# HYBRID CONSTRUCTION DOCUMENT CLASSIFICATION MODEL USING MACHINE LEARNING (ML) AND TEXT SEGMENTATION METHODOLOGY

Tarek Mahfouz, PhD / Assistant Professor, tmahfouz@bsu.edu
*Department of Technology, Ball State University (BSU), Muncie, Indiana, USA*

## ABSTRACT

The dynamic nature of the construction industry yields enormous documents that are generated in an unstructured format like technical specifications, meeting minutes, daily reports, claims, and construction litigation cases. With the increasing level of sophistication and growing speed of the industry, the efficient use of these documents became inevitably needed. This paper proposes a hybrid automated construction document classifier utilizing Machine Learning (ML) and Text Segmentation. The current research builds on previous study performed by the author that utilized Support Vector Machines (SVM) for automating construction document classification. To that end, the current paper presents the enhanced results of performing a pre-processing step of text segmentation of construction documents. Lengthy construction documents like claims typically address different topics or different aspects of the same topic within one document. This issue decreases the accuracy of the SVM classifiers. Consequently, the pre-processing step aims at defining texts that are related to different topic within the same document. The adopted research methodology (1) gathered and utilized a corpus of 500 Different Site Conditions (DSC) cases from the Federal Court of New York; (2) developed a tokenizing and parsing algorithm for the used documents through C++; (3) implemented text segmentation adopted from Hearst's TextTiling algorithm; (4) developed SVM automated classification models; and (5) compared the outputs to results attained in previous works. The outcomes of this research are expected to enhance automated decision support tools developed for the construction industry.

**Keywords:** Document Classification, Text Segmentation, Machine Learning (ML), TextTiling, Support Vector Machines (SVM).

## 1. INTRODUCTION

The construction industry is a major contributor to the development of nations' economies. The US Census data showed that the total construction spending in 2007 was about $ 14 trillion (US Census 2010). In the US, 4.3% of the gross domestic product (GDP) is derived from the construction industry (US Bureau of Economic Analysis 2010). Worldwide, this industry is undergoing fast advancements in construction methods and strategies, technologies, machinery, and materials, which increase the sophistication and complexity of construction project. These two characteristics of the industry created strong need for increasing the collaboration between diversified parties that may not exist in the same geographic region (Caldas et al. 2002). Such aspect yields the production of massive amount of documents in diversified formats.

Over the last two decades, researchers have extensively utilized artificial intelligence (AI) techniques for managing the knowledge contained in these documents. Researches cover a wide spectrum including enhancing information models, document integration, inter-organizational systems, and expert systems (Labidi 1997). Consequently, automated and semi-automated tools were developed to enable the utilization of textual data expressed in natural language through text mining, document clustering, controlled vocabularies, and web-based models (Ioannou and Liu 1993, Yang et. al 1998, Caldas et. al 2002, Caldas and Soibelman 2003, NG et al. 2006, Mahfouz et al. 2010, and Mahfouz and Kandil 2010a). Although those studies resulted in significant contribution, they did not address one of the major problems related to these documents. Construction documents like technical

specifications, meeting minutes, daily reports, claims, and construction litigation cases are lengthy and address several topics or different aspects of the same topic. This aspect affects the accuracy of the previously mentioned tools.

As a result, this paper represents a continuation step in a line of research aiming at developing a comprehensive and robust methodology for automated document classification in the construction industry. The current research proposes a hybrid automated document classifier through text segmentation and Machine Learning (ML). Construction documents, which are represented in natural language, can be characterized as a sequence of sub-topical discussion of related or unrelated concepts in a coherent manner. Consequently, text segmentation can be defined as the process of defining changes and shifts in topics or sub-topics within a document. The aim of this paper is to investigate the suitability of using text segmentation methodologies, adopted from TextTiling that was developed by Marti Hearst, to define topic boundaries in construction documents. To that end, the proposed research methodology (1) gathered and utilized a corpus of 500 Different Site Conditions (DSC) cases from the Federal Court of New York and 30 claims compiled from different projects around the world; (2) developed a tokenizing algorithm in C++ to parse the used cases; (3) implemented text segmentation adopted from Hearst's TextTiling algorithm; (4) developed SVM automated classification models; and (5) compared the outputs to results attained in previous works. Within the current research, topics are considered to be legal construction dispute existing in a case and boundaries are identified at shifts between them based on existing text. The outcomes of this research are expected to enhance automated document classification and decision support tools developed for the construction industry. The rest of the body of this paper describes the followings.

- Literature Review;
- Methodology;
- Results and discussion; and
- Conclusion

## 2. LITERATURE REVIEW

Most construction information models work with structured data like CAD models and scheduling databases. However, a major portion of crucial construction knowledge is stored in semi-structured or unstructured format (Caldas et al. 2002). Examples of these documents include, but not limited to, contract documents, change orders, and meeting minutes. These documents are normally stored as text files. Facilitating the use of these documents through integrated methods has become a necessity. A number of research studies have tackled this drawback. Ioannou and Liu (1993) proposed a computerized database for classifying, documenting, storing and retrieving documents on rising construction technologies. Kosovac et al. (2000) investigated the use of controlled vocabularies for the representation of unstructured data. In 2000, Wood provided a method for the hierarchical structuring of concepts extracted from textual design documents. Scherer and Reul (2002) utilized text mining techniques to classify structured project documents. Caldas et al. (2002) and Caldas and Soibelman (2003) used information retrieval via text mining techniques to facilitate information management and permit knowledge discovery through automated categorization of various construction documents according to their associated project component. Xie et al. (2003) provided an integrated model for retrieving construction project documents to facilitate decision-making, logical judgment, and control for project managers. Caldas et al. (2005) proposed a methodology for incorporating construction project documents into project management information systems using semi-automated support integration to improve overall project control. To facilitate and improve design reuse, Demian and Fruchter (2005) investigated the use of different text analysis methodologies to highlight and quantify potential similarities among objects from an archive of building models. Ng et al. (2006) implemented Knowledge Discovery in Databases (KDD) through a text mining algorithm to define the relationships between type and location of different university facilities, and the nature of the required maintenance reported in the Facility Condition Assessment database. In recent researches, Mahfouz and Kandil 2010a, and Mahfouz et al. 2010 developed automated construction document classifiers using Support Vector Machines (SVM) and Latent Semantic Analysis (LSA). Within these researches, it was found that the accuracy of the developed models is affected by the fact that documents like meeting minutes and construction litigations address more than one topic within one document.

In 2011, Mahfouz proposed a text segmentation methodology to define shifts between different topics included in a textual document. According to Yarri 1997, text segmentation methodologies are classified into one of two categories. The first is lexical cohesion, which is based on similarity of vocabulary. Text passages with closely related and similar words are more likely to relate to similar topic (Reynar, 1994). Researches in that realm included word repetition methodologies, context vector analysis, word frequency models and semantic similarities (Reynar, 1994, Hearst, 1994, Reynar, 1999, and Morris and Hirst, 1991 respectively). The second methodology relates to hybrid methods that combines lexical cohesion with indicators of topic shifts. Hybrid systems utilized probabilistic models (Reynar, 1998) and machine learning models like decision trees (Litman and Passonneau, 1995). All of the previously mentioned researches attained improvements in this field. However, they were tested on broadcast news, science text collections like Stargezer, and spoken dialogues. Consequently, the current paper builds on previous researches performed by the author by developing a hybrid classifier that utilizes text segmentation and support vector machines.

## 3. METHODOLOGY

The following sections of the paper describe the different steps of developing, implementing, and validating the models. To that end, the adopted research methodology constitutes of three folds. The first relates to the development of Support vector Machine (SVM) document classifier. The aim of the SVM model is to automatically identify the topics pertinent to a document. The second fold is concerned with implementing text segmentation algorithm (Mahfouz 2011) that identifies topic shifts within a document. This step allows for breaking down a long document into a set of smaller documents, based on the defined shifts. Each of these documents can then be classified using the SVM classifier. The third fold relates to comparing the outputs of the two previous fold to quantify the benefits gained, if any, from the hybrid classifier. To achieve the above-mentioned folds, five (5) main stages, as illustrated in Figure 1, are implemented. These stages are defined as (1) Corpus Development; (2) Tokenizing; (3) Text Segmentation Algorithm Implementation; (4) Feature Space Development; and (5) Model Design, Implementation, and evaluation.

### 1.1 Corpus Development

The current research task is concerned with lengthy unstructured construction documents represented in natural language. To that end, a set of 500 Differing Site Conditions (DSC) cases were utilized for the model development and testing. These cases were gathered from the Federal Court in New York due to the abundant amount of cases. They were compiled using LexisNexis, a web legal retrieval system.

### 1.2 Tokenizing

Although each document implicitly includes the required knowledge to perform the segmentation task, in the form of words and phrases, it also includes textual representations that are not related to the topic and can deteriorate the performance of the model. As a result, an initial preparation step is needed. The processing step of the tested documents will include data cleaning, data integration, and data reduction (Ng. et al. 2006, Mahfouz et al. 2010b, and Mahfouz 2011). For more illustrations, textual representation of documents might include frequent words that carry no meaning, misspelled words, outliers, noise, and inconsistent data. While data processing is performed on each textual case representation separately, data integration is performed over the entire dataset. In this step, the entire processed dataset is stored in a coherent manner that facilitates their use for further analysis. While the integrated data might be very large, data reduction can decrease the data size by aggregating and eliminating redundant features. An algorithm in C++ was developed to perform the aforementioned steps. The basic principle of the developed program is to break down each document within the collected data set into sentences and represent each sentence as a vector of word frequencies.

The parsing and extraction steps implemented by the algorithm are as follows: (1) Extract all words in a document; (2) Eliminate non-content-bearing words, also known as stopwords (Scherer and Reul 2000); (3) Reduce each word to its “root” or “stem” eliminating plurals, tenses, prefixes, and suffixes; (4) For each document, count the number of occurrences of each word; and (5) allocates the

occurrence number of each word to a sequential sentence number. The output of the implementation of this algorithm is w unique words remain in d unique sentences; a unique identifier is assigned between 1 and w to each remaining word, and a unique identifier between 1 and d to each sentence resulting in a term-frequency (tf) matrix.

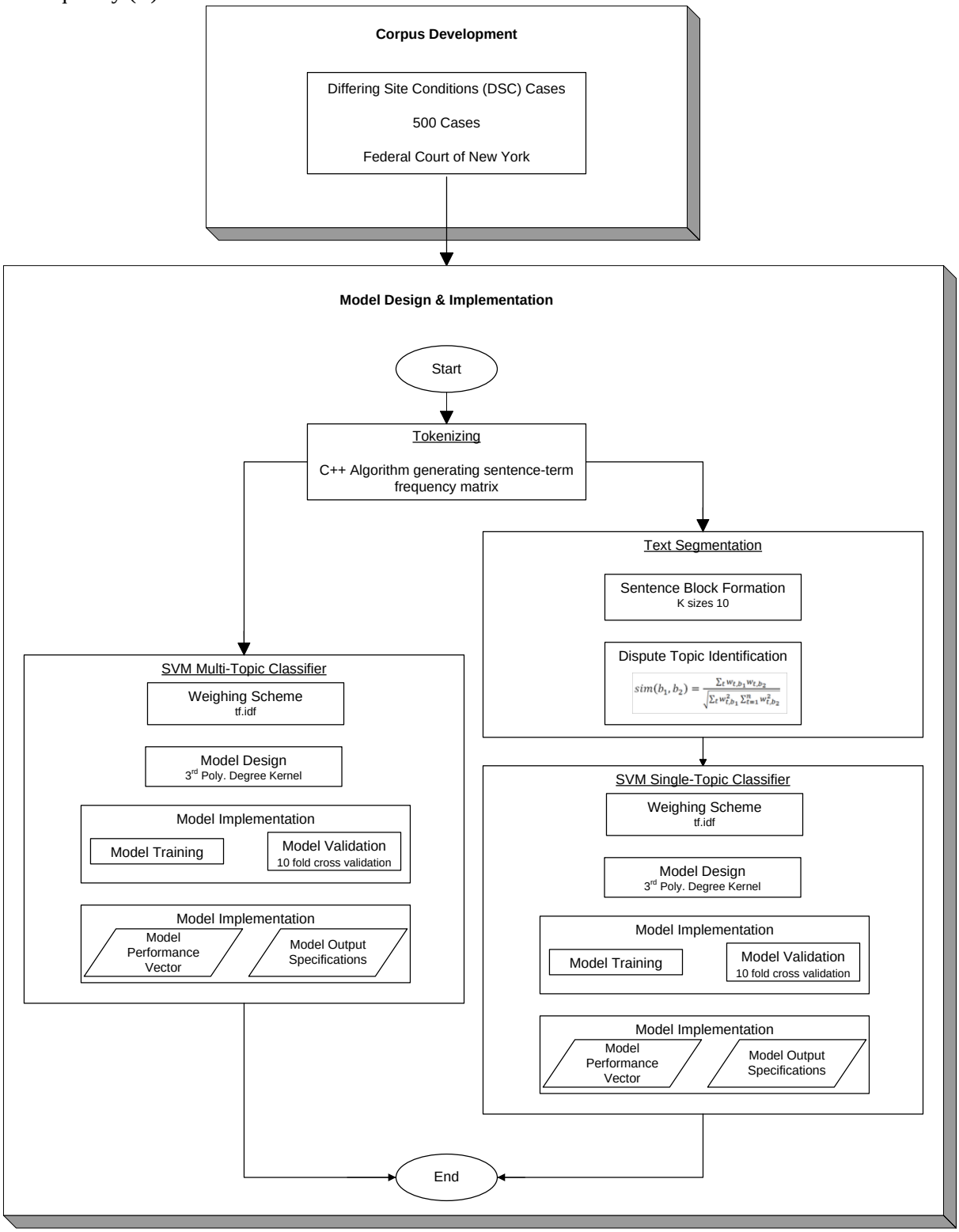


Figure 1: Research methodology.

### 1.3 Text Segmentation Implementation

The developed algorithm for the current sub-task is adopted from TextTiling algorithm developed by Marti Hearst, for its proven superiority in the literature review (Reynar, 1994, and Hearst 1994). The implementation steps of the algorithm are two folds. The first relates to calculating similarity measures between the different tokenized sentences. The algorithm starts by breaking the tokenized sentences into blocks of size k. Each block includes sequential sentences that appeared in close proximity in the original text. It is assumed that these blocks will include sentences related to the same specific topic. In addition, the blocks are formulated to decrease the computation of the algorithm. The size k is pertinent to each domain. As a result, it should vary from one application to the other to attain better segmentation accuracy. Hearst (1994) illustrate that k should represent the average number of sentences with respect to the number of paragraphs included in the document. "In practice, a value of k=6 works well for many texts" (Hearst 1994). The adopted k size for the current sub-task is 10 due to its enhanced performance proven in the author's previous researches. For further details on the k size evaluations and performance, please refer to Mahfouz 2011. Following the formulation of the blocks, cosine similarity measures are calculated between each two successive blocks in accordance with Equation 1. If the similarity measure between two blocks is high, they are assumed to be related to the same specific topic.

$$sim(b_1, b_2) = \frac{\sum_t w_{t,b_1} w_{t,b_2}}{\sqrt{\sum_t w_{t,b_1}^2 \sum_{t=1}^{n} w_{t,b_2}^2}} \quad (1)$$

Where t is the number of all tokenized terms, $b_1$ is first block, $b_2$ is second block, $wt,b_1$ is the weight assigned to term t In the first block, and $wt,b_2$ is the weight assigned to term t In the second block. Term weights under the current research are considered as the term frequency attained through the tokenization step.

The second fold is related to defining topic boundaries within the texted document. The algorithm iterates through all calculated similarity measures in a sequential order performing the followings.

1. A sequential number between 1 and n (number of blocks – 1) is assigned to each calculated similarity measure.
2. For i representing a similarity measure calculated between block i and block i+1, the value is compared to the two successive calculated measure to its right i+1 and left i-1.
3. If the measures are increasing, the blocks are assumed to be related to the same topic.
4. The algorithm keeps iterating through the calculated measures until a drop in the value is noticed. A topic boundary is assigned between blocks if the dropped measure is less than 75%.
5. The following steps are repeated until all calculated similarity measures are checked.

### 1.4 Feature Space Development

A mere representation of significant words in the form of (tf) is not sufficient to accurately extract the required knowledge from the case corpus. For example, a word like contract might exist in all processed documents in high (tf). However, a decision must be made about whether this word would help define a topic of concern or not. Consequently, an appropriate weighting mechanism must be implemented to create a representative matrix of these documents within the entire dataset. Literature in the field of ML and text mining illustrated the effectiveness of alternate term weighting schemes like logarithmic term frequency (ltf) (Equation 2), augmented weighted term frequency (atf) (Equation 3), and term frequency inverse document frequency (tf.idf) (Equation 4).

$$ltf_{i,d} = 1 + \log(tf_{i,d}) ; tf_{i,d} > 0 \quad (1)$$

$$atf_{i,d} = 0.5 + \frac{0.5 \times tf_{i,d}}{\max_t(tf_{i,d})} \quad (3)$$

$$tf.idf_{i,d} = (1 + \log(tf_{i,d})) \times \log\left(\frac{N}{df_i}\right) \text{ if } tf_{i,d} > 0 \quad (4)$$

The four above mentioned weighting schemes were utilized in earlier research tasks performed by the author. These researches illustrated the superiority of tf.idf weighing scheme over the others (Mahfouz and Kandil 2010a, Mahfouz and Kandil 2010b, and Mahfouz et al. 2010). As a result, tf.idf

weighing was adopted for the current research. The developed algorithm implements the required calculations as per Equation 4 to formulate the final matrix of the set of documents (Figure 2).

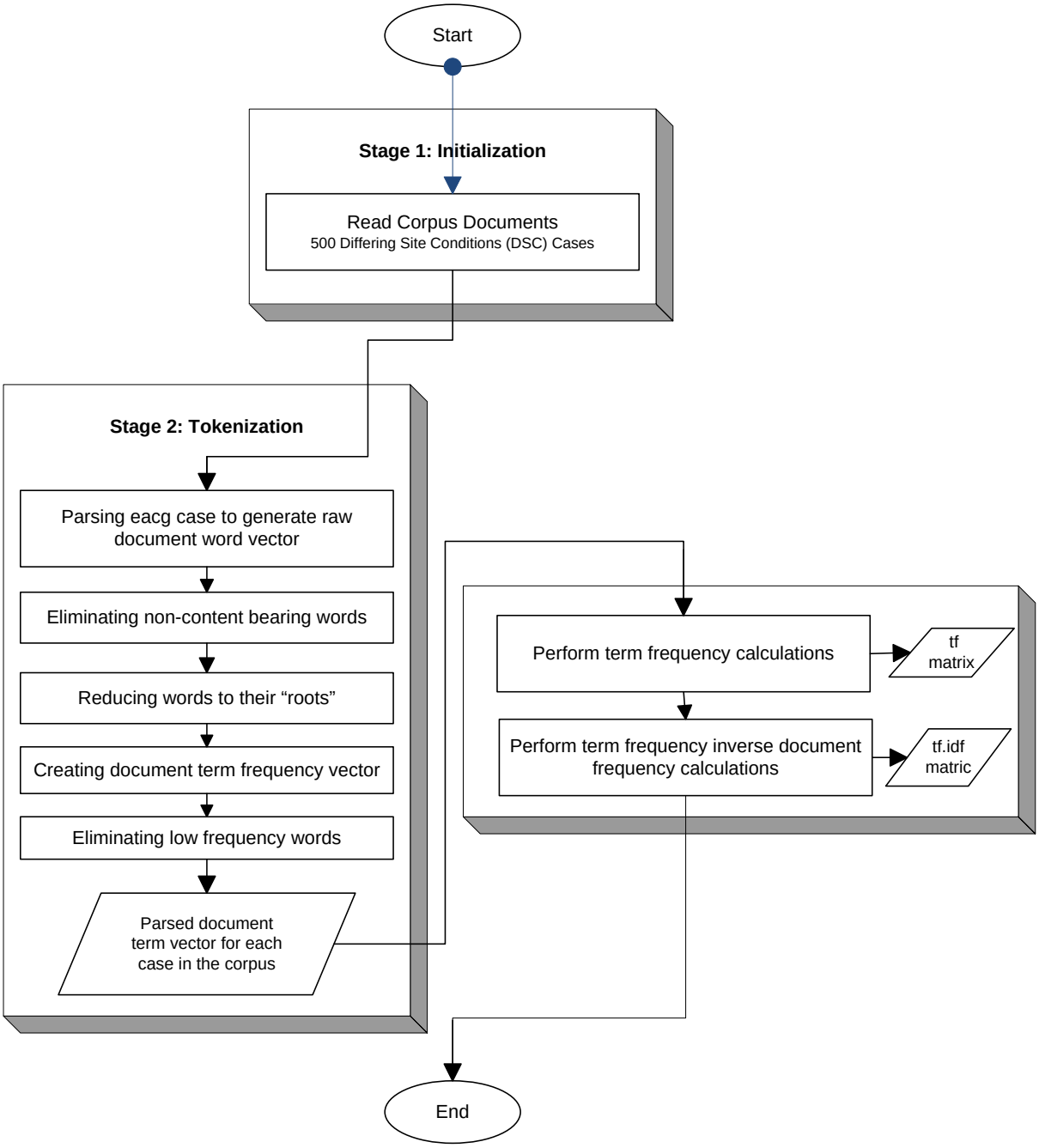


Figure 2: Weighing scheme implementation.

### 1.5 Model Design, Implementation, and evaluation

The following is a descriptive background of the Support Vector Machines concept. SVM classification aims to find a surface that best separates a set of training data points into classes in a high dimensional space. In the current research, it aims at defining the construction subject pertinent to each of the training documents based on the word representation in its content. In its simplest linear form, a SVM finds a hyper-plane that separates a set of positive examples (documents belonging to a legal construction dispute) from the set of negative examples (documents not belonging to the same legal construction dispute) with a maximum margin. Binary classification is performed by using a real

valued hypothesis function, Equation 5, where input x (document) is assigned to the positive class (legal construction dispute) if $f(x) \geq 0$; otherwise, it is assigned to the negative class.

$$Y = \langle w.x \rangle + b \quad (5)$$

For a binary linear separation problem a hyper-plane is assigned to be $f(x) = 0$. With respect to Equation 5, the vector w (weight vector) and b (functional bias) are the parameters that control the function of the separation hyper-plan (refer to Figure 3). In addition, x is the feature vector which may have different representations based on the nature of problem. Within the context of the current research, the input feature space X constitutes of the training DSC cases that are defined by the vectors x and o in Figure 3.

In the development of the proposed SVM models a problem emerges if the data are not linearly separable. Assigning DSC cases to specific legal construction dispute cannot be represented by a simple linear combination of its content words. Consequently, a more sophisticated higher dimension space is needed for the representation of the current problem in order for it to be linearly separable. As the literature in this field suggests, Kernel representations provides a solution to this problem by transforming the data into a higher dimensional feature space to enhance the computational power of linear machine learning (Mangasarian and Musicant 1999). As shown in Equation 5, the representation of a case in the feature space for linear machine learning is achieved as a dot product of the tf.idf vector (x) and the weight vector (w). By introducing the appropriate Kernel function, cases are mapped to a higher feature space (Equation 6 and Figure 3) transforming the prediction problem from a linearly inseparable to a linearly separable one. In this manner, the input space X is mapped into a new higher feature space $F=\{\emptyset(x)|x$ where $\emptyset$ is the kernel transformation function.

$$x=(X_1, \ldots., X_n) \rightarrow \Phi(X)=(\Phi_1 X_1, \ldots., \Phi_n X_n) \quad (6)$$

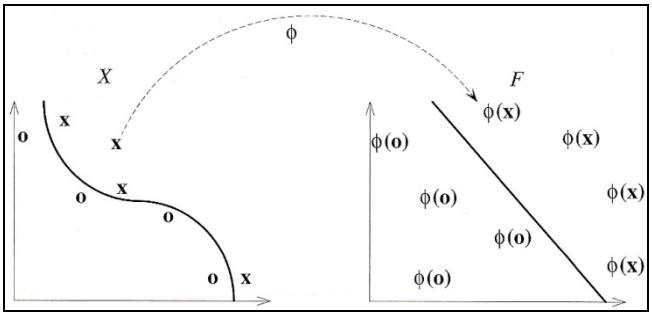


Figure 3: SVM Kernel Transformation and Classification

Previous researches performed by the author illustrate the strength of third ($3^{rd}$) polynomial degree kernel transformation in problems with similar nature (Mahfouz and Kandil 2010a, Mahfouz et al. 2010, and Mahfouz 2009). Consequently, the proposed research methodology developed and compared the outputs of third ($3^{rd}$) polynomial degree SVM model while implementing and not implementing text segmentation. Validation of the best-developed model was based on prediction accuracy. Since the analysis is aiming at automatically classifying each DSC case to a specific legal construction dispute, the following approach was adopted.

- The initial SVM classifier, while not implementing text segmentation, was developed as a multiple classifier. For more elaboration, each case was manually tagged with the existing legal construction disputes within its text. In the training stage, the SVM classifier learns the latent relation between the existing weighted word matrix and the tagged disputes. The learning process is performed on a 10 fold cross validation mechanism. In other words, the set of training cases is divided into 10% and 90% portions in each fold. The model is trained on the 90% and tested on the other 10% cases. The process is done in an iterative manner until the model is trained and tested over the whole set of cases. The prediction accuracy of the model is developed as the average accuracy attained among all folds and the Kappa as the measure of agreement between all folds.
- The second SVM classifier, while implementing text segmentation, was developed as a single topic classifier. After defining the topic shifts identified by text segmentation algorithm, each

case is divided into a set of text files based on these shifts. Each case is manually tagged with its appropriate dispute topic, after which the SVM classifier is trained and tested in a similar manner described above. The accuracy of the classifier is calculated as the number of cases accurately predicted to be concerned with all disputes in the original case to the total number of tested cases.

## 4. RESULTS AND DISCUSSION

The outcomes of the implementation of the aforementioned methodology are illustrated in Tables 1 and Figures 4 and 5. In earlier researches related to text segmentation (Mahfouz 2011), it was noticed that the accuracy of the model in predicting appropriate shifts drops as the number of disputes increases. Consequently, clustered analysis was performed. The utilized data set was broken down into three (3) subsets. The first was concerned with cases including up to two (2) dispute topics. The second included cases with more than two (2) dispute topics but less than five (5). The third was designated to cases with five (5) or more dispute topics. Figure 4 below defines the percentage distribution of each subgroup.

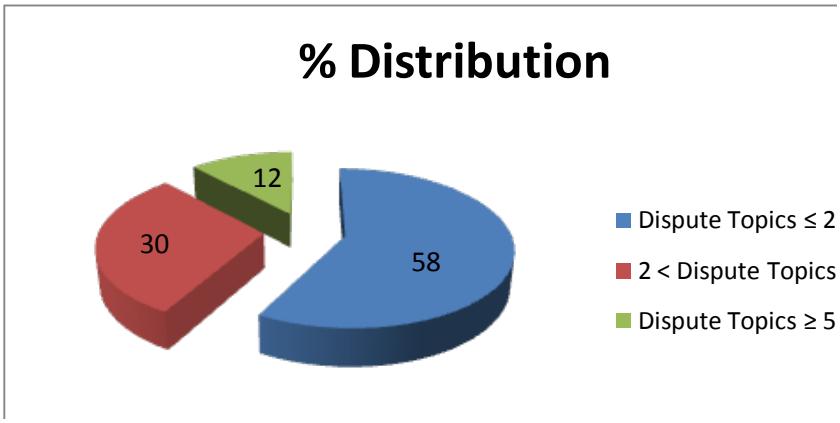


Figure 4: Percentage distribution of DSC cases

Table 1 illustrates the average attained prediction accuracy of the two classifiers within each category of cases.

Table 1: Average prediction accuracy results

| | Average Accuracy (%) | | |
|---|---|---|---|
| **Classifier Type** | **Dispute Topics ≤ 2** | **2 < Dispute Topics > 5** | **Dispute Topics ≥ 5** |
| Multiple Classifier | 83% | 79% | 78% |
| Hybrid Classifier | 80% | 82% | 83.5% |

Further examination of the results illustrates that followings.

- For DSC cases with up to two (2) dispute topics, a decrease of three percent (3%) in the prediction accuracy was achieved due to the use of the hybrid system. This could be attributed to two aspects. First, the majority of cases (40%) within that category included one (1) dispute. This aspect enhances the prediction due to the active learning feature of SVM. It becomes easier to define the hyper-plane between positive and negative cases. The second is related to the nature of the text segmentation algorithm. It forces a separation between topic shifts based on cosine similarity between (tf) vectors. Consequently, false shifts might be detected between text passages discussing the same dispute but in different word formats. This is especially applicable to litigations. Humans usually adopt a specific writing styles for introductions that might differ from the body or conclusions. As a result, the choice of words vary between these parts resulting in false detection of dispute shifts.
- For DSC cases with more than two (2) dispute topics, an average increase of four and a quarter percent (4.25%) in the prediction accuracy was attained. This could be attributed to

lengthy nature of the cases. The analysis utilized a set of 500 cases, which generated a large number of features reaching to more than 2500 words. The fact that the number of cases is less than twice the number of features deteriorates the active learning feature of SVM. "Active learning forces the SVM algorithm to restrict learning to the most informative training examples and not to attempt to use the entire body of data" (Oracle 2009). On the other hand, the separation of lengthy cases into shorter ones based on topic shifts increased the number of cases analysed and decreased the number of features resulting in better performance in the SVM classifier.

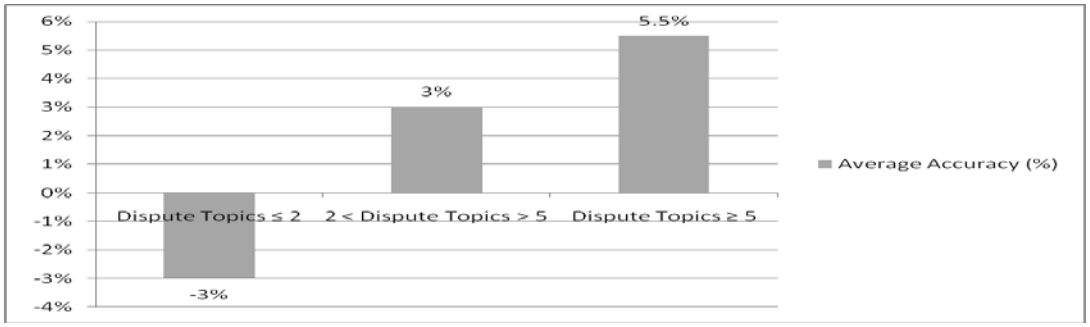


Figure 5: Percentage improvement due to the use of Hybrid classifier

## 5. CONCLUSION

The paper proposed a methodology for automating construction document classification through hybrid classifier. The analysis utilized a set of 500 DSC cases gathered from the Federal Court of New York filed between 1912 and 2009. To that end, the adopted research methodology developed and compared the outputs of two third ($3^{rd}$) polynomial degree kernel SVM classifiers. The first was a multi-topic classifier. The second was a hybrid classifier that made use of a text segmentation algorithm and a single topic classifier. The outcomes of this research task highlight the followings.

- The task in hand is a complex research task due to the nature of the analysed documents. Litigation cases are represented in natural language using legal terminologies, which makes it difficult for the algorithm to separate topics based on similarity measures.
- As the number of dispute topics increases, the performance of the hybrid model increases too.
- The highest prediction accuracy of 83.5% was achieved using the hybrid classifier.

This research task represents a continuation step in a line of research aiming at developing a comprehensive and accurate construction document classification. It is conjectured that this research line will help in relieving the negatives associated with the lengthy analysis of documents.

## ACKNOWLEDGMENTS

The author would like to acknowledge Dr. Marti Hearst, Professor in the UC Berkeley School of Information with an affiliate position in the Computer Science Division, for providing access to her earlier research work in the realm.

## REFERENCES

Caldas, C. H., and Soibelman, L. (2003) "Automating hierarchical document classification for construction management information systems." *Automation in Construction*, 12(4), 395-406.

Caldas, C. H., Soibelman, L., and Gasser, L. (2005) "Methodology for the integration of project documents in model-based information systems." *Journal of Computing in Civil Engineering*, 19(1), 25-33.

Caldas, C. H., Soibelman, L., and Han, J. (2002) "Automated classification of construction project documents." *Journal of Computing in Civil Engineering*, 16(4), 234-243.

Demian, P., and Fruchter, R. (2005) "Measuring relevance in support of design reuse from archives of building product models." *Journal of Computing in Civil Engineering*, 19(2), 119-136.

Hearst, M. (1994) “Multi-paragraph segmentation of expository text.” *Proceedings of the ACL' 94.* Las Crees, NM.

Ioannou, P. G., and Liu, L. Y. (1993) “Advanced construction technology system—ACTS.” *Journal of Construction Engineering and Management*, 119(2), 288-306.

Kosovac, B., Froese, T., and Vanier, D. (2000) “Integrating heterogeneous data representations in model-based AEC/FM systems.” *Proceedings CIT 2000*, Reykjavik, Iceland, 1, 556-566.

Labidi, S. (1997) “Managing multi-expertise design of effective cooperative knowledge-based system.” *Proceedings of 1997 IEEE Knowledge & Data Engineering Exchange Workshop*, IEEE, Piscataway, NJ, 10-18.

Litman, D. and Passonneau, R. (1995) “Combining multiple knowledge sources for discourse segmentation.” In *Proceedings of the 33rd Annual Meeting o/the ACL.*

Mahfouz, T. (2011) “Text Segmentation Methodology for Unstructured Construction Documents.” *Proceedings of The 2011 CSCE Conference, The 3rd International/9th Construction Specialty Conference*, June 2011; Ottawa, Canada.

Mahfouz, T. (2009). Construction legal support for differing site conditions (DSC) through statistical modeling and machine learning (ML). Ph.D. thesis, Civil, Construction, and Environmental Engineering (CCEE), Iowa State University.

Mahfouz, T. and Kandil, A. (2010a) “Unstructured Construction Document Classification Model through Latent Semantic Analysis (LSA).” *Proceedings of The CIB W78 2010 Conference, The 27th International Conference – Application of IT in the AEC Industry*, November 2010; Cairo, Egypt.

Mahfouz, Tarek and Kandil, Amr (2010b) “Automated Outcome Prediction Model for Differing Site Conditions through Support Vector Machines” In the *Proceedings of the International Conference on Computing in Civil and Building Engineering (ICCCBE-2010)*, Nottingham, UK.

Mahfouz, T., James, J., and Kandil, A. (2010) “A Machine Learning Approach For Automated Document Classification: A Comparison between SVM and LSA Performances.” *The International Journal of Engineering Research and Innovation (IJERI)*, fall 2010.

Morris, J. and Hirst, G. (1991) “Lexical cohesion computed by thesaural relations as an indicator of the structure of text.” *Computational Linguistic*s, (17):21-48.

Ng, H. S., Toukourou, A., and Soibelman, L. (2006) “Knowledge discovery in a facility condition assessment database using text clustering.” *Journal of Computing in Civil Engineering*, 12(1), 50-59.

Oracle. < http://www.oracle.com/index.html> (Accessed 2009)

Reynar, J. (1994) “An automatic method of finding topic boundaries.” *Proceedings of ACL'94 (Student session).*

Reynar, J. (1998). *Topic segmentation: Algorithms and applications.* Ph.D. thesis, Computer and Information Science, University of Pennsylvania.

Reynar, J. (1999) “Statistical models for topic segmentation.” In *Proceedings of the 37th Annual Meeting of the ACL*, pages 357-364. 20-26th June, Maryland, USA.

Scherer, R. J., and Reul, S. (2000) “Retrieval of project knowledge from heterogeneous AEC documents.” *Proceedings of the Eight International Conference on Computer in Civil and Building Engineering*, Palo Alto, Calif., 812-819.

U.S. Bureau of Economics < http://www.bea.gov/> (Accessed 2011).

US Census Bureau < http://www.census.gov/const/www/c30index.html> (Accessed 2010).

Wood, W. H. (2000). “The development of modes in textual design data.” *Proc., Eight International Conference on Computer in Civil and Building Engineering*, Palo Alto, Calif., 882-889.

Xie, H., Isaa, R. A., and O’Brien W. (2003) “User model and configurable visitor for construction project information retrieval.” *4th Joint International Symposium on Information Technology in Civil Engineering*, ASCE, Nashville, Tennessee, 47.

Yaari, Y. (1997) “Segmentation of expository texts by hierarchical agglomerative clustering.” *Proceedings of RANLP'97.* Bulgaria.

Yang, M. C., Wood, W. H., and Cutkosky, M. R. (1998) “Data mining for thesaurus generation in informal design information retrieval.” *Proceedings of the International Computing Congress*, ASCE, Reston, Va., 189-200.