LDA v. LSA: A Comparison of Two Computational Text Analysis Tools for the Functional Categorization of Patents

Toni Cvitanic[1], Bumsoo Lee[1], Hyeon Ik Song[1], Katherine Fu[1], and David Rosen[1]

[1]Georgia Institute of Technology, Atlanta, GA, USA
(tcvitanic3, blee300, hyeoniksong) @gatech.edu
(katherine.fu, david.rosen) @me.gatech.edu

Abstract. One means to support for design-by-analogy (DbA) in practice involves giving designers efficient access to source analogies as inspiration to solve problems. The patent database has been used for many DbA support efforts, as it is a pre-existing repository of catalogued technology. Latent Semantic Analysis (LSA) has been shown to be an effective computational text processing method for extracting meaningful similarities between patents for useful functional exploration during DbA. However, this has only been shown to be useful at a small-scale (100 patents). Considering the vastness of the patent database and realistic exploration at a large-scale, it is important to consider how these computational analyses change with orders of magnitude more data. We present analysis of 1,000 random mechanical patents, comparing the ability of LSA to Latent Dirichlet Allocation (LDA) to categorize patents into meaningful groups. Resulting implications for large(r) scale data mining of patents for DbA support are detailed.

Keywords: Design-by-analogy • Patent Analysis • Latent Semantic Analysis • Latent Dirichlet Allocation • Function-based Analogy

1 Introduction

Exposure to appropriate analogies during early stage design has been shown to increase the novelty, quality, and originality of generated solutions to a given engineering design problem [1-4]. Finding appropriate analogies for a given design problem is the largest challenge to practical implementation of DbA. There have been numerous efforts to address this challenge with computational support for targeted access to design repositories, which will be reviewed next. The major research gap is in the scale of implementation, the size of the repository being accessed. To address this gap, we compare two computational approaches to processing design repository content (patents) for categorization and similarity judgment, with the goal of both (1) evaluating the methods in direct juxtaposition to one another, and (2) developing a method to examine the effectiveness of data synthesis techniques at a large scale. In the context of the Case-Based Reasoning (CBR) Workshop on Computational Analogy, this work directly addresses methods for identifying and retrieving analogies, similarity measures for analogy, analogical distance metrics, and data mining techniques for textual CBR.

2 Background

2.1 Patent-Based Design and DbA Tools

Patents are often used as input for design tools and repositories because of the large amount of information captured by the patent database, already deemed novel and useful in nature by its inherent patentability [5]. Patents have been used to develop

In Proceedings of the ICCBR 2016 Workshops. Atlanta, Georgia, United States of America

conceptual graphs of claims [6] and dependency structures and content relationships [7]. Patents have been mapped to: extract the implicit structure in the data to support DbA [8-10], to understand overlap in IP portfolios for mergers and acquisitions [11], to search through patents for DbA support [12], to assist with patent infringement analysis in the biomedical arena [13], and to build a taxonomy from the data [14]. TRIZ, the Theory of Inventive Problem Solving, is one the major efforts involving the use of the patent database to support design [15], and a number of tools have been developed based upon the original efforts of Genrich Altshuller [15-26]. The computational analysis presented in this paper contributes to these efforts by showing a direct comparison of two leading computational text analysis that can and do serve as the basis of many of these and future patent based design tools.

2.2 Latent Dirichlet Allocation (LDA)

Within the large field of "data mining," a body of knowledge has emerged that provides methods for managing large document archives (text corpus). Tools have been developed that can summarize a corpus, classify articles into categories, identify common themes, and help users find relevant articles. A specific class of methods, called topic modeling, is particularly promising for its potential to form a readily explorable database of patents, or other documents, for use in DbA. As one of the leaders in this area notes [27], "topic modeling algorithms are statistical methods that analyze the words of the original texts to discover the themes that run through them, how those themes are connected to each other, and how they change over time."

Topic modeling grew from LSA in several directions. Inputs to methods typically include a word-document matrix that records the number of times a particular word is included in one document. In the early 2000's, a different approach called Latent Dirichlet Allocation (LDA) [28] was developed, where the basic idea is "that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words."

Many variants of LDA have been developed over the years. Of note, supervised LDA methods enable the user to specify some topics and the corpus analysis seeks to include these seeded topics in its overall probabilistic model [29, 30]. Another extension is the use of nonparametric Bayesian methods to determine hierarchies of topics from LDA results [31, 32]. More recently, several researchers have investigated variants of PCA and other least-squares regression formulations for topic modeling, including sparse matrix formulations. El Ghaoui et al. [33] compared LASSO regression [34] and sparse PCA [35] to LDA and found comparable efficacy at topic modeling, but that LASSO and sparse LDA were significantly more efficient. Another group investigated Non-negative Matrix Factorization (NMF) [36] for interactive topic modeling and found computational performance sufficiently fast [37].

2.3 Latent Semantic Analysis (LSA)

LSA is a computational text analysis tool that builds a semantic space from a corpus of text. This semantic space is then used to compute the similarity between words, sentences, paragraphs, or whole documents for a wide variety of purposes [38-41]. Note that this semantic space is a high-dimensional vector space (typically 300 or more dimensions) with little inspectable value to humans; additional methods are needed to create that inspectable structure. After performing LSA, the results can be compared directly to LDA output, or can become input for further algorithmic

processing to understand the similarity values in a different way.

In ref. [9], the functional content (verbs) and surface content (nouns) of patents were processed and mapped separately, yielding structures that have the potential to develop a better understanding of the functional and surface similarity of patents, for the sake of analogical knowledge transfer. Structures created with this methodology yield spaces of patents that are meaningfully arranged into labeled clusters, and labeled regions, based on their functional similarity or surface content similarity. Examples show that cross-domain analogies and transfer of knowledge based on functional similarity can be extracted from the function based structures, and even from the surface content based structures as well.

More generally, LSA has mixed reception due to its inability to match observed data, for example predicting human word associations. This is due to the nature of the spatial representation that is intrinsic to LSA, forcing symmetry in similarity of words and imposition of the triangle inequality, among others. While these criticisms are valuable, they are at the word-to-word comparison level, which may or may not become trivial with very large corpuses and repository sizes.

3 Research Methods

3.1 Theoretical Approach

LSA gives a direct comparison between different patents in the form of a cosine similarity matrix, where document similarities range from -1 (the two documents are complete opposites) to 1 (the two documents are the same). However, LDA works a bit differently, in that it assigns the words of a document to different topics, and has no output that directly compares documents. However, using a document vector technique, described in a subsequent section on the implementation of LDA, it is possible to use the data output from LDA to build a matrix of document similarities.

For the purposes of comparison, the actual values within the document-similarity matrices obtained from LSA and LDA are not important. In order to compare the two methods, only the order of similarity between documents was used. This was done by organizing the document-similarity matrices so that for a given column, every row down, starting from the second, represents a document that is less similar to the document in the first row than all of the documents above it (see Fig. 1).

By comparing the order in which documents were rated on similarity between LSA and LDA, it is possible to judge how similar or different the results of the two methods are. In the case that the two methods yield substantially different results, a qualitative analysis can be done to determine if one method better sorts based on functionality. There are many ways to go about this, but one effective check is to look at the top 50 rows in the document-similarity matrices, and count the average number of patents with the same core functions (determined by first author, not automated), then see which method yielded a greater number.

3.2 Data Selection

Patents were selected from a set of all US CPC patents found in the bulk data storage system of the United States Patent and Trademark Office (USPTO) at

Document 1	Document 2	Document 3	Document 4
Most Similar	Most Similar	Most Similar	Most Similar
2nd Most Similar	2nd Most Similar	2nd Most Similar	2nd Most Similar
3rd Most Similar	3rd Most Similar	3rd Most Similar	3rd Most Similar

Fig 1. Example of Document Comparison Matrix

https://data.uspto.gov/data2/patent/classification/cpc/. For this study, only patents from the CPC section F, the section for mechanical patents, were used. Any patents that were cross-listed under multiple CPC sections were removed from the study's dataset in order to reduce the scope of the data for document matching, in an effort to get more coherent results from both the LDA and LSA methods. In addition, any withdrawn patents were removed from the dataset. Finally, any patent number that is below 3,930,270 is not accessible on the USPTO search online and was removed.

Once the study dataset was finalized, four patents were selected manually for a small-scale test. For the large-scale test, 996 patents were selected using a pseudo-random number generator built into MatLab. 996 patents were chosen to have 1000 patents, including the four from small-scale test.

3.3 Data Pre-Processing

Both LDA and LSA take a word by document matrix as an input. Each row represents a word from the entire dataset, and each column represents a patent. Each location in the matrix has a number that corresponds to the number of times the word designated by the row appeared in the document designated by the column. Before this word by document matrix was created, however, some pre-processing was done on the data.

First, a program was created to read the patents and retain only words from the abstract, description, and claims sections. These sections are the most representative of the mechanical nature of a patent. In addition, symbols and numbers were removed from the dataset. Next, the entire dataset was run through a spellchecker to remove any misspelled words. Then, words contained in a list of "stop words" were removed, which are words deemed to have no value in describing the mechanical qualities of a patent. For even further reduction, any words common to 90% or more of the patents were removed, further reducing words that do not distinguish one patent from another. The 90% cutoff was chosen through experimentation. When lower than ~80%, words that are important mechanical descriptors were excluded. The cutoff was set to 90% to include a margin of error.

3.4 Latent Semantic Analysis (LSA)

LSA gives a direct output of document similarities in the form of a cosine similarity matrix. Values range from -1 to 1, where -1 represents two documents that are complete opposites, and 1 represents two of the same document. This output is sufficient to create a matrix whose columns each represent a document and whose rows contain documents in their order of similarity to the document associated with the column they are in. This matrix is the desired output for this study, and no further processing is needed once it is obtained.

3.5 Latent Dirichlet Allocation (LDA)

Unlike LSA, LDA does not directly output document similarities. Instead, LDA outputs a matrix, z, whose rows represent all the words in the dataset, and columns represent all the documents. Each value in the matrix represents a topic that the word represented by the row and column is assigned to by the LDA algorithm. The user specifies the total number of topics that the words are sorted into, and each value in the matrix ranges between 0 and the user-defined number of topics.

LDA was run with different numbers of topics until a good topic range was found for the dataset. This range is determined by looking at the word-topic assignments

output for each number of topics. If individual topics are judged, subjectively, to contain groups of words that should belong to more than one topic, then the algorithm is run again with more topics. If there are many empty or sparsely populated topics, the algorithm is run with fewer topics. For the small-scale test, experiments were run with k = 2, 4, and 6 to see what number of topics is appropriate for the comparison. For the large-scale test (1000 patents), 150 topics provided the best sorting.

In order to compare documents, it is necessary to represent a document's subject matter by the topics found within that document. For this study, this was done using the "document vector" method. In this method, each document is represented as a vector whose length is equal to the total number of topics. Each component of the vector represents a topic, so the first component represents topic 1, the second, topic 2, and so on. Each component of the document vector is then assigned a value that is equal to the number of words in the document that were assigned to that topic. So, if a document had 20 words assigned to topic 3, the third component of the vector would have a value of 20. Next, the vector is normalized by dividing it by the total number of words in the document it represents. In order to compare two documents, one subtracts their document vectors, then takes the magnitude of the resulting vector, the L2 norm. The lower the magnitude of this resulting vector, the more similar the documents are.

The magnitudes of the differences of these document vectors can be considered similarity scores, where a lower score corresponds to a higher similarity. Having these scores, it is possible to create a matrix which orders documents based on their similarity, the same way it was done for the LSA output.

3.6 Data Post-Processing

The final step is to compare the document similarity matrices output by LSA and LDA. If only minor differences can be found between them, it can be concluded that LSA and LDA are more or less equal in their ability to sort mechanical patents. However, if the two matrices differ significantly, the more effective algorithm is determined by looking at the top 50 documents in each column of the matrices, and counting the number of documents with the same core functions. The core functions of a mechanical patent must be subjectively determined.

3.7 Color Coded Comparison for Large Scale Test

In order to compare the document similarity matrices outputs from LSA and LDA algorithms, column with same reference documents from LSA and LDA output matrices were individually compared. Each column of the matrices was divided into groups of 100, starting from most similar to least similar. One group each from LSA and LDA that is in the same ranking group are directly compared by how many number of the same documents are in that group. The group gets assigned with a color according to the percentage of similarity and each document in that group shows the same color in the document similarity color matrix. Each color with range of percentage match is shown in Fig. 4. Since the most number of matches in one group was under 35, each color has 3 percent range except for the last dark green color.

4 Results

4.1 Small Scale Test Case

For the small-scale test case, LSA and LDA algorithms were performed on full patent text, functional (verb-based) patent text, and surface (noun-based) patent text to

compare the results from LSA to LDA vice versa. Patents chosen for this test case were two pairs of functionally similar technologies, as show in Fig. 2, with Docs 1 and 2 relating to archery, and Docs 3 and 4 relating to power generation. By performing this very small-scale test case, we hoped to be able to dissect why LSA and LDA might behave differently in their categorization of patents. LDA algorithm was performed with three different number of topics, 2, 4, and 6. The result from LDA with 4 topics was most similar to the result of a LSA. There was no particular pattern or similarity between the results from LSA and LDA with topic number of 2 and 6, which indicates that the number of topics is a crucial parameter to the categorizations.

Doc #	Patent #	Patent Title
1	3942506	Demountable archery bow
2	3957027	Take-down and folding bow
3	7174710	Nelson flywheel power plant improvement
4	7363760	Thermodynamic free walking beam engine

Fig 2. Patents Included in Small Scale Test Case

The results from the small-scale test case are shown in Fig. 3A, 3B, and 3C. The first row of each table is named "reference document" in this paper, as all the subsequent documents are ordered below it depending on their similarity to that reference document. Full patent text comparison between two algorithms shows the best match with a minor discrepancy in the last two rows of the second column as shown in Fig. 3A. The order of Docs 3 and 4 is switched in the two methods, which, given that they are both power generation technologies, is not alarming. The functional patent text comparison in Fig. 3B shows the next best match. Although there is a discrepancy in every column of the matrix, it is interesting to note that the most similar document in each column is the least similar document in different methods, while the two other documents are in the same order. In the first column of the table, Doc 2 in LSA is the most similar text to Doc 1 while it is the least similar text in LDA's result (as shown by the red outlines in Fig. 3B). The same pattern applies to all columns. The surface patent text comparison in Fig. 3C shows no similarity or pattern between the results of the two methods. Although the LSA result in Fig. 3C is identical to the result of the full patent text LSA results in Fig. 3A, there are too many dissimilarities to compare to the LDA results in Fig. 3C.

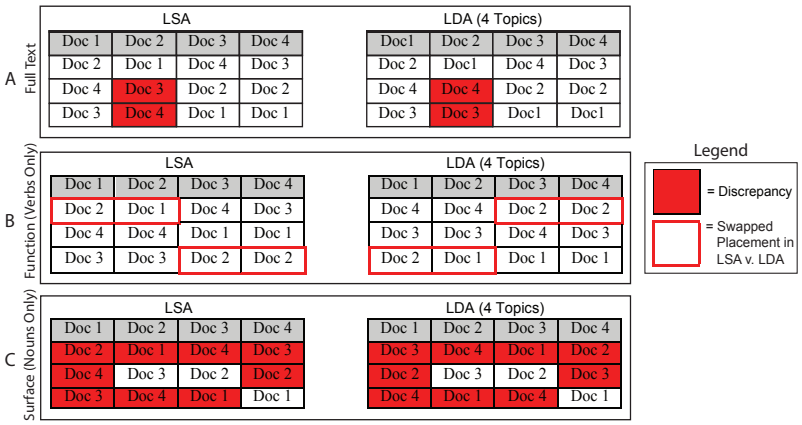

Fig. 3 Small Scale Test Case with Four Patents A) Using full text of patents, B) Using Only Functional Content of Patents (Verbs), C) Using Only Surface Content of Patents (Nouns)

4.2 Large Scale Test Case

In large-scale test case, 1000 random mechanical patent documents, including the 4 patent documents used earlier in small-scale test case, were selected to perform LSA and LDA algorithms. In the large-scale test, the LDA algorithm was performed with 150 topics. The results are shown in Fig. 4A, B and C. For all three types of text comparisons, the results show more green color that is above 30 percent match, in the first group of 100 as compared to the rest. The first group in each column is the group of top 100 ranked documents that each algorithm ranked to be more similar to the reference document than the rest. Also, for all cases, more similarity appeared in the first and the last groups, and less similarity appeared in the middle region.

For in-depth analysis, the results for the large-scale test were analyzed to determine whether they are consistent with those of the small-scale test. The reference document of the fifth column is Doc 1 from the small-scale test. The LSA results of the small and large-scale test agreed in terms of the order of the four selected documents. However, the LSA result in the large-scale test was not so effective in sorting the patents by the functional similarity. Doc 2, which is thought to be the most similar document to the reference document, was 231st similar document for large-scale test.

Instead, the LDA result for the functional patent text was better at sorting the functionally related documents in the first group. Same as the first column of LSA result, the reference document is Doc 1, which describes the functional component of a bow. The fifth column also includes two more bow related documents in the first group 100, specifically in ranked 20th and 23rd. However, this was only true for this column and no similar pattern was observed in the other three examples.

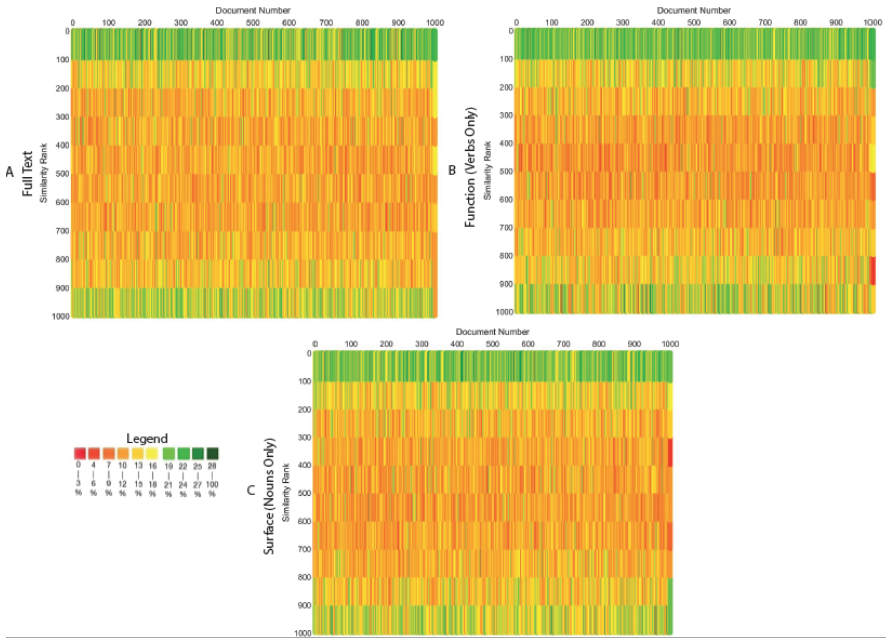

Fig. 4 Large-Scale Test Case with 1,000 Patents A) Using full text of patents, B) Using Only Functional Content of Patents (Verbs), C) Using Only Surface Content of Patents (Nouns)

5 Discussion

5.1 Comparison of LSA and LDA for Small Scale Test Results

LDA requires a defined number of topics as an input parameter. For the small-scale test, Fig. 3A and 3B indicate that LSA and LDA with 4 topics gives similar results for the full patent text and functional patent text respectively. The consistency in both cases suggests that using 4 topics is more appropriate than 2 or 6 topics as the input parameter when LDA is performed with 4 patent documents. It is still unknown whether LDA is effective in categorizing patent documents and how an appropriate number of topics can be determined. Therefore, the empirical finding in the small-scale test could be important in deciding whether LDA is appropriate for analyzing patents. Given that Fu et al. succeeded in applying LSA as effective method for categorizing patents at a small scale, the underlying hypothesis is that it could be more effective than LDA at large scale.

The functional text comparison in the small-scale test shows an interesting pattern in the order of the doc-doc similarity matrix. Although all columns in the matrix shown in Fig. 3B show discrepancies, the results resemble each other if the order of the most similar document and least similar document in a column are switched. The fact that the same rule applies to every column in Fig. 3B shows that there are similar documents in the middle region of the comparison matrix, while completely different documents in the regions farther away from the middle. When the documents are analyzed by function, LSA is more accurate than LDA in sorting them. For instance, Doc 2 should have matched functionally with Document 1 as they both describe the component of a bow. However, this is only true for results for LSA. Further research on small- and large-scale tests is required draw conclusions about these algorithms. Unlike the comparison of the functional text in the small-scale test, the surface text does not show any similarity in between the results of the two methods.

5.2 Comparison of LSA and LDA for Large Scale Test Results

For all cases, the similar documents are more apparent in the top and bottom groups of 100 patent documents. The fact that both methods agree on the most and least similar documents can help designers to look at the two groups for near-field or far-field analogies. Depending on the goal of the designer, they could analyze similar documents or dissimilar documents during design ideation. However, the conclusion that the groups are internally similar among the patents contained within them is tenuous, as the best percentage match is approximately 30% and the rest are mostly below 10%. This may be due to the lack of well-established methods to choose the number of LDA topics, or to the diverse nature of language and particularly of articulation of technologies within patents. Especially for the large-scale test, it is unrealistic to test different numbers of topics until the best result is achieved.

5.3 Future Directions

Future work includes examining the data more closely to understand why and how patents are categorized, and how that changes with scale. A method to determine the best number of topics for the LDA algorithm is much needed. Ultimately, the goal is to make a recommendation regarding the underlying method that should be used to analyze and categorize patents based on their textual content, but further work must be done prior to that recommendation.

By mining the textual content of the patent database at an increasing scale, we can start to access the wealth of knowledge contained in the historical records of invention and technology. The computational techniques compared in this paper provide a way to quantitatively evaluate similarity (and thus distance) between source analogies. In the future, when deployed at a large scale with interactive data visualization, these techniques will open up computationally supported analogy to a much larger audience.

References

1. H. Casakin, Goldschmidt, G., "Expertise and the Use of Visual Analogy: Implications for Design Education," Design Studies, vol. 20, pp. 13-175, 1999.
2. B. T. Christensen, Schunn, C. D., "The Relationship of Analogical Distance to Analogical Function and Preinventive Structure: The Case of Engineering Design," Mem. & Cog., vol. 35, pp. 29-38, 2007.
3. I. Tseng, J. Moss, J. Cagan, and K. Kotovsky, "The role of timing and analogical similarity in the stimulation of idea generation in design," Design Stud., vol. 29, pp. 203-221, 2008.
4. P. Leclercq and A. Heylighen, "5.8 Analogies per Hour," in Art. Int. in Des. '02, Springer Netherlands, 2002, pp. 285-303.
5. I. Kang, S. Na, J. Kim, J. Lee, "Cluster-based Patent Retrieval," Inf. Proc.&Mgmt, 43, 2007.
6. S.-Y. Yang and V.-W. Soo, "Extract conceptual graphs from plain texts in patent claims," Engineering Applications of Artificial Intelligence, vol. 25, pp. 874-887, 2012.
7. G. Ferraro and L. Wanner, "Towards the derivation of verbal content relations from patent claims using deep syntactic structures," Knowledge-Based Sys., vol. 24, p. 1233-44, 2011.
8. K. Fu, "Discovering and Exploring Structure in Design Databases and Its Role in Stimulating Design by Analogy," Ph.D. Dissertation, Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA, 2012.
9. K. Fu, J. Cagan, K. Kotovsky, and K. Wood, "Discovering Structure in Design Databases Through Function and Surface Based Mapping," Journal of Mech. Design, In Press, 2013.
10. K. Fu, J. Chan, J. Cagan, K. Kotovsky, C. Schunn, and K. Wood, "The Meaning of "Near" and "Far": The Impact of Structuring Design Databases and the Effect of Distance of Analogy on Design Output," ASME Journal of Mechanical Design, In Press, 2012.
11. M. Moehrle and A. Geritz, "Developing acquisition strategies based on patent maps," presented at the 13th IAMOT, Washington, D.C., 2004.
12. S. Koch, H. Bosch, M. Giereth, and T. Ertl, "Iterative Integration of Visual Insights during Patent Search and Analysis," presented at the IEEE Symposium on Visual Analytics Science and Technology, Atlantic City, NJ, USA, 2009.
13. S. Mukherjea, B. Bhuvan, and P. Kankar, "Information Retrieval and Knowledge Discovery Utilizing a BioMedical Patent Semantic Web," IEEE Trans. on Know. & Data Eng., vol. 17, pp. 1099-1110, 2005.
14. S. Chakrabarti, B. Dom, R. Agrawal, and P. Raghavan, "Scalable Feature Selection, Classification and Signature Generation for Organizing Large Text Databases into Hierarchical Topic Taxonomies," The VLDB Journal vol. 7, pp. 163-178, 1998.
15. G. S. Altshuller and R. B. Shapiro, "On the psychology of inventive creation (in Russian)," The Psychological Issues, vol. 6, pp. 37-39, 1956.
16. R. Duran-Novoa, N. Leon-Rovira, H. Aguayo-Tellez, and D. Said, "Inventive Problem Solving Based on Dialectical Negation, Using Evolutionary Algorithms and TRIZ Heuristics," Computers in Industry, vol. 62, pp. 437-445, 2011.
17. N. V. Hernandez, L. C. Schmidt, and G. E. Okudan, "Systematic Ideation Effectiveness Study of TRIZ," presented at the ASME IDETC/CIE, Chicago, IL, USA, 2012.
18. N. V. Hernandez, L. C. Schmidt, and G. E. Okudan, "Experimental Assessment of TRIZ Effectiveness in Idea Generation," presented at ASEE AC, San Antonio, TX, USA, 2012.
19. V. Krasnoslobodtsev and R. Langevin, "TRIZ Application in Development of Climbing

Robots," presented at the First TRIZ Symposium, Japan, 2005.
20. Y. Liang, R. Tan, and J. Ma, "Patent Analysis with Text Mining for TRIZ," presented at the IEEE ICMIT, Bangkok, Thailand, 2008.
21. T. Nakagawa, "Creative Problem-Solving Methodologies TRIZ/USIT: Overview of my 14 Years in Research, Education, and Promotion," The Bulletin of the Cultural and Natural Sciences in Osaka Gakuin University, vol. 64, March 2012.
22. A. A. Nix, B. Sherret, and R. B. Stone, "A Function Based Approach to TRIZ," presented at the ASME IDETC/CIE, Washington, D.C., USA, 2011.
23. R. Zhang, J. Cha, and Y. Lu, "A Conceptual Design Model Using Axiomatic Design, Functional Basis and TRIZ," presented at the Proceedings of the 2007 IEEE IEEM, 2007.
24. R. Houssin and A. Coulibaly, "An Approach to Solve Contradiction Problems for Safety Integration in Innovative Design Process," Comp. in Industry, vol. 62, p. 398-406, 2011.
25. D. P. Moreno, M. C. Yang, A. Hernandez, and K. L. Wood, "Creativity in Transactional Design Problems: Non-Intuitive Findings of an Expert Study Using Scamper," presented at the Int. Design Conference, Human Behav. and Des., Dubrovnik, Croatia, 2014.
26. A. Dong, W., H. A., Agogino, A. M., "A Document Analysis Method for Characterizing Design Team Performance," Journal of Mechanical Design vol. 31, pp. 011010-8, 2004.
27. D. M. Blei, "Probabilistic Topic Models," Comm. of the ACM, vol. 55, p. 77-84, 2012.
28. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," J. Mach. Learn. Res., vol. 3, pp. 993-1022, 2003.
29. J. McAuliffe & D. Blei, "Supervised topic models," Adv. Neur. Inf. Proc. Sys., 121-8, 2008.
30. J. Jagarlamudi, H. Daume, and R. Udupa, "Incorporating Lexical Priors into Topic Models," presented at the EACL '12, Avignon, France, 2012.
31. D. M. Blei, T. L. Griffiths, and M. I. Jordan, "The Nested Chinese Restaurant Process and Bayesian Nonparametric Inference of Topic Hierarchies," J. ACM, vol. 57, 2010.
32. D. Newman, A. Asuncion, P. Smyth, and M. Welling, "Distributed Algorithms for Topic Models," J. Mach. Learn. Res., vol. 10, pp. 1801-1828, 2009.
33. L. El Ghaoui, V. Pham, G.-C. Li, V.-A. Duong, A. Srivastava, and K. Bhaduri, "Understanding Large Text Corpora via Sparse Machine Learning," Stat. Anal. & Data Mining, vol. 6, pp. 221-242, 2013.
34. R. Tibshirani, "Regression shrinkage and selection via the LASSO," J R Stat Soc Ser B, vol. 58, pp. 267–288, 1996.
35. H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," J Comp. Graph. Stat., vol. 15, pp. 265–286, 2006.
36. D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," Nature, vol. 401, pp. 788-791, 1999.
37. J. Choo, C. Lee, D. K. Reddy, and H. Park, "UTOPIAN: User-Driven Topic Modeling BasedonInteractive Nonnegative Matrix Factorization," IEEE Trans. Vis. and Comp. Graph., vol. 19, pp. 1992-2001, 2013.
38. S. Deerwester, S. T. Dumais, G. W. Furnas, and T. K. Landauer, "Indexing by Latent Semantic Analysis," J. of the Amer. Soc. for Inf. Sci. vol. 41, pp. 391–407, 1990.
39. P. W. Foltz, W. Kintsch, and T. K. Landauer, "The Measurement of Textual Coherence with Latent Semantic Analysis," Discourse Processes vol. 25, pp. 285-307, 1998.
40. T. Landauer, P. W. Foltz, and D. Laham, "An Introduction to Latent Semantic Analysis," Discourse Processes vol. 25, pp. 259-284, 1998.
41. T. Landauer, Dumais, S.,"A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge," Psych. Rev., 211-40, 1997.