Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

Extending Building Information Models Semi-Automatically Using Semantic Natural Language Processing Techniques

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

Abstract

Automated compliance checking (ACC) of building designs requires automated extraction of information from building information models (BIMs). However, current Industry Foundation Classes (IFC)-based BIMs provide limited support for ACC, because they lack the necessary information that is needed to perform compliance checking (CC). In this paper, the authors propose a new method for extending the IFC schema to incorporate CC-related information, in an objective and semi-automated manner. The method utilizes semantic natural language processing (NLP) techniques and machine learning techniques to extract concepts from documents that are related to CC (e.g., building codes) and match the extracted concepts to concepts in the IFC class hierarchy. The proposed method includes a set of methods/algorithms that are combined into one computational platform: (1) a method for concept extraction that utilizes pattern-matching-based rules to extract regulatory concepts from CC-related regulatory documents, (2) a method for concept matching and semantic similarity (SS) assessment to select the most related IFC concepts to the extracted regulatory concepts, and (3) a machine learning classification method for predicting the relationship between the extracted regulatory concepts and their most related IFC concepts. The proposed method enables the extension of the IFC schema, in an objective way, using any construction regulatory document. To test and evaluate the proposed method, two chapters were randomly selected from the International Building Code

[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; Tel: +1-217-333-6620; Fax: +1-217- 265-8039.

(IBC) 2006 and 2009. Chapter 12 of IBC 2006 was used for training/development and Chapter 19 of IBC 2009 was used for testing and evaluation. Comparing to manually-developed gold-standards, 91.7% F1-measure, 84.5% adoption rate, and 87.9% precision were achieved for regulatory concept extraction, IFC concept selection, and relationship classification, respectively.

CE Database subject headings: Project management; Construction management; Building information models; Information management; Computer applications; Artificial intelligence.

Author keywords: Automated compliance checking; Automated information extraction; Building information modeling (BIM); Natural language processing; Semantic systems; Knowledge modeling; IFC extension; Automated construction management systems.

Introduction

In comparison to the traditional manual process, automated compliance checking (ACC) of construction projects is expected to reduce the time, cost, and errors of compliance checking (Nguyen and Kim 2011; Kasim et al. 2013; Zhang and El-Gohary 2013). ACC has been pursued both in academia and industry. Among the existing construction regulatory ACC efforts, building information models (BIMs) were mostly utilized as the representation of project information (Eastman et al. 2009). Due to the lack of a fully developed all-inclusive BIM data/information schema that can sufficiently represent project information for ACC needs in different areas (e.g., fire safety, structural safety, and sustainability), existing ACC efforts typically went into one of two directions: either creating their own BIM or extending existing BIMs.

One of the important ACC projects, the Construction and Real Estate NETwork (CORENET) project of Singapore (Eastman et al. 2009), developed their own semantic objects in FORNAX library (i.e., a C++ library) to represent building design information. In the U.S., the General

Services Administration (GSA) design rule checking efforts defined the BIM modeling requirements in a well-documented building information modeling guide and allowed users to choose their own BIM authoring tool to define building models according to the guide (Eastman et al. 2009). In addition, many of the existing research efforts proposed or implemented the idea of extending BIMs to fulfill their specific information needs. For instance, Nguyen and Kim (2011) and Sinha et al. (2013) extended existing BIMs in Autodesk Revit Architecture by creating new project parameters such as “area of opening in firewall” and “width of opening in firewall”; Kasim et al. (2013) extended existing BIMs through adding new data items into industry foundation classes (IFC)-represented BIMs directly; Nawari (2011) proposed the development of appropriate Information Delivery Manuals (IDM) and Model View Definitions (MVDs) for the ACC domain to achieve the required level of detail on IFC-represented BIMs; and Tan et al. (2010) extended IFC in eXtensible markup language (ifcXML) to develop an extended building information modeling (EBIM) in XML.

These existing efforts to extend BIMs for ACC deepened the understanding of BIM modeling requirements for ACC. However, the model extension methods were mostly ad-hoc and subjective (i.e., relying on subjective developments or extensions by individual software developers and/or researchers); and the resulting models were usually still missing essential compliance checking (CC)-related information that are needed to achieve complete automation in CC (Martins and Monteiro 2013; Niemeijer et al. 2009). In addition, such ad-hoc and subjective developments/extensions lack generality and objectivity, which are essential to full automation of CC at a broader scale. As a result, a more generalized and objective method is needed to extend BIMs for facilitating ACC.

To address this gap, in this paper, a new method for extending the IFC schema with regulatory requirement information, in an objective and semi-automated manner, is proposed. The method utilizes semantic natural language processing (NLP) techniques and machine learning (ML) techniques to extract concepts from documents that are related to CC (e.g., building codes) and match the extracted concepts to concepts in the IFC class hierarchy to extend the IFC schema. The method includes developing a set of algorithms/methods and combing them into one computational platform: (1) a pattern matching-based concept extraction method utilizing a set of Part-Of-Speech (POS) patterns to extract regulatory concepts from CC-related documents, (2) a semantic similarity (SS)-based ranking method utilizing a newly-proposed equation to measure SS between concepts to identify the most related IFC concepts to the extracted regulatory concepts, and (3) a ML-based classifier to predict the relationship between the extracted regulatory concepts and their most related IFC concepts. This paper focuses on presenting each method/algorithm and their evaluation results.

Background

Building Information Modeling (BIM) and IFC

According to the National Building Information Model Standard Project Committee (National Institute of Building Sciences 2014), a building information model (BIM) is "a digital representation of physical and functional characteristics of a facility. BIM is believed to improve interoperability through structured information and coordinated information flow during a building life cycle and between different disciplines (Hamil 2012). However, although BIM is intended to be fully interoperable, in reality different BIM softwares and platforms are not yet realizing full compatibility and seamless information exchange hitherto, which prevents BIM from realizing its full potential (Young et al. 2009).

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

Standardization is a primary way to improve interoperability. The current main standardization efforts in BIM include Industry Foundation Classes (IFC) and the CIMSteel Integration Standards (CIS/2) (Isikdag et al. 2007). The IFC represent the main data model to describe building and construction industry data. The IFC schema specifications are written using the EXPRESS data definition language (ISO 10303-11 by the ISO TC 184/SC4 committee) (BuildingSmart 2014). The IFC schema is the data exchange standard to facilitate interoperability in the construction industry (BuildingSmart 2014). The CIS/2 is a product model and data exchange file format for structural steel project information (AISC 2014). Both IFC and CIS/2 models are defined using standard STEP description methods, which is the official "Standard for Exchange of Product model data" – ISO 10303. In contrast to CIS/2 which is focusing on modeling information of structural steel framework, IFC schema is designed to cover all subdomains and phases of building and construction industry. Thus, IFC attracted more attention in BIM research. IFC schema is neutral and platform independent. IFC schema is registered as ISO/PAS 16739 and is becoming an official international standard ISO/IS 16739.

Semantic Models and WordNet

In general, "semantics" studies the meanings of the words (Fritz 2006). A semantic model defines data/information entities and relationships between the entities (Hanis and Noller 2011). Specialization and decomposition are two main hierarchical relations in a semantic model. Specialization refers to the relationship between a superclass and its subclass (Dietrich and Urban 2011). Decomposition refers to the relationship between a whole object (class) and the parts that belong to the object (class) (Klas and Schrefl 1995; Böhms et al. 2009).

Ontology is the most widely explored and adopted type of semantic model. It is defined as "an explicit specification of a conceptualization", where "a conceptualization is an abstract simplified view of the world that we wish to represent for some purpose" (Gruber 1995; El-Gohary and El-Diraby 2010). An ontology models domain knowledge in the form of concept hierarchies, relationships (between concepts), and axioms to help define the semantic meaning of the conceptualization (El-Gohary and El-Diraby 2010). Ontological models were shown to improve the performance of various information processing tasks such as text classification (e.g., Zhou and El-Gohary 2014) and information extraction (e.g., Zhang and El-Gohary 2013; Soysal et al. 2010).

WordNet was frequently utilized in semantic research efforts. It is a large lexical database of English where the four types of POS words (nouns, verbs, adjectives, and adverbs) are grouped into sets of cognitive synonyms (synsets) (Fellbaum 2005). In WordNet, each of the four POS categories is organized into a subnet and the synsets are linked to each other using one or more of the following conceptual semantic and lexical relations: synonymy, hyponymy (sub-super or is-a relation), meronymy (part-whole relation), and antonymy (Fellbaum 2005). Because of the abundant, explicitly-defined and well-structured conceptual semantic relations between word senses in WordNet, WordNet has been widely used in semantic research, as a "lexical database" (Shehata 2009; Kamps et al. 2004), a "lexical dictionary" (Varelas et al. 2005), a "semantic dictionary" (Simpson and Dao 2010), or a "domain-independent background knowledge model" (Suchanek et al. 2007). The lexical relations in WordNet can assist in semantic text processing. The hyponymy and meronymy relations in WordNet correspond well to the is-a and part-whole relations in semantic models. In addition, a synonymy relation carries an "equivalency" relation between semantic classes.

In spite of the generally accepted assumption that semantic relations are domain dependent (Orna-Montesinos 2010), WordNet, as a resource for providing semantic relations across different domains, is still widely used for domain specific text/knowledge processing tasks. This may be caused by a lack of domain-specific semantic relation resource with a comparative coverage as that can be achieved using WordNet. For example, in OmniClass (a popular classification system for the construction industry), common concepts in building codes (e.g., “cross ventilation”) may be difficult to find matches. In contrast, the dictionary-level coverage in WordNet contains the semantic relations for each word, and enabling semantic analysis for any multi-term concept in a compositional manner. Although being few, WordNet has been used for text/knowledge analysis or processing in Architecture, Engineering and Construction domain, such as in (Orna-Montesinos 2010) and (Li 2010).

Semantic Similarity

Semantic Similarity (SS) is the conceptual/meaning distance between two entities such as concepts, words, or documents (Slimani 2013). SS plays an important role in information and knowledge processing tasks such as information retrieval (Rodrı´guez and Egenhofer 2003), text clustering (Song et al. 2014), and ontology alignment (Jiang et al. 2014).

SS could be quantitatively estimated using different measures. Some popular measures are: (1) Shortest Path Similarity, which utilizes the shortest path connecting two concepts in a taxonomy (i.e., concept hierarchy) (Resnik 1995); (2) Leacock-Chodorow Similarity, which utilizes the shortest path connecting two concepts in a taxonomy while penalizing long shortest path according to the depth of the taxonomy (Resnik 1995); (3) Resnik Similarity, which utilizes the information content measure from information theory to measure the information shared by two concepts using the information content of the two concepts’ least common subsumer (Resnik 1995); (4) Jiang-Conrath Similarity, which utilizes the information content of the two concepts in addition to that of their least common subsumer in the taxonomy; and (5) Lin Similarity, which utilizes the ratio between the information content of the least common subsumer (in the taxonomy) of the two concepts and the sum of the information contents of the two concepts. Shortest Path Similarity is simple and intuitive, it approximates the conceptual distance between concepts by the number of edges in-between. The main limitation of Shortest Path Similarity is its inability to take specificity of concepts into the measurement, which lead to the same similarity results between concept pairs at shallow level of a taxonomy and concept pairs at deep level of the taxonomy as long as the counts of number of edges for both concept pairs are the same. This limitation is compensated for in Leacock-Chodorow similarity by taking the maximum depth of the taxonomy into the measurement. Thus in Leacock-Chodorow similarity concept pairs deeper in the taxonomy (i.e. more specific) would have larger similarity score than concept pairs shallow in the taxonomy, with the number of edges being equal. Resnik Similarity, Jiang-Conrath Similarity, and Lin Similarity are information content-based similarity measures. They do not have the specificity problem of shortest path similarity because the information content reflects specificity by definition. Resnik Similarity sometimes is considered coarse because different concept pairs could have the same least common consumer. Jiang-Conrath Similarity and Lin Similarity improve upon Resnik Similarity by taking the information content of the concepts from the pair into measurement in addition to their least common consumer. The main limitation for information content-based measures, however, is the need of a corpus in addition to the taxonomy, which lead to different results based on different corpuses. When using the measures herein to evaluate SS, WordNet was widely used as the taxonomy.

Machine Learning Algorithms

In any machine learning (ML) application, different ML algorithms are usually tried out and tested. Some of the most commonly-used ML algorithms are summarized in Table 1.

Insert Table 1

Naïve Bayes is a simple statistical ML algorithm. It applies Bayes' rule to compute conditional probabilities of predictions given evidence. It is the simplest type of algorithm among the commonly-used ML algorithms. However, Naïve Bayes could outperform more complex learning algorithms in some cases (Domingos 2012). Perceptron is a linear learning algorithm where predictions are made based on a linear combination of feature vectors (Rosenblatt 1958). Perceptron is applicable to problems that are linearly separable. The application process of perceptron is iterative: a prediction vector is iteratively constructed based on each instance in the training dataset (Freund and Schapire 1999). Decision Tree ML algorithms use a tree to map instances into predictions. In a Decision Tree model, each non-leaf node represents one feature, each branch of the tree represents a different value for a feature, and each leave node represents a class of prediction. Decision Tree is a flexible algorithm that could grow with increased amount of training data (Domingos 2012). K-Nearest Neigbor (k-NN) is a similarity-based ML algorithm. K-NN predicts the class of an instance using the instance's k nearest instances by assigning it the majority class of those k instances' classes (Cover 1967; Domingos 2012). Support Vector Machines (SVM) is a kernel-based ML algorithm that has significant computational advantages over standard statistical algorithms. Kernel methods is a technique for constructing nonlinear features so that nonlinear functional relationships could be represented using a linear model. A linear model is much simpler comparing to a nonlinear model, both theoretically and practically, giving SVM its computational advantages (Cristianini and Shawe-Taylor 2000). SVM was found to outperform other ML algorithms in many applications such as text classification (e.g., Salama and El-Gohary 2013).

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

ML is one type of machine-based reasoning (i.e., inductive reasoning), where the various types of ML algorithms induct knowledge from input data (Domingos 2012). In any machine-based reasoning, successful reasoning depends on appropriate representations (Bundy 2013). What features should be used to represent the data in a ML problem is, thus, an important decision.

State of the Art and Knowledge Gaps

Several research efforts extended the IFC schema for various purposes, such as building life cycle management (Vanlande 2008), cost estimation (Ma et al. 2011a), enterprise historical information retrieval (Ma et al. 2011b), parametric bridge models information exchange (Ji et al. 2013), and virtual construction systems data sharing (Zhang et al. 2014). However, most of the extension efforts extended the schema in an arbitrary and ad-hoc manner, which lacks objectivity and generality. Despite the potential of using ontology alignment and ontology mapping techniques (using SSs) in developing a generalized IFC extension method, to the best of the authors' knowledge there was little empirical exploration of this approach. The work of Delgado et al. (2013) and Pan et al. (2008) are the closest to this approach.

Delgado et al. (2013) evaluated 15 ontology matching techniques in matching geospatial ontologies with BIM-related ontologies (including an ontology for IFC) to discover correspondences of concepts between each pair of ontologies (e.g., between CityGML ontology and IFC ontology). The 15 techniques were classified into three categories: string-based techniques, WordNet-based techniques, and matching systems techniques. The alignment between CityGML ontology and IFC ontology is conceptually and technically similar to extending IFC. The main evaluation metrics were precision, recall, and F1-measure. Precision was defined as the correctly found correspondences divided by the total number of correspondences found. Recall was defined as the correctly found correspondences divided by

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

the total number of correspondences that should be found. F1-measure was defined as the harmonic mean of precision and recall. In their experimental results: (1) String-based techniques showed the best performance (100% precision, 57.1% recall, and 23.2% F1-measure) among the three tested techniques; (2) Within the WordNet-based techniques, the synonym distance technique showed 41.6% precision, 14.2% recall, and 21.2% F-measure; and (3) Within the matching systems techniques, the Association Rule Ontology Matching Approach (AROMA) showed 40% precision, 5.7% recall, and 10% F-measure. These results show that further research is needed to investigate whether the use of other semantic relations in WordNet (such as hyponymy), in addition to synonymy, would result in higher levels of performance.

Pan et al. (2008) conducted semi-automated mapping of architectural, engineering, and construction (AEC) ontologies, including an IFC ontology, using relatedness analysis techniques. In their ontology mapping, three types of features were used to provide expert guidance: (1) corpus-based features: co-occurrence frequencies between two concepts, (2) attribute-based features: attribute value structures of ontologies, and (3) name-based features: stemmed terms of the concept names to use for direct term-based matching. Further research is needed to explore how different types of semantic relations among concepts could be leveraged in IFC concept mapping.

Proposed Method for Semi-Automated IFC Extension with Regulatory Concepts

The proposed method for IFC extension with regulatory concepts is semi-automated. It utilizes automation techniques in all tasks for reducing the required manual effort. The proposed method includes three phases (Fig. 1): regulatory concept extraction, IFC concept selection, and relationship classification.

Insert Figure 1

Regulatory Concept Extraction

Proposed Concept Extraction Approach

To conduct ACC in a fully automated way, all concepts related to regulatory requirements in a relevant regulatory document must be incorporated into a BIM schema (e.g., IFC schema). The regulatory concept extraction phase aims to automatically extract all concepts from a selected, relevant regulatory document. The proposed extraction method utilizes pattern-matching-based extraction rules. After all concepts are automatically extracted from a textual regulatory document, a user manually removes concepts that are not related (or irrelevant) to regulatory requirements (about one and half pages of words for one chapter).

Concept Extraction Rules

The left-hand side of a rule defines the pattern to be matched and the right-hand side defines the concept that should be extracted. The patterns are composed of POS features (i.e., POS tags). Fig. 2 shows an example concept extraction (CE) rule and its corresponding meaning. The rule could extract four term concepts like "thermally isolated sunroom addition." Ten selected POS tags from Penn Treebank tag set are also listed in Fig. 2. Only flattened patterns are utilized in the CE rules to avoid recursive parsing. This decision is in contrast to grammar-based recursive parsing where the same rule could be applied multiple times for exploring a hierarchical structure as parsing results, because the purpose of the rules herein is only for identifying base noun phrases instead of exploring the internal structures of the phrases. This is analogous to chunking instead of finding constituents. Flattened patterns are patterns that include only terminal symbols (i.e., symbols that cannot be further broken down), which are analogous to leaf nodes in a tree-like structure. For example, the following P1, P2, P3, and P4 patterns are flattened because they only contain POS tags (i.e., "NN", which is the POS tag for singular noun). Non-flattened patterns, on the other hand, are patterns that include non-terminal symbols (i.e., symbols that could be further broken down). For example, the “NN NP” pattern in rule R2 is non-flattened because it contains non-terminal symbols (i.e., “NP”, which is a phrase level tag for noun phrase that could be further broken down). Recursive parsing is avoided, in the proposed method, because: (1) recursive parsing increases time complexities of parsing algorithms. For example in Fig. 3, to match the pattern P4, the number of trials for applying rules R1 and R2 using recursive parsing are minimum 4 and maximum 8, higher than the number of trials for applying rules R1, R3, R4, and R5 using non-recursive parsing which are minimum 1 and maximum 4; and (2) recursive parsing is less flexible. For example, if only P1 and P3 should be matched while P2 and P4 should not, this is easy to achieve through applying R1 and R4 using non-recursive parsing, whereas it is difficult to achieve using recursive parsing.

Insert Figure 2

Insert Figure 3

<u>Development of POS Pattern Set</u>

The development of the set of POS patterns to use in the CE rules is conducted following the algorithm shown in Fig. 4. The algorithm is executed after a gold standard of regulatory concepts is created and after the POS tags for all sentences in the development text are generated. The development text is a sample of regulatory text (Chapter 12 of IBC 2006, in this paper) that is used to identify common POS patterns in the text for developing the POS pattern set. The algorithm incrementally processes concepts in the gold standard using two levels of loops: the outer loop accesses each sentence in the gold standard and the inner loop accesses each concept in the sentence being accessed. In the processing of each concept, the POS pattern for the concept is first tentatively collected into the POS pattern set. Then the POS pattern set is used to extract concepts from all sentences. The recall and F1-measure are then calculated for the result. If the recall and F1-measure increase comparing to the previous recall and F1-measure (without the tentatively added POS pattern), then the addition of the POS pattern into the POS pattern set is committed. This process iterates through all concepts in all sentences. The algorithm iteratively improved recall and F1-measure of extraction by incorporating more POS patterns.

Insert Figure 4

Exclusion Word Removal

Exclusion words are defined, here, as words (unigram, bigram, or multigram) that match certain POS patterns in the CE rules but should not be extracted as concepts. The POS tags of these exclusion words usually introduce ambiguity because they carry more than one lexical or functional category/meaning, which may introduce false positives (i.e., incorrectly extracted concepts) in concept extraction. For example, “VBG” (POS tag for both “verb gerund” and “present participle”) is useful to extract “verb gerund” concepts like “opening”, but it introduces false positives when incorrectly extracting “present participle” words like “having” as concepts. To avoid introducing false positives during concept extraction, an exclusion word list is used.

IFC Concept Selection

Proposed Concept Selection Approach

The IFC concept selection phase aims to (1) automatically find the most related concept(s) in the IFC schema (called F-concept(s) hereafter) to each extracted regulatory concept (called R-concept hereafter) and (2) accordingly, allow the user to select the F-concept(s) for each R-concept. In this paper, the extension of the IFC schema is an incremental process; each R-concept is added to the IFC schema one by one, incrementally. As a result, an R-concept that gets selected (and thus added) to the IFC schema becomes part of the schema (i.e., becomes an F-concept for the following automated selection step). The automated IFC concept selection method includes four steps/techniques (as shown in Fig. 5): (1) Step 1: stemming, which reduces words to their stems; (2) Step 2: term-based matching, which aims to find all F-concepts that share term(s) with an R-concept; (3) Step 3: semantic-based matching, which aims to find all semantically related F-concepts to an R-concept. Semantic-based matching is used, to add a deeper level of searching, if the term-based matching fails to find candidate concepts; and (4) Step 4: SS scoring and ranking, which measures the SS between each candidate F-concept (from Step 2 and Step 3) and the R-concept, and accordingly ranks all candidate F-concepts related to that one single R-concept for final F-concept user selection. The same process is repeated for all R-concepts and their related candidate F-concepts.

Insert Figure 5

<u>Stemming</u>

Stemming is utilized in both term-based and semantic-based matching. Concepts are stemmed before matching to avoid incorrect mismatching due to variant word forms (rather than variant meaning). For example, with stemming applied, “foot” could be matched to “feet” (the stem of “feet” is “foot”).

<u>Term-Based Matching</u>

For term-based matching, three types of matching are used, based on the following two heuristic rules, H1 and H2: (1) “First Term Term-Based Matching”: the first term in the R-concept is terminologically matched against all F-concepts to find related F-concepts, (2) “Last Term Term-Based Matching”: the last term in the R-concept is terminologically matched against all F-concepts to find related F-concepts, and (3) “First and Last Term Term-Based Matching”: the first and last terms in the R-concept are terminologically matched against all F-concepts to find related F-concepts. Which type of matching to use depends on two main factors: (1) the number of terms in the concept name of the R-concept, whether the concept name is unigram (i.e., concept name with only one term), bigram (i.e., concept name with two terms), or multigram (i.e., concept name with three or more terms); and (2) the types of POS patterns in the concept name of the R-concept, whether the pattern is "N" (i.e., a POS pattern with only one POS tag and the POS tag is a noun), "NN" (i.e., a POS pattern starting with a noun and ending with a noun), or "JN" [i.e., a POS pattern starting with a prenominal modifier (e.g., adjective) and ending with a noun]. The matching strategy is illustrated in Fig. 6. For example, the R-concept "interior space" is a bigram, and the POS pattern ("JJ NN") matches "JN", thus "Last Term Match" is used to find matching concepts that contain the term "space" (i.e., the last term in the R-concept).

- H1: The term that has a nominal POS tag (i.e., noun) is the primary meaning-carrying term in a multi-term concept name.
- H2: The terms that have non-nominal POS tags (e.g., "JJ") are the secondary meaning-carrying terms in a multi-term concept name, which add to or constrain the meaning as modifiers.

Insert Figure 6

Semantic-Based Matching

In semantic-based matching, the semantic relations of WordNet (Fellbaum 2005) are utilized to find concept matches beyond term-based matching. Three types of these relations are used: hypernymy, hyponymy, and synonymy. These three types were selected because they are most relevant to the superclass-subclass structure of the IFC class hierarchy. Hypernymy is a semantic relation where one concept is the hypernym (i.e., superclass) of the other. For example, "room" is a hypernym of "kitchen". Hyponymy is the opposite of hypernymy where one concept is the hyponym (i.e., subclass) of the other. For example, "kitchen" is a hyponym of "room". Synonymy is the semantic relation between different concepts who share the same meaning. For example, "gypsum board", "drywall", and "plasterboard" all share the same meaning of "a board made of gypsum plaster core bonded to layers of paper or fiberboard." Three types of matching are used, which semantically match the first term, the last term, or the first term and last term in the R-concept, respectively, against all F-concepts to find related F-concepts: "first term semantic-based matching", "last term semantic-based matching", and "first and last term semantic-based matching". To conduct the semantic matching, the hypernyms, hyponyms, and synonyms of the first/last term are determined, based on WordNet, and then term-matched against all F-concepts to find related F-concepts. Similar to term-based matching, which type of matching to use depends on (1) the number of terms in the concept name of the R-concept and (2) the POS pattern types in the concept name of the R-concept. The matching strategy is illustrated in Fig. 6. "Search" represents "semantic-based matching" in Figure 6.

<u>Semantic Similarity Scoring and Ranking</u>

The proposed SS scoring method follows heuristic rules H3, H4, and H5.

- H3: In a multi-term concept name, the contribution of each term's carried meaning to the meaning of the whole concept decreases from right to left; the first term contributes the least to the meaning of the whole concept.
- H4: The difference in length between two concept names (where the length is measured in number of terms) is indicative of the closeness of the two concepts in a concept hierarchy; the smaller the difference, the closer the two concepts are, and vice versa.

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

Sibling concepts are, thus, likely to have a small difference between their concept name lengths.

- H5: The length of a concept name is related to its level in a concept hierarchy. The shorter the length of a concept name is, the more general the concept is; and thus the higher its level in a concept hierarchy. The longer the length of a concept name is, the more specific the concept is; and thus the lower its level in a concept hierarchy. A superconcept is, thus, likely to have a shorter concept name length than its subconcept.

Based on these heuristic rules, Eq. (1) and Eq. (2) are proposed as two alternative functions for SS scoring, where SS_{RF1} and SS_{RF2} are the concept-level SS scores between an R-concept and an F-concept, SS_{RmFk} is the term-level SS score between the m^{th} term in the R-concept and the k^{th} term in the F-concept, m is the ordinal number for the term Rm in R-concept, k is the ordinal number for the term Fk in F-concept, L_F is the length of F-concept measured in number of terms, and L_R is the length of R-concept measured in number of terms.

$$SS_{RF1} = \frac{1}{L_F} \sum_{k=1}^{k=L_F} \frac{2k}{L_F(L_F+1)} SS_{RmFk} \tag{1}$$

$$SS_{RF2} = \frac{1}{|L_R - L_F| + 1} \sum_{k=1}^{k=L_F} \frac{2k}{L_F(L_F+1)} SS_{RmFk} \tag{2}$$

Any existing term pair SS measure, such as Shortest Path Similarity measure or Leacock-Chodorow Similarity measure, can be used (after testing) to compute SS_{RmFk}. In Eq. (1) and Eq. (2), each term-level SS score (i.e., SS_{RmFk}) is discounted using the factor $\frac{2k}{L_F(L_F+1)}$. This term-level discount factor is based on heuristic rule H3. The concept-level semantic similarity score between the R-concept and the F-concept (i.e., SS_{RF}) is determined by further discounting the summation of all discounted term-level SS scores (of all term pairs formed between the matching term of R-concept and each term of the F-concept). In Eq. (1), the concept-level discount factor is $\frac{1}{L_F}$, which linearly discounts the summation using the length of the F-concept. This discount favors concepts at higher levels in a concept hierarchy and follows heuristic rule H5 to identify higher-level concepts based on the lengths of concept names. In Eq. (2), the concept-level discount factor is $\frac{1}{|L_R - L_F|+1}$, based on the absolute length difference between the concept names of R-concept and F-concept. This discount favors concepts at similar levels in a concept hierarchy and follows heuristic rule H4.

Accordingly, the proposed SS scoring method is summarized in Fig. 5. Combinations of different concept-level SS scoring functions (i.e., Eq. 1 and Eq. 2) and term-level SS scoring functions (i.e., existing similarity measures such as Shortest Path Similarity) should be experimentally tested to select the best-performing combination. Separate testing is conducted for term-based matched F-concepts (i.e., F-concepts found using term-based matching, from Step 2) and semantic-based matched F-concepts (i.e., F-concepts found using semantic-based matching, from Step 3). The authors' experimental testing and results are presented and discussed in the Experimental Testing and Results section.

For SS ranking, all candidate F-concepts related to one single R-concept are ranked according to their SS scores, in order of decreasing score. A threshold value or a maximum permitted value is further used to filter the most related F-concept(s) among the candidate concepts. The threshold is the minimum SS score below which a candidate F-concept is considered semantically not related (and thus ineligible for selection for this R-concept). The maximum permitted value is a natural number (default is 1) that defines at most how many number of F-concepts could be selected for a single R-concept. Both, threshold value and maximum permitted value, are set by the user. For example, using term-based matching, many F-concepts were found to match “exterior wall” through the matching term “wall”, such as “wall” and “curtain wall”. Then, the SS scores were computed between “exterior wall” and each of the matched F-concepts, such as “wall and “wall”, and “wall” and “curtain wall”. The candidate F-concepts were ranked according to the SS scores and the highest scored candidates were automatically selected, according to the default maximum permitted value. If the maximum permitted value is set to 1 and Eq. (1) is used, “wall” is selected because of its highest SS score. Following a similar process, but using semantic-based matching, “railing” was selected as the match to “grab bars.”

Relationship Classification

Proposed Classification Approach

The relationship classification phase aims to classify the relationship between each pair of R-concept and F-concept. ML techniques are used to automatically predict the relationship between a concept pair based on the concept features of the pair.

Types of Relationships

Four types of relationships are considered (Table 2): (1) equivalent concept, indicating that the R-concept and the F-concept are equivalent; (2) superconcept, indicating that the R-concept is a superconcept of the F-concept; (3) subconcept, indicating that the R-concept is a subconcept of the F-concept; and (4) associated concept, indicating that the R-concept and the F-concept are associated (bidirectional relationship).

Insert Table 2

Types of Features

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

The authors identified the following initial set of eight features, which includes a mix of syntactic (i.e., related to syntax and grammar) and semantic (i.e., related to context and meaning) features (see Table 3): (1) RTermNum: the number of terms in the concept name of the R-concept, whether the concept name is unigram, bigram, or multigram; (2) RTermPOS: the type of POS pattern in the concept name of the R-concept, whether the pattern is “N”, “NN”, or “JN”; (3) RMatchType: the match type of R-concept, in terms of which term in the R-concept name matches a term in the F-concept name, whether it is the “first” or “last” term in the R-concept name; (4) RelMatchType: the match type between R-concept and F-concept, whether it is “term-based” match, “synonym”-based match (i.e., the matched term in the F-concept name is a synonym of the matching term in the R-concept name), “hyponym”-based match, or “hypernym”-based match; (5) FMatchType: the match type of F-concept, in terms of which term in the F-concept name matches the matching term in the R-concept name, whether it is “first”, “middle”, or “last”; (6) FTermNum: the number of terms in the concept name of the F-concept, whether the concept name is unigram, bigram, or multigram; (7) FTermPOS: the type of POS pattern in the concept name of the F-concept, whether the pattern is “N”, “NN”, or “JN”; and (8) DOM: the degree of match, which is represented as a Boolean value describing if the R-concept and the F-concept match term by term, with stemming applied, where one represents match and zero represents no match. These features were identified based on the following heuristic rules:

- H5 (see above).
- H6: The type of POS pattern in the name of a concept affects its meaning; and since the concept names are all noun phrases, the most distinguishing POS pattern is whether the concept has a modifier(s), and if yes, whether the modifier(s) is/are nominal (i.e., noun or noun sequences).

- H7: The match type, in terms of which term in each concept name is matched, affects the relationship between the matched concepts.
- H8: The match type, in terms of the type of relationship between the matched terms in both concepts, affects the relationship between the matched concepts.
- H9: If, in the same domain, two concept names match term by term (with stemming applied), then the two concepts are likely to be equivalent.

Table 4 shows some example concept pairs and their features. The final set of features is determined after conducting feature selection (as further discussed in the Experimental Testing and Results section).

Insert Table 3

Insert Table 4

Experimental Testing and Results

The proposed semi-automated IFC extension method was tested on extending the IFC class hierarchy (based on schema version IFC2X3_TC1) using regulatory concepts from IBC. Two chapters, Chapter 12 of IBC 2006 and Chapter 19 of IBC 2009, were randomly selected. Chapter 12 was used for: (1) developing the set of POS patterns for use in regulatory concept extraction (Phase 1), (2) selecting the best combination of SS scoring function and SS measure for IFC concept selection (Phase 2), and (3) training the ML classifier for relationship classification (Phase 3). Chapter 19 was used for testing and evaluating each of the following sub-methods/algorithms: regulatory concept extraction, IFC concept selection, and relationship classification. Each sub-method/algorithm was tested separately.

Regulatory Concept Extraction

<u>Gold Standard</u>

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

The gold standards of R-concepts for Chapter 12 of IBC 2006 and Chapter 19 of IBC 2009 were manually developed by the authors. An R-concept is a concept in regulatory documents that defines a "thing" (e.g., subject, object, abstract concept). The criteria for identifying R-concepts in the gold standard is that the concept should be as specific as possible (i.e., including all information related to the concept) without determiners and post modifiers. The longest span for each noun phrase according to this criteria, thus, was manually recognized and extracted as an R-concept. The longest span could be multiple terms (e.g., "continuously operated mechanical operation") or one term (e.g., "ventilation") depending on its appearance in text. For example, concepts in the list L1 were recognized and extracted from Sentence S1. The gold standards of Chapter 12 and Chapter 19 include 368 and 821 concepts, respectively. The concepts in the gold standard will be compared with concepts extracted by the algorithm for evaluating the algorithm in terms of precision and recall of extracted concepts.

- S1: "Wall segments with a horizontal length-to-thickness ratio less than 2.5 shall be designed as columns."
- L1: ['wall_segments', 'horizontal_length-to-thickness_ratio', 'columns']

Algorithm Implementation

The proposed regulatory concept extraction method was implemented in Python programming language (v.2.7.3). The Stanford Parser (version 3.4) (Toutanova et al. 2003) was selected and used to generate the POS tags for each word. The Stanford Parser used Penn Treebank tag set which includes 36 tags. Ten, out of the 36 tags, were used (shown in Fig. 2).

Evaluation

Regulatory concept extraction was evaluated in terms of precision, recall, and F1-measure. The definitions of these measures are similar to those in ontology matching except for the "found correspondences" are replaced by "extracted concepts." A higher recall is more important than precision because the overall method of IFC extension is semi-automated; precision errors could be detected and eliminated by the user during user concept selection.

Development Results and Analysis

The development of the set of POS patterns to use in the CE rules was conducted following the algorithm shown in Fig. 4. Fig. 7 shows the final set of POS patterns, which consists of 39 patterns. These 39 POS patterns were used as conditions for 39 CE rules, one POS pattern for one CE rule. For example, the pattern "JJ" "JJ" "JJ" "NN" was used for a CE rule which extracts three consecutive adjectives followed by a singular/mass noun as a concept, such as in the concept "minimum net glazed area."

Table 5 shows the performance of extracting R-concepts from the development text (Chapter 12 of IBC 2006). Through error analysis two sources of errors were found: (1) POS tagging error, which accounted for 38.1% of the errors. For example, "herein" was incorrectly tagged as "NN" instead of the correct tag "RB", and was, thus, incorrectly extracted; and (2) ambiguity of the POS tag "VBG" between gerund and present participle, which accounted for 61.9% of the errors. For example, "being" was a present participle thus not representing a concept, but it was extracted because the POS tag "VBG" was included in the POS patterns for representing gerund. While addressing error type (1) depends on improvement of POS taggers, error source (2) was addressed by adding the false positive present participle terms (e.g., "having," "being," "involving") to the exclusion word list. Membership in the exclusion word list prevents a word/phrase from being extracted in spite of matching a POS pattern in the set. The performance of regulatory concept extraction using the exclusion word list is shown in Table 5. Precision increased, from 93.4% to 97.1%, without decreasing recall.

Insert Figure 7

Insert Table 5

<u>Testing Results and Analysis</u>

The regulatory concept extraction algorithm was tested on Chapter 19 of IBC 2009. The precision, recall, and F1-measure are 89.4%, 94.2%, and 91.7%, and 88.7%, 94.2%, and 91.4%, with and without the use of exclusion word list, respectively. Table 6 shows the performance results. Through error analysis, when using the exclusion word list, four sources of errors were found: (1) POS tagging errors, which accounted for 20.0% of the errors. For example, "corresponding" was incorrectly tagged as "NN" (as opposed to "VBG"); and, thus, "force_level_corresponding" was incorrectly extracted as a concept; (2) ambiguity of POS tag "VBG" between gerund and present participle, which accounted for 7.9% of the errors. For example, "excluding" was incorrectly extracted as a concept because the POS tag for present participle was "VBG" (although it does not represent a meaningful nominal concept); (3) word continuation using hyphen, which accounted for 27.9% of the errors. For example, "pro_vide" was incorrectly extracted as a concept because the word continuation in "pro-vide" led to "pro" and "vide" be tagged as two words with the tags "JJ" and "NN"; and (4) missing POS patterns, which accounted for 44.3% of the errors. For example, "concrete_breakout_strength" and "breakout_strength_requirements" were incorrectly extracted as two concepts (instead of one concept, "concrete_breakout_strength_requirements") because the POS pattern *"JJ" "JJ" "JJ" "NN" "NNS"* was missing.

Preventing errors from source (1) requires improvement of POS taggers. Preventing errors from source (3) requires a better word continuation representation manner instead of using hyphen, in order to avoid confusion with hyphens used for conjoining noun modifiers. Preventing errors from sources (2) and (4) could be partially prevented by further developing the exclusion word list and POS pattern set, respectively. The use of the developed exclusion word list (to prevent errors from source (2)) prevented 6 instances of false positives and increased precision from 89.0% to 89.6%. More terms could be added, iteratively, to the exclusion word list to further enhance performance. Similarly, errors from source (4) could be prevented by adding more patterns to the POS pattern set until all possible POS patterns are included. While theoretically this POS pattern set is infinite (e.g., infinite number of "JJ" before a "NN"), in practice this POS pattern set is quite limited [e.g., words with more than 7 prenominal modifiers (e.g., white thin high strong stone north exterior ancient wall) are seldom (if not never) seen].

To test the effect of iterative development of the exclusion word list and POS pattern set, three more experiments were conducted to: (1) add the false positive present participle terms (identified as a result of initial testing) to the exclusion word list and use it in further testing; (2) add the missing POS patterns (identified as a result of initial testing) to the pattern set and use it in further testing; and (3) use both, the extended exclusion word list and the extended POS pattern set, in further testing. Table 7 shows the performance results of the three experiments. The results show that the use of the extended exclusion word list and the POS pattern set both improve the performance of concept extraction, with the latter showing a larger improvement.

Insert Table 6

Insert Table 7

IFC Concept Selection

<u>Gold Standard</u>

The gold standards of F-concepts for Chapter 12 of IBC 2006 and Chapter 19 of IBC 2009 were manually developed by the authors. The F-concepts were initially identified using the matching and ranking algorithms and then manually filtered. The gold standards of Chapter 12 and Chapter 19 include 343 and 588 F-concepts, respectively.

Algorithm Implementation

The proposed IFC concept selection method and algorithms were implemented in Python programming language (v.2.7.3). The Porter Stemmer (Porter 1980) was used for stemming. The “re” (regular expression) module in python was utilized to support the matching algorithms. The hypernymy, hyponymy, and synonymy relations in WordNet were utilized through the Natural Language Toolkit (NLTK) (Bird et al. 2009) WordNet interface in python.

Evaluation

IFC concept selection was evaluated in terms of adoption rate. Adoption rate is defined as the number of automatically selected F-concepts that were adopted divided by the total number of automatically selected F-concepts.

Development Results and Analysis

For term-based matched F-concepts, Table 8 shows the results of testing combinations of different concept-level SS scoring functions and term-level SS scoring functions. Table 9 shows some example concepts that were extracted and matched using the different combinations. For concept-level SS scoring, Eq.1 and Eq. 2 were tested. As shown in Table 8, Eq. (1) consistently outperformed Eq. (2). Eq. (1) prefers shorter F-concepts and, thus, tends to select F-concepts that are higher in the concept hierarchy (most likely a superclass). In comparison, Eq. (2) prefers F-concepts with similar length to the R-concept and, thus, tends to select F-concepts that are at a similar level in the concept hierarchy to the R-concept. However, an F-concept located at a similar level to the R-concept may deviate a lot in meaning because concepts at similar level in a concept hierarchy could belong to different branches of the hierarchy. A matched higher-level F-concept, thus, usually has higher relatedness to the R-concept than a matched similar-level F-concept. For example, using Shortest Path Similarity (for term-level SS scoring), Eq. (1) resulted in matching of "net_free_ventilating_area" and "quantity_area", whereas Eq. (2) resulted in the matching of "net_free_ventilating_area" and "annotation_fill_area_occurrence". "Quantity_area" was correctly a superconcept of "net_free_ventilating_area" and was adopted. On the other hand, the meaning of "annotation_fill_area_occurrence" was far from that of "net_free_ventilating_area" despite being at a similar level in the concept hierarchy. Based on these experimental results, Eq. (1) was selected for concept-level SS scoring for term-based matched F-concepts.

For term-level SS scoring, the following five existing SS measures were tested: Shortest Path Similarity, Jiang-Conrath Similarity, Leacock-Chodorow Similarity, Resnik Similarity, and Lin Similarity (see Background section). The Shortest Path Similarity is the simplest among the five tested measures, and achieved the best adoption rate of 86.5%. The Shortest Path Similarity and Leacock-Chodorow Similarity are based on shortest path between two concepts in a taxonomy. The other three SS measures are based on information content of the two concepts' least common subsume (i.e., the lowest-level concept that is a superconcept of both concepts). The performance drop from the Shortest Path Similarity to the other similarity measures (except for Leacock_Chodorow Similarity) shows the advantage of a shortest path measure in comparison to an information content of the least common subsumer measure. Empirically, this is because the length of path between two concepts is more distinctive than the information content of their least common subsumer. For example, in the concept hierarchy of Fig. 8, the shortest paths between C2 and C5, C4 and C5, C7 and C5 are different, but their least common subsumers are all the same (i.e., C1). For shortest path measures, the Leacock-Chodorow Similarity takes the depth of the taxonomy into consideration, in addition to the use of shortest path. The performance drop from the Shortest Path Similarity to the Leacock-Chodorow Similarity indicates that the absolute taxonomy depth is not a distinctive feature in the context of concept matching. Based on these experimental results, the Shortest Path Similarity was selected for term-level SS scoring for term-based matched F-concepts.

Insert Figure 8

Insert Table 8

Insert Table 9

For semantic-based matched F-concepts, Table 10 shows the results of testing combinations of different concept-level SS scoring functions and term-level SS scoring functions. Table 11 shows some examples of concepts that were extracted and matched using the different combinations. As shown in Table 10, for concept-level SS scoring, Eq. (1) and Eq. (2) did not show any variability in performance. Since both functions performed equally, for consistency with term-based matching of F-concepts, Eq. 1 was selected for concept-level SS scoring for semantic-based matched F-concepts.

For term-level SS scoring, the Shortest Path Similarity outperformed all other SS measures. This is consistent with the results obtained for term-based matched F-concepts. Based on the experimental results, the Shortest Path Similarity was selected for term-level SS scoring for semantic-based matched F-concepts.

Thus, the same term-level SS scoring function (Shortest Path Similarity) and concept-level SS scoring function (Eq. 1) were selected for both term-based matching and semantic-based matching algorithms. This shows consistency of performance across both types of matching.

Insert Table 10

Insert Table 11

<u>Testing Results and Analysis</u>

The proposed IFC concept selection method and algorithms [using Eq. (1) and Shortest Path Similarity] were tested in automatically selecting F-concepts for the extracted R-concepts (from Phase I). The testing results are summarized in Table 12. The total adoption rate is 84.5%. The adoption rates for term-based and semantic-based matched F-concepts are 84.8% and 82.7%, respectively, both which are close to the training performance (87.1% and 82.5%, respectively). This shows initial stability in the performance of the proposed IFC concept selection method.

Insert Table 12

Relationship Classification

<u>Gold Standard</u>

The aim of the classifier is to predict the relationship between each pair of R-concept and F-concept. Two gold standards, one for training and one for testing, were manually developed by the authors and three other graduate students in construction management. The training and testing gold standards included pairs of concepts from Chapter 12 of IBC 2006 and Chapter 19 of IBC 2009, respectively. The training data set was used for feature selection, ML algorithm selection, and classifier training, and the testing data set for evaluating the classifier's performance. In each gold standard, the relationship between each R-concept and F-concept was defined. Four types of relationships were defined, as per Table 2.

<u>Algorithm Implementation</u>

The proposed relationship classification algorithms were developed and tested in Waikato Environment for Knowledge Analysis (Weka) data mining software system (Hall et al. 2009). A program for generating the ML features was developed using Python programming language (v.2.7.3). The following ML algorithms were tested: (1) weka.classifiers.bayes.NaiveBayes for Naïve Bayes; (2) weka.classifiers.trees.J48 for Decision Tree; (3) weka.classifiers.lazy.IBk for k-NN; and (4) weka.classifiers.functions.SMO for SVM. Tenfold cross-validation was applied to each training experiment, which randomly split the data to training subset and testing subset ten times and averaged the results from the ten trials of training and testing.

Evaluation

Relationship classification was evaluated in two ways: (1) the performance across all relationships was evaluated, together, in terms of precision, and (2) the performance for each type of relationship was evaluated, separately, in terms of precision, recall, and F1-measure. In the first case, precision is defined as the number of correctly classified concept pairs divided by the total number of classified concept pairs. In the second case, precision is defined as the number of correctly classified concept pairs in a relationship type divided by the total number of concept pairs that are classified into that relationship type. Recall is defined as the correctly classified concept pairs in a relationship type divided by the total number of concept pairs that should be classified into that relationship type. F1-measure is the harmonic mean of precision and recall.

ML Algorithm Selection, Feature Selection, and Classifier Training

The training data set was used for feature selection and classifier training. The results of testing the four ML algorithms are summarized in Table 13. While three out of the four ML algorithms achieved a precision greater than 85%, k-NN achieved the best precision of 90.98%. Decision Tree ranked second in performance, with 86.07% precision.

A "leave-one-out" feature analysis was used for feature selection. Feature selection, in this paper, aims at selecting – based on performance – a subset (or the full set) of the complete/initial feature set (the eight features, see Table 3) for use in representing the concepts. The "leave-one-out" feature analysis is a method to analyze the contribution of each feature by comparing the performance with and without that feature. The analysis was conducted using the top-two performing ML algorithms (Decision Tree and k-NN). The feature analysis results are summarized in Table 14. The bold highlighted values indicate the precision values that outperformed the baseline precision (underlined, where all eight features were used). The results show that four out of the eight features (RTermNum, RTermPOS, RelMatchType, FTermNum) were not discriminating when using Decision Tree, and one out of the eight features (FTermNum) was not discriminating when using k-NN. Using only the discriminating features (i.e., Features RMatchType, FMatchType, FTermPOS, and DOM for Decision Tree, and Features RTermNum, RTermPOS, RMatchType, RelMatchType, FMatchType, FTermPOS, and DOM for k-NN), Decision Tree achieved a precision of 87.43% and k-NN achieved a precision of 91.26%. This difference shows that, in comparison to Decision Tree, k-NN was able to achieve a higher performance with a larger feature size. This may indicate that the additional features used by k-NN provided better discriminating ability to the classifier. As such, based on the experimental results, the above-mentioned seven discriminating features and the k-NN algorithm were selected for training the classifier.

The results also show that the following four features were discriminating for both algorithms: RMatchType, FMatchType, FTermPOS, and DOM. DOM was discriminating because a term-by-term match could provide a strong indication of concept equivalency. The fact that RMatchType and FMatchType were discriminating shows that the arrangement of terms could

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

affect the meanings of concepts and that the locations of the matching terms in a concept pair could affect the relationship between the two concepts in the pair. In addition to these four features, the following three features were discriminating for k-NN: RTermNum, RTermPOS, and RelMatchType. The fact that these features were discriminating in k-NN but not in Decision Tree may attributed to the different types of ML algorithms. More importantly, the fact that the RelMatchType is discriminating shows that the semantic features could benefit the task of concept relationship classification and result in further improvement of precision.

Insert Table 13

Insert Table 14

<u>Testing Results and Analysis</u>

The testing data set was used for testing and evaluating the performance of the classifier. The testing results are summarized in Table 15. The overall precision across all relationships is 87.94%. This is close to the overall training precision (90.98%), which shows the initial stability in the performance of the relationship classifier. The subconcept relationship type achieved the best precision of 93.4% and best recall of 93.4%. The analysis of the results shows that in many cases the R-concept was a bigram or multigram (e.g., “structural concrete”) whose last term matched with the only term in a unigram F-concept (e.g., “concrete”). This pattern has a strong predictive effect. Comparing to the subconcept relationship type, the superconcept relationship type shares a similar pattern but did not achieve a performance as high. The precision and recall for the superconcept relationship type were 88.5% and 75.4%, respectively. One observation was that the classifier tends to prefer subconcept relationship types over superconcept relationship types, when both the R-concept and the F-concept were bigram or multigram. For example, there were six cases where a superconcept relationship was incorrectly classified as a subconcept

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

relationship, but zero cases where a subconcept relationship was incorrectly classified as a superconcept relationship. This could be due to the fact that there were only two instances of bigram/multigram concept pairs with superconcept relationship in the training data set. The equivalent relationship type achieved a precision of 91.9% and recall of 86.1%. The associated relationship type achieved a precision of 62.5% and recall of 80.0%, which is the lowest among the four types of relationships. This is probably because: (1) the size of the training data was limited for this relationship type, and (2) the associated relationship includes more semantic types than the other types of relationships and has more variability in the expression of concepts. Thus, while the data set might provide enough variability for concepts related to the other relationship types, the associated relationship may require more data. Overall, the precision is 87.94%, which is considered a good performance [within the range of 80% to 90% (Spiliopoulos et al. 2010)].

Insert Table 15

Limitations and Future Work

Two limitations of this work are acknowledged, which the authors plan to address as part of their ongoing/future research. First, due to the large amount of manual effort required in developing the gold standard for each phase, the proposed method was only tested on one Chapter of IBC 2009. Similar good performance is expected on other chapters of IBC and other regulatory documents. However, different performance results might be obtained due to the possible variability of contents across different chapters of IBC 2009 or across different types of regulatory documents. As such, in their future work, the authors plan to test the proposed method on more chapters of IBC 2009 and on other types of regulatory documents (e.g., EPA regulations). Second, only unigram (single terms) semantic-based matching was used for finding

Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

semantically related F-concepts to an R-concepts. While the combinatorial nature of term meanings [i.e., the meanings of single terms (e.g., "exterior" and "door") in a concept name are combined to form the overall meaning of the whole concept (e.g., "exterior door")] renders this unigram method effective, there may be cases where bigram (pairs of terms) or multigram (groups of three or more terms) matching could be effective. As such, in future work, the authors plan to extend the semantic-based matching method to incorporate semantic relations between bigram and multigram to test whether such bigram or mutligram considerations could further improve the performance of concept matching.

Contributions to the Body of Knowledge

This study contributes to the body of knowledge in four main ways. First, this research offers a method for automated concept extraction that utilizes POS-pattern-matching-based rules to extract regulatory concepts from natural language regulatory documents. The set of POS patterns that was developed captures natural language knowledge, which allows for the recognition of concepts based on the lexical and functional categories of their terms. The pattern set includes only flattened patterns to avoid recursive parsing, which allows for efficient computation. The set of POS patterns are also generalized, and thus can be used to extract concepts in other domains. Second, this research offers a matching-based method for identifying and selecting the most related IFC concepts to the extracted regulatory concepts. The proposed method leverages both syntactic and semantic knowledge, which allows for the recognition of related concept pairs based on the syntactic and semantic similarities of their terms. As part of this method, two new concept-level semantic similarity (SS) scoring functions are offered. In the context of schema extension, existing SS scoring functions allow for measuring SS at the term-level. These proposed two functions further allow for measuring SS at the concept-level. Third, this research offers an automated machine learning classification method for classifying the relationships between the extracted regulatory concepts and their most related IFC concepts. The classification results show that semantic features could benefit the task of relationship classification and result in further improvement of precision. The proposed method is also generalized and can be used to classify the relationships between any two concepts, based on eight syntactic and semantic features of their terms, into the following four types: “equivalent concept”, “superconcept”, “subconcept”, and “associated concept”. Fourth, the experimental results show that the three proposed methods could be effectively combined in a sequential way for extending the IFC schema with regulatory concepts from regulatory documents. This offers a new method for objectively extending the IFC schema with domain-specific concepts that are extracted from natural language documents. The proposed combined method is also generalized and can be used to extend the IFC schema with other types of concepts (e.g., environmental concepts) from other types of documents (e.g., environmental documents) or to extend other types of class hierarchies (e.g., of an ontology) in the construction domain or in other domains.

Conclusions

This paper presented a new method for extending the IFC schema with regulatory concepts from relevant regulatory documents for supporting automated compliance checking. The proposed method utilizes semantic natural language processing (NLP) techniques and machine learning techniques, and is composed of three primary methods that are combined into one computational platform: (1) a method for concept extraction that utilizes POS-pattern-matching-based rules to extract regulatory concepts from regulatory documents, (2) a method for identifying and selecting the most related IFC concepts to the extracted regulatory concept, which utilizes term-based and semantic-based matching algorithms to find candidate related IFC concepts and a semantic similarity (SS) scoring and ranking algorithm to measure the SS between each candidate IFC concept a regulatory concept, and (3) a machine learning classification method for predicting the relationship between the extracted regulatory concepts and their most related IFC concepts based on the syntactic and semantic features of their terms. The proposed IFC extension method was evaluated on extending the IFC schema with regulatory concepts from Chapter 19 of IBC 2009. Each of the three methods were evaluated separately, and achieved 91.7%, 84.5%, and 87.94% F1-measure, adoption rate, and precision, respectively. The performance results indicate that the proposed IFC extension method is potentially effective. The results also show that semantic features of the concept terms and their interrelationships are helpful in IFC extension and result in performance improvement. In their future work, the authors plan to: (1) test the proposed method on other chapters of IBC 2009 and other construction regulatory documents (e.g., EPA regulations); and (2) extend the semantic-based matching method to incorporate semantic relations between bigram and multi-term to test whether such bigram or multigram considerations could further improve the performance of concept matching.

Acknowledgement

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

AISC. (2014). “Technology integration.” <http://www.aisc.org/content.aspx?id=26044>. (Aug 12, 2014).

Bird, S., Klein, E., and Loper, E. (2009). “Natural language processing with Python.” O’Reilly Media Inc., Sebastopol, CA.

Böhms, M., Bonsma, P., Bourdeau, M., and Kazi, A.S. (2009). “Semantic product modelling and configuration: challenges and opportunities.” *J. Inf. Techno. Constr.*, 14, 507-525.

BuildingSmart. (2014). “Industry Foundation Classes (IFC) data model.” < http://www.buildingsmart.org/standards/ifc/model-industry-foundation-classes-ifc> (Aug 12, 2014).

Bundy, A. (2013). “The interaction of representation and reasoning.” *Proc., R. Soc. A*, 469(2157), 1-18.

Cover, T.M. (1967). “Nearest neighbor pattern classification.” *IEEE Transactions on Information Theory*, 13(1), 21-27.

Cristianini, N., and Shawe-Taylor, J. (2000). “An introduction to support vector machines and other kernel-based learning methods.” Cambridge University Press, 1st edition, Cambridge, U.K.

Dietrich, S.W., and Urban, S.D. (2011). “Fundamentals of object databases: object-oriented and object-relational design.” *Synthesis Lectures on Data Management*, Morgan & Claypool Publishers, San Rafael, California.

Domingos, P. (2012). “A few useful things to know about machine learning.” *Communications of the ACM*, 55(10), 78-87.

Delgado, F., Martínez-González, M.M., and Finat, J. (2013). “An evaluation of ontology matching techniques on geospatial ontologies.” *Int. J. Geogr. Inf. Sci.*, 27(12), 2279-2301.

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

El-Gohary, N.M., and El-Diraby, T.E. (2010). “Domain ontology for processes in infrastructure and construction.” *J. Constr. Eng. Manage.*, 136(7), 730–744.

Fellbaum, C. (2005). “WordNet and wordnets.” In: Brown, Keith et al. (eds.), *Encyclopedia of Language and Linguistics*, Second Edition, Oxford: Elsevier, 665-670.

Freund, Y., and Schapire, R. (1999). “Large margin classification using the perceptron algorithm.” *Mach. Learn.*, 37(3), 277-296.

Fritz, D. (2006). “The Semantic Model: A basis for understanding and implementing data warehouse requirements.” <http://www.tdan.com/view-articles/4044> (Aug 12, 2014).

Gruber, T.R. (1995). “Toward principles for the design of ontologies used for knowledge sharing.” *Int. J. Hum.-Comput. St.*, 43, 907-928.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I.H. (2009). “The WEKA data mining software: an update.” *SIGKDD Explor.*, 11(1), 10-18.

Hamil, S. (2012). “Building information modelling and interoperability.” < http://www.thenbs.com/topics/bim/articles/bimAndInteroperability.asp> (Aug 12, 2014).

Hanis, T., and Noller, D. (2011). “The role of semantic models in smarter industrial operations.” *Oper.*, IBM Corporation, Armonk, New York.

Isikdag, U., Aouad, G., Underwood, J, and Wu, S. (2007). “Building information models: a review on storage and exchange mechanisms.” *Proc., 24th W78 Conf. & 5th ITCEDU Workshop & 14th EG-ICE Workshop, Bringing ITC Knowledge to Work*, International Council for Research and Innovation in Building and Construction (CIB), Rotterdam, Netherlands, 135-144.

Ji, Y., Borrmann, A., Beetz, J., and Obergrießer, M. (2013). ”Exchange of parametric bridge models using a neutral data format.” *J. Comput. Civ. Eng.*, 27(6), 593–606.

Jiang, Y., Wang, X, and Zheng, H. (2014). "A semantic similarity measure based on information distance for ontology alignment." *Inf.. Sci.*, 278(2014), 76-87.

Kasim, T., Li, H., Rezgui, Y., and Beach, T. (2013). "Automated sustainability compliance checking process: proof of concept." *Proc., 13th Int. Conf. Constr. App. Vir. Real.*, Teesside University, Tees Valley, UK, 11-21.

Kamps, J., Marx, M., Mokken, R.J., and Rijke, M. (2004). "Using WordNet to measure semantic orientations of adjectives." *Proc., LREC-04,* European Language Resources Association (ELRA), Paris, France, 1115-1118.

Khemlani, L. (2005). "CORENET e-PlanCheck: Singapore's automated code checking system." AECbytes "Building the Future" Article, <http://www.novacitynets.com/pdf/aecbytes_20052610.pdf> (Aug 12, 2014).

Klas, W., and Schrefl, M. (1995). "Metaclasses and their application - Data model tailoring and database integration." *Lect. Notes. Comput. Sc.*, 943, Springer-Verlag, Berlin Heidelberg, Germany, 1-7.

Li, T. (2010). "Practice and exploration of ontology creation algorithms." < http://www.cs.ubc.ca/~carenini/TEACHING/CPSC503-14/FINAL-REPORTS-10/CPSC503_Project_Report_Tianyu.pdf > (Mar. 21st, 2015).

Ma, Z., Lu, N., and Song, W. (2011a). "Identification and representation of information resources for construction firms." *Adv. Eng. Inform..*, 25(4), 2011, 612-624.

Ma, Z., Wei, Z., Song, W, Lou, Z. (2011b). "Application and extension of the IFC standard in construction cost estimating for tendering in China." *Autom. Constr.*, 20(2), 196-204.

Martins, J.P., and Monteiro, A. (2013). "LicA: A BIM based automated code-checking application for water distribution systems." *Autom. Constr.*, 29(2013), 12–23.

National Institute of Building Sciences. (2014). “National BIM standard – United States Version 2.” < http://www.nationalbimstandard.org/faq.php#faq1> (Aug 12, 2014).

Nawari, N.O. (2011). “Automating codes conformance in structural domain.” *Proc., Comput. Civ. Eng.*, ASCE, Reston, VA, 569-577.

Nguyen,T., and Kim, J. (2011). “Building code compliance checking using BIM technology.” *Proc., 2011 Winter Simulation Conference (WSC)*, IEEE, New York, NY, 3395-3400.

Niemeijer, R.A., Vries, B. D., and Beetz, J. (2009). “Check-mate: automatic constraint checking of IFC models.” In A Dikbas, E Ergen & H Giritli (Eds.), *Manag. IT in Constr. Manag. Constr. for Tomorrow*, CRC Press, London, UK, 479-486.

Orna-Montesinos, C. (2010). “Hyponymy relations in construction textbooks: a corpus-based analysis.” *Linguistic and Translation Studies in Scientific Communication, Linguistic Insights*, 86, 96-114.

Pan, J., Cheng, C.J., Lau, G.T., and Law, K.H. (2008). “Utilizing statistical semantic similarity techniques for ontology mapping - with applications to AEC standard models.” *Tsinghua Sci. and Technol.*, 13(S1), 217-222.

Porter, M. (1980). “An algorithm for suffix stripping.” *Program (Autom. Libr. and Inf. Syst.)*, 14(3), 130-137.

Resnik, P. (1995). “Using information content to evaluate semantic similarity in a taxonomy.” *Proc., IJCAI’95*, IJCAI, Inc., Somerset, NJ, 448-453.

Rodrı´guez, M.A., and Egenhofer, M.J. (2003). “Determining semantic similarity among entity classes from different ontologies.” *IEEE T. Knowl. Data En.*, 15(2), 442-456.

Rosenblatt, F. (1958). “The perceptron: a probabilistic model for information storage and organization in the brain.” *Psychol. Rev.*, 65(6), 1958.

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 (Feb. 23, 2013).

Shehata, S. (2009). “A WordNet-based semantic model for enhancing text clustering.” *2009 IEEE Int. Conf. Data Mining. Workshops*, IEEE, Piscataway, NJ, 477-482.

Simpson, T., and Dao, T. (2010). “WordNet-based semantic similarity measurement.” < http://www.codeproject.com/Articles/11835/WordNet-based-semantic-similarity-measurement> (Aug 14, 2014).

Sinha, S., Sawhney, A., Borrmann, A., and Ritter, F. (2013). “Extracting information from building information models for energy code compliance of building envelope.” *COBRA 2013 Conf.*, International Council for Research and Innovation in Building and Construction (CIB), Rotterdam, Netherlands.

Song, W., Liang, J.Z., and Park, S.C. (2014). “Fuzzy control GA with a novel hybrid semantic similarity strategy for text clustering.” *Inform. Sciences*, 273(2014), 156-170.

Soysal, E., Cicekli, I., and Baykal, N. (2010). “Design and evaluation of an ontology based information extraction system for radiological reports.” *Comput. in Biology and Med.*, 40(11-12), 900-911.

Spiliopoulos, V., Vouros, G.A., and Karkaletsis, V. (2010). “On the discovery of subsumption relations for the alignment of ontologies.” *Web Semant.*, 182, 1-20.Slimani, T. (2013). “Description and evaluation of semantic similarity measures approaches.” *Int. J. Comput. Appl.*, 80(10), 25-33.

Suchanek, M., Kasneci, G., and Weikum, G. (2007). “YAGO: A core of semantic knowledge unifying WordNet and Wikipedia.” *Proc., WWW 2007*, Association for Computing Machinery, New York, NY, 697-706.

Tan, X., Hammad, A., and Fazio, P. (2010). “Automated code compliance checking for building envelope design.” *J. Comput. Civ. Eng.*, 10.1061/1192 (ASCE)0887-3801(2010)24:2(203), 203–211.

Toutanova, K., Klein, D., Manning, C., and Singer, Y. (2003). “Feature-rich part-of-speech tagging with a cyclic dependency network.” *Proc., HLT-NAACL 2003*, 252-259.

Vanlande, R., Nicolle, C., and Cruz, C. (2008). “IFC and building lifecycle management.” *Autom. Constr.*, 18(1), 70-78.

Varelas, G., Voutsakis, E., and Raftopoulou, P. (2005). “Semantic similarity methods in WordNet and their application to information retrieval on the web.” *Proc., 7th annual ACM intl. workshop on Web inform. and data manage. (WIDM '05)*, Association for Computing Machinery, New York, NY, 10-16.

Young, N.W., Jr., Jones, S.A., Bernstein, H.M., and Gudgel, J.E. (2009). “The business value of BIM: getting building information modeling to the bottom line.” The McGraw-Hill Companies, New York, NY.

Zhang, J., and El-Gohary, N. (2013). “Semantic NLP-based information extraction from construction regulatory documents for automated compliance checking.” *J. Comput. Civ. Eng.*, Accepted and published online ahead of print.

Zhang, J., Yu, F., Li, D., and Hu, Z. (2014). “Development and implementation of an Industry Foundation Classes-based graphic information model for virtual construction.” *Comput.-Aided Civ. Inf.*, 29(2014), 60-74.

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

Zhou, P., and El-Gohary, N. (2014). "Ontology-based multi-label text classification for enhanced information retrieval for supporting automated environmental compliance checking." *Proc., 2014 ASCE Constr. Res. Congress (CRC),* ASCE, Reston, VA, 2238-2245.

Tables

Table 1. Commonly-used Machine Learning Algorithms

	Machine learning algorithm				
	Naïve Bayes	Perceptron	Decision Tree	k-NN	SVM
Key feature	simple but effective	linear	flexible	similarity-based	kernel-based

Table 2. Types of Relationships Considered

Relationship type	Relationship interpretation
Equivalent concept	R[1] is equivalent to F[2]
Superconcept	R is superconcept of F
Subconcept	R is subconcept of F
Associated concept	R and with F are associated

[1] R means R-concept
[2] F means F-concept

Table 3. The Syntactic and Semantic Features used for the Relationship Classifier

Feature name	RTermNum	RTermPOS	RMatch Type	RelMatch Type	FMatchType	FTermNum	FTerm POS	DOM
Possible values	unigram, bigram, multigram	N, NN, JN	first, last	synonym, hypernym, hyponym, term-based	first, middle, last	unigram, bigram, multigram	N, NN, JN	1, 0

Table 4. Example R-concepts, Matched F-concepts, and Their Feature Values

R-concept	F-concept	RTerm Num	RTerm POS	RMatch Type	RelMatch Type	FMatch Type	FTermNum	FTerm POS	DOM
construction	construction resource	unigram	N	first	term-based	first	bigram	NN	0
floor joist	beam	bigram	N	last	synonym	first	unigram	NN	0
preconstruction	preconstruction	bigram	NN	first	term-	first	bigram	NN	1

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

testing	test				based				
skylight	window	unigram	N	first	synonym	first	unigram	N	0
water-proof joint	structural connection	bigram	NN	last	hypernym	last	bigram	JN	0

Table 5. Performance of Extracting Regulatory Concepts from Development Text (Chapter 12 of IBC 2006)

Method	Number in gold standard	Number extracted	Number correctly extracted	Precision	Recall	F1-measure
Without exclusion word list	368	391	365	93.4%	99.2%	96.2%
With exclusion word list	368	376	365	97.1%	99.2%	98.1%

Table 6. Performance of Extracting Regulatory Concepts from Testing Text (Chapter 19 of IBC 2009)

Method	Number in gold standard	Number extracted	Number correctly extracted	Precision	Recall	F1-measure
Without exclusion word list	821	871	773	88.7%	94.2%	91.4%
With exclusion word list	821	865	773	**89.4%**	94.2%	**91.7%**

Table 7. Performance of Regulatory Concept Extraction after Improvements

Method	Number in gold standard	Number extracted	Number correctly extracted	Precision	Recall	F1-measure
Baseline Condition (from Table 6)	821	865	773	89.4%	94.2%	91.7%
With extended exclusion word list	821	856	774	90.4%	94.3%	92.3%
With extended POS pattern set	821	860	784	91.2%	95.5%	93.3%
With both extended exclusion word list and extended POS pattern set	821	851	785	**92.2%**	**95.6%**	**94.0%**

Table 8. Performances of Different SS Scoring Methods for Term-Based Matched F-Concepts

Proposed concept-level SS scoring function	Term-level SS scoring function	Number of related F-concepts found	Number of related F-concepts adopted	Adoption rate
Eq. (1)	Shortest Path Similarity	286	249	**87.1%**
Eq. (2)	Shortest Path Similarity	286	225	78.7%
Eq. (1)	Jiang-Conrath Similarity	286	244	85.3%
Eq. (2)	Jiang-Conrath Similarity	286	224	78.3%
Eq. (1)	Leacock-Chodorow Similarity	286	237	82.9%
Eq. (2)	Leacock-Chodorow Similarity	286	202	70.6%
Eq. (1)	Resnik Similarity	286	246	86.0%
Eq. (2)	Resnik Similarity	286	228	79.7%
Eq. (1)	Lin Similarity	286	246	86.0%
Eq. (2)	Lin Similarity	286	224	78.3%

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

Table 9. Examples of Matched R-Concepts and F-Concepts Using Different SS Scoring Methods for Term-Based Matched F-Concepts

Extracted R-concept	Proposed concept-level SS scoring function	Matched F-concept using Shortest Path Similarity[1]	Matched F-concept using Leacock-Chodorow Similarity[1]	Matched F-concept using Jiang-Conrath Similarity[1]
adjacent dwelling unit	Eq. (1)	dwelling unit	*unit assignment*	*derived unit*
	Eq. (2)	*context dependent unit*	*context dependent unit*	*context dependent unit*
square foot	Eq. (1)	feet	*footing*	feet
	Eq. (2)	feet	*footing*	feet
international energy conservation code	Eq. (1)	code	code	code
	Eq. (2)	international mechanical code	international mechanical code	international mechanical code
required ventilating area	Eq. (1)	area	area	area
	Eq. (2)	*annotation fill area*	*annotation fill area*	*annotation fill area*
exterior walls	Eq. (1)	wall	wall	wall
	Eq. (2)	curtain wall	curtain wall	curtain wall

[1] italicized concepts were not adopted

Table 10. Performances of Different SS Scoring Methods for Semantic-Based Matched F-Concepts

Proposed concept-level SS scoring function	Term-level SS scoring function	Number of related F-concepts found	Number of related F-concepts adopted	Adoption rate
Eq. (1)	Shortest Path Similarity	114	94	**82.5%**
Eq. (2)	Shortest Path Similarity	114	94	**82.5%**
Eq. (1)	Jiang-Conrath Similarity	114	92	80.7%
Eq. (2)	Jiang-Conrath Similarity	114	92	80.7%
Eq. (1)	Leacock-Chodorow Similarity	114	93	81.6%
Eq. (2)	Leacock-Chodorow Similarity	114	93	81.6%
Eq. (1)	Resnik Similarity	114	93	81.6%
Eq. (2)	Resnik Similarity	114	93	81.6%
Eq. (1)	Lin Similarity	114	93	81.6%
Eq. (2)	Lin Similarity	114	93	81.6%

Table 11. Examples of Matched R-Concepts and F-Concepts Using Different SS Scoring Methods for Semantic-Based Matched F-Concepts

Extracted R-concept	Proposed concept-level SS scoring Function	Matched F-concept using Shortest Path Similarity[1]	Matched F-Concept using Leacock-Chodorow Similarity[1]	Matched F-Concept using Jiang-Conrath Similarity[1]
corrosion-resistant wire cloth screening	Eq. (1)	hardware cloth	hardware cloth	hardware cloth
	Eq. (2)	hardware cloth	hardware cloth	hardware cloth
grab bars	Eq. (1)	railing	railing	railing
	Eq. (2)	railing	railing	railing
outdoors	Eq. (1)	outside horizontal clear space	outside horizontal clear space	outside horizontal clear space
	Eq. (2)	outside horizontal clear space	outside horizontal clear space	outside horizontal clear space
installed	Eq. (1)	*contaminant sources*	*contaminant source*	*light source*

shower heads	Eq. (2)	*contaminant sources*	*contaminant source*	*light source*

[1] italicized concepts were not adopted

Table 12. Testing Results of IFC Concept Selection Method

Concept matching type	Concept-level SS scoring function	Term-level SS scoring function	Number of related F-concepts found	Number of related F-concepts adopted	Adoption rate
Term-based matching	Eq. (1)	Shortest Path Similarity	598	507	84.8%
Semantic-based matching			98	81	82.7%
Total			696	588	84.5%

Table 13. Results of Testing Different ML Algorithms

Metric	ML algorithm			
	Naïve Bayes	Decision Tree	k-NN	SVM
Total number of relationship instances	366	366	366	366
Number of correctly classified relationship instances	279	315	333	314
Precision	76.23%	86.07%	**90.98%**	85.79%

Table 14. Leave-One-Out Feature Analysis Precision Results

ML algorithm	Excluded feature								
	None	RTermNum	RTermPOS	RMatchType	RelMatchType	FMatchType	FTermNum	FTermPOS	DOM
Decision Tree	86.07%	**86.89%**	**86.61%**	81.15%	**86.61%**	84.70%	**86.89%**	86.07%	81.98%
k-NN	90.98%	89.07%	89.89%	86.34%	88.80%	86.89%	**91.26%**	90.44%	88.25%

Table 15. Relationship Classifier Testing Results

Relationship type	Number of relationship instances in gold standard	Number of classified relationship instances	Number of correctly classified relationship instances	Precision	Recall	F1-Measure
Equivalent concept	79	74	68	91.9%	86.1%	88.9%
Subconcept	241	241	225	93.4%	93.4%	93.4%
Superconcept	61	52	46	88.5%	75.4%	81.4%
Associated concept	50	64	40	62.5%	80.0%	70.2%
Total	431	431	379	87.94%	87.94%	87.94%

Figure 1. Proposed IFC extension method

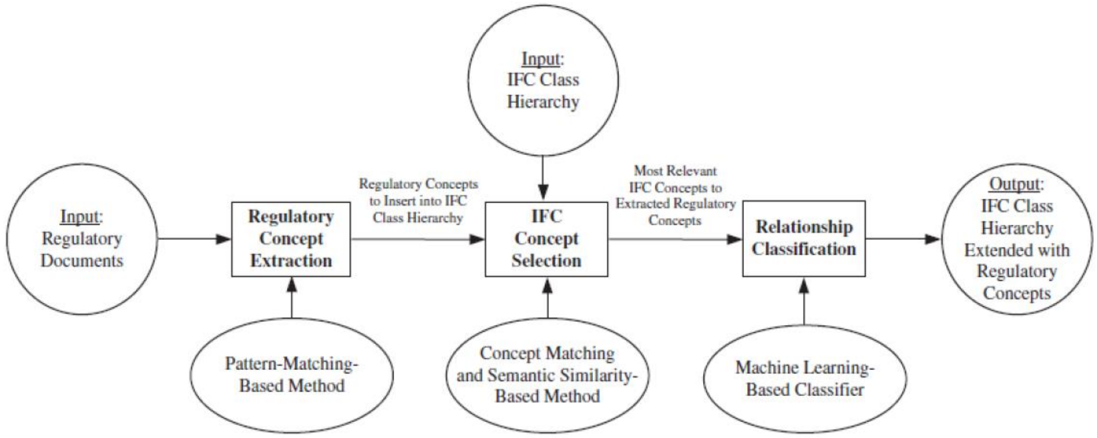


Figure 2. Example concept extraction rule and its meaning

Concept Extraction Rule: RB VBN NN NN -> Extract the four matched terms

Meaning: If four consecutive terms are adverb, past participle verb, singular/mass noun, and singular/mass noun, then these four terms should be extracted as a concept.

POS Tag	Meaning
NN	Singular or mass noun
NNS	Plural noun
NNP	Singular proper noun
NNPS	Plural proper noun
JJ	Adjective
RB	Adverb
VBN	Past participle verb
VBP	Non-3rd person singular present verb
VBD	Past tense verb
VBG	Gerund or present participle verb

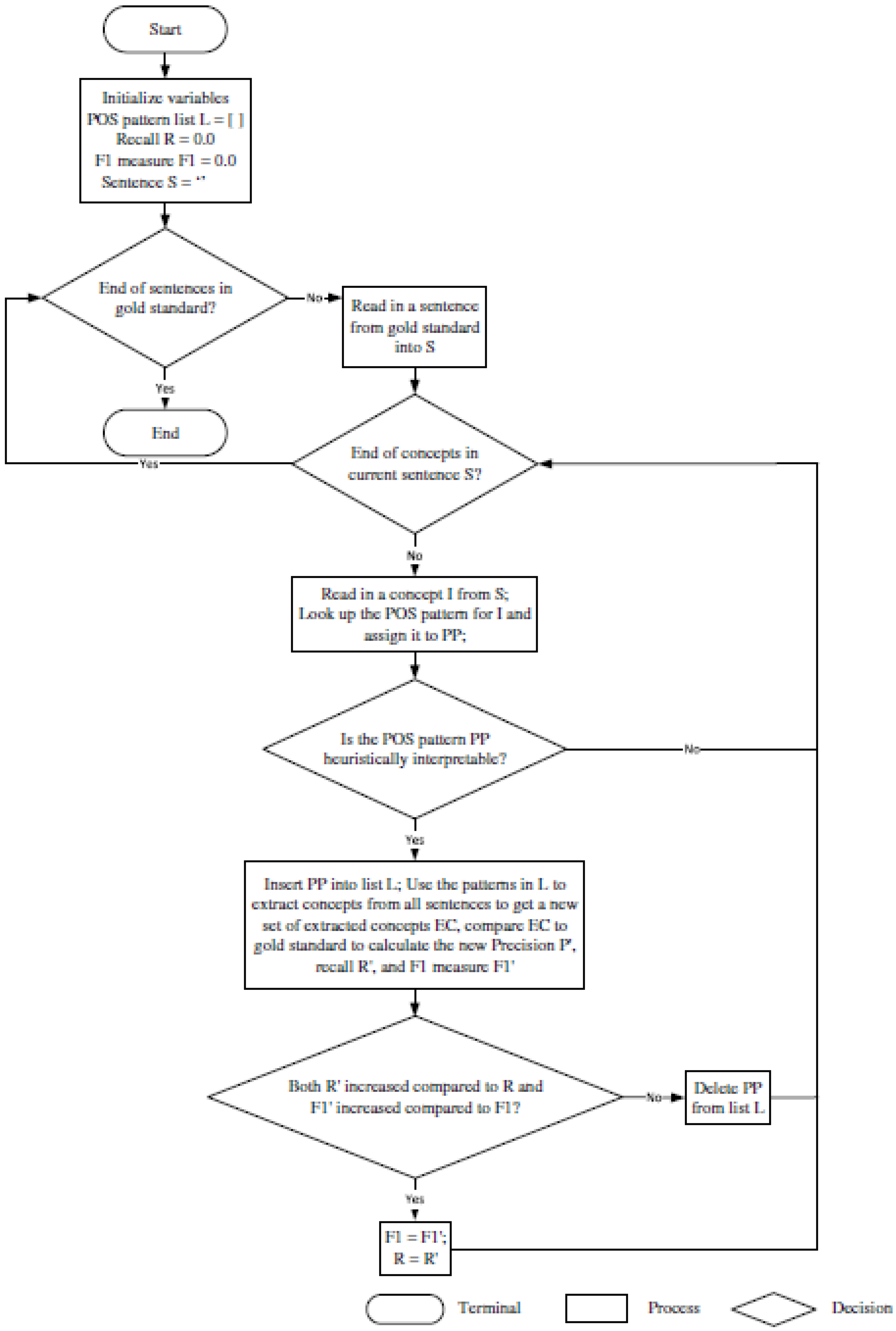
Figure 3. Sample of Patterns and Concept Extraction Rules

Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.

- R1: NP -> NN NN
- R2: NP -> NN NP (non-flattened)
- R3: NP -> NN NN NN
- R4: NP -> NN NN NN NN
- R5: NP -> NN NN NN NN NN
- P1: NN NN
- P2: NN NN NN
- P3: NN NN NN NN
- P4: NN NN NN NN NN

Figure 4. Flow chart of the POS pattern set development algorithm

Final published version is found in the ASCE Library here: http://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000536
Zhang, J. and El-Gohary, N. (2016). "Extending Building Information Models Semiautomatically Using Semantic Natural Language Processing Techniques." J. Comput. Civ. Eng., 10.1061/(ASCE)CP.1943-5487.0000536, C4016004.



1041
1042
1043 Figure 5. IFC concept selection method
1044

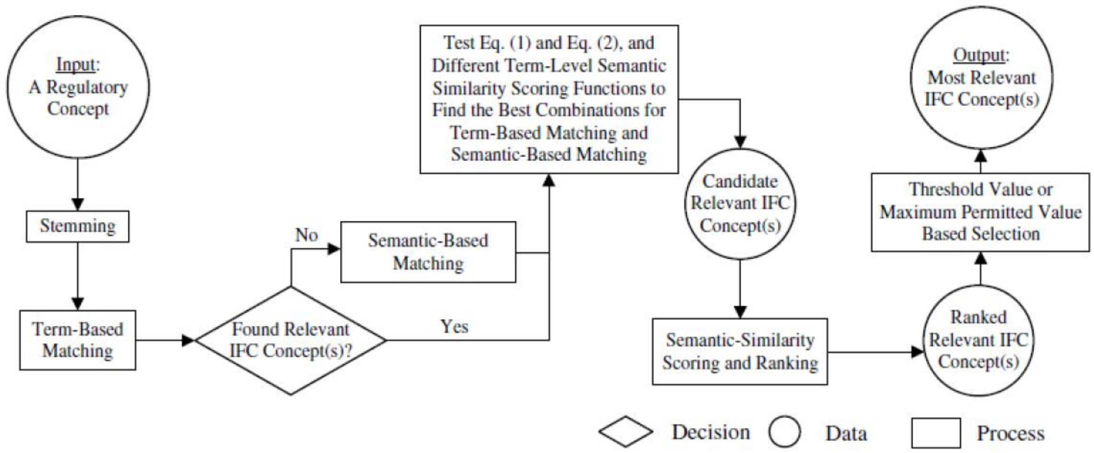


Figure 6. Term-based and semantic-based matching strategy

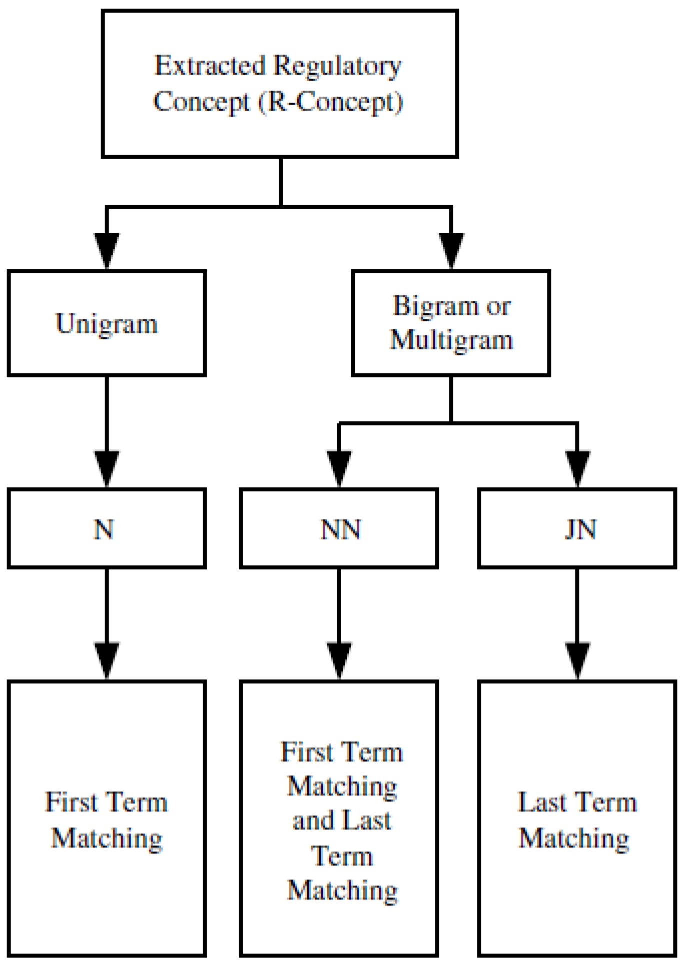


Figure 7. Set of POS patterns developed

NN	VBN NN	VBN NN NN
NNP	VBN NNS	VBN NN NNS
NNS	JJ JJ NN	JJ JJ JJ NN
VBG	JJ JJ NNS	JJ JJ NN NN
JJ NN	JJ NN NN	JJ NN NN NN
JJS NN	JJ NN NNS	NN NN NN NNS
JJ NNS	NN NN NN	NNP NN NN NN
NN NN	NN NN NNS	NNP NN NN NNS
NN NNS	NNP NN NN	NNP NNP NNP NNP
NNP NNP	NNP NNP NNP	RB VBN JJ NN
NNP NNS	NNP VBD NNS	RB VBN NN NN
VBG NN	NN VBG NN	RB VBN NNP NN
VBG NNS	RB JJ NNS	JJ VBN JJ JJ VBG NN

Figure 8. A sample concept hierarchy structure