Lexical-Morphological Modeling for Legal Text Analysis

Danilo S. Carvalho[1*], Minh-Tien Nguyen[1,2], Chien-Xuan Tran[1], and Minh-Le Nguyen[1]

[1] School of Information Science, Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Nomi, Ishikawa, 923-1292, Japan
[2] Hung Yen University of Education and Technology (UTEHY), Hung Yen, Vietnam
{danilo, tiennm, chien-tran, nguyenml}@jaist.ac.jp

Abstract. In the context of the Competition on Legal Information Extraction/Entailment (COLIEE), we propose a method comprising the necessary steps for finding relevant documents to a legal question and deciding on textual entailment evidence to provide a correct answer. The proposed method is based on the combination of several lexical and morphological characteristics, to build a language model and a set of features for Machine Learning algorithms. We provide a detailed study on the proposed method performance and failure cases, indicating that it is competitive with state-of-the-art approaches on Legal Information Retrieval and Question Answering, while not needing extensive training data nor expert produced knowledge.

1 Introduction

Answering legal questions has been a long-standing challenge in the Information Systems research landscape. A topic that is drawing progressively more attention, as we experience an explosive growth in legal document availability on the World Wide Web and specialized systems. This growth is not accompanied by a matching increase in information analysis capabilities, which points to a severe under-utilization of the available resources and a potential for information quality issues [1]. As a consequence, professionals of law, in particular, have been put into increasingly pressure, since having the relevant and correct information is a vital step in legal case solving and thus is closely tied to the matter of professional ethics and liability. This problem is often referred as "information crisis" of law.

The ability to retrieve relevant and correct information given a legal query has improved over time, with the combination of expert Knowledge Engineering and NLP methods. On the other hand, the ability to answer questions in the legal domain is of special difficulty, due to the need of reasoning over different types of information, such as past decisions, laws and facts. Furthermore, the way legal concepts are applied in the language often differs from common usage,

* Supported by CNPq – Brazil scholarship grant

and differences in laws and procedures from each country prevent the creation of comprehensive and coherent international law corpora. Common legal ontologies are among the efforts to facilitate automatic legal reasoning, but have not seen strong development in the past years [2]. In this context, *Textual Entailment Recognition* plays a very important role, as a set hypothesis presented in a question will most certainly have answers in the previously cited types of information (decisions, laws, facts). The Recognition of Textual Entailment (RTE) challenge series [3], although not specific to the legal domain, is a recognized benchmark for methods that can be adapted to legal texts.

To effectively answer legal questions, one fundamental set of information that must be available is the law, presented as the collection of codes, sections, articles and paragraphs that should be unequivocally referenced when a hypothesis is raised as part of a legal inquiry. Therefore, adequate representation of law corpora is the basis of a functional system for legal question answering. The representation problem is often associated with ontologies and other annotated knowledge bases, but these methods are costly and more difficult to automate when compared to fully text-based approaches, such as bag-of-words, n-gram and topic models.

In this work, we propose a fully text-based method for legal text analysis, in the context of the Competition on Legal Information Extraction/Entailment (COLIEE), covering both the tasks of Information Extraction and Question Answering. The goal is the retrieval of relevant law articles to a given yes/no legal question and the use of the retrieved articles to correctly answer the question in a completely automated way. The proposed method is based on a broad lexical and morphological analysis of the English translated Japanese Civil Code, comprising tokenization, POS-tagging, lemmatization, word clustering and a set of lexical statistics. Such analysis allows the construction of a mixed size n-gram model and a set of features appropriate for Machine Learning algorithms. The n-gram model is used for Relevance Analysis on the legal corpus, with respect to the queries provided, and the Machine Learning features are used for Textual Entailment classification. A study on success and fail cases is provided, with common baseline practices and related works used as means of performance comparison.

The remaining of this work is structured as follows: Section 2 presents the related works and relevant results; Section 3 details the Legal Question Answering problem and the COLIEE competition shared task; Section 4 explains our approach to the competition problem; Section 5 presents the experimental setting, results and discussion; Finally, Section 6 offers some concluding remarks.

2 Related Works

Liu, Chen and Ho [3] presented the three-phase prediction (TPP) method for retrieval of relevant statutes in Taiwan's criminal law, given general language queries. The method is a hierarchical ranking approach to law corpora, featuring

[3] www.aclweb.org/aclwiki/index.php?title=Recognizing_Textual_Entailment

a combination of several Information Retrieval techniques, as well as Machine Learning and Feature selection ones. Results were evaluated in terms of recall, achieving from 0.52 to 0.91, from the top 3 to 10 retrieved results, respectively.

Inkpen et al. showed one of the first successful models for RTE by SVMs [4]. Later, Castillo proposed a system for solving RTE by using SVMs, in which training data includes RTE-3, annotated data set of RTE-4, and development set of RTE-5 [5]. 32 features were used and the training model achieved the best F-measure of 0.69 in two-way and 0.67 in three-way task. Most of features from [5] were used in this study along with *Word2Vec*[16] similarity instead of *WordNet*.

Nguyen et al. [6] conducted a study of RTE on a Vietnamese version of RTE-3 [7] translated from [8]. The author used SVMs based on 15 features including two groups: distance and statistical features. A voting combined three classifiers built on three feature groups (distance, statistical, and combined features) was used to judge entailment relation. The method obtained 0.684 of F-measure in two-way task. In this study, we use all features in [6] and add additional features: Word2Vec, term frequency - inverse document frequency (TF-IDF), or rules.

Research dedicated to Question Answering (QA) in legal text has seen less attention when compared to general QA. Tran et al. solved legal text QA by using inference [9]. The author used requisite-effectuation structures of legal sentences and similarity measures to find out correct answers without training data and achieved 60.8% of accuracy with 51 articles on Japanese National Pension Law.

Kim et al. proposed a hybrid method containing simple rules and unsupervised learning using deep linguistic features to solve RTE in civil laws [10]. The author also constructed a knowledge base for negation and antonym words which would be used for classifying simple questions. To deal with difficult questions, the author used morphological, syntactic and lexical analysis to identify premises and conclusions. The accuracy was 68.36% with easy questions and 60.02 with difficult ones.

3 Legal Question Answering

Legal Question Answering (LQA) is to find out and provide *"correct answers"* given by a legal question for users. An overview of LQA is shown in Fig. 1.

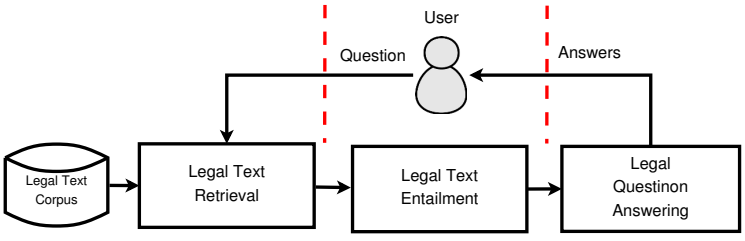

Fig. 1. The model of legal text question answering system

LQA can be described in three tasks: 1) retrieving relevant articles, i.e., the ones containing the answer; 2) finding correct evidence in the relevant articles that allows answering the question; and 3) answering the question. While the first task is essentially IR, the second can be considered as a form of RTE, in which given a question, a LQA system has to decide whether and how a relevant article can answer the question. The final one is the combination of the two previous tasks.

Legal texts are very different in comparison to general ones e.g., new articles due to their characteristics. Firstly, they have specific logical sentence structures e.g., requisite and effectuation [11]. Secondly, words and writing style are used in a strict form because law documents require high correctness and should avoid ambiguity. An alternative aspect is that law documents are written in a highly abstract level [12]; therefore, they usually require reference of readers to understand law articles and to answer a law question. The use of references leads to a situation in which there is not much, or in some cases, even no word overlapping between a law question and its relevant articles.

In this work, LQA tasks are considered into the context of COLIEE, a competition on legal information extraction/entailment which was first held in 2014, in association with Workshop on Juris-informatics (JURISIN). COLIEE 2015 [4] is the second competition consisting of three phases:

- Phase One: retrieving relevant articles from all Japanese Civil Code Articles given a set of YES/NO questions.
- Phase Two: confirming the entailment relationship between the question and retrieved articles.
- Phase Three: combination of Phase One and Phase Two, the system will retrieve list of relevant articles given a query, and then decide the entailment relationship between retrieved articles and provided question.

The Japanese Civil Code is composed by a collection of numbered articles, each one containing a set of declarations pertaining to a specific topic of the law, e.g., labor contracts, mortgages.

Information Retrieval Task: Relevance Analysis

The first phase consists on an explicit IR task, for which the goal is to retrieve the relevant articles that can be used to correctly answer a given yes/no question. The challenge in this task is to determine the relative relevance, i.e., Relevance Analysis (RA), of an article to the query presented in the question. Different articles dealing with the same topic often have similar wording and is common for questions not to refer to topic keywords or refer to alternative versions of them. Furthermore, the restricted size of Japanese Civil Code means that obtaining reliable linguistic information from the articles is difficult and most questions will present new language structures that can range from useful to necessary for answering.

[4] webdocs.cs.ualberta.ca/~miyoung2/COLIEE2015/

Simple Question Answering Task: Textual Entailment

The goal of Textual Entailment (TE) is to decide whether a legal query/question can be answered by a set of relevant articles retrieved by RA. This task can be accomplished by recognizing textual entailment (RTE), in which the query/question is treated as an hypothesis and relevant articles are textual information. Given a question Q and a set of relevant articles A, ($A = \{a_1, ..., a_n\}$), if Q is answered by a_i ($1 \leqslant i \leqslant n$), then a_i entails Q [8,13] and a pair *(Q, a_i)* is assigned label YES; otherwise is NO. Assuming only relevant articles are provided, the challenge in this task is to identify the textual features that characterize conforming and conflicting information. To this end, the linguistic challenges of phase one also apply.

4 Proposed Approach

In order to be able to perform both Relevance Analysis and textual entailment independently in phases one and two, and jointly in phase three, IR and classifier methods were developed separately. First, both the legal corpus and training data are analyzed and combined into representation models. The models are then used to rank articles or classify answers according to the task. The representation model used for Relevance Analysis is a mixed size n-gram model and the one used for textual entailment is a multi-feature vector for Machine Learning. Figure 2 shows the overall view of the proposed method.

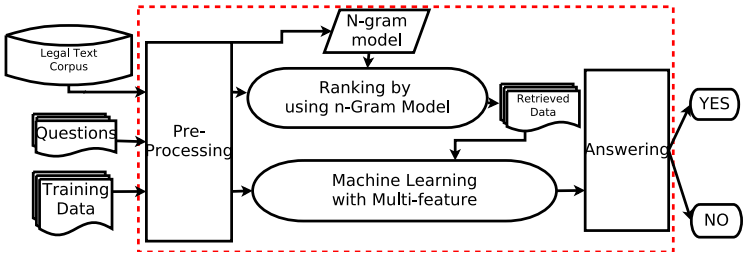

Fig. 2. Model overview

4.1 Relevance Analysis

A detailed analysis of the Civil Code and training data revealed that lexical and syntactic overlapping may vary to a high degree between questions and articles, and also between articles concerning the same topic. However, certain morphological features, such as lemmas, retain a higher level of consistency among topics. For this reason, the adopted representation model was a mixed size n-gram model, with $n : [1, k]$, i.e., terms made by sequences up to k words, in which the terms are lemmatized. For simplicity, the Relevance Analysis method hereon described was named R_2NC (Ranking Related N-gram Collections). A summarized view of the process is shown on Figure 3.

The steps to build the model are detailed as follows:

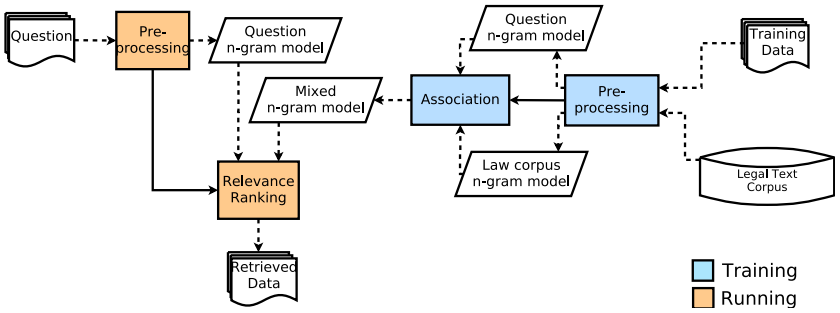

Fig. 3. The process of legal text retrieval

1. Collect the entire content for each article, including section title;
2. Check references between articles and annotate accordingly;
3. Tokenize and POS-tag;
4. Remove stopwords;
5. Lemmatize words;
6. Generate n-grams;
7. Expand the n-gram set, by including references n-grams;
8. Associate article number and references;
9. Store the model.

Except for step 4, each step is responsible for adding new information to the model. The information is obtained either from the text, e.g., section title, references, or from morphological analysis, e.g., POS-tags, lemmas. If an article have references, its n-gram set is expanded with the references' n-grams. This is done so that all the necessary information for interpretation of any single article is self-contained. Besides the n-grams, links between the articles are also stored. To include the training data information, the same process is repeated for the questions, and n-gram sets of the trained questions are used to expand the associated articles n-gram model.

To determine the relative relevance of an article with regard to the content of a question, a ranking approach was adopted. First, the n-gram set of the question is obtained by applying steps 1-6, using the question content instead of article. Then, for each article in the Civil Code, a relevance score is calculated using the following formula:

$$score = \frac{\sum_{\forall t} idf(t)}{I_q \times |q\_ng\_set| + I_{art} \times |art\_ng\_set|}, \quad t \in (q\_ng\_set \cap art\_ng\_set) \quad (1)$$

where q_ng_set is the set of n-grams for the question, art_ng_set is the set of n-grams for the article in the stored model, I_q is the relative significance of the question n-gram set size and I_{art} is the relative significance of the article n-gram set size. $idf(t)$ is the Inverse Document Frequency for the term t over the articles collection

$$idf(t) = log\frac{N}{df_t} \tag{2}$$

where N is the total number of articles and df_t is the number of articles in which t appears.

The formula (1) is a variation of the traditional *TF-IDF* scoring method, disregarding term frequency and giving different weights for the two types of document being evaluated: articles and questions, according to their size. I_q and I_{art} are parameters to be adjusted according to the corpus characteristics. This formula was developed after initial experiments with a TF-IDF based classifier showed poor results for this task and further observation showed that TF did not contribute for article relevance in many cases. With TF removed, document size becomes a more relevant feature and must be considered in the scoring.

From this point, the articles are sorted by descending score and the 10 best are selected for filtering. The filtering step consists in fetching the best scoring article and verifying if it exceeds a parameter threshold $confidence\_thresh$. If it does, all the articles in the list that are referred by the first and exceed a parameter threshold $reference\_thresh$ are also fetched. The fetched articles compose the final list of relevant articles to the input question. All R_2NC parameters I_q, I_{art}, $confidence\_thresh$, $reference\_thresh$ and also k, the maximum n-gram size, were adjusted empirically on the training data.

4.2 Textual Entailment

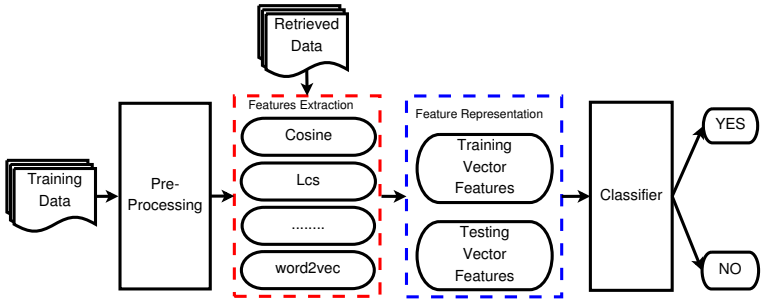

Fig. 4. The process of legal textual entailment recognition

Textual Entailment (TE) in law domain can be represented in form of binary classification [14,5,4]. Given a civil law question and a relevant article, TE relation is assigned by YES label if the article contains information for answering the question; otherwise is NO. More precisely, we aim to solve TE by using machine learning in form of ensemble methods suggested in [6].

TE process is represented in Fig. 4, in which training data is pre-processed by sentence and word segmentation, and removing stop words; next, features are extracted from both training and retrieved data (from the RA step); subsequently, the training and retrieved data are represented in vector space model,

in which training vectors will be used to train a classification model; finally, the model judges entailment relation on retrieved data.

To train the model, a straightforward method is to extract features which represent relevant characteristics of the data. A data observation was conducted to capture the characteristics. The observation is illustrated in Tab. 1.

Table 1. Data statistic in phase two

	# pairs	**# sentences**	**# tokens**	**% uni-gram word overlapping**
Training Set	267	273	36.562	58.80

The observation suggested that word overlapping and word similarity between a question and an article can be useful to capture TE relation. Based on the observation, we proposed the use of 15 lexical features in the form of two groups: distance and statistical features due to its common characteristics. The features are shown in Tab. 2. Note that features in Section 4.1 can be also used for this task; however, due to time constraint, these features will be investigated in the future.

Table 2. The feature groups; Avg is average; Q is a question, S is a sentence

	Feature	**Description**
Distance	Manhattan	Manhattan distance from two text fragments
	Euclidean	Euclidean distance from two text fragments
	Cosine similarity	Cosine similarity distance
	Matching coefficient	Matching coefficient of two text fragments
	Dice coefficient	Dice coefficient of two text fragments
	Jaccard	Jaccard distance of two text fragments
	Jaro	Jaro distance of two text fragments
	Damerau-Levenshtein	Damerau Levenshtein distance of two text fragments
	Levenshtein	Levenshtein distance of two text fragments
Statistical	Lcs	The longest common sub string of two text fragments
	Average of TF-IDF	Term frequency-inverse document frequency
	Avg-TF of Q and S	Avg-TF of words in a Q appearing in a S
	Avg-TF of S and Q	Avg-TF of words in a S appearing in a S
	Word overlapping	# word overlapping in a Q appearing in a article
	Average of Word2Vec	Average of word2vec similarity

After extracting features, pairs in the training dataset were represented by feature vectors with two pre-defined classes: entailment (YES) or no-entailment (NO). These vectors were used to find hypothesis for the classification model. The model would be used to judge the entailment relation on testing dataset.

5 Experiments and Results

5.1 Experimental Setup

The dataset was obtained from the published data for the COLIEE shared task [5], consisting in a text file with the Japanese Civil Code and a set of XML files with

[5] webdocs.cs.ualberta.ca/~miyoung2/COLIEE2015/

training and testing data for phases one to three. The training set for the three tasks contains 267 pairs (question, relevant articles). Experiments where divided in phases one and two only, dealing with Information Retrieval and Textual Entailment methods respectively. Each experiment comprised: i) data analysis, ii) model and parameter adjustments and iii) test runs.

For the IR task, data analysis suggested that the problem of restricted linguistic information could be overcome by including external corpora into the n-gram generation process. Thus, attempts were made to expand the R_2NC n-gram model with three different corpora, as follows:

- *News*: One billion word collection of news articles (in English). Text tokenized and cleaned from markups [15].
- *CA—LA_Law*: Collection of Civil Codes from U.S. states of California [6] and Louisiana [7]. Contains 3420 cleaned and tokenized articles, with about 1.7 million words in total.
- *JPN_Law*: Collection of all Civil law articles of Japan's constitution [8]. Contains 642 cleaned and tokenized articles, with about 13.5 million words.

Terms in corpora were clustered by means of n-gram cosine similarity using *Word2Vec* [16]. For a given article content, the n-gram cluster with the highest cosine similarity to the text was included into its n-gram set. Tokenization and lemmatization were done using *NLTK*[9] (v. 3.0.2) and POS-tagging was done using *Stanford Tagger*[10] (v. 3.5.2). Experiment results are shown in Table 3.

Parameters were adjusted for each test run and the best results are reported. The final parameter values used in the competition are $k = 3$, $I_q = 0.965$, $I_{art} = 0.035$, $confidence\_thresh = 0.32$ and $reference\_thresh = 0.2$.

For the TE task, AdaBoost [17] was used to train the model, where: *classifier* was the standard *DecisionStump*, with 10 iterations, $seed = 1$, no *re-sampling* and $weightthreshold = 100$.

5.2 Baselines

As for the second edition of COLIEE, there is still no definite baseline for the competition dataset. However, common baseline practices and related works could be used for evaluating performance on each task. For phase one, a relationship can be drawn between R_2NC and TPP [3], the latter achieving a recall of 0.52 for the top 3 retrieved statutes and 0.91 for the top 10.

Support Vector Machine (SVM) [18] in the LibSVM [11] implementation, integrated in Weka[12] was used as the baseline for phase 2. The parameters of SVM are $C = 1$, $\gamma = 0$, *kernel Type = radial basis function*. Another baseline was using SVMs as weak learners for AdaBoost, instead of the standard DecisionStump.

[6] leginfo.legislature.ca.gov
[7] legis.la.gov
[8] www.japaneselawtranslation.go.jp
[9] www.nltk.org
[10] nlp.stanford.edu/software/tagger.shtml
[11] www.csie.ntu.edu.tw/c̃jlin/libsvm/
[12] weka.wikispaces.com

5.3 Evaluation Method

Given the limited training data available, leave-one-out validation was used to evaluate the performance of the model in both tasks on the training dataset with three measures: precision (P), recall (R) and F-measure (F) as in Eq. (3), (4) and (5). In phase two, accuracy (A) measurement is also used as in Eq. (6).

$$P = \frac{Cr}{Rt} \quad (3) \qquad R = \frac{Cr}{Rl} \quad (4) \qquad F = \frac{2(P * R)}{P + R} \quad (5) \qquad A = \frac{Cq}{Q} \quad (6)$$

where Cr counts the correctly retrieved articles for all queries, Rt counts the retrieved articles for all queries, Rl counts the relevant articles for all queries, Cq counts the queries correctly confirmed as true or false and Q counts all the queries.

5.4 Results

Relevance analysis results were as follows:

Table 3. Experiment results for phase one (Information Retrieval) with R_2NC. Model and parameter adjustments were made for each test run. The best results are presented.

Ext. corpus	Precision	Recall	F-measure
None	0.568	0.516	**0.54**
News	0.461	0.485	0.472
CA—LA_Law	0.476	0.498	0.486
JPN_Law	0.541	0.52	0.53

Additionally, recall was 0.643 and 0.77 for the top 3 and top 10 retrieved articles respectively. This indicates that R_2NC is expected to be competitive with state-of-the-art approaches to relevance analysis in legal documents. However, the proposed method is much simpler when compared to *TPP* [3] and operates with considerably less training data: 266 documents for R_2NC against 1518 documents for *TPP*. R_2NC design also makes it difficult for the model to be overtrained beyond the parameter adjustment, since no training data is counted more than one time and the method is single-shot, as opposed to convergence-based. The test with no external corpus was repeated with traditional TF-IDF scoring, yielding 0.51 F-score. An important observation is that the results got worse when expanding the n-gram model with external data. This could indicate that the external corpora contains a fair amount of noise, or that the questions are highly corpus oriented, and the relevant information is at a abstraction level not reachable by morphological analysis. The drop in F-measure as the external data becomes semantically farther from the Civil Code corpus and the fact that using Japan Law corpus improved the recall support the latter point of view.

Results of TE in Tab. 4 indicate that our method significantly outperforms the baselines 0.084 (8.4%) and 0.113 (11.3%) of F-measure, respectively. More importantly, the precision and accuracy of our method also achieves high improvements in comparison the baselines. This concludes that our method is expected to be efficient for solving TE in legal domain.

Table 4. The performance our method vs. SVMs, where our method uses *DecisionStump* and SVMs uses *Radial basis function* kernel; with the strongest features

	Precision	**Recall**	**F-measure**	**Accuracy (%)**
Our method	0.621	0.614	0.597	**61.42**
SVMs	0.537	0.543	0.513	*54.30*
Our method	0.621	0.614	0.597	**61.42**
AdaBoost with SVMs	0.485	0.491	0.484	*49.06*

An alternative interesting point is that Word2Vec similarity contributes to improve the performance of our model. In reality, as stated in Section 3, legal documents usually require an inference to understand and answer a question; therefore, semantic similarity from Word2Vec can help to improve the performance. The results also show the efficiency of lexical features.

The performance of TE model, however, is not comparable with the same task in common data i.e., news articles [5,6] due to the characteristics of law dataset in LQA stated in Section 3. The performance is also not improved so much even when many features in both phase one and two were combined. Moreover, the observations from Relevance Analysis issues also apply here. This suggests that sophisticated features e.g., semantic inference or semantic rules should be considered.

5.5 Feature Evaluation

Further evaluation of feature impact on TE model was also conducted. In the evaluation, each feature was removed and the remain ones were kept to find out the most effective features. The most effective features are shown in Tab. 5.

Table 5. Top 4 influential features, italic is statistical features

Features	**Influential value**	**Features**	**Influential value**
Euclidean	0.005	*Lcs*	0.0001
Damerau-Levenshtein	0.154	Average of Word2Vec	0.024

Results in Tab. 5 show that all effective features contribute to the method. Note that both *Damerau-Levenshtein* and *Euclidean* are distance features whereas *the longest common substring* is statistical feature. The results support that in legal texts, there is not much word overlapping between a question and relevant articles. An interesting aspect is that Word2Vec similarity has a big positive impact to the model. This supports the conclusion on similarity stated in Section 5.4.

5.6 Error Analysis and Discussion

Investigation of the ranked list obtained with R_2NC in phase one (see Section 4.1) revealed that in most cases, relevant articles ranked lower than second had keywords not present in the corpus nor in the trained models. This reinforces the view that the questions are highly directed, albeit in a conceptual

level. Relevant articles that ranked lower than 15th (approx. 20%) were found to require a relatively high level of abstraction to obtain an interpretation that could link to the corresponding question. Table 6 shows an example of complex relevance relationship.

Table 6. Example of low ranking, high relevance article and corresponding question

ID	Article	Question	Ranked in
H18-2-1	Article 697(1)A person who commences the management of a business for another person without being obligated to do so (hereinafter in this Chapter referred to as "Manager") must manage that business (hereinafter referred to as "Management of Business") in accordance with the nature of the business, using the method that best conforms to the interests of that another person (the principal).(2)The Manager must engage in Management of Business in accordance with the intentions of the principal if the Manager knows, or is able to conjecture that intention.	In cases where a person plans to prevent crime in their own house by fixing the fence of a neighboring house, that person is found as having intent towards the other person.	424th

Table 7. Examples of entailment judgment; P is predicted and A is annotated

ID	Article	Question	P	A
H18-2-4	(Managers' Claims for Reimbursement of Costs)Article 702(1)If a Manager has incurred useful expenses for a principal, the Manager may claim reimbursement of those costs from the principal.(2)The provisions of Paragraph 2 of Article 650 shall apply mutatis mutandis to cases where a Manager has incurred useful obligations on behalf of the principal.(3)If a Manager has engaged in the Management of Business against the intention of the principal, the provisions of the preceding two paragraphs shall apply mutatis mutandis, solely to the extent the principal is actually enriched.	In cases where a person repairs the fence of a neighboring house after it collapsed due to a typhoon, but the neighbor had intended to replace the fence with a concrete-block wall in the near future, if a separate typhoon causes the repaired sections to collapse the following week, reimbursement of repair fees can no longer be demanded.	YES	YES
H18-26-1	(Renunciation of Shares and Death of Co-owners)Article 255 If one of co-owners renounces his/her share or dies without an heir, his/her share shall vest in other co-owners.	In cases where person A and person B co-own building X at a ratio of 1:1, if person A dies and had no heirs or persons with special connection, ownership of building X belongs to person B.	NO	YES

Table 7 shows that our method gives correct decisions e.g., ID H18-2-4. In this example, there several common words leading our model can correctly judge the TE relation e.g., *reimbursement*. In addition, several words can be inferred from the questions by using Word2Vec similarity e.g., *person* $\sim$ *manager*, *fees* $\sim$ *costs or expenses*. This supports our observation that TE can be solved by using lexical features and word similarity. An alternative interesting point is that even H18-2-4 contains a few common words and needs a reference to understand and answer the question; however, our method can still predict the TE relation correctly. This indicates the efficiency of our model as well as the features, especially word similarity feature.

On the other hand, the pair H18-26-1 exemplifies a case in which our model predicted NO while TE relation was annotated YES even when the question and

answer share some common words. This shows the limitation of our feature set in cases where the question and answer are short. In this case, after removing stop words, a few remaining words may not be enough to capture the TE relation. Moreover, lack of important words e.g., *building*, *connection* or *belong* reveals a big challenge for our method to decide the TE relation. This suggests that a keyword enriching mechanism such as term expansion used in phase one could improve the results.

6 Conclusion

This paper explores the challenging issue of building a QA system in the legal domain. We propose a model including three stages: legal text retrieval, legal textual entailment and legal text answering. In the first stage, a mixed size n-gram model built from morphological analysis is used to find out relevant articles corresponding to a legal question; next, pairs of questions and retrieved articles are judged by a machine learning algorithm trained on lexical features, to decide whether the questions can be answered positively or negatively by the retrieved articles; and finally, correct answers would be provided for users in the final stage. The contributions of this work in IR and TE task are: 1) a simple, yet effective language model for law corpora coupled with a Relevance Analysis method (R_2NC) capable of exploiting such model; 2) a set of RTE features, including Word2Vec similarity for Machine Learning algorithms. With a F-measure of 0.54 for the first retrieved articles and recall of 0.64 for the top 3, R_2NC appears as competitive when compared to state-of-the-art similar work, in spite of being much more simple and applicable with less training data. By combining RTE features and Word2Vec similarity, our method for LQA also significantly outperforms the baselines 0.084 (8.4%) and 0.113 (11.3%) of F-measure.

For future directions, information on a higher abstraction level, e.g., syntactic mappings, could be used to improve the language model for the IR task. In the TE task, since a sentence in a legal article is usually long, a sophisticated method of sentence partition e.g., requisite and effectuation should be considered. In feature extraction, features in IR should be combined with lexical features in TE and investigated to improve the quality of the judgment. Moreover, capturing contradictions in the TE relation by current statistical features is a big challenge. To solve this issue, semantic rules should be defined and incorporated into the feature extraction. Finally, we would like to investigate and apply sentence similarity calculation by Sent2Vec to improve the performance of the TE.

Acknowledgements

This work is supported partly by the grant of NII Research Cooperation and AIST's Research grant.

References

1. Robert C. Berring: "The heart of legal information: The crumbling infrastructure of legal research". *Legal information and the development of American law*. St. Paul, MN: Thomson/West, 2008.

2. Rinke Hoekstra, Joost Breuker, Marcello Di Bello and Alexander Boer: "The LKIF Core ontology of basic legal concepts". Proc. of the Workshop on Legal Ontologies and Artificial Intelligence Techniques (LOAIT 2007), 2007.
3. Yi-Hung Liu, Yen-Liang Chen and Wu-Liang Ho: "Predicting associated statutes for legal problems". *Information Processing & Management* 51.1: 194-211, 2015.
4. Diana Inkpen, Darren Kipp and Vivi Nastase: "Machine Learning Experiments for Textual Entailment. *Proceedings of the Second Challenge Workshop Recognising Textual Entailment*: 17-20, 2006.
5. Julio Javier Castillo: "An approach to Recognizing Textual Entailment and TE Search Task using SVM. *Procesamiento del Lenguaje Natural*, 44: 139-145, 2010.
6. Minh-Tien Nguyen, Quang-Thuy Ha, Thi-Dung Nguyen, Tri-Thanh Nguyen and Le-Minh Nguyen: "Recognizing Textual Entailment in Vietnamese Text: An Experiment Study." KSE, 2015 (accepted).
7. Quang Nhat Minh Pham, Le Minh Nguyen and Akira Shimazu: "Using Machine Translation for Recognizing Textual Entailment in Vietnamese Language." RIVF: 1-6, 2012.
8. Danilo Giampiccolo, Bernardo Magnini, Ido Dagan and Bill Dolan: "The third PASCAL recognising textual entailment challenge". *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*. Association for Computational Linguistics: 1-9, 2007.
9. Oanh Thi Tran, Bach Xuan Ngo, Minh Le Nguyen and Akira Shimazu: "Answering Legal Questions by Mining Reference Information". New Frontiers in Artificial Intelligence. Springer International Publishing: 214-229, 2014.
10. Mi-Young Kim, Ying Xu, Randy Goebel and Ken Satoh: "Answering Yes/No Questions in Legal Bar Exams". New Frontiers in Artificial Intelligence. Springer International Publishing: 199-213, 2014.
11. Bach Xuan Ngo, Minh Le Nguyen and Akira Shimazu: "RRE Task: The Task of Recognition of Requisite Part and Effectuation Part in Law Sentences." J. IJCPOL 23(2): 109-130, 2010.
12. Oanh Thi Tran, Bach Xuan Ngo, Minh Le Nguyen and Akira Shimazu: "Reference Resolution in Legal Texts." In Proc. of ICAIL: 101-110, 2013.
13. Ido Dagan, Bill Dolan, Bernardo Magnini and Dan Roth: "Recognizing textual entailment: Rational, evaluation and approaches - Erratum". *Natural Language Engineering* 16(1): 105-105, 2010.
14. Rui Wang: *"Intrinsic and Extrinsic Approaches to Recognizing Textual Entailment.* Saarland University, ISBN 978-3-933218-32-2: 1-219, 2011.
15. Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn and Tony Robinson: "One billion word benchmark for measuring progress in statistical language modeling", arXiv preprint arXiv:1312.3005, 2013.
16. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado and Jeffrey Dean: "Distributed Representations of Words and Phrases and their Compositionality". In Proceedings of NIPS, 2013.
17. Yoav Freund and Robert E. Schapire: "A decision-theoretic generalization of online learning and an application to boosting". Journal of computer and system sciences 55.1: 119-139, 1997.
18. Corinna Cortes and Vladimir Vapnik: "Support-Vector Networks". *Machine Learning* 20(3): 273-297, 1995.