodt et al. [3,4]. In order to reduce the categories an iterative approach leveraging domain knowledge would have been needed. The researchers found the human iterative approach excessively time consuming and had trouble exercising stable categories. This suggested that without adequate domain knowledge, it would be difficult to communicate these categories in any meaningful way; the whole process seemed unlikely to produce worthwhile results and was abandoned.

In most human topic assignments, the people involved have training in the specific distinctions between the types of documents they are likely to run across. However, developing ad hoc topics to distinguish similar products in a corpus requires a level of sophistication and ability to form consensus that our workers did not let emerge.

2.2.2 Machine Learning Classification

This research uses the approach of applying K-means clustering analysis to term-frequency inverse-document-frequency (TF-IDF) coding of DPS dissertations. We then compare the TF-IDF analysis to the topic analysis of their abstracts in IBM Watson Discovery.

2.2.3 TF-IDF

TF-IDF for "term frequency - inverse document frequency" is a characterization tool for text documents. Each abstract is regarded as a "bag of words", as if the meaning of each abstract were implicit in the words used in that abstract and the order of those words were unimportant. Each individual word is then deemphasized according to how often it occurs in the collected abstracts overall. The formula for TF-IDF can be seen in Figure 1.

TFIDF

For a term i in document j:

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

tf_{ij} = number of occurrences of i in j
df_i = number of documents containing i
N = total number of documents

Figure 1. TF-IDF Formula

2.2.4 K-Means Clustering

"K-means" is an unsupervised learning algorithm that solves clustering problem. It is one of the most basic clustering algorithms and works with numeric data only. The algorithm is composed of the following steps: (1) Pick a number (K) of cluster centers at random.; (2) Assign every item to its nearest cluster center.; (3) Move each cluster center to the mean of its assigned items.; (4) Repeat the prior two steps until convergence. After initialization, the k-means algorithm iterates between the following two steps: (1) Assign each data point to the closest centroid, as seen in Figure 2., and,

$$z_i \leftarrow \operatorname{argmin}_j \left\| \mu_j - \mathbf{x}_i \right\|^2$$

Figure 2. Assign each data point to the closest centroid

 DOI: https://doi.org/10.30564/jcsr.v2i4.2230

(2) Revise centroids as the mean of the assigned data points, as seen in Figure 3.

$$\mu_j \leftarrow \frac{1}{n_j} \sum_{i:z_i=j} \mathbf{x}_i$$

Figure 3. Revise centroids as the mean of the assigned data points

The algorithm has convergence when the cluster assignments no longer change. There is no assurance that the cluster assignments are optimal using K-means. Clusters will be reasonable, however may not be robust to different start point selection. In k-means, the number of clusters must be selected beforehand. The algorithm is very sensitive to outliers. It can be proved that the running of the algorithm will always terminate. How can we tell if the k-means algorithm is converging? We can look at the cluster assignments and see if they stabilize over time. In fact, we'll be running the algorithm until the cluster assignments stop changing at all. To be extra safe, and to assess the clustering performance, we'll be looking at an additional criteria: the sum of all squared distances between data points and centroids, as defined in Figure 4.

$$J(\mathcal{Z}, \mu) = \sum_{j=1}^{k} \sum_{i:z_i=j} \|\mathbf{x}_i - \mu_j\|^2$$

Figure 4. Assessing convergence

2.2.5 IBM Watson Discovery

IBM Watson Discovery is a cloud platform which ingests and standardizes user data, providing services such as sentiment analysis, named entity extraction and concept tagging through an API [20]. In addition to providing chatbots and other workflow enhancements, Watson Discovery provides Smart Document Understanding, a clustering solution [21].

3. Results

We applied K-means analysis to term-frequency inverse-document-frequency (TF-IDF) coding of 114 of Pace University's DPS dissertations and then compare the output of that analysis to the topic analysis of their abstracts in IBM Watson Discovery.

3.1 IBM Watson Discovery Classification

We chose to examine the dissertations through machine learning using IBM Watson Discovery using both standard classification as well as enriched classification. This analysis extends the work of Ellrodt et al. [3,4] and Freeman et al. [5], where the dissertation abstracts were classified using a TF-IDF algorithm.

The IBM Watson Discovery system produced the top six enriched text concepts show in Table 1 as: Scientific method (14), Algorithm (11), Management (11), Computer (10), Mathematics (10), and Agile software development (9). The enriched text key-words were: Research (29), Dissertation (16), Model (11), Study (11), Approach (10), and Addition (9). None of this second list of individual words relate to any specific topic in the computing field, which is a comparative weakness of this approach to the TF-IDF analysis.

The system sentiment analysis examines sentence word choices with respect to sentiments. On the default configuration IBM labeled the 114 dissertation abstracts as follow: 85% (97) positive, 3% (3) neutral, and 12% (14) negative based on the word used. Interestingly, industry word choices like "false negative" triggered the negative analysis categorization.

Table 1. Watson Topic Analysis for Abstracts

Topic	Assigned Papers
Scientific method	14
Algorithm	11
Management	11
Computer	10
Mathematics	10
Agile software development	9
Software engineering	9
Education	8
Internet	8
Computer program	7
Computer science	7
Waterfall model	7

3.2 IBM Watson Full-Text Analysis

Using IBM Watson Discovery, we performed full-text PDF analysis, extending the work of Freeman et al. [5] and Haigler et al. [6]. Full text PDFs of the 114 dissertations were uploaded to Watson and evaluated via the basic Watson Discovery Natural Language Understanding (NLU) engine. The basic engine yields results such as sentiment analysis, related concepts and top entities.

The results of related concepts are listed using enriched text produced by Watson. The top six listed are: Software engineering (13), Agile software development (12), Computer (12), Software development (12), Biometrics (9),

DOI: https://doi.org/10.30564/jcsr.v2i4.2230

and Extreme Programming (9). This resulted in a different categorization of the dissertations than categorization of the abstracts alone.

IBM sentiment analysis labeled the 114 dissertation full text as follow: 82% positive, 2% neutral, and 16% negative.

3.3 Amazon EC2 Full-Text Analysis: K-Means with TF-IDF

The selection of k - the cluster count - is the primary hyperparameter for a k-means model. In order to select k, we let k vary from one to 25 - more than a fifth of the document count - and observed the heterogeneity. We assess the heterogeneity of a single cluster as the sum of all squared distances between data points in that cluster and its centroids.

Heterogeneity should decrease more quickly after the optimal point, at which point the model is overfit. The heterogeneity is plotted in Figure 1.

This plot does not have a clear point after which heterogeneity decreases, and so some further treatment is used to expose the optimal cluster count k. Therefore, we apply a log transformation and de-slope the output to look for discontinuity.

It is evident in Figure 5 that heterogeneity is discontinuous at a cluster count of eight and above. Therefore, we treat eight as the optimal value of k across all four experiments, which is unique from the work of Ellrodt et al. [3,4], Freeman et al. [5] and Haigler et al. [6,7].

Table 2 shows the keywords and cluster labels for k = 8.

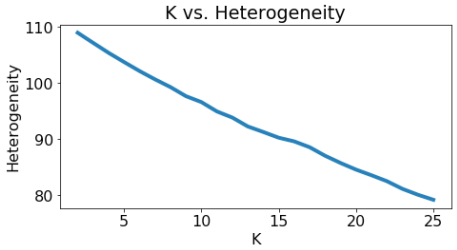

Figure 5. Heterogeneity v cluster count of non-stemmed full text dissertations

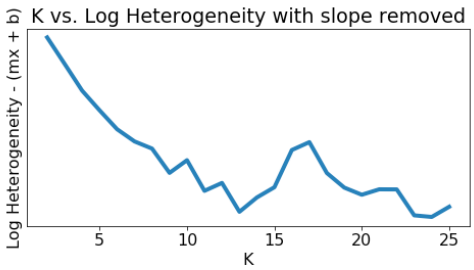

Figure 6. De-sloped log heterogeneity by cluster count of non-stemmed full-text dissertations

The results of running the TF-IDF/K-means clustering for eight clusters, produced the results seen in Table 2. Depicted in the table are the following: the cluster number, number of dissertations clustered into the category, top five key-words identified by the algorithms, our human interpretation of the category and the dissertation papers clustered in this category. Github [22] shows the abstract paper number title mapping given in the tables.

Table 2. K-Means with fulltext dissertations non-stemmed, k=8

#	*Instance Count*	*Top Five Keywords*	*Category*	*Paper Number*
1	17	cloud:0.168 compliance:0.068 security:0.067 hipaa:0.048 csp:0.047	Data security	16, 18, 19, 20, 22, 31, 49, 57, 66, 69, 72, 75, 77, 90, 99, 108
2	26	students:0.084 instructor:0.041 teaching:0.037 cics:0.037 abstraction:0.034	Education	2, 9, 10, 15, 24, 27, 32, 33, 35, 54, 55, 58, 63, 68, 71, 82, 83, 84, 89, 94, 102, 104, 105, 111, 112, 114
3	14	agile:0.172 team:0.116 retrospective:0.094 kms:0.068 pba:0.060	Agile software practices	3, 8, 26, 40, 42, 56, 62, 64, 67, 74, 76, 85, 93, 101
4	10	int:0.180 sa:0.128 annealing:0.071 fitness:0.070 patch:0.066	Optimization	1, 4, 6, 21, 36, 38, 39, 70, 103, 106
5	5	irs:0.219 impute:0.143 pottery:0.137 loyalty:0.123 recommender:0.084	Machine learning categorization	11, 51, 52, 88, 107
6	13	schematron:0.173 xml:0.164 owl:0.092 ontology:0.089 rdf:0.088	Ontology	7, 12, 28, 29, 37, 48, 53, 86, 87, 91, 92, 95, 98
7	19	keystroke:0.132 biometric:0.085 roc:0.057 classifier:0.052 svm:0.042	Biometrics	13, 23, 25, 30, 34, 41, 44, 46, 50, 59, 60, 65, 73, 78, 79, 80, 81, 97, 113
8	10	pda:0.097 channel:0.092 wireless:0.080 uumi:0.068 callback:0.061	Distributed software architecture	5, 14, 17, 43, 45, 47, 61, 96, 109, 110

The categories listed in Table 2 are meaningful human-assigned text labels to summarize the top five keywords extracted by the k-means on TF-IDF analysis. The categories we selected are discussed below in Table 3.

DOI: https://doi.org/10.30564/jcsr.v2i4.2230

Table 3. Meaningful Human-Assigned Cluster Text Labels

• ***Data security*** is a current topic in the computing world.
• ***Education*** is focus on the students, many of whom are trying to get credentials to advance an academic career.
• ***Agile software practices*** are a focus of the program content.
• ***Optimization*** is discussed in several classes and encouraged as a dissertation focus by faculty.
• ***Machine learning categorization*** is another focus of the program class content.
• ***Ontology*** is referred to in some courses and encouraged as a dissertation area.
• ***Biometrics*** is a particular focus of research at Pace University.
• ***Distributed software architecture*** is a mainstay of many students' work lives and supported by the program as a dissertation area.

Figure 7 is a visualization for the full non-stemmed dissertations cluster counts where k=8. In contrast, Figure 8 is a visualization for the non-stemmed abstracts cluster counts.

Cleaning data is a common methodology to improve the quality of natural language processing results. We employed the *NLTK* toolkit to clean the data (e.g. remove stop words, Porter stem, remove non-ASCII characters, etc.) Table 4 shows the keywords, and dissertations for eight clusters.

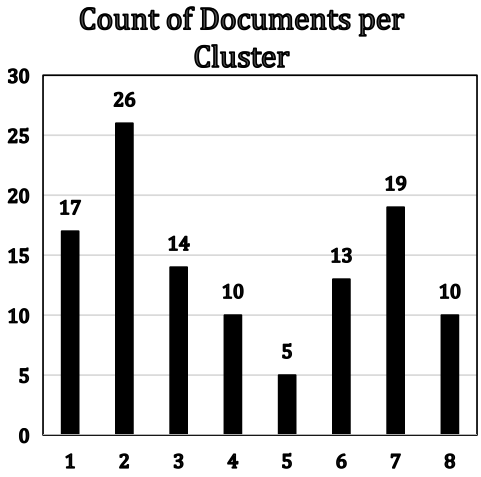

Figure 7. Counts of non-stemmed Full -Text Documents Per Cluster

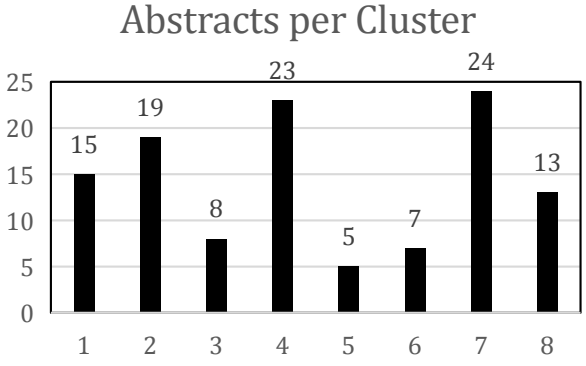

Figure 8. Counts of Abstracts per Cluster

Table 4. K-Means fulltext stemmed, k=8 where highlighted numbers changed from non-stemmed categories

#	*Instance Count*	*Top Five Keywords*	*Category*	*Paper Number*
1	14	cloud:0.225 csp:0.091 cic:0.069 packet:0.067 ip:0.063	Cloud Computing	**5, 16, 20, 36, 45, 47, 55, 63, 71, 72, 75, 90, 110, 114**
2	13	student:0.179 instructor:0.117 teach:0.090 game:0.078 cs:0.074	Education	10, 15, 27, **28**, **32**, 58, 68, **74**, 84, 89, 94, 102, 111
3	30	keystrok:0.094 biometr:0.069 imput:0.045 roc:0.039 distanc:0.031	Biometrics	**9, 11,** 13, **14, 21**, 23, **24**, 25, 30, 34, **44**, 46, **49**, 50, **51**, **54**, 59, 60, 65, 73, 78, 79, 80, 81, **83, 96**, 97, **101, 104**, 113
4	22	agil:0.109 team:0.088 retrospect:0.074 complianc:0.051 secur:0.046	Agile Software Development	8, **18, 19,** 26, **31**, 40, **52**, 56, 57, 62, 64, **66**, 67, **76, 77, 82**, 85, **87, 99, 100, 106, 108**
5	12	schematron:0.193 xml:0.190 schema:0.080 xs:0.069 recip:0.067	Data Validation	**3, 7, 12, 17, 29, 35, 48, 53, 61, 88, 92, 95**
6	12	int:0.163 sa:0.107 km:0.080 ler:0.073 ga:0.072	Optimization	1, 4, 6, 38, 39, **41, 42**, 70, **93**, 103, **105, 112**
7	3	pda:0.371 pervas:0.159 itamm:0.147 button:0.131 lotu:0.128	Human Computer Interaction	**22, 43, 109**
8	8	ontolog:0.158 rdf:0.122 owl:0.112 drug:0.104 pir:0.103	Medical Ontologies	**2, 33, 37, 69, 86, 91, 98, 107**

The heterogeneity for the stemmed full-text dissertations, to contrast with the above non-stemmed heterogeneity, is shown in Figure 9.

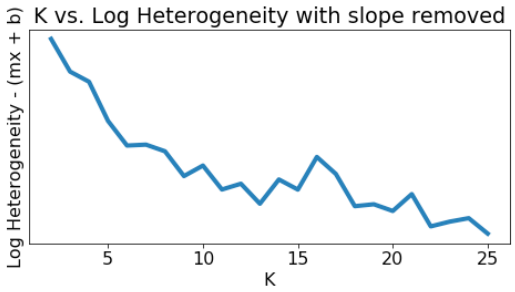

Figure 9. Heterogeneity v cluster count of stemmed full-text dissertations

The bolded dissertations listed in Table 4 show the category change from when a dissertation is categorized in a non-stemmed and stemmed-format. As we can see from the table, over half (i.e. 71 of 114) have changed categories after applying the stemming indicating that stemming is an essential standardization prior to categoriatzation.

Ellrodt et al. [1,2] and Freeman et al. [3] examined text classification of these same 114 doctoral dissertations; however, they report that the heterogeneity is discontinuous at a cluster

DOI: https://doi.org/10.30564/jcsr.v2i4.2230

count of six and above. Therefore, the earlier abstract-only analysis treated six as the optimal value of k (i.e. categories).

Thus, to contrast to the clustering of eight-categories for the stemmed and non-stemmed 114 dissertations abstract, in Table 5, we clustered the non-stemmed dissertation abstracts into eight categories (i.e. k=8).

Table 5. K-Means with dissertation abstracts non-stemmed, k=8

#	*Instance Count*	*Top Five Keywords*	*Category*	*Paper Number*
1	15	keystroke:0.104 biometric:0.071 authentication:0.058 input:0.052 beta:0.052	Biometrics	4, 5, 16, 17, 34, 35, 61, 63, 70, 76, 80, 89, 91, 100, 101
2	19	agile:0.105 development:0.069 software:0.057 team:0.046 outsourcing:0.041	Agile Software Development	2, 14, 31, 39, 45, 46, 51, 52, 60, 62, 67, 69, 71, 74, 86, 96, 97, 102, 111
3	8	loyalty:0.079 user:0.073 sparse:0.065 capacity:0.059 unusual:0.057	Optimization	8, 15, 21, 22, 23, 41, 77, 104
4	23	security:0.051 components:0.037 cloud:0.037 application:0.033 requirements:0.033	Security	1, 11, 13, 19, 20, 24, 25, 32, 36, 40, 42, 47, 48, 50, 59, 73, 75, 78, 81, 87, 93, 113, 114
5	5	shape:0.159 pottery:0.143 images:0.102 classification:0.100 shapes:0.098	Machine Learning	28, 105, 107, 108, 110
6	7	students:0.124 erp:0.116 computer:0.077 schillinger:0.076 programming:0.070	Education	6, 7, 18, 29, 54, 90, 95
7	24	factors:0.059 cloud:0.044 genetic:0.040 algorithm:0.033 problems:0.033	Genetic Algorithms	1, 9, 12, 26, 27, 30, 37, 38, 43, 44, 49, 57, 58, 65, 66, 68, 79, 84, 85, 94, 99, 103, 106, 112
8	13	xml:0.164 documents:0.074 document:0.069 semantic:0.067 constraints:0.058	Ontologies	10, 33, 53, 55, 56, 64, 72, 82, 83, 88, 92, 98, 109

The heterogeneity for the non-stemmed abstract dissertations, to contrast with the above non-stemmed heterogeneity, is shown in Figure 10 and log heterogeneity in Figure 11.

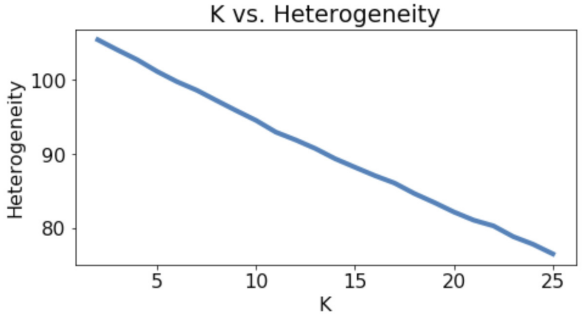

Figure 10. Heterogeneity v cluster count of non-stemmed abstract dissertations

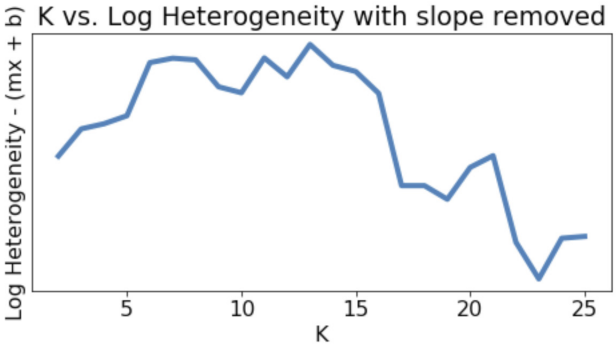

Figure 11. Developed log heterogeneity by cluster count of non-stemmed abstract dissertations

Thus, to contrast to the clustering of eight-categories for the stemmed and non-stemmed 114 dissertations dissertations, in Table 6, we clustered the stemmed dissertation abstracts into eight categories (i.e. k=8). The bolded dissertation abstract paper numbers changed categories from the stemmed fulltext categories.

Table 6. K-Means abstracts stemmed, k=8, where highlighted numbers changed from stemmed full categories

#	*Instance Count*	*Top Five Keywords*	*Category*	*Paper Number*
1	9	factor:0.094 shape:0.091 technolog:0.073 search:0.069 name:0.067	Machine Learning	**14, 25, 39, 44, 62, 63, 76, 78, 105**
2	18	secur:0.139 cloud:0.066 complianc:0.065 knowledg:0.057 risk:0.045	Cloud Computing Security	**8, 35, 52, 55, 58, 60, 65, 67, 71, 75, 77, 90, 100, 101, 108, 109, 110, 111**
3	21	agil:0.092 student:0.089 scienc:0.054 softwar:0.051 learn:0.048	Agile Software Development	**1**, **2**, **5,** 6, 7, 12, **13**, **27**, **29**, **30**, **43**, **49**, **59**, **66, 69, 85, 87, 92, 94, 99,** 102
4	12	algorithm:0.122 genet:0.086 problem:0.084 ann:0.083 optim:0.077	Genetic Algorithms	**10, 22, 24, 28, 42, 45, 47, 56, 74, 82, 84, 96**
5	7	estim:0.216 project:0.170 retrospect:0.107 elf:0.099 binari:0.092	Project Mangement	**21, 50, 64, 73, 81, 83, 93**
6	17	keystrok:0.111 biometr:0.088 featur:0.084 text:0.064 classif:0.056	Biometrics	**0**, **3**, **4**, **15**, 26, **32, 33, 41,** 61, 68, 72, 80, **86,** 88, **97,** 103, **104**
7	2	insur:0.281 cybersecur:0.256 gi:0.216 risk:0.201 financi:0.193	Security	**40, 57**
8	26	xml:0.088 busi:0.047 site:0.043 agent:0.042 constraint:0.042	Data Validation	**9, 11, 16, 17, 18, 19, 20, 23, 31, 34, 36, 37, 38, 46, 48, 51, 53, 54, 70, 79, 89, 91, 95, 98, 106, 107**

The heterogeneity for the stemmed full-text dissertations, to contrast with the above non-stemmed log heterogeneity, is shown in Figure 12.

DOI: https://doi.org/10.30564/jcsr.v2i4.2230

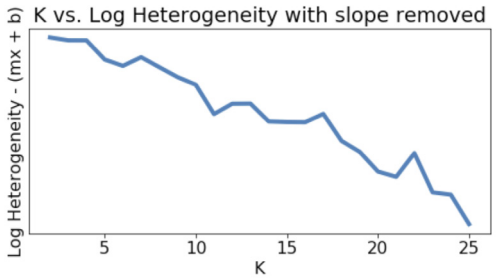

Figure 12. Developed log heterogeneity by cluster count of stemmed abstract dissertations

As we can see from Table 6 that all but eleven dissertations change categories indicating that abstract analysis alone is not a good indicator of content or dissertation category.

4. Discussion

Of the tools we examined, IBM Watson Discovery allowed for the quickest startup process. It easily ingested and categorized all 114 dissertations in less than 30 minutes from start to finish. In contrast, the k-means analysis using Turi's GraphLab Create TF-IDF coding was a more tedious process to set up the environment which required programming and debugging of the code, converting each dissertation into a text format, ingesting the dissertation, and finally running the algorithm on a large AWS EC2 server. Once running on the large server, the acual algorithm took approximately a minute to run.

Each of the machine learning algorithms produce different categories of dissertation research topics. The Watson Discovery system enriches the text using a natural language understanding module and the resulting categories shown in Table 1 are different from the categories produced by k-means as shown in Table 2, Table 4, Table 5, and Table 6. Overall, the dissertation abstracts appear to not be a good representation of the full dissertation. A future work study would involve an IRB-approved survey to ack the authors their own human interpretation for their own dissertation category.

In all cases we found that EFWP students tend to favor the emerging technologies they face in industry. The Pace University DPS program allows the student to select their research topic-then match an advisor versus the traditional Ph.D. program where students research tends to follow that of their academic advisor.

Future work involves further examination of the machine generated cluster category of each dissertation as compared to the authors actual intent. This work can further be extended to examine the year of the dissertation defense to determine if they are aligned with industry technology trends of the time.

5. Conclusion

There exists very little research on research itself from the perspective of text analysis. In this research we have performed cluster analysis on both the fulltext and abstracts for the first 114 dissertations defended in Pace University's DPS program to see what topics have been the doctoral focus of senior and experienced fulltime working professionals (EFWPs). We found that many students tend to focus their research on industry trends first; and, then, find an adviser. As such, the DPS dissertation research is typically different than the Ph.D. program model where students focus on their advisor's topics. We also showed that data preprocessing including stemming did slightly change the clusters identified by the machine learning algorithms. We also showed that fulltext analysis produces different categories than abstract analysis indicating that abstract analysis alone may not be sufficient for categorizing dissertation research. Future work such as examining longitudinal-trends, innovation, accountability, and automatic keyword generation can be further developed from our research. Lastly, we showed that machine learning on the abstract alone were not good indicators on dissertation content. As more and more text becomes digitally available, we must continue to develop methodologies to build semantic understandings from the available data.

Supplementary Data/Information

A full list of the dissertation full-text analysis mapping identifiers to their abstract-analysis identifiers can be found on GitHub [22].

References

[1] Susan M. Merritt, Allen Stix, Judith E. Sullivan, Fred Grossman, Charles C. Tappert, and David A. Sachs. Developing a professional doctorate in computing: a fifth-year assessment. In Working group reports from ITiCSE on Innovation and technology in computer science education (ITiCSE-WGR '04). ACM, New York, NY, USA, 2004: 42-46. DOI: http://dx.doi.org/10.1145/1044550.1041654

[2] Fred Grossman, Charles Tappert, Joe Bergin, and Susan M. Merritt. A research doctorate for computing professionals. Commun. ACM 54, 2011, 133-141. DOI: https://doi.org/10.1145/1924421.1924450

[3] L. R. Ellrodt, I. C. Freeman, A. J. Haigler, S. E. Schmeelk. Doctor of Professional Studies in Computing: A Categorization of Applied Industry Research. in 2018 IEEE Frontiers in Education Conference (FIE), 2018: 1-6.

DOI: https://doi.org/10.30564/jcsr.v2i4.2230

[4] Lisa R. Ellrodt, Ion C. Freeman, Ashley J. Haigler, Lynne E. Larkin, Suzanna E. Schmeelk, Ronald G. Williams. Pace University DPS in Computing Studies: A Categorization of Applied Industry Research. The Michael L. Gargano 16th Annual Research Day. Pace University. May 2018 Retrieved from: http://csis.pace.edu/~ctappert/srd/index.htm

[5] I. Freeman, A. Haigler, S. Schmeelk, L. Ellrodt, T. Fields. What are they Researching? Examining Industry-Based Doctoral Dissertation Research through the Lens of Machine Learning. In 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), 2018: 1338-1340.

[6] Ashley J. Haigler, Suzanna E. Schmeelk, Tonya L. Fields, Lisa R. Ellrodt, Ion C. Freeman. Employing Machine-Learning to Understand Research Trends of Full-Time Working Professionals. The Michael L. Gargano 17th Annual Research Day. Pace University, 2019. Retrieved from: http://csis.pace.edu/~ctappert/srd2019/schedule.htm

[7] Ashley J. Haigler, Suzanna E. Schmeelk, Tonya L. Fields, Lisa R. Ellrodt, Ion C. Freeman. Educational Needs in Computing of Experienced Full-Time Working Professionals. Future of Education. Florence, Italy, 2019.

[8] Dhillon, Paramvir, Amandeep Walia. A Study on Clustering Based Methods. International Journal of Advanced Research in Computer Science, vol. 8, no. 4, May 2017, 8(4): 1-5, 0967-5697.

[9] J. A. Hartigan, M. A. Wong. Algorithm AS 136: A K-Means Clustering Algorithm. Journal of the Royal Statistical Society, 1979, Series C. 28(1): 100-108.

[10] Claire Fautsch, Jacques Savoy. Adapting the tf idf vector-space model to domain specific information retrieval. In Proceedings of the 2010 ACM Symposium on Applied Computing (SAC'10). Association for Computing Machinery, New York, NY, USA, 2010, 1708-1712. DOI: https://doi.org/10.1145/1774088.1774454

[11] Duan, Xiaolin Gui, Mingan Wei, You Wu. A Resume Recommendation Algorithm Based on K-means++ and Part-of-speech TF-IDF. In Proceedings of the 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM 2019). Association for Computing Machinery, New York, NY, USA, 2019, Article 50: 1-5. DOI: https://doi.org/10.1145/3358331.3358381

[12] Ted Tao Yuan, Zezhong Zhang. Merchandise Recommendation for Retail Events with Word Embedding Weighted Tf-idf and Dynamic Query Expansion. In The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '18). Association for Computing Machinery, New York, NY, USA, 2018, 1347-1348. DOI: https://doi.org/10.1145/3209978.3210202

[13] Yao, L., Mao, C., Luo, Y. Graph convolutional networks for text classification. In Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33: 7370-7377.

[14] Kumar M., Alshehri M., AlGhamdi R., Sharma P., Deep V., 2020. A DE-ANN Inspired Skin Cancer Detection Approach using Fuzzy C-Means Clustering. Mobile Network and Applications, 2020. DOI: https://doi.org/10.1007/s11036-020-01550-2

[15] Sinoara, R.A., Camacho-Collados, J., Rossi, R.G., Navigli, R., Rezende, S.O. Knowledge-enhanced document embeddings for text classification. Knowledge-Based Systems, 2019, 163: 955-971.

[16] Aggarwal A., Rani A., Kumar M. A Robust Method to Authenticate License Plates using Segmentation and ROI Based Approach. Smart and Sustainable Built Environment, 2019. DOI: https://doi.org/10.1108/SASBE-07-2019-0083

[17]Stephanie Yung. All the Queens Voices: An Oral History, Visualized. A data visualization of Queens Memory program`s 400+ oral history interviews collected in Queens, New York." Thesis, Parsons School of Design at The New School. New York, 2019. Retrieved from: http://parsons.nyc/thesis-2019/#13

[18]Ellie Frymire. An Exploration of the Social Movement #metoo. Thesis, Parsons School of Design at The New School. New York, 2018. Retrieved from: https://parsons.nyc/thesis-2018/#4

[19] Daniel Jurafsky and James H. Martin. Speech and Language Processing (2nd Edition). Prentice-Hall, Inc., USA, 2009.

[20] C. Forrest. IBM launches Watson Discovery Service for big data analytics at scale. TechRepublic. Retrieved from: https://www.techrepublic.com/article/ibm-launches-watson-discovery-service-for-big-data-analytics-at-scale/

[21] IBM. Corporation. Watson Discovery. IBM. Retrieved from: https://www.ibm.com/watson/services/discovery/

[22] Suzanna E. Schmeelk, Tonya L. Fields, Lisa R. Ellrodt, Ion C. Freeman, Ashley J. Haigler. GitHub: JSCR Machine Learning: Researching the Research, 2020. Retrieved from: https://github.com/schmeelk/jcsr_ml_researchingtheresearch

 DOI: https://doi.org/10.30564/jcsr.v2i4.2230