Ontology-Based Multilabel Text Classification of Construction Regulatory Documents

Peng Zhou, S.M.ASCE[1]; and Nora El-Gohary, A.M.ASCE[2]

Abstract: In order to fully automate the environmental regulatory compliance checking process, rules should be automatically extracted from applicable environmental regulatory textual documents, such as energy conservation codes. In the authors' automated compliance checking (ACC) approach, prior to rule extraction, the text is first classified into predefined categories to only retrieve relevant clauses and filter out irrelevant ones, thereby improving the efficiency and accuracy of rule extraction. Machine learning (ML) techniques have been commonly used for text classification (TC). Nonontology-based, ML-based TC has, generally, performed well. However, given the need for an exceptionally high performance in TC to support high performance in ACC, further TC performance improvement is needed. To address this need, an ontology-based TC algorithm is proposed to further improve the classification performance by utilizing the semantic features of the text. A domain ontology for conceptualizing the environmental knowledge was used. The proposed ontology-based TC algorithm was tested on 25 environmental regulatory documents, evaluated using four evaluation metrics, and compared with the authors' previously utilized ML-based approach. Based on the testing data, the results show that the ontology-based approach consistently outperformed the ML-based approach, under all evaluation metrics. **DOI: 10.1061/(ASCE)CP.1943-5487.0000530.**

Introduction

Manual compliance checking is costly and time consuming (Eastman et al. 2009). Automated compliance checking (ACC) aims to address this practical gap by reducing the cost and time of checking the compliance of construction projects to regulatory requirements. A number of important research efforts have been recently undertaken in the area of ACC, including checking of building designs (Eastman et al. 2009), building envelope performance (Tan et al. 2010), building structural design (Nawari 2012), construction quality (Zhong et al. 2012), building fire safety (Dimyadi et al. 2014), and formwork constructability (Jiang and Leicht 2015). Larger ongoing research efforts that are led by industry organizations include the Autocodes project that is led by Fiatech, which aims to automate the regulatory compliance review process with a focus on checking building accessibility and egress, fire and life safety, and mechanical and engineering (Fiatech 2014). Despite the importance of these efforts, existing ACC methods and systems are not fully automated. Existing methods and systems require manual effort to extract regulatory requirements from textual regulatory documents (e.g., codes) and encode the extracted requirements in a computer-processable rule format.

To address this gap, in the authors' previous work, a natural language processing (NLP)-enabled approach was proposed for ACC in construction to support automated text processing and analysis for automated rule extraction from regulatory documents. A rule-based semantic information extraction algorithm was proposed to automatically extract regulatory information from building codes (Zhang and El-Gohary 2013); and a rule-based semantic information transformation algorithm was proposed to automatically transform the extracted information into a computer-processable logic rule format (Zhang and El-Gohary 2015).

In order to further improve the efficiency and accuracy of information extraction and transformation, the authors further proposed a nonontology-based supervised machine learning (ML)-based multilabel text classification (TC) algorithm to classify regulatory clauses prior to information extraction. The algorithm aims to only retrieve relevant clauses, thereby avoiding inefficiency and errors in information extraction resulting from unnecessary processing of irrelevant text. The classification problem is multilabel, as opposed to single label, because one clause could be assigned more than one label (e.g., one clause could be relevant to both "thermal insulation topic" and "air leakage topic"). As commonly conducted in nonontology-based, ML-based TC, this multilabel classification problem was transformed to multiple single-label classification problems, where—for each classification problem—one clause is assigned at most one label. A nonontology-based, ML-based TC algorithm uses an ML algorithm [e.g., support vector machines (SVM)] to learn the relationships between text and labels based on previous data. In supervised ML, human supervision is provided in the form of training data [i.e., a series of labelled text units (data) that provide guidance in learning].

Nonontology-based, supervised ML-based TC has, generally, performed well (e.g., Pawar and Gawande 2012; Salama and El-Gohary 2013; Zhou and El-Gohary 2015). However, given the need for an exceptionally high performance (100% recall with high precision) in TC for supporting high performance in ACC, further TC performance improvement is needed; ideally a 100% recall of all relevant clauses is needed to avoid consequent compliance reasoning errors.

Ontologies could be explored as means to help capture text semantics for enhancing TC performance. However, there have been no research efforts for using ontology-based TC in the construction

[1] Graduate Student, Dept. of Civil and Environmental Engineering, Univ. of Illinois at Urbana-Champaign, 205 N. Mathews Ave., Urbana, IL 61801.

[2] Assistant Professor, Dept. of Civil and Environmental Engineering, Univ. of Illinois at Urbana-Champaign, 205 N. Mathews Ave., Urbana, IL 61801 (corresponding author). E-mail: gohary@illinois.edu

Note. This manuscript was submitted on October 14, 2014; approved on July 16, 2015; published online on September 30, 2015. Discussion period open until February 29, 2016; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Computing in Civil Engineering***, © ASCE, ISSN 0887-3801.

domain. Outside of the construction domain, ontology-based efforts (1) rely on supervised ML, which is human effort intensive; (2) require transformation of multilabel TC problems to multiple single-label problems, which involves extra data preparation and classifier building effort; and/or (3) show no single outperforming ontology-based method or algorithm, which indicates that it is difficult to reuse an existing ontology-based TC algorithm from one domain to the other. To address these needs and gaps, an ontology-based TC algorithm is proposed as a potential approach to further improve the classification performance by utilizing the semantic features of the text.

As such, this paper builds on the authors' previous work in the area of TC in three primary ways. First, instead of taking a nonontology-based TC approach, an ontology-based semantic TC approach for the construction domain is explored. The concepts and relationships of the ontology help in recognizing the semantic features of the text. In comparison to the previously-used nonontology-based, ML-based algorithm (Zhou and El-Gohary 2015), in the proposed ontology-based approach, a document (or clause) is represented in terms of semantic concepts and relations, rather than just terms (words). Second, the multilabel classification problem is addressed in a direct way, instead of transforming the multilabel classification problem to multiple single-label classification problems (as commonly used in ML-based TC). Third, no human supervision is involved (i.e., training data are provided without labeling). In the remainder of this paper, the proposed ontology-based TC approach for classifying environmental regulatory clauses according to a set of predefined semantic labels is presented and its performance is compared with that of the authors' previous nonontology-based, supervised ML-based approach (Zhou and El-Gohary 2015).

Background

Multilabel Text Classification Problems

NLP aims to enable computers to analyze and process natural language in a meaningful way to facilitate a range of tasks (e.g., automated machine translation) (Manning and Schutze 1999). TC, a subfield of NLP, aims to classify documents (like paragraphs or clauses) to one or more categories (Manning and Schutze 1999). A category is represented by a label, and may refer to a class or concept. A TC problem could be categorized as a multilabel or single-label TC (Tsoumakas and Katakis 2007). Multilabel TC can assign more than one label to a document, while single-label TC can only assign one label to each document. In this research, a multilabel TC problem is addressed, since multiple labels could be assigned to one clause. For example, the following clause was assigned the labels "air leakage topic" and "thermal insulation topic," because it contains requirements for high pressure ducts in terms of thermal insulation and sealing to prevent air leakage: "C403.2.7.1.2 Medium-pressure duct systems. All ducts and plenums designed to operate at a static pressure greater than 2 inches water gauge (w.g.) (500 Pa) but less than 3 in. w.g. (750 Pa) shall be insulated and sealed in accordance with Section C403.2.7. Pressure classifications specific to the duct system shall be clearly indicated on the construction documents in accordance with the International Mechanical Code" (ICC 2012).

There are two approaches to address multilabel TC problems: (1) an indirect approach, called problem transformation method (PTM), which assumes independence of labels and transforms a multilabel TC problem to multiple single-label TC problems (Tsoumakas and Katakis 2007); and (2) a direct approach, called algorithm adaptation method (AAM), which adapts algorithms to directly handle the multilabel TC problem (Tsoumakas and Katakis 2007). Using a PTM approach, one classifier is built for each label, where each classifier is independent of the other classifiers. Examples of TC work adopting PTM include Caldas et al. (2002), Kovacevic et al. (2008), and Mahfouz (2011). Using an AAM approach, only one classifier is built for all labels. Examples of TC work using AAM include Brinker and Hüllermeier (2007), Zhang and Zhou (2007), and Spyromitros et al. (2008). The advantages of AAM are (1) the ability to predict a set of labels at one time; and (2) avoiding the assumption of label independence, which is not valid in many cases because labels are usually interrelated in real world (Manning et al. 2009). Considering interrelationships between labels may improve the TC performance (Sorower 2010).

Machine Learning Techniques for Text Classification

ML refers to the ability of a computer to learn from data or past experience (Manning and Schutze 1999). ML-based TC classifies an unknown document (testing data) based on experience learned from classifying known documents (training data).

ML can be categorized into three primary types:

1. Supervised ML: Human guidance is involved, in the form of labelled training data, to guide the learning process;
2. Unsupervised ML: No human guidance is involved. Using unlabelled training data, the classifier tries to identify some unknown categories which the data can be clustered into; and
3. Semi-supervised ML: Only partial human guidance is provided by labeling a fraction of the training data.

Compared with unsupervised and semi-supervised ML techniques, supervised ML techniques require higher manual effort for preparing the training data but can typically yield better performance because of the extra human guidance (Sebastiani 2002). Some commonly used supervised and semi-supervised ML algorithms include SVM, Naïve Bayes (NB), k-nearest neighbors (kNN), and decision trees (DT) (Aggarwal and Zhai 2012). SVM is the most commonly used supervised ML algorithm. It maps the labelled data into a feature space and tries to find the best separators that distinguish all categories. The testing data are then mapped to the feature space and classified by the found separators (Aggarwal and Zhai 2012). Some commonly used unsupervised ML algorithms (Aggarwal and Zhai 2012) include k-Means and hierarchical algorithm (Aggarwal and Zhai 2012). Some less commonly used ML algorithms include labelled-latent Dirichlet allocation (LDA) (Ramage et al. 2009).

ML can also be classified into two main types: shallow learning and deep learning. Shallow learning can only learn simple functions with a linear combination of parameters from the training data (Bengio and LeCun 2007). Shallow learning algorithms (e.g., SVM) have been successful in TC, but their limited modeling and representational power make them unable to learn complex functions such as those involved in text semantics (Bengio and LeCun 2007). In contrast, deep learning can learn complex functions (cascaded by multiple single functions) with a nonlinear combination of parameters from the training data (Bengio and LeCun 2007). Deep learning attempts to model the data based on the theory of distributed representations from ML. Distributed representations assume that the data are generated by some hidden factors. Deep learning further assumes that these hidden factors are organized into a multilevel hierarchy. Therefore, deep learning models the data in a multilevel hierarchy (Bengio et al. 2013).

The most commonly used algorithm for implementing deep learning is the neural network algorithm (Bengio et al. 2003). A neural network algorithm models the iterative learning process of the human brain that learns from known information (i.e., unlabelled data) and infers new unknown information based on the learned knowledge (e.g., predicting the next word of a partial sentence based on the embedded linguistic characteristics and patterns learned from seeing a large number of sentences) (Bengio et al. 2003). Examples of neural network algorithms include the feedforward neural network algorithm (Bengio et al. 2003) and the recurrent neural network algorithm (Mikolov et al. 2010). The most state-of-the-art and best-performing algorithm is the hierarchical softmax skip-gram algorithm (Mikolov et al. 2013a, b). The hierarchical softmax skip-gram was developed to improve the accuracy of distributed representations on large datasets. It tries to learn word vector representations from the training data and predict surrounding words of the current word in a sentence based on the corresponding learned word vectors (Mikolov et al. 2013a). The learned word vectors could predict semantic relationships of words/concepts (e.g., automatic-turn versus manual-shut, lumen-luminaire versus watts-lamp, weld-gasket versus fasten-caulk) based on cosine distance of vectors.

Ontology-Based Techniques for Semantic Text Classification

Semantic TC refers to using the semantics of text to facilitate TC. An ontology is a knowledge conceptualization that captures the semantics of a domain in the form of concepts, relationships, and axioms (El-Gohary and El-Diraby 2010). An ontology can, thus, help in capturing the semantics of the text. In general, ontologies may support TC in two main ways: (1) use an ontology to represent the features of the documents and then use an ML algorithm to classify documents based on their features. For example, in Lee et al. (2009), term features are extracted from documents and mapped to the concepts of the ontology. Documents originally represented by term features get represented by ontology concept features instead. These concept features are then used for ML-based TC; and (2) use an ontology to represent the categories in terms of concept features and then use the concept features of each category to classify the documents (represented in either concept features or term features) based on either concept-to-concept or concept-to-term semantic similarity scores. For example, Yu et al. (2006) use a combination of a linguistic ontology and statistical information (such as word frequency) for TC. The ontology covers concepts that describe the syntactic features [e.g., part of speech (POS) tag of a word] and semantic features of words (e.g., semantic tag of a word). These syntactic and semantic features of words [what Yu et al. (2006) call "linguistic ontology knowledge"] are then learned based on a set of labelled training data. For TC, the keywords of documents are extracted and the documents are classified based on the linguistic ontology knowledge of its keywords. Yang et al. (2008) use concept vectors for TC. A category is represented in terms of a vector of concept-value pairs, where (1) the concept is derived from an ontology, and (2) the value is defined based on their term frequency inverse document frequency (TFIDF) scores [TFIDF aims to weigh a word in a document in terms of the total count of that word in the document and the total number of documents in the whole document collection containing that word (Aggarwal and Zhai 2012)]. A testing document is represented in terms of a vector of keyword-TFIDF pairs. The documents are then classified based on the similarities of document vectors to category vectors. In contrast to the first example, the second example may be classified as an unsupervised ontology-based effort because labelled training data are not needed. Compared with supervised ML-based TC, unsupervised ontology-based TC, thus, provides the opportunity of eliminating the massive manual effort required for labeling training data.

State of the Art and Knowledge Gaps in Text Classification

ML techniques have commonly been used for TC (e.g., Caldas et al. 2002; Kovacevic et al. 2008; Mahfouz 2011; Salama and El-Gohary 2013; Zhou and El-Gohary 2015). While generally successful, nonontology-based, ML-based TC usually discards semantic text information (e.g., meaning of words) although it is potentially very useful in identifying the correct label(s) of a document. Some nonontology-based, ML-based TC algorithms try to partially and indirectly capture some semantic text information (e.g., using Bigram model to capture relationships of adjacent words in a sentence in terms of conditional probability). However, the probabilistic and statistical methods usually achieve unsatisfactory performance in capturing the semantics of the text (Zhou and El-Gohary 2015). Semantic-based TC has, thus, been introduced to capture and take advantage of the semantics of the text for improving the TC performance. The use of ontologies in TC has, therefore, recently attracted much research effort.

In this regard, two main research gaps are identified. First, there have been no research efforts for using ontology-based TC in the construction domain. This is a lost opportunity for exploring the use of domain semantics to improve the performance of TC-based applications in construction. Second, outside of the construction domain, ontology-based TC efforts: (1) rely on supervised ML for training the classifier—using labelled training data—to learn the rules for labeling any given text (e.g., Vogrinčič and Bosnić 2011; Lee et al. 2009; He et al. 2004). This involves much manual effort in labeling the training data; (2) can only deal with single-label classification problems (e.g., Yang et al. 2008; Wei et al. 2006; Yu et al. 2006; Song et al. 2005; He et al. 2004) or are unable to deal with a multilabel TC problem directly (e.g., Waraporn et al. 2010). This requires transformation to multiple single-label problems; and/or (3) show inconsistent results for ontology-based TC in comparison with nonontology-based, ML-based TC. Some efforts (e.g., Fang et al. 2007) compared ontology-based TC with SVM-ML-based TC and showed that SVM-ML-based TC outperformed. Some efforts (e.g., Yang et al. 2008; Song et al. 2005; He et al. 2004) compared ontology-based TC with multiple ML algorithms for TC and showed that ontology-based TC outperformed only some of these ML algorithms [e.g., only NB in Yang et al. (2008)]. Other efforts (e.g., Yu et al. 2006) showed that the ontology-based approach outperformed the nonontology-based, ML-based approach using multiple algorithms like NB, kNN and SVM, but only reported enhanced performance in terms of precision (Yu et al. 2006). These inconsistent results indicate that there is no single outperforming ontology-based method or algorithm, and, thus, that it is difficult to reuse an existing ontology-based TC algorithm from one domain to the other.

Proposed Methodology for Ontology-Based Text Classification of Environmental Regulatory Documents

To address these knowledge gaps, in this paper, an ontology-based multilabel TC approach for semantic TC in the construction domain is proposed. In comparison to existing ontology-based TC methods, the proposed method is different in three primary ways. First, instead of using supervised ML (e.g., He et al. 2004; Lee et al. 2009; Vogrinčič and Bosnić 2011; Wijewickrema and Gamage 2013)

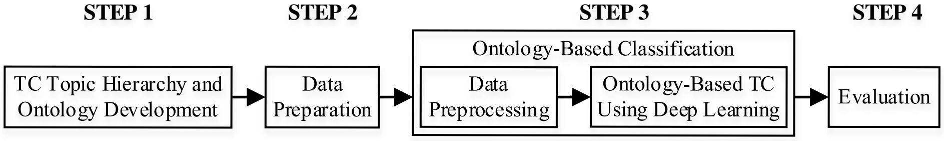

STEP 1
STEP 2
STEP 3
STEP 4
TC Topic Hierarchy and Ontology Development
Data Preparation
Ontology-Based Classification
Data Preprocessing
Ontology-Based TC Using Deep Learning
Evaluation

Fig. 1. Proposed ontology-based text classification methodology

for training the classifier to learn the rules of labeling, unsupervised ML is used for learning the semantic similarity between a term of a clause and a concept related to a topic from a set of training clauses. As a result, only the testing data are labelled, which saves much manual effort that would have been required to label the training data. Second, instead of using a problem transformation approach (e.g., He et al. 2004; Lee et al. 2009; Wijewickrema and Gamage 2013), a direct multilabel ontology-based TC method is used. As a result, (1) only one pair of training and testing data needs to be prepared; and (2) only one classifier needs to be built. This reduces the data preparation and classifier building effort. Third, instead of using shallow learning (e.g., Lee et al. 2009), deep learning is used to better represent the complexity that exists in text semantics. This aims to enhance the performance of TC.

A four-phase methodology for ontology-based domain-specific TC is proposed, as shown in Fig. 1. The labels are defined based on a hierarchy of topics. For each topic, a subontology is built to model the concepts and relationships that are related to this topic. Then, a deep learning algorithm is applied to learn the similarities between each clause (based on the terms in the clause) and each topic (based on the ontological concepts related to this topic) for classifying each clause into zero or more topics.

TC Topic Hierarchy and Ontology Development

A topic hierarchy was first developed to identify the labels that will be used for TC. In this paper, the analysis is focused on the "energy efficiency topic," which is a subtopic of "environmental topic" (as per Fig. 2). For developing the topic hierarchy, the established methodologies for taxonomy development (e.g., El-Gohary and El-Diraby 2010) were followed. The methodology includes two primary steps: (1) identification of the main concepts in the domain of interest: the concepts were extracted based on a review of the main relevant environmental regulatory documents (e.g., the 2012 International Energy Conservation Code and the 2010 California Energy Code); and (2) organization of the identified concepts into a hierarchy of concepts: the concepts were structured into a taxonomy using a combination of a top-down (starting by defining the most abstract concepts) and a bottom-up approach (starting by defining the most specific concepts). The "commercial building energy efficiency topic" subhierarchy is shown in Fig. 2. Six of the ten leaf nodes (subtopics in the taxonomical topic hierarchy) were used as labels for classification.

For modeling the semantic information associated with each topic, the hierarchy was extended into an application ontology. For each topic, a subontology was built to model the concepts and relationships that are related to this topic. For developing the ontology, the ontology development methodology by El-Gohary and El-Diraby (2010) was benchmarked. The main steps that were used to develop the ontology include: (1) purpose and scope definition: the purpose of the ontology is to support semantic TC and the scope is limited to "commercial building energy efficiency"; (2) taxonomy building: the same methodology as that used for building the TC topic hierarchy was followed (as described above). For the identification of the main concepts (the first step in taxonomy development), the scope was focused on identifying the main concepts that are related to each of the six leaf node concepts in the TC topic hierarchy. For example, the concepts related to the "lighting control topic" include "multilevel lighting control," "daylighting control," and "demand responsive control"; (3) relation modeling: the nonhierarchical relationships between concepts were identified and modeled to describe the semantic links between concepts. For example, "is_controlled_by" links the concepts "luminaire control" and "motion sensor"; and (4) ontology coding: the concepts and relations were represented using unified modeling language (UML) class diagrams. For example, Figs. 3 and 4 show the subontologies for the "lighting system control" and "lighting power" topics, respectively.

Data Preparation

Approximately 2,400 clauses were collected from 25 regulatory documents (Fig. 5). The documents were manually selected

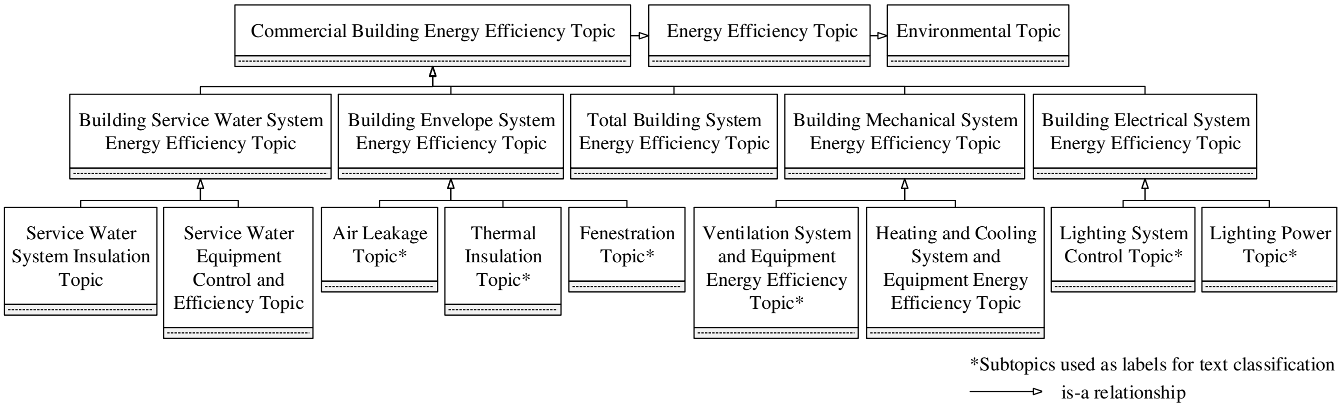

Commercial Building Energy Efficiency Topic
Energy Efficiency Topic
Environmental Topic
Building Service Water System Energy Efficiency Topic
Building Envelope System Energy Efficiency Topic
Total Building System Energy Efficiency Topic
Building Mechanical System Energy Efficiency Topic
Building Electrical System Energy Efficiency Topic
Service Water System Insulation Topic
Service Water Equipment Control and Efficiency Topic
Air Leakage Topic*
Thermal Insulation Topic*
Fenestration Topic*
Ventilation System and Equipment Energy Efficiency Topic*
Heating and Cooling System and Equipment Energy Efficiency Topic
Lighting System Control Topic*
Lighting Power Topic*
*Subtopics used as labels for text classification
is-a relationship

Fig. 2. Text classification topic hierarchy

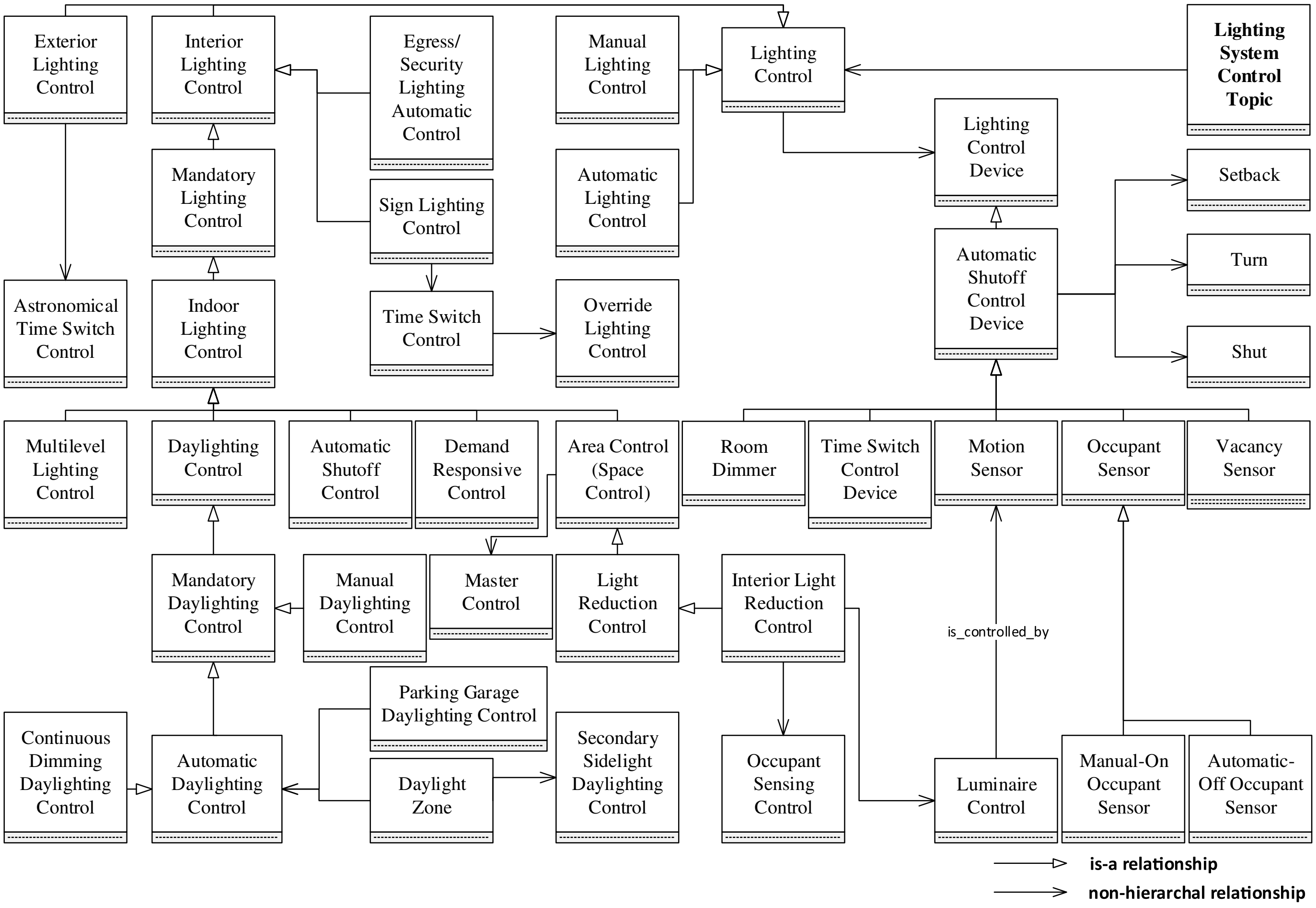

Exterior Lighting Control
Interior Lighting Control
Egress/ Security Lighting Automatic Control
Manual Lighting Control
Lighting Control
Lighting System Control Topic
Lighting Control Device
Mandatory Lighting Control
Sign Lighting Control
Automatic Lighting Control
Setback
Automatic Shutoff Control Device
Turn
Astronomical Time Switch Control
Indoor Lighting Control
Time Switch Control
Override Lighting Control
Shut
Multilevel Lighting Control
Daylighting Control
Automatic Shutoff Control
Demand Responsive Control
Area Control (Space Control)
Room Dimmer
Time Switch Control Device
Motion Sensor
Occupant Sensor
Vacancy Sensor
Mandatory Daylighting Control
Manual Daylighting Control
Master Control
Light Reduction Control
Interior Light Reduction Control
is_controlled_by
Parking Garage Daylighting Control
Continuous Dimming Daylighting Control
Automatic Daylighting Control
Daylight Zone
Secondary Sidelight Daylighting Control
Occupant Sensing Control
Luminaire Control
Manual-On Occupant Sensor
Automatic-Off Occupant Sensor
is-a relationship
non-hierarchal relationship

Fig. 3. Partial subontology for "Lighting System Control Topic"

because they all contain energy efficiency requirements for commercial buildings, which is the scope of this research. All original documents were downloaded in PDF format. The clauses were then extracted manually from the documents and each clause was represented in a separate text (.txt) format file. A clause is defined as a paragraph of text that contains at least one requirement. For example, the following is a clause that was extracted from the 2012 International Energy Conservation Code (ICC 2012): "C403.2.7.1.2 Medium-pressure duct systems. All ducts and plenums designed to operate at a static pressure greater than 2 inches water gauge (w.g.) (500 Pa) but less than 3 inches w.g. (750 Pa) shall be insulated and sealed in accordance with Section C403.2.7. Pressure classifications specific to the duct system shall be clearly indicated on the construction documents in accordance with the International Mechanical Code."

In collecting the data (clauses), data sufficiency in terms of quantity and quality was taken into account. The performance of the proposed methodology highly depends on the quantity and quality of data (Mikolov et al. 2013a) due to the use of deep learning (in Step 3). The use of large quantities of data can facilitate accurate learning of semantics. Good quality data refers to the words in the sentences being logical and coherent. Based on the experimental results, the good performance results indicate that data are sufficient in terms of quantity and quality.

The collected clauses were split into two sets, a training set and a testing set, at a ratio 5:1. Because the proposed methodology uses unsupervised ML, only the testing set was manually labelled for use in performance evaluation (in Step 4). The gold standard (the labelled testing set) was developed manually. Based on content, the first author assigned each testing clause zero or more of the six labels. The labeling was then checked by two other researchers. Full agreement on labeling was achieved.

Ontology-Based Classification

Data Preprocessing

Data preprocessing is the process of transforming the raw text into the required format. Three steps of data preprocessing were implemented: (1) Tokenization: Tokenization aims to segment the text into words or tokens, meanwhile eliminating characters such as punctuation and transforming words to their lowercase form. For example, "building, Thermal Insulation" are tokenized to "building thermal insulation"); (2) Stemming: Stemming aims to strip off word suffixes (in some cases prefixes) to its root or stem. For example, "insulation" and "insulated" can both be mapped to "insul." Stemming reduces the number of features by combining words sharing the same stem. It is usually effective in improving the performance of classification (e.g., reaching 5% gain in average precision for English) (Manning et al. 2009). In the proposed ontology-based methodology, stemming was implemented using the Porter2 stemming algorithm in Python programming language; and (3) Stopword removal: Stopwords refer to those high-frequency and low-content words that are not discriminative in classification like "am," "is," "a," "the," and "of." Removing

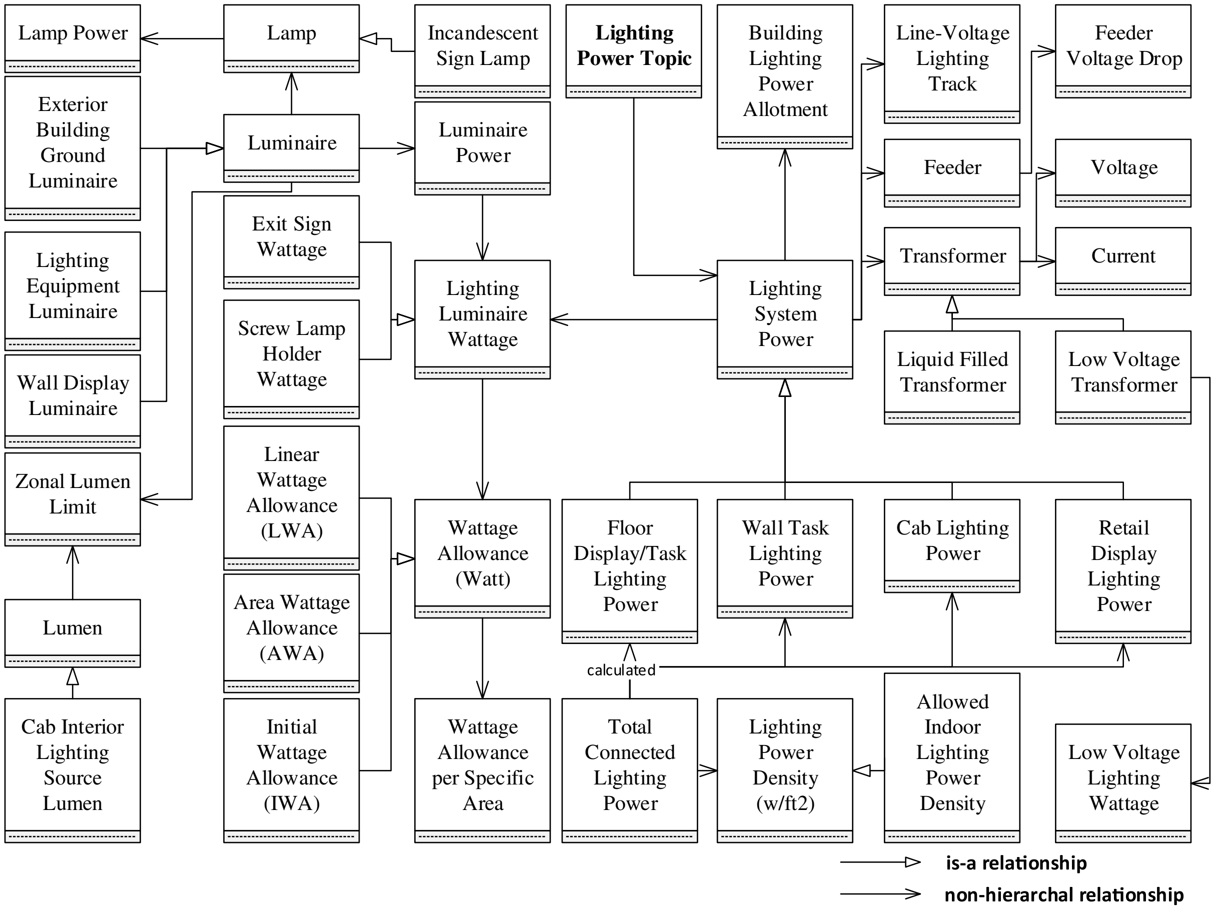
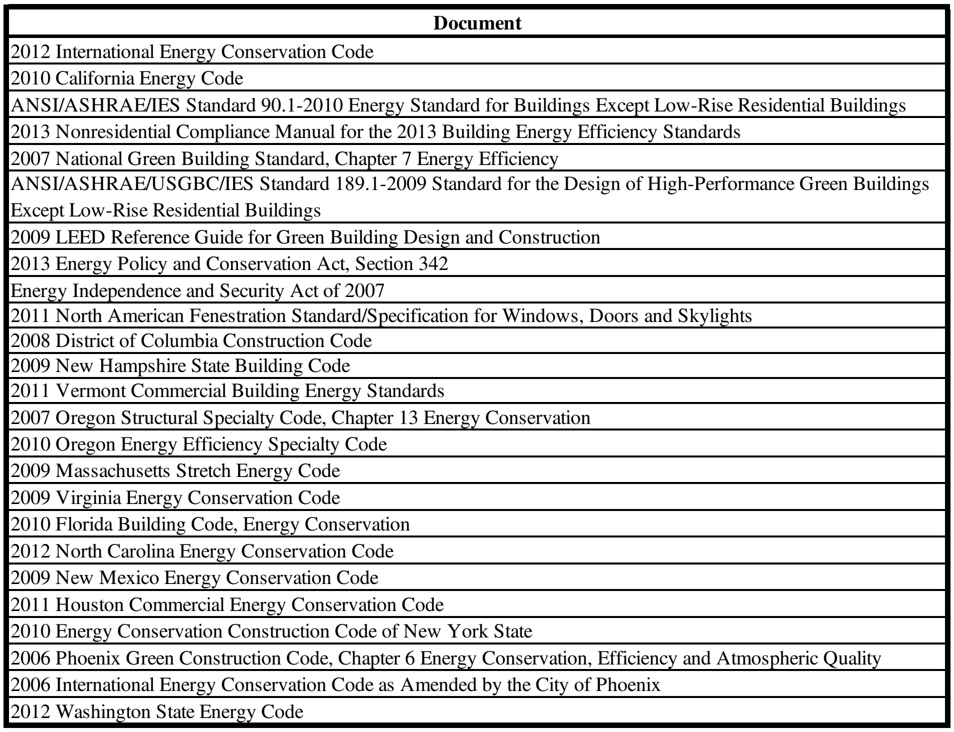

Lamp Power
Lamp
Incandescent Sign Lamp
Lighting Power Topic
Building Lighting Power Allotment
Line-Voltage Lighting Track
Feeder Voltage Drop
Exterior Building Ground Luminaire
Luminaire
Luminaire Power
Feeder
Voltage
Lighting Equipment Luminaire
Exit Sign Wattage
Transformer
Current
Screw Lamp Holder Wattage
Lighting Luminaire Wattage
Lighting System Power
Wall Display Luminaire
Liquid Filled Transformer
Low Voltage Transformer
Zonal Lumen Limit
Linear Wattage Allowance (LWA)
Wattage Allowance (Watt)
Floor Display/Task Lighting Power
Wall Task Lighting Power
Cab Lighting Power
Retail Display Lighting Power
Lumen
Area Wattage Allowance (AWA)
calculated
Cab Interior Lighting Source Lumen
Initial Wattage Allowance (IWA)
Wattage Allowance per Specific Area
Total Connected Lighting Power
Lighting Power Density (w/ft2)
Allowed Indoor Lighting Power Density
Low Voltage Lighting Wattage
is-a relationship
non-hierarchal relationship

Fig. 4. Partial subontology for "Lighting Power Topic"



Document
2012 International Energy Conservation Code
2010 California Energy Code
ANSI/ASHRAE/IES Standard 90.1-2010 Energy Standard for Buildings Except Low-Rise Residential Buildings
2013 Nonresidential Compliance Manual for the 2013 Building Energy Efficiency Standards
2007 National Green Building Standard, Chapter 7 Energy Efficiency
ANSI/ASHRAE/USGBC/IES Standard 189.1-2009 Standard for the Design of High-Performance Green Buildings Except Low-Rise Residential Buildings
2009 LEED Reference Guide for Green Building Design and Construction
2013 Energy Policy and Conservation Act, Section 342
Energy Independence and Security Act of 2007
2011 North American Fenestration Standard/Specification for Windows, Doors and Skylights
2008 District of Columbia Construction Code
2009 New Hampshire State Building Code
2011 Vermont Commercial Building Energy Standards
2007 Oregon Structural Specialty Code, Chapter 13 Energy Conservation
2010 Oregon Energy Efficiency Specialty Code
2009 Massachusetts Stretch Energy Code
2009 Virginia Energy Conservation Code
2010 Florida Building Code, Energy Conservation
2012 North Carolina Energy Conservation Code
2009 New Mexico Energy Conservation Code
2011 Houston Commercial Energy Conservation Code
2010 Energy Conservation Construction Code of New York State
2006 Phoenix Green Construction Code, Chapter 6 Energy Conservation, Efficiency and Atmospheric Quality
2006 International Energy Conservation Code as Amended by the City of Phoenix
2012 Washington State Energy Code

Fig. 5. Regulatory document list

stopwords from a document can, thus, reveal the content-bearing, discriminative words. The similarity between the discriminative terms of a document and the concepts of each subontology can then be measured for classification. A Python preprocessing program was coded for implementing the above-mentioned three subtasks. The input to the program is two sets of raw .txt files (training and testing sets, as obtained from Step 2), and the output is two preprocessed datasets (training dataset and testing dataset). The training and testing datasets are the input to the ontology-based TC algorithm (discussed in the "Ontology-Based TC Using Deep Learning: Proposed Method" subsection).

Ontology-Based TC Using Deep Learning: Proposed Method

The proposed classification method is similarity-based. After the datasets are preprocessed, a deep learning algorithm is applied on the training dataset to learn the distributed representations of terms and concepts for capturing the similarities between each term in a clause and each concept in the ontology. Accordingly, the similarity between a clause and a topic is quantified. The assignment of a label to a clause is then determined based on similarity values and two experimentally-defined similarity thresholds.

The hierarchical softmax skip-gram algorithm is used for deep learning and computing the similarities between each term in a clause and each concept in the ontology; it was selected because of its best-reported performance (Mikolov et al. 2013a, b). The algorithm learns the distributed representations of terms (in the clauses that exist in the training data) and concepts (in the ontology) from training data. A distributed representation of a term (or concept) is a real-valued vector of features that characterize the meaning of the term (or concept). The feature vector includes syntactic features (e.g., morphological category like gender of noun) and semantic features (e.g., hypernymy relation like room-bedroom) (Mikolov et al. 2013b). After learning the distributed representations of terms and concepts, the similarity between each term in a testing clause (i.e., a clause in the testing dataset) and each concept in the ontology is measured by the cosine similarity of their vectors (Mikolov et al. 2013c). Cosine similarity measures the angle between vectors and a smaller angle indicates higher similarity (Harispe et al. 2013).

The similarities between each term in a testing clause and each concept related to a topic are then summed up for each clause-topic pair to compute the total similarity (TS) between a clause and a topic. All topics with a positive TS with a clause are selected as potential labels for that clause; and the topics with a negative TS are filtered out. Zero similarity is used as the threshold for selecting potential labels by assuming that, in this application, any clause that is relevant to a topic must have a positive TS to that topic. Under this assumption, all true labels of a clause are a subset of its potential labels. The experimental results show that this assumption is valid in this application.

If there is exactly one topic with a positive TS with a clause, then that topic is assigned as the only label for that clause. If there are no topics with a positive TS with a clause, then no topics are assigned to that clause. If there are more than one topic with a positive TS with a clause, then the topic with the highest TS is assigned as the primary label for that clause (the corresponding topic is then referred to as a primary topic). In this case, based on the primary topic, two thresholds are used for assigning the remaining labels (referred to as secondary labels) for that clause: (1) the total similarity difference (TSD) [Eq. (1)] between the primary topic and the other topic(s) (potential secondary topics) are measured. Then, a TSD threshold is used to further identify the secondary labels. Topics with a TSD equal to or less than the threshold are assigned to the clause as its secondary labels. The larger the TSD threshold, the more secondary labels are assigned to the clause. In this case, more true labels are likely to be recalled, but incorrect labels may also be assigned to the clause; and (2) the total similarity percentage difference (TSPD) [Eq. (2)] between the primary topic and the potential secondary topic(s) are measured. Then, a TSPD threshold is used to further identify the secondary labels. Topics with a TSPD equal to or less than the threshold are assigned to the clause as its secondary labels. Both thresholds values are set experimentally for maximizing the overall performance. TSD and TSPD values are calculated as per Eqs. (1) and (2), where TSD_{sd} = TSD of topic s for clause d; TSPD_{sd} = TSPD of topic s for clause d; TS_{pd} = TS of topic p for clause d; and TS_{sd} = TS of topic s for clause d, topic s is a potential secondary topic of clause d, and topic p is the primary topic of clause d.

$$\mathrm{TSD}_{\mathrm{sd}} = \mathrm{TS}_{pd} - \mathrm{TS}_{sd} \tag{1}$$

$$\mathrm{TSPD}_{sd} = \frac{\mathrm{TS}_{pd} - \mathrm{TS}_{sd}}{\mathrm{TS}_{pd}} \tag{2}$$

In using both threshold values, two assumptions are made. First, the strength or weakness of relevance of a clause to multiple topics could be reflected and ordered by their corresponding TS values. The topic with the strongest relevance (i.e., highest TS) is the primary label, and all other relevant topics are the secondary labels. Second, for a certain clause, the TS values of all relevant secondary topics should be closer to the TS value of the primary topic than those of irrelevant topics, thereby having these relevant topics falling in a certain TSPD range. The final labels assigned to a clause are selected based on one of the two threshold values, whichever is the strictest. Since each topic addresses a different aspect of energy efficiency, specific threshold values should be set for each topic.

An illustrative example showing label assignments for a given clause based on similarity and threshold values is provided in Table 1. The TS value of each topic was computed, as shown in Table 1. Accordingly, "air leakage topic" was assigned as the primary label of that clause, because of its highest TS value (1,723.4). Accordingly, the TSD and TSPD values of each potential secondary topic were computed, as shown in Table 1. For example, the "thermal insulation topic" has TSD and TSPD values of 396.5 and 23.0%, respectively. Based on these values, the "thermal insulation topic" was assigned as a secondary label of that clause because both of its TSD and TSPD values fall below the threshold values (486.9 and 37.8% TSD threshold and TSPD threshold, respectively). All other potential secondary topics were not assigned because they did not meet the threshold values.

Implementation of the Proposed Method

To implement the proposed method, *Generate Similar* (Gensim) (Rehurek and Sojka 2010), a Python programming language version of the softmax skip-gram algorithm (by Mikolov et al. (2013b), was used for deep learning and for computing the similarities between each term in a testing clause and each concept in the ontology. The input to the tool includes (1) the training data, split as a set of sentences as required by the tool; (2) testing clauses; and (3) ontology concepts related to each topic. Some input parameters that were experimentally set include: (1) the dimensionality of word vectors was set as 200; (2) the maximum distance between the current and predicted word in a sentence was set as two; (3) after removing stopwords, all remaining words that have a frequency lower than five were ignored; and (4) two worker threads (a parameter that is related to the training speed of multicore machines) were used. For a more detailed description of these parameters, the readers are referred to Rehurek and Sojka (2010). The output of the tool

Table 1. Example of Classifying a Testing Clause

Topic	Total similarity[a]	Total similarity difference[a]	Total similarity percentage difference (%)[a]	Total similarity difference threshold[a]	Total similarity percentage difference threshold (%)[a]	Result[a]
Air leakage topic	1,723.4	0.0	0.0	486.9	37.8	Assigned (primary)
Thermal insulation topic	1,326.9	396.5	23.0	486.9	37.8	Assigned (secondary)
Fenestration topic	406.0	1,317.4	76.4	486.9	37.8	Not assigned
Ventilation system and equipment energy efficiency topic	304.2	1,419.2	82.4	486.9	37.8	Not assigned
Lighting power topic	222.0	1,501.4	87.1	486.9	37.8	Not assigned
Lighting system control topic	48.7	1,674.7	97.2	486.9	37.8	Not assigned

[a]For the following clause: "503.2.7 Duct and plenum insulation and sealing. All supply and return air ducts and plenums shall be insulated with a minimum of R-5 insulation when located inside the building thermal envelope and a minimum of R-8 insulation when located outside the building thermal envelope in accordance with Table 503.2.7. When located within a building envelope assembly, the duct or plenum shall be separated from the building exterior or unconditioned or exempt spaces by a minimum of R-8 insulation. Exceptions: 1. When located within equipment. 2. When the design temperature difference between the interior and exterior of the duct or plenum does not exceed 15°F (8°C). All ducts, air handlers and filter boxes shall be sealed in accordance with the Mechanical Code and SMACNA Method A." (Houston City Council 2011).

is the similarity values between each term in a testing clause and each concept in the ontology.

Another program was developed to (1) compute the TS value between each testing clause and each topic based on the similarity values (from the previous program); (2) assign the primary labels of the testing clauses; (3) compute the TSD and TSPD values for each potential secondary topic of a testing clause; and (4) assign the secondary labels to the testing clauses based on whether their TSD and TSPD values meet the threshold values. The program was coded in Python.

Evaluation

Since the proposed ontology-based TC algorithm can deal with multilabel classification problems directly, multilabel classification evaluation metrics were used. Four types of evaluation metrics were utilized. Although the metrics are different, they all use redefined recall and precision measures to evaluate the overall performance. In general, recall measures the number of correctly predicted true labels [(true positive (tp)] as a percentage of the total number of true labels [tp plus false negative (fn)]; and precision measures the number of correctly predicted true labels (tp) as a percentage of the total number of predicted labels [tp plus false positive (fp)]. Typically, there is a tradeoff between recall and precision, because the more true labels are recalled, the higher the risk of making precision errors. In the subject application, recall is given a higher priority than precision because missing to recall one relevant clause—and thus missing to check compliance of the project with this clause—might result in noncompliance detection errors.

Multilabel evaluation metrics can be categorized into two main types: example-based metrics and label-based metrics (Tsoumakas et al. 2010; Madjarov et al. 2012). Using example-based metrics, the performance (recall and precision) of classification for each test document (clause, in this research) is calculated, and the overall performance is obtained by calculating the mean performance over all test documents. Example-based recall and precision are calculated using Eqs. (3) and (4) (Madjarov et al. 2012), where tp_i = number of labels predicted correctly as positive for a testing document i; fp_i = number of labels predicted incorrectly as positive for a testing document i; fn_i = number of labels predicted incorrectly as negative for a testing document i; and N = total number of test documents

$$\text{Example–based Recall} = \frac{1}{N}\sum_{i=1}^{N}\frac{|tp_i|}{|tp_i + fn_i|} \tag{3}$$

$$\text{Example–based Precision} = \frac{1}{N}\sum_{i=1}^{N}\frac{|tp_i|}{|tp_i + fp_i|} \tag{4}$$

Using label-based metrics, the classification performance is calculated for each category, and the overall performance is obtained by calculating the mean performance across all categories. Six label-based metrics are used: micro-recall, micro-precision, macro-recall, macro-precision, weighted-recall, and weighted-precision. The six metrics are calculated using Eqs. (5)–(10) (Madjarov et al. 2012; Pedregosa et al. 2011), where tp_j = number of test documents labelled correctly as positive for a category j; fp_j = number of test documents labelled incorrectly as positive for a category j; fn_j = number of test documents labelled incorrectly as negative for a category j; C = total number of categories (in this application, $C = 6$, since there are six topics in total); and $(tp_j + fn_j)$ = total number of true labels for all test documents for category j. During the evaluation of each category, the label-based metrics temporarily treat the category as positive and all other categories as negative. Micro-recall and micro-precision measure the overall performance by counting the total number of tp, fp, and fn across all categories. Macro-recall and macro-precision calculate the recall and precision by counting the total number of tp, fp, and fn for each category, and then use the arithmetic mean performance across all categories to obtain the overall performance. The weighted-based metrics are very similar to the macro-based metrics, except that the former use the total number of true labels for all test documents across each category as a weight to obtain a weighted mean performance

$$\text{Micro–Recall} = \frac{\sum_{j=1}^{C} tp_j}{\sum_{j=1}^{C} tp_j + \sum_{j=1}^{C} fn_j} \tag{5}$$

$$\text{Micro–Precision} = \frac{\sum_{j=1}^{C} tp_j}{\sum_{j=1}^{C} tp_j + \sum_{j=1}^{C} fp_j} \tag{6}$$

$$\text{Macro–Recall} = \frac{1}{C}\sum_{j=1}^{C}\frac{tp_j}{tp_j + fn_j} \tag{7}$$

$$\text{Macro–Precision} = \frac{1}{C}\sum_{j=1}^{C}\frac{tp_j}{tp_j + fp_j} \tag{8}$$

$$\text{Weighted–Recall} = \frac{1}{\sum_{j=1}^{C}(tp_j + fn_j)} \sum_{j=1}^{C} (tp_j + fn_j) \frac{tp_j}{tp_j + fn_j} \quad (9)$$

$$\text{Weighted–Precision} = \frac{1}{\sum_{j=1}^{C}(tp_j + fn_j)} \times \sum_{j=1}^{C} (tp_j + fn_j) \frac{tp_j}{tp_j + fp_j} \quad (10)$$

These different types of metrics can be used in combination to indicate performance from multiple perspectives. Although related, these metrics measure the performance in different ways (Madjarov et al. 2012; Pedregosa et al. 2011). Example-based metrics treat all documents with equal weight regardless of the different number of labels the documents may have. For example, a two-label document with only one correctly predicted label and a six-label document with three correctly predicted labels have equal example-based recall (i.e., $1/2 = 3/6$), although the two documents contribute to the absolute number of errors differently. In contrast to the example-based metrics, label-based metrics take this difference (i.e., number of labels for each document) into account. Among the three types of label-based metrics, micro-based metrics do not consider which category the incorrect labels come from but instead consider all errors from all categories equally in performance assessment. In contrast, macro-based metrics consider which category the incorrect labels come from and assess the overall performance in terms of the performance for each category. In comparison to macro-based metrics, weighted-based metrics further weigh the performance for each category in terms of the total number of true labels in that category. Micro-based metrics are, thus, the most stringent, because an error from any category contributes equally to the total performance, whereas in example-based, macro-based, and weighted-based metrics an error could be discounted during averaging/weighting. A high performance variance across macro-based metrics and weighted-based metrics may indicate that the dataset suffers from a label imbalance problem. Label imbalance problems are common in multilabel classification; they occur when some categories have much more documents than other categories (e.g., 1 versus 100) (Chawla et al. 2004; Charte et al. 2013).

Experimental Results and Analysis

The proposed ontology-based TC algorithm was tested on the six topics using the four types of evaluation metrics. The overall performance under each metric and their corresponding thresholds are summarized in Table 2. Among the four types of metrics, the example-based metric yielded the highest performance at 98.69% recall and 92.70% precision, while the micro-based metric showed the least performance at 97.32% recall and 86.51% precision. These results are consistent with the fact that micro-based metrics are the most stringent.

Evaluation of the Proposed Ontology-Based TC Algorithm

The overall performance under each multilabel evaluation metric may (1) illustrate the strengths and weaknesses of the proposed methodology in classifying environmental regulatory documents, thereby providing clues for refining the methodology for further performance improvement; and (2) indicate some characteristics of environmental regulatory documents for future comparison with those of documents of other types (e.g., contract documents) and other domains (e.g., safety).

Threshold Analysis

A threshold analysis was conducted. The thresholds (TSD threshold and TSPD threshold) of each topic are keys in determining the assignment of multiple labels to a clause. Analyzing the thresholds of the topics may thus help in understanding the characteristics/relationships of the topics and may reveal some clues for further performance improvement. Based on the analysis, it is observed that thresholds of topics may reflect the existing semantic overlaps/relationships among the topics and, thus, the organization of topics in the hierarchy. Three observations are made. First, a semantic overlap/relationship between two topics may be reflected by the closeness of their threshold values. For example, the semantic relationships between the "air leakage topic" and the "thermal insulation topic" (e.g., that air leakage in a building envelope directly influences its quality of thermal insulation) was reflected by the closeness of their TSD threshold values (486.9 and 370.1, respectively). Similarly, both "lighting system control topic" and "lighting power topic" used the same TSD threshold value (79.8). The close/same threshold values are expected due to the existing semantic relationships between the two topics. For example, installing an automatic shut-off in the lighting system can save energy to help meet the requirements of lighting power. This may be substantiated by the fact that these two topics are located in the same branch of the topic hierarchy. Second, the semantic distinctiveness (i.e., less overlap) of a topic may be reflected by its small threshold values compared to other topics. For example, the relatively small TSD thresholds (the smallest at 79.8) of the "lighting system control

Table 2. Performance of Ontology-Based TC Approach

	Ontology-based approach										Machine learning-based approach	
	Total similarity difference threshold	Total similarity percentage difference threshold (%)	Example-based metric		Micro-based metric		Macro-based metric		Weighted-based metric			
Topic			Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)	Recall (%)	Precision (%)
Air leakage topic	486.9	37.8	98.69	92.70	97.32	86.51	97.65	90.44	97.32	89.01	97.30	84.30
Fenestration topic	120.2	44.0	98.69	92.70	97.32	86.51	97.65	90.44	97.32	89.01	97.30	84.30
Lighting power topic	79.8	4.3	98.69	92.70	97.32	86.51	97.65	90.44	97.32	89.01	97.30	84.30
Lighting system control topic	79.8	17.2	98.69	92.70	97.32	86.51	97.65	90.44	97.32	89.01	97.30	84.30
Thermal insulation topic	370.1	0.5	98.69	92.70	97.32	86.51	97.65	90.44	97.32	89.01	97.30	84.30
Ventilation system and equipment energy efficiency topic	140.1	18.6	98.69	92.70	97.32	86.51	97.65	90.44	97.32	89.01	97.30	84.30

topic" and the "lighting power topic" indicate that these two topics are likely not semantically related to the other topics. This may be reflected by the fact that these two topics are the only two topics in their hierarchy branch. Similarly, the TSD and TSPD thresholds of the "ventilation system and equipment energy efficiency topic" (140.1 and 18.6%) were small compared to the other topics, indicating that this topic is likely a semantically isolated topic. This may be substantiated by the fact that it is a sole topic in its hierarchy branch, thereby sharing less semantic overlaps with the other topics. Less semantic overlaps with other topics makes the classification of clauses related to this topic easier, which is partially reflected by its high performance, namely 100% and 97.8% macro-based recall and precision, respectively. Third, the semantic dominance of a primary topic, compared to secondary topics, may be reflected by its large TSD and TSPD thresholds. For example, the largest TSD threshold (486.9) and the second largest TSPD threshold (37.8%) were used for the "air leakage topic." The large thresholds indicate that the total similarities of the secondary topics to the labelled clauses are relatively much smaller than that of the primary topic. This further indicates that the "air leakage topic" is relatively more relevant to these clauses than their secondary topics.

Performance Analysis

The proposed ontology-based TC methodology achieved overall recall values from 97.32 to 98.69% and overall precision values from 86.51 to 92.70%. First, compared to other ontology-based TC efforts, the proposed methodology: (1) outperforms some efforts (e.g., Fang et al. 2007; Wei et al. 2006) in both recall and precision; (2) outperforms some efforts in a limited way. For example, the proposed algorithm outperforms some efforts (Song et al. 2005; Yang et al. 2008) in recall under all four metrics but in precision under only example-based and macro-based metrics; and (3) is outperformed by some efforts in a limited way. For example, He et al. (2004) only addressed a single-label binary classification problem, though they achieved both recall and precision values of over 97%. Second, in terms of performance improvement compared with the nonontology-based, supervised ML-based TC [proposed in the authors' previous work (Zhou and El-Gohary 2015)], the proposed approach shows improvement in both recall and precision (average improvement of 0.5% in recall and 5.4% in precision), rather than a trade-off improvement [e.g., recall was improved at the expense of precision in Wei et al. (2006), Fang et al. (2007), and Yang et al. (2008)]. The proposed algorithm shows more improvement in precision compared with recall because many incorrect labels are filtered out during the assignment of secondary labels.

Among all four types of evaluation metrics, (1) the example-based metrics showed the highest performance, at 98.69% recall and 92.70% precision. This indicates a high performance level; (2) the most stringent metric—the micro-based metrics—showed 97.32% recall and 86.51% precision. This provides the most conservative performance estimate for comparison with the nonontology-based, ML-based approach; (3) the macro-based metrics showed 97.65% recall and 90.44% precision by evaluating the performance in terms of each category; and (4) the weighted-based metrics showed 97.32% recall and 89.01% precision. The small difference between the macro-based and weighted-based metrics (a difference of 0.33% in recall and 1.43% in precision) may indicate that the dataset does not suffer from label imbalance problems.

An error analysis was conducted to identify the sources of errors. Precision errors come from incorrectly assigning false labels to some clauses. A semantic relationship between two or more topics may result in misclassification among these topics. For example, the following clause was labelled incorrectly with "thermal insulation topic," in addition to the correct label "air leakage topic," since any air leakage in the building thermal envelope may compromise the performance of thermal envelope insulation: "C402.4.8 Recessed lighting. Recessed luminaires installed in the building thermal envelope shall be sealed to limit air leakage between conditioned and unconditioned spaces" (ICC 2012).

Recall errors come from incorrectly missing to assign true labels to some clauses. A semantic dominance of a primary topic, compared to secondary topics, may result in missing secondary labels. For example, one secondary label ("thermal insulation topic") was missed for both of the following two clauses, because their primary topics ("air leakage topic" and "fenestration topic" for Clauses 1 and 2, respectively) dominated semantically: (1) "C403.2.7.3.3 High-pressure duct systems. Ducts designed to operate at static pressures in excess of 3 inches water gauge (w.g.) (750 Pa) shall be insulated and sealed in accordance with Section C403.2.7." (ICC 2012); and (2) "502.3.2 Maximum U-factor and SHGC. For vertical fenestration and skylights, the maximum U-factor and solar heat gain coefficient (SHGC) shall be as specified in Table 502.3." (ICC 2009).

Performance Comparison: Ontology-Based Approach versus ML-Based Approach

The performance of the proposed ontology-based approach was further compared with that of the nonontology-based, supervised ML-based approach [proposed in the authors' previous work (Zhou and El-Gohary 2015)], in terms of recall and precision, as illustrated in Fig. 6. In this work (Zhou and El-Gohary 2015), SVM was selected based on performance after testing ten commonly used ML algorithms, including SVM (implemented in both linear and rbf kernel), DT [implemented by classification and regression trees algorithm (Breiman et al. 1984)], NB (implemented by three variances of algorithms: Gaussian NB, multinomial NB, Bernoulli NB), kNN, radius-based neighbors, nearest centroid, random forest, and gradient boosted regression trees (Aggarwal and Zhai 2012; Breiman 2001; Friedman 2001).

Under the four evaluation metrics, the ontology-based approach achieved recall values from 97.32 to 98.69%, and precision values from 86.51 to 92.70%. Thus, based on the testing data, it consistently outperformed the nonontology-based, supervised ML-based approach that had achieved 97.30% and 84.30% overall average recall and precision, respectively. This shows that the proposed ontology-based approach is potentially successful in utilizing the semantics of the text for improving the performance of TC.

Limitations and Future Work

Two main limitations of the work are acknowledged. First, the performance of the proposed approach—and of ontology-based approaches in general—depends on the quality of ontologies used. In their future work, the authors will test the proposed approach in classifying environmental regulatory documents using other ontologies. Second, the proposed methodology was tested on only six topics. In their future work, the authors will test the methodology in classifying environmental regulatory clauses based on more environmental topics.

In future work, the authors will also continue to refine the proposed ontology-based methodology. For example, a more complex threshold function could be used for assigning labels, considering that the TSD threshold value may relate to the length of each clause. Since a longer clause usually contains more concepts, its TS to a topic may be much larger than the TS of a shorter clause to the same

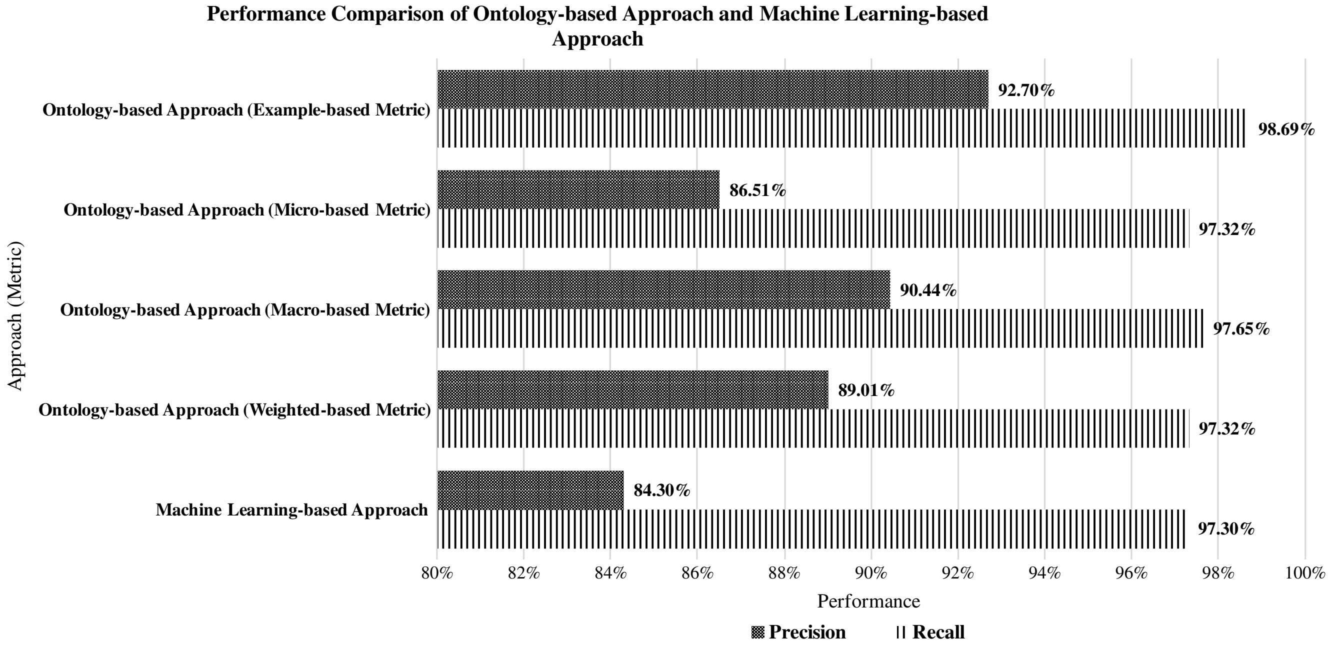


Fig. 6. Performance comparison of ontology-based approach and machine learning-based approach

topic. Therefore, the TSD of other topics for a longer clause may be much larger than that of a shorter clause.

For recommendations for future research, other researchers may also adapt the proposed methodology for classifying documents in other domains. Such adaptation research could focus on three main areas. First, testing the proposed methodology in classifying other types of documents [e.g., Occupational Safety and Health Administration (OSHA) standards] based on other types of topics (e.g., safety topics). Second, determining the threshold values of the different topics. Third, investigating how the characteristics (e.g., depth and breadth) of an ontology could affect the level of performance of the proposed methodology.

Contribution to the Body of Knowledge

This work contributes to the body of knowledge on two main levels. First, a new ontology-based TC methodology for classifying environmental regulatory documents in the construction domain is proposed. This work offers a leading initiative; it is the first ontology-based TC effort in the construction domain. In comparison to the commonly used nonontology-based, supervised ML-based approach [e.g., the authors' previous work (Zhou and El-Gohary 2015)], the proposed ontology-based approach

1. Utilizes the knowledge of the domain (in the form of an ontology) to capture the semantics of the text for enhanced classification: In nonontology-based, ML-based TC, some semantics are captured partially and indirectly using statistical and probabilistic methods. For example, the conditional probability of adjacent words in a sentence may indicate some useful word relationships for classification (e.g., using a Bigram model). However, such limited semantic information is not sufficient in adequately capturing the semantics of the text. In comparison, an ontology-based approach allows for capturing deeper semantic information by representing each category in terms of concepts and relationships.
2. Outperforms in terms of recall and precision: The experimental results showed that the proposed algorithm achieved higher recall and precision in comparison to that proposed in Zhou and El-Gohary (2015).
3. Eliminates the need for labeling training data, since unsupervised deep learning is implemented for exploring similarities: This makes the proposed approach more practical to use in real-life applications where most of the data are unlabelled.
4. Reduces the efforts of data preprocessing: The proposed algorithm only requires data in .txt format. In contrast, nonontology-based ML-based TC requires preprocessing of data into numeric vectors using text representation methods like the bag of words model.
5. Is easier to adapt for classifying other types of documents: Using the proposed methodology, the major effort in TC lies in developing an ontology (if one is not readily available) and labeling the testing dataset. In contrast, nonontology-based, ML-based TC requires experimental testing of different text representation methods, term weighting schemes, ML algorithms, etc.

Second, an improved method for ontology-based TC is offered. In comparison to existing ontology-based TC methods, the proposed method (1) uses unsupervised instead of supervised ML, which saves the manual effort needed in labeling the training data; (2) deals with the multilabel classification problem in a direct way instead of conducting problem transformation, which reduces the data preparation and classifier building effort; and (3) uses deep instead of shallow learning, which aims to better represent the complexity that exists in text semantics.

Conclusions

This paper proposed an ontology-based, multilabel TC approach for classifying environmental regulatory clauses for supporting ACC in construction. A domain ontology was developed for representing the hierarchy of environmental topics and the concepts and relationships associated with each topic. An unsupervised deep learning technique was used to learn the similarities between each clause (based on the terms in the clause) and each topic (based on the ontological concepts related to this topic) for classifying each clause into zero or more topics according to two experimentally set similarity thresholds. Four types of multilabel classification evaluation metrics were used to measure the performance of the proposed approach. Based on the testing data, across the four types of metrics, the proposed algorithm achieved overall recall and precision values from 97.32 to 98.69% and from 86.51 to 92.70%, respectively.

The experiment results also indicate the following: (1) topics may semantically overlap because of their interrelationships. For example, "air leakage topic" and "thermal insulation topic" overlap because an air leakage in the building envelope is likely to affect the performance of building thermal insulation; (2) the semantic overlaps among topics may be manifested through their positions in the topic hierarchy. For example, the semantically overlapping "air leakage topic" and "thermal insulation topic" are located under the same branch in the hierarchy; (3) threshold values may help identify semantic relationships among topics. For example, close threshold values of two topics may indicate that two topics are semantically overlapping or related to each other; relatively small threshold values of one topic compared to other topics may indicate semantic distinctiveness of that topic (i.e., less overlaps with the other topics); and relatively large thresholds of a primary topic compared to the secondary topics may indicate semantic dominance of the primary topic compared to the secondary topics. Therefore, finding the "right" threshold values are key in achieving optimal classification performance; and (4) different thresholds should be used for different topics. Since each topic is associated with different semantics, the thresholds values should be customized for each topic to determine the optimal assignment of clauses to that topic. The thresholds should be set experimentally for maximizing performance.

Compared with existing ontology-based TC methodologies, the proposed methodology uses an unsupervised deep learning algorithm for capturing the semantics behind the words and addresses the multilabel classification problem in a direct way without transformation to multiple single-label ones. Compared with the nonontology-based, supervised ML-based approach (proposed in the authors' previous work), the proposed ontology-based approach outperforms based on four evaluation metrics, reduces the efforts of data preprocessing and classifier building, and is easier to adapt for classifying other types of documents.

The proposed ontology-based TC approach could be generalized to other domains such as safety regulatory documents. The same methodology could be employed and tested, but a different ontology—one that is relevant to the domain of application (e.g., safety ontology)—would be needed. Like any other ontology-based method, the performance of the proposed methodology could vary depending on the quality of the ontology; and like any other ML-based algorithm, the performance of the proposed method could vary depending on the size of the training and testing data sets.

In future work, the authors will further test the proposed methodology in classifying other types of construction documents (e.g., contract specifications). Further refinement of the methodology may be proposed based on the testing results.

Acknowledgments

The authors would like to thank the National Science Foundation (NSF). This material is based upon work supported by NSF under Grant No. 1201170. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF.

References

Aggarwal, C., and Zhai, C. (2012). "A survey of text classification algorithms." *Mining text data*, Springer, New York, 163–222.

Bengio, Y., Courville, A., and Vincent, P. (2013). "Representation learning: A review and new perspectives." *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8), 1798–1828.

Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). "A neural probabilistic language model." *J. Mach. Learn. Res.*, 3, 1137–1155.

Bengio, Y., and LeCun, Y. (2007). "Scaling learning algorithms towards AI." *Large-Scale Kernel Mach.*, 34(5), 1–41.

Breiman, L. (2001). "Random forests." *Mach. Learn.*, 45(1), 5–32.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and regression trees*, Chapman and Hall/CRC, Boca Raton, FL.

Brinker, K., and Hüllermeier, E. (2007). "Case-based multilabel ranking." *Proc., 20th Int. Joint Conf. on Artificial Intelligence (IJCAI'07)*, Morgan Kaufmann, San Francisco, 702–707.

Caldas, C. H., Soibelman, L., and Han, J. (2002). "Automated classification of construction project documents." *J. Comput. Civ. Eng.*, 10.1061/(ASCE)0887-3801(2002)16:4(234), 234–243.

Charte, F., Rivera, A., Jesus, M. J., and Herrera, F. (2013). "A first approach to deal with imbalance in multi-label datasets." *Lecture notes in computer science (LNCS)*, Springer, Berlin, 150–160.

Chawla, N. V., Japkowicz, N., and Kotcz, A. (2004). "Editorial: Special issue on learning from imbalanced data sets." *J. SIGKDD Explor. Newsl.*, 6(1), 1–6.

Dimyadi, J., Clifton, C., Spearpoint, M., and Amor, R. (2014). "Regulatory knowledge encoding guidelines for automated compliance audit of building engineering design." *2014 Int. Conf. on Computing in Civil and Building Engineering*, ASCE, Reston, VA, 536–543.

Eastman, C., Lee, J., Jeong, Y., and Lee, J. (2009). "Automatic rule-based checking of building designs." *Automat. Constr.*, 18(8), 1011–1033.

El-Gohary, N., and El-Diraby, T. (2010). "Domain ontology for processes in infrastructure and construction." *J. Constr. Eng. Manage.*, 10.1061/(ASCE)CO.1943-7862.0000178, 730–744.

Fang, J., Guo, L., Wang, X. D., and Yang, N. (2007). "Ontology-based automatic classification and ranking for web documents." *Proc., 4th Int. Conf. Fuzzy Systems and Knowledge Discovery*, Vol. 3, IEEE, Washington, DC, 627–631.

Fiatech. (2014). "AutoCodes project." ⟨http://www.fiatech.org/project-management/projects/593-automated-code-plan-checking-tool-proof-of-concept⟩ (Jan. 21, 2015).

Friedman, J. H. (2001). "Greedy function approximation: A gradient boosting machine." *Ann. Stat.*, 29(5), 1189–1232.

Harispe, S., Ranwez, S., Stefan, J., and Montmain, J. (2013). "Semantic measures for the comparison of units of language, concepts or instances from text and knowledge representation analysis." ⟨http://arxiv.org/pdf/1310.1285.pdf⟩.

He, Q., Qui, L., Zhao, G., and Wang, S. (2004). "Text categorization based on domain ontology." *Lecture notes in computer science (LNCS)*, Springer, Berlin, 319–324.

Houston City Council. (2011). "2011 Houston commercial energy conservation code." ⟨http://edocs.publicworks.houstontx.gov/documents/divisions/planning/enforcement/2009_iecc.pdf⟩ (Mar. 24, 2015).

ICC (International Code Council). (2009). "2009 Virginia energy conservation code." ⟨http://www.ecodes.biz/ecodes_support/free_resources/virginia2009/09energy/09energy_main.html⟩ (Mar. 23, 2015).

ICC (International Code Council). (2012). "2012 international energy conservation code." ⟨http://publicecodes.cyberregs.com/icod/iecc/2012/⟩ (Mar. 21, 2015).

Jiang, L., and Leicht, R. (2015). "Automated rule-based constructability checking: Case study of formwork." *J. Manage. Eng.*, 10.1061/(ASCE)ME.1943-5479.0000304, A4014004.

Kovacevic, M., Nie, J. Y., and Davidson, C. (2008). "Providing answers to questions from automatically collected web pages for intelligent decision making in the construction sector." *J. Comput. Civ. Eng.*, 10.1061/(ASCE)0887-3801(2008)22:1(3), 3–13.

Lee, Y. H., Tsao, W. J., and Chu, T. H. (2009). "Use of ontology to support concept-based text categorization." *Lectures notes in business information processing*, Springer, Berlin, 201–213.

Madjarov, G., Kocev, D., Gjorgjevikj, D., and Džeroski, S. (2012). "An extensive experimental comparison of methods for multi-label learning." *Pattern Recognit.*, 45(9), 3084–3104.

Mahfouz, T. (2011). "Unstructured construction document classification model through support vector machine (SVM)." *Proc., Int. Workshop on Computing in Civil Engineering*, ASCE, Reston, VA, 126–133.

Manning, C., Raghawan, P., and Schutze, H. (2009). *An introduction to information retrieval*, Cambridge University Press, Cambridge, U.K.

Manning, C., and Schutze, H. (1999). *Foundations of statistical natural language processing*, MIT Press, Cambridge, MA.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). "Efficient estimation of word representations in vector space." *Proc., Int. Conf. Learning Representations (ICLR)*, NEC Laboratories America, Princeton, NJ.

Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., and Khudanpur, S. (2010). "Recurrent neural network based language model." *Proc., 11th Annual Conf. of the Int. Speech Communication Association (INTERSPEECH 2010)*, International Speech Communication Association (ISCA), Baixas, France, 1045–1048.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). "Distributed representations of words and phrases and their compositionality." *Adv. Neural Inform. Process. Syst.*, 3111–3119.

Mikolov, T., Yih, W., and Zweig, G. (2013c). "Linguistic regularities in continuous space word representations." *Proc., 2013 Conf. North American Chapter of the Association for Computational Linguistics—Human Language Technologies (NAACL-HLT-2013)*, Association for Computational Linguistics, Stroudsburg, PA, 746–751.

Nawari, O. N. (2012). "Automating codes conformance." *J. Archit. Eng.*, 10.1061/(ASCE)AE.1943-5568.0000049, 315–323.

Pawar, Y. P., and Gawande, S. H. (2012). "A comparative study on different types of approaches to text categorization." *Int. J. Mach. Learn. Comput.*, 2(4), 423–426.

Pedregosa, F., et al. (2011). "Scikit-learn: Machine learning in python." *J. Mach. Learn. Res.*, 12, 2825–2830.

Ramage, D., Hall, D., Nallapati, R., and Manning, C. D. (2009). "Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora." *Proc., 2009 Conf. on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Stroudsburg, PA, 248–256.

Rehurek, R., and Sojka, P. (2010). "Software framework for topic modelling with large corpora." *Proc., LREC 2010 Workshop on New Challenges for NLP Frameworks*, European Language Resources Association (ELRA), Paris, 45–50.

Salama, D., and El-Gohary, N. (2013). "Semantic text classification for supporting automated compliance checking in construction." *J. Comput. Civ. Eng.*, 10.1061/(ASCE)CP.1943-5487.0000301, 04014106.

Sebastiani, F. (2002). "Machine learning in automated text categorization." *J. ACM Comput. Surv.*, 34(1), 1–47.

Song, M. H., Lim, S. Y., Kang, D. J., and Lee, S. J. (2005). "Automatic classification of web pages based on the concept of domain ontology." *Proc., 12th Asia-Pacific Software Engineering Conf.*, IEEE, Washington, DC, 15–17.

Sorower, M. S. (2010). *A literature survey on algorithms for multi-label learning*, Oregon State Univ., Corvallis, OR.

Spyromitros, E., Tsoumakas, G., and Vlahavas, I. (2008). "An empirical study of lazy multilabel classification algorithms." *Lecture notes in computer science (LNCS)*, Springer, Berlin, 401–406.

Tan, X., Hammad, A., and Fazio, P. (2010). "Automated code compliance checking for building envelope design." *J. Comput. Civ. Eng.*, 10.1061/(ASCE)0887-3801(2010)24:2(203), 203–211.

Tsoumakas, G., and Katakis, I. (2007). "Multi-label classification: An overview." *Int. J. Data Warehouse. Min.*, 3(3), 1–13.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2010). "Mining multi-label data." *Data mining and knowledge discovery handbook*, Springer, Berlin, 667–685.

Vogrinčič, S., and Bosnić, Z. (2011). "Ontology-based multi-label classification of economic articles." *Comput. Sci. Inf. Syst.*, 8(1), 101–119.

Waraporn, P., Meesad, P., and Clayton, G. (2010). "Ontology-supported processing of clinical text using medical knowledge integration for multi-label classification of diagnosis coding." *Int. J. Comput. Sci. Inf. Security*, 7(3), 30–35.

Wei, G. Y., Yu, J., Ling, Y., and Liu, J. (2006). "Design and implementation of an ontology algorithm for web documents classification." *Lecture notes in computing science (LNCS)*, Springer, Berlin, 649–658.

Wijewickrema, C. M., and Gamage, R. (2013). "An ontology based fully automatic document classification system using an existing semi-automatic system." *Proc., IFLA 2013 World Library and Information Congress*, International Federation of Library Associations and Institutions (IFLA), Netherlands.

Yang, X. Q., Sun, N., Zhang, Y., and Kong, D. R. (2008). "General framework for text classification based on domain ontology." *Proc., 3rd Int. Workshop Semantic Media Adaptation and Personalization (SMAP)*, IEEE, Washington, DC, 147–152.

Yu, F., Zheng, D. Q., Zhao, T. J., Li, S., and Yu, H. (2006). "Text classification based on a combination of ontology with statistical method." *Proc., 5th Int. Conf. Machine Learning and Cybernetics*, IEEE, Washington, DC, 13–16.

Zhang, J., and El-Gohary, N. (2013). "Semantic NLP-based information extraction from construction regulatory documents for automated compliance checking." *J. Comput. Civ. Eng.*, 10.1061/(ASCE)CP.1943-5487.0000346, 04015014.

Zhang, J., and El-Gohary, N. (2015). "Automated information transformation for automated regulatory compliance checking in construction." *J. Comput. Civ. Eng.*, 10.1061/(ASCE)CP.1943-5487.0000427, B4015001.

Zhang, M. L., and Zhou, Z. H. (2007). "ML-KNN: A lazy learning approach to multi-label learning." *Pattern Recognit.*, 40(7), 2038–2048.

Zhong, B. T., Ding, L. Y., Luo, H. B., Zhou, Y., Hu, Y. Z., and Hu, H. M. (2012). "Ontology-based semantic modeling of regulation constraint for automated construction quality compliance checking." *Automat. Constr.*, 28(2012), 58–70.

Zhou, P., and El-Gohary, N. (2015). "Domain-specific hierarchical text classification for supporting automated environmental compliance checking." *J. Comput. Civ. Eng*, 10.1061/(ASCE)CP.1943-5487.0000513, 04015057.