Interim Report

Project Title: Information Extraction from Hong Kong Court Cases

Author:

Yeung Tsz Lok (3035366788),

Yu Tung Chuen (3035377127),

Yang Lingqin (3035330404)

Supervisor: Prof. Ben Kao

Date of Submission: 2 February. 2020

Abstract

Studying past rulings by tribunals plays an essential role in the daily routine of legal professionals. However, such a task consumes a fair amount of effort for comprehension and is deemed to be tedious, in general. This project aims to apply Natural Language Processing (NLP) technologies for automatic information extraction in the context of Hong Kong court cases, reducing the prohibitive cost of reviewing. Beyond information extraction, the project team believes that there is a demand for tools across various disciplines to harness the exhaustively parsed court case data, either in visualization, prediction, or classification. Nevertheless, developing applications based on parsed court case data is still tentative. The main determinant of the proceedings of the project team lies in the quality of parsed data obtained from the information extraction stage.

Acknowledgment

The project team would like to thank our advisor Professor Benjamin Kao for general guidance and support, explaining the rationale and importance of the project, providing high-level research and development directions.

Last but not least, we would like to thank our families and friends for their support and love.

Table of Contents

List of Figures

List of tables

List of Abbreviations

CRF	Conditional Random Field
GRU	Gated Recurrent Unit
HMM	Hidden Markov Model
IR	Information Retrieval
NER	Named Entity Recognition
NLP	Natural Language Processing
Protobuf	Protocol Buffer
Q&A System	Question Answering System
RNN	Recurrent Neural Network

1. Introduction

The laws of Hong Kong are primarily constitute of the Basic Law, Common Law, Rule of Equity, and Statue Law. The theory of Stare Decisis (Latin for “Let the decision stand”.) which Common Law is based on, stipulates that the rulings of judges are bounded by factually similar prior decisions [1]. Thus, scrutinizing court cases is an essential research process for legal professions. The research routine involves processing a sizeable sum of text and synthesizes the information court rulings entailed. Due to the dull and cumbersome nature of reviewing court cases, the common practice would be narrowing the focus to a finite number of precedents, hoping the most relevant findings would be accepted in court and their research captures sufficiently broad picture happened in the past. The research group believes that current natural language processing techniques is adequate to assist lawyers in the research process and could potentially automate the extraction process.

Developing technologies to assist human language tasks is an active research field within the domain of computer science. From search to automatic machine translation, methods that aim to make computers process human languages fall under the category of Natural Language Processing [2]. The advancement of NLP has been rapid, several leaps in the arena were witnessed. One of the breakthroughs is the neural-based methods, which was popularized in the last decade, it further reduces manual effort at the stage of designing language models [3], increased adaptability among language models. The task Named Entity Recognition (NER) is an epitome of the advancements in NLP task benchmarks. NER, a task of identifying entities, such as person, organization, location, etc. After applying neural-based methods, researchers could obtain a 90% of accuracy in English, a level which computer scientists consider solved the problem [4]. The development of NLP creates opportunities for computers to assist task with greater complexities — text processing with low degree of logical reasoning. Automating the process of reviewing court cases might be feasible under such circumstance.

With the task of analyzing court rulings automated, legal professionals would liberate from low value-added paperwork and their endeavors could focus on delivering value from high-level thinking, which could enhance productivity in general. Beyond such, legal practitioners could gain better insight, from the full landscape of past rulings, instead of only a selected review of court cases due to limited time and resources.

This study aims to design and implement software systems that extract relevant information from court cases, based on the currently available stack of NLP technologies. In phase 1, this project will concentrate on drug trafficking cases or other related types of court cases. For instance, the relating factors are the amount of drugs, the type of drugs, charges, mitigating factors, etc.

In the following, the report will briefly introduce related works and publications revolving in NLP and legal information extraction, methodologies planned to adopt in this project, tentative schedule of the project, and preliminary results obtained from experiments in the past semester.

2. Related Work

This chapter is dedicated to reviewing the prior works of information extraction in the legal field and its neighbouring work, and NLP technologies that have an affinity with the project, namely, vector representation of words, and Machine Comprehension Models. The following sections will discuss the findings of the literature review, then elucidate the conceptions of each NLP technology under existing plans.

2.1. Prior Works

Information extraction is an active research field among computer science, nevertheless, only a finite number of publications intersect with the legal domain. As a result, the project team lowered the screening criteria for literature review and categorized prior work into 3 main categories: 1.) Directly related but with subordinate lower technical ambition, 2.) Indirectly related within intersections of the realm of law and NLP, 3.) Applicable NLP technologies.

Existing publications and projects of information extraction on court cases are based on rule-based systems, a fixed set of predefined logic [5]. Typically focused on mining metadata, for instance, date, name of judges, etc. The emphasis of this project intend to extract information embedded within unstructured text in court cases. Hence, software developed to accomplish such a goal must be able to adapt to the variability of language usage within the rulings. Though, the prior work could still serve as a reference, when we preprocess the data and perform metadata extraction.

2.2. Vector Representation of Words

The vector representation of words, *or embeddings*, is a cornerstone of modern natural language processing, referring to words are being represented in real-valued vectors. The notion of representing words as vectors is that vectors could be processed by mathematical operators, which are the building blocks of computers. The conversion of words to vector

mainly stems from the distributional hypothesis: *linguistic items with similar distributions have similar meanings* [6]. It was famously articulated by the leading figure in linguistics, J. R. Firth, "You shall know a word by the company it keeps" [7]. In short, words that have similar meanings are likely to share similar context. As a result, words with similar context will be assigned with similar vector values by word embedding models.

The similarity between two words could be easily obtained numerically from word vectors. The cosine similarity is a prevalent measurement for gauging the semantic distance between words. For two words w_1 and w_2, the cosine similarity is given as follows:

$$\mathbf{similarity}(\mathbf{w_1}, \mathbf{w_2}) = \cos\boldsymbol{\theta} = \frac{\mathbf{w_1} \cdot \mathbf{w_2}}{||\mathbf{w_1}||\,||\mathbf{w_2}||}$$

The semantic similarity could be compared at a scientific and data-driven base with the help of word vectors. Nevertheless, word vectors have an apparent pitfall — unable to identify ambiguity of words. For instance, "*bank*" in a sentence could be referred to as financial institutions that provide credits, in the meantime, point to the river "*bank*". Thankfully, the introduction of contextualized word embeddings resolved the aforementioned problem. Contextualized word representation models look into the context and assign a value to words according to their relevant meaning, eschewing the loophole that trapped previous word representation models [8].

Word embeddings provide increased adaptability to this project in the process of search, enable computers to capture words and phrases with equivalent or similar semantic meanings. The project team believes word vectors would help in raising the rate and accuracy of extraction.

2.3. Machine Comprehension Models

Machine Comprehension (MRC), is a task attempts to teach machines to answer user proposed questions, of open domain or domain-specific. The approach to this task could be generalized into 2 genres, 1.) Information Retrieval-based factoid Question Answering, and,

2.) Knowledge-based Question Answering. Information retrieval methods are analogous to search, the IR method searches for relevant corpora and paragraphs from a large number of documents, then, extracting an answer from the text retrieved. Knowledge-based methods work like a database query, questions are converted from a semantic format into a structured query, as a result, the answer could be simply retrieved from a structured database [9]. Despite QA systems are now able to answer trivia questions, challenges are still lying ahead, current implementations are still eminently brittle to noise [10], which will be discussed in section 6.3, *The Performance of the QA System*.

3. Methodology

This chapter presents a practical solution to legal information extraction on drug trafficking court cases. Due to the robust performance of contextualized embeddings and machine comprehension models in Question and Answer domain, we adopt an ensemble approach, first to represent words as word embeddings, and then to use the embeddings in machine comprehension process, specifically Bidirectional Attention Flow (BiDAF) Model.

3.1. Problem Stated

Given a documented court case on drug trafficking, our task is to extract relevant information out and to answer corresponding questions. A sample case of the original data and the machine comprehension result is shown in Figure 1., where the input is a paragraph of the case and the output is the answer to the question asked.

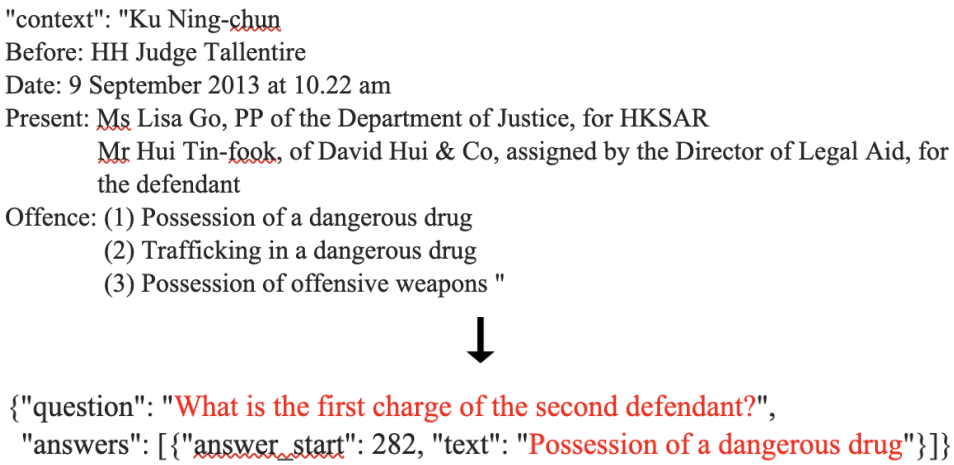

Figure 1 Example Court Case Question and Answering Data

3.2. Feature Extraction

The features to be extracted are features that can affect the judgement. They are selected by our supervisor's group with the help of legal professionals.

The Main features to be extracted are listed below with examples.

- Charge: "Trafficking in a dangerous drug".
- Ordinance: "section 4(1)(a) and (3) of the Dangerous Drugs Ordinance, Chapter 134" Since the ordinance is usually embedded into a special format, it could be easily extracted by named-entity recognition.
- Drug: "methamphetamine hydrochloride: 4.86 kilogrammes". Type of drug and the amount of corresponding drug is required. This is one of the hardest parts, since the numerical relationship of drug and its weight is hard to detect.
- Defendant Background: Age, name, occupation, education status, family, etc.
- Mitigating facts: Such facts include self-consume, good personality, remorse etc. Since they are implicitly recorded in the court cases, context comprehension is needed to extract.
- Aggravating facts: Such facts include international element, poor criminal record, etc. Context comprehension is needed to extract.
- Sentence: "28 years 9 months imprisonment". Since it is usually formatted, it could be easily extracted.

3.3. Contextualized Word Embeddings

Contextualized word embeddings is a task attempting to capture word semantics in different contexts to provide meaningful representations for words and the corresponding context. Such methods could help us better map the words into vectors of real values for later machine comprehension.

ELMo - Embeddings from Language Models

As indicated by name, ELMo is mainly structured by two parts, namely Bidirectional long short-term memory (BiLSTM) based language model and Task-specific word representation [11].

ELMo adopts a stacked multi-layer BiLSTM [11]. The bidirectional architecture allows the model to do both forward and backward reading, and the multi-layer structure allows the model to learn different characteristics of the language through different layers.

What is special about ELMo is that it is task-agnostic. The hidden state generated by BiLSTM is utilized in a specific given end task to generate vector representations of words, as shown in Figure 2.

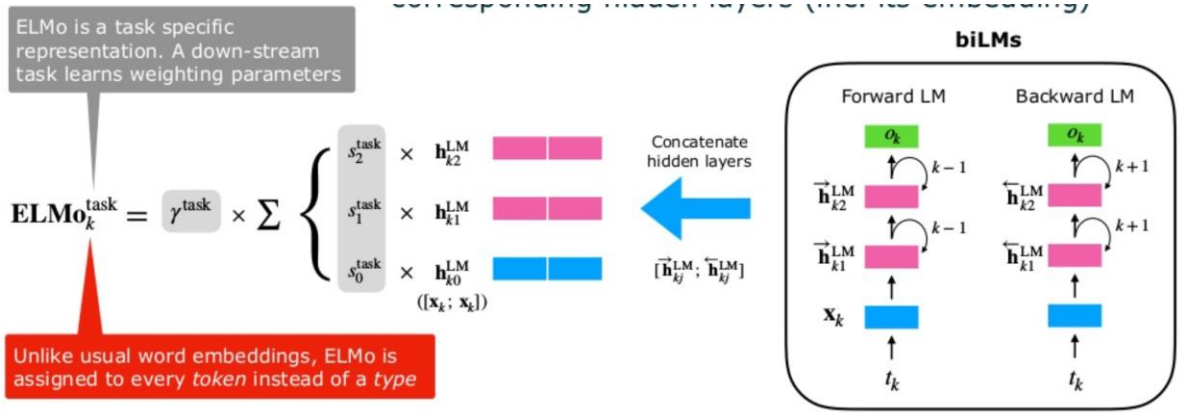

Figure 2 Diagram of the workflow of ELMo

BERT - Bidirectional Encoder Representations for Transformers

BERT, standing for Bidirectional Encoder Representations from Transformers, is a recent language representation model designed to pre-train deep bidirectional representations from unlabelled text [8]. It is a state-of-the-art model (see Figure 3) on many language tasks.

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

Figure 3 Performance of BERT in SQuAD 1.1

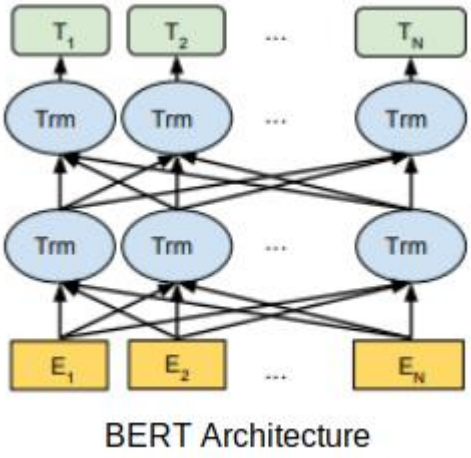

Figure 4 Architecture of BERT Word Vector Model

BERT is composed of transformers (see Figure 4), a mechanism that learns through context in text. A transformer usually consists of an encoder-decoder architecture, where encoder reads the text input and the decoder predicts [8]. In the context of BERT, it mainly focuses on encoder. The two innovations of BERT are that it is pre-trained through Masked Language Modelling (MLM), which masks part of the text for prediction, and Nest Sentence Prediction (NSP), which pairs consecutive sentences for prediction [8]. The MLM mechanism enables BERT to read from both left and right simultaneously (see Figure 5) and the NSP mechanism (see Figure 5) allows BERT to better understand the relationship among sentences. Such well-designed architecture promises its robustness through all kinds of language tasks and precise predictions.

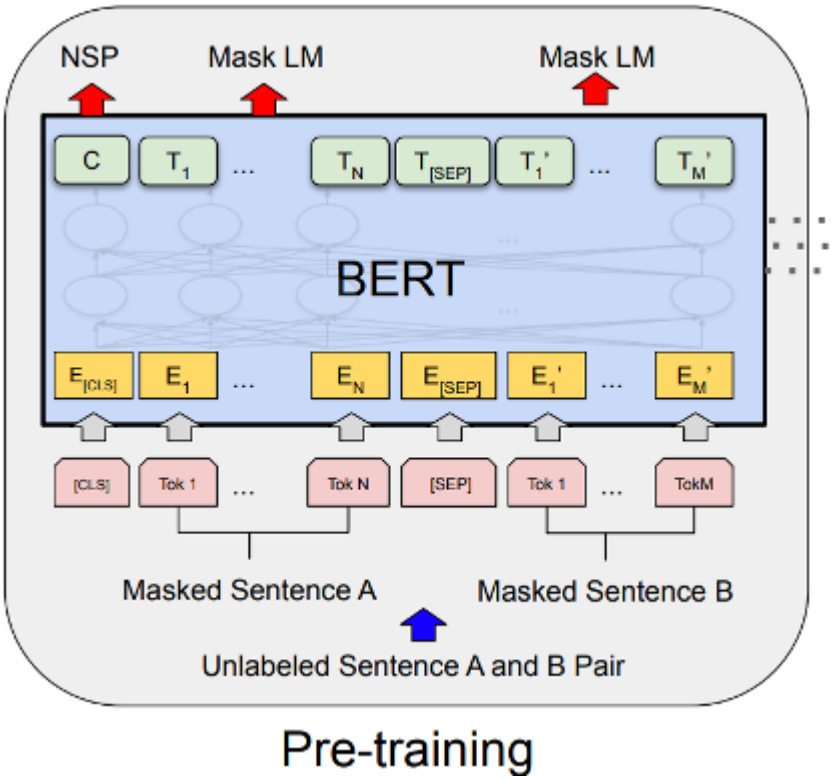

Figure 5 Mechanism of BERT

ALBERT – A Lite BERT

ALBERT (A Lite BERT) is an improved version of BERT that emphasis on scalability by introducing parameter-reduction techniques to reduce computational cost. The reduced computational cost is extremely beneficial in this project as computational resources are limited.

On top of BERT, ALBERT achieves parameter-reduction by introducing factorization of the embedding parametrization. Instead of the original design in BERT, the embedding matrix in ALBERT is split into WordPiece embedding and hidden-layer embeddings, where WordPiece embedding provides context-independent representations and hidden-layer embeddings provides context-dependent representations. This effectively reduce the embedding parameters.

In addition, ALBERT also eliminates the redundancy observed in BERT where multiple independent layers often carry out similar operations with different parameters. By

introducing cross-layer parameter sharing, parameters are shared across all the layers which significantly improve parameter efficiency.

These two design decisions of ALBERT allow for about 90% parameter reduction at the cost of little performance drop. The parameter reduction along with the minimal performance drop is essential for saving computational cost as well as scaling up the model, which the latter is proven to provide a performance gain that offset the performance loss caused by parameter-reduction by a large margin.

3.4. Bi-Directional Attention Flow Model (BiDAF)

The research team currently is using a Bidirectional Attention Flow (BiDAF) Model [12] for the task of machine comprehension. This model is being considered as the state of the art of its kind, outperforming other models substantially when the model was debuted.
BiDAF has a complex structure, hence, the research team has summarized three of the most important high-level features to illustrate the nature of BiDAF. First, it is a Long Short-Term Memory (LSTM) architecture. Secondly, it has a bi-directional stricture. Last but not least, from its name implies that it contains attention flow mechanism. The features aforementioned above are all designed for improving the performance and avoiding certain pitfalls exists in previously designed models. The details of the model will be elaborated as follows.

LSTM is a special kind of RNNs, an RNN that would discard irrelevant information and learn key features. RNNs are sequence models, their structures are genuinely flexible. They could take each word as input dynamically, adjusting the structure to fit the length of sentences or even passage. The formation of LSTM has an idiosyncratic nature compare with other RNNs. As shown in figure 4, the previous output of a state is passed on to the next state; hence, each state is taking all previous states as inputs. Overloading with a massive amount of data, the beauty of LSTM lies in its ability to forget irrelevant information. Equip with different gates within the network, LSTM could learn to divide relevant information into long term and short term, discarding the rest [13]. As a result, LSTM could be trained without being overloaded by a vast amount of data and being enforced to capture the most relevant features.

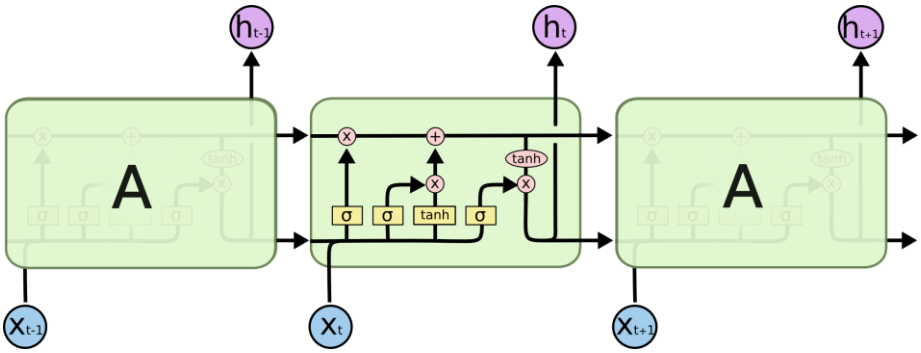

Figure 6 The structure of LSTM

Source: http://colah.github.io/posts/2015-08-Understanding-LSTMs

Bidirectional structure in LSTM is another breakthrough that aims to maximize the amount of data a model could process. Most neural networks are forward structured, in fact, from a technical point of view, all NNs are designed in a forward structure. The Achilles' heel of such a design is its inability to capture backward relationships. For example, referring to a previously mentioned concepts in a passage. The aforementioned referral creates a connection between the current sentence and the concept previously appeared sentence. Such sentence structures are eminently common to occur in writings. The incapability of forwarding neural networks substantially degrade the performance of MRC models. As a remedy, two RNNs are in use, as shown in figure 5. One matching relevant answer from left to right (L2R), another matching relevant answer from right to left (R2L). The key to merging the two outputs of these RNNs is a new algorithm "Synchronous Bidirectional Beam Search" [12]. The details of the algorithm are not the focus of our research, hence, interested readers might want to refer to the journal article cited below. In summary, the bidirectional structure in LSTM empowers the network to recognize connections between words in both forward and backward directions.

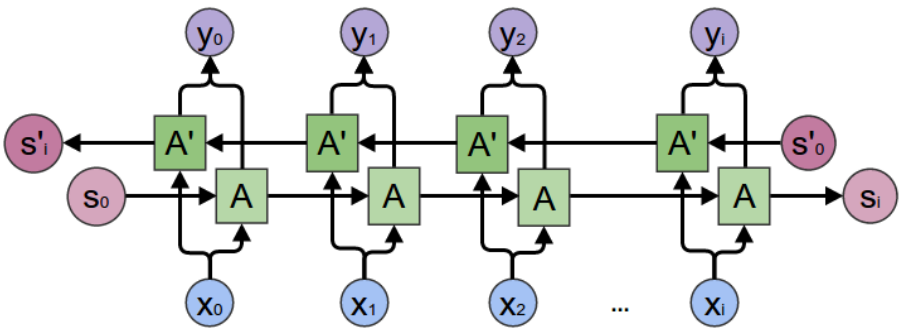

Figure 7 The architecture of bi-LSTM

Source: https://towardsdatascience.com/understanding-bidirectional-rnn-in-pytorch-5bd25a5dd66

Attention flow mechanism improves accuracy and robustness of the task machine comprehension, by mimicking human attention – focuses on relevant key points in sentences. RNNs, in contrast, scan through the whole passage given for identifying the answer in a passage. Even LSTM needs to read through the whole passage, though it discards irrelevant information. In comparison, human works differently, we read only chunks of the passage and digest the read chunks, then move on to another chunk and connect the two. The reading approach of human consumes much less memory and requires processing a significantly less amount of data. This observation sparks the idea of attention flow mechanism in LSTM particular.

Not all words worth equal attention, or equal weights in a model. Attention is trained to help models focus on the most relevant section of the text to render an answer from the question received. The attention flow mechanism of BiDAF model has two components, namely, Context-to-query attention (C2Q), Query-to-context (Q2C) attention. C2Q computes the relevance of each query word to each context word. Similarly, Q2C computes the relevance of each context word to each query word. The two results are in matrix form and will be merged by a multilayer perceptron [12]. Then feed the modelling layer, as shown in figure 6, for yielding the final result. By focusing on the main points in sentences, models are less likely to be confused by an enormous load of data, resulting in higher robustness. Attention flow essentially assigns higher weights to relevant words; thus, better performance could be achieved.

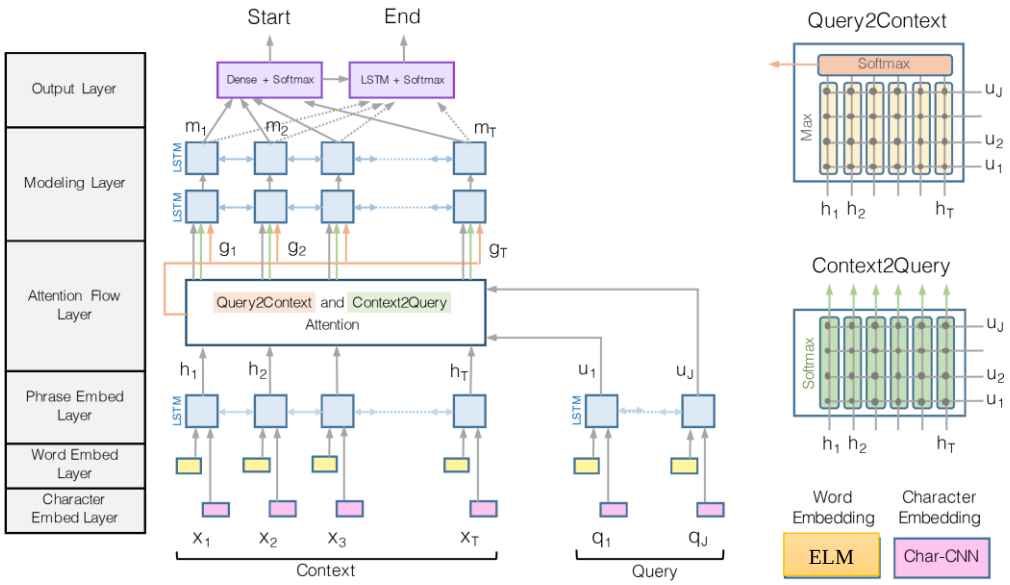

Figure 8 The architecture of the BiDAF model.

Source: https://allenai.github.io/bi-att-flow/

The embedding currently in use is a contextualized word embedding previously mentioned section 2.2, named Embeddings from Language Models (ELMo) [14]. It is closely related to the NER model in this project and the BiDAF model. They share similar structure and ideas. In short, the model used for training ELMo embeddings uses 2 layers of bidirectional LSTM. The details will be omitted for the sake of simplicity. The implication of using ELMo is significant. For the reason, that word embeddings plays a central role in NLP and the introduction of ELMo outperformed several benchmarks at that time [14]. The following figure lists an incomplete number of tasks, ELMo has achieved state-of-the-art (SOTA) performance. Stanford Question Answering Dataset (SQuAD) is the most recognized question answering performance measurement metric. ELMo's improvement on this metric implies that ELMo would likely improve machine comprehension combining with the BiDAF model.

Task	Previous SOTA		Our baseline	ELMo + Baseline	Increase (Absolute/Relative)
SQuAD	SAN	84.4	81.1	85.8	4.7 / 24.9%
SNLI	Chen et al (2017)	88.6	88.0	88.7 +/- 0.17	0.7 / 5.8%
SRL	He et al (2017)	81.7	81.4	84.6	3.2 / 17.2%
Coref	Lee et al (2017)	67.2	67.2	70.4	3.2 / 9.8%
NER	Peters et al (2017)	91.93 +/- 0.19	90.15	92.22 +/- 0.10	2.06 / 21%
Sentiment (5-class)	McCann et al (2017)	53.7	51.4	54.7 +/- 0.5	3.3 / 6.8%

Figure 9 Key Results of ELMo

Source: https://allennlp.org/elmo

The process of extraction will be presented with further details in section 6.3; thus, will be omitted in the following.

4. Preliminary Results and Analysis

The following section is dedicated to illustrating the experiments conducted in this project along with subsequent evaluations and analysis.

4.1. Experiment Set up

Dataset.
Stanford **Qu**estion **A**nswer **D**ataset (SQuAD) 1.1 is a standard reading comprehension dataset, mostly consists of questions from crowdworkers on Wikipedia articles. Segment of text, or *span*, from the reading passage are taken out from the corresponding reading passage.

HKU Labelled Court Case, court case data, is a dataset consists of labeled Hong Kong judgement. Labels are like the format of SQuAD, consists of segment of text, or *span*, from the corresponding judgement. In this project, labels are transformed to question form to fit the format of machine comprehension model.

Model Details.
Our model is highly similar to QA section of the ELMo embedding paper [14]. The model embeds tokens by concatenating ELMo word vectors and a character-derived embedding from a convolution neural network, which is part of BiDAF. The concatenated word embedding tokens, are then pass on to the BiDAF model described in the last section. Lastly, a linear layer is being fed with the results of BiDAF for predicting the start and end index of the answer.

4.2. Preliminary Results

After feeding with court case training data, the new trained model was evaluated with the validation dataset of court cases data. The metrics used for evaluation and the result are listed in the following.

Metrics.

Start Accuracy (Start Acc):

This metric measures the percentage of the start index matching the ground truth answer.

End Accuracy (End Acc)

This metric measures the percentage of the end index matching the ground truth answer.

Span Accuracy (Span Acc):

This metric measures the percentage of both start and end index matching the ground truth answer.

Exact Match (EM):

The EM metric measures the percentage of matching output string of the model and the answer string.

F1 Score (F1):

The F1 score measures the balance of precision and recall, more specifically, the overlap between the prediction and ground truth answer.

$$F_1 = \left(\frac{2}{\text{recall}^{-1} + \text{precision}^{-1}}\right) = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

Loss Value (Loss):

The loss value obtained from Adam optimizer during training.

Result.

Court Case Training Dataset

Start Acc	End Acc	Span Acc	Exact Match (EM)	F1	Loss
77.2%	**76.6**%	**66.0**%	**72.0**%	**86.0%**	**1.299**37

Figure 10 Evaluation Result of Court Case Training Dataset

Court Case Validation Dataset

Start Acc	End Acc	Span Acc	Exact Match (EM)	F1	Loss
74.9%	**74.4%**	**63.8**%	**70.6**%	**85.5**%	**1.526**60

Figure 11 Evaluation Result of Court Case Validation Dataset

Baseline: SQuAD 1.1 Dataset

EM	F1
81.0%	**87.4**%

Figure 12 Original Performance on SQuAD 1.1 dev dataset

4.3. Analysis

The result of training is slightly fall behind the expectation of the research team. ELMo + BiDAF model obtained an 81% of EM and 87 F1 score on SQuAD dataset. In comparison, the model trained with court case data was far behind on the EM metric with only 70.6% and a close match on F1 with 85.5%. This implies that newly trained model has most answers approximately correct, rather than exact fit.

The research team believes that court cases data are much more convoluted than SQuAD data, which lead to decline in EM metric. The SQuAD dataset consists of researchers cleansed Wikipedia question and answer data with a consistent style of answer. Nevertheless, court cases data were labelled with more lenient rules. Hence, there is reason to suspect that

the culprits of the significant decline of EM metric are the inconsistent manner of labelling and the convoluted structure of court cases data.

4.4. Limitations, Risks, and Assumptions

There 3 major limitations constraining the progress of the current project. 1.) Bounded computational resources, and 2.) limited knowledge in the legal domain.

4.4.1. Bounded Computational Resources

Due to the nature of the high dimensionality of language, training language model is computationally demanding, especially word embeddings. It was estimated that training the base model of BERT alone would cost $3000-6000. Hence, the research team may not able to afford the exorbitant price of training cost.

4.4.2. Limited Knowledge in the legal domain

All team members are originated from the realm of Computer Science, none of us possess any forms of access to legal training. As a result, there may exist a discrepancy between legal professionals and the project team of the focal points on court cases. In the worse scenario, the misinterpretation might lead the development direction to an unfruitful circumstance. Hence, the project team will proceed in trepidation, conduct research on available information and resources to reduce the probability of an unfavourable outcome.

5. Preliminary Results and Analysis

There are two major future works that each focuses on different aspects of the machine comprehension model.

5.1. Experiment on Other New Word Vectors

The only word embedding model used at this stage is ELMo, as it is integrated along with the AllenNLP suite, and thus, is chosen as the starting point of building the architecture. However, the ELMo model was proposed in March 2018 [14]. Since then, more language models have been developed and have achieved an improved theoretical performance upon EMLo. Two language models released in year 2019 are of interests, namely BERT and ALBERT, which the latter is also another improvement upon the former.

In the future stage of the project, time will be dedicated to swapping out ELMo for better performing language model and evaluating the performance gain. Since BERT is superseded by ALBERT, ALBERT will be directly incorporated into the BiDAF model as the embedding layer, and evaluation will be performed on the same ground as how ELMo was set up.

5.2. Comprehension model improvement

There are two improvements that could be made to machine comprehension model, namely multi-paragraph reading comprehension and handling unanswerable questions.

5.2.1. Multi-Paragraph Reading Comprehension

The existing problem the project aims to solve is to extract a list of defined features given a documented court case. However, court cases contain multiple long paragraphs that often span multiple pages. If the entire document is taken at once as an input sequence, the computational cost is expensive and training is not efficient. In the upcoming stage of the project, investigation and experimentation will be carried out to examine ways to alleviate the problem. A paper from 2018 by Clark and Gardner provides an insight on how to build an effective Multi-Paragraph Reading Comprehension model [15]. Two approaches can be taken, namely pipelined method and confidence method. The methods mainly revolve at selecting a paragraph according to certain metrics that is most likely to contain an answer in prior to feeding it to the model.

Time will be taken to examine and experiment how this approach can be implemented and evaluate whether this approach would be beneficial in the context of extracting features from court cases.

5.2.2. Handling unanswerable questions

Currently, the list of features to extract from each case is exhaustive, meaning all the features in the problem definition are attempted to be extracted from each case. This causes problems of false-positive prediction when there is actually no answer to the features for certain cases. For example, there may not be any aggravating factors in a court case. A recent paper provides a read and verify approach to handle cases that no answers can be inferred [16].

Conclusions

Reviewing court cases is a routine task of legal practitioners that consumes a fair amount of effort. The rationale of this project lies on the notion of automating such a task would greatly improve the productivity of legal professionals. At preparation stage, the project reviewed related publications and applicable NLP technologies, such as word embeddings, NER tagging, QA system. The research team proposed two systems to facilitate information extraction, static system and dynamic systems for extracting relevant information form court cases. The project team has also evaluated the limitations of current technologies, thus, would strive to mitigate the possible impacts of the constraints aforementioned. A detailed working plan is also listed in section 4. Subsequently, the report presented the preliminary results and corresponding evaluation of the current progress. The results of the majority of testing were satisfactory in general. Nevertheless, further research and developments are still needed for completing the project, in particular, experimenting on new word vectors. The team wishes to proceed to the model training stage and complete this task before the end of this year.

References

[1] P. Wesley-Smith, The Sources of Hong Kong Law, Hong Kong: Hong Kong University Press, 1994.

[2] J. Eisenstein, Introduction to Natural Language Processing, MIT Press, 2019.

[3] S. Ruder, "Frontiers of Natural Language Processing," 2018. [Online]. Available: https://drive.google.com/open?id=15ehMIJ7wY9A7RSmyJPNmrBMuC7se0PMP. [Accessed 1 February 2020].

[4] Vikas Yadav, Steven Bethard, "A Survey on Recent Advances in Named Entity Recognition from Deep Learning models," in *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, New Mexico, USA, 2018.

[5] Tin Tin Cheng, Jeffrey Cua, Mark Davies Tan, Kenneth Gerard Yao, Rachel E. O. Roxas, "Information extraction from legal documents," *Eighth International Symposium on Natural Language Processing*, pp. 157-162, 2009.

[6] M. Faruqui, "Proposal : Beyond the Distributional Hypothesis," 2015.

[7] M. Sahlgren, "The distributional hypothesis.," *Italian Journal of Disability Studies*, no. 20, pp. 33-53, 2008.

[8] Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova., "Bert: Pre-training of deep bidirectional transformers for language understanding," 11 Oct 2018. [Online]. Available: https://arxiv.org/abs/1810.04805. [Accessed 31 January 2020].

[9] Daniel Jurafsky and James H. Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition," New York, 2008.

[10] Christopher Clark, Matt Gardner, "Simple and Effective Multi-Paragraph Reading Comprehension," 29 October 2017. [Online]. Available: https://arxiv.org/abs/1710.10723. [Accessed 31 January 2020].

[11] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer, "Deep contextualized word representations," in *NAACL*, 2018.

[12] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, Hannaneh Hajishirzi, "Bidirectional Attention Flow for Machine Comprehension," in *ICLR*, 2017.

[13] Gers, Felix A., Jürgen Schmidhuber, and Fred Cummins, "Learning to forget: Continual prediction with LSTM," in *ICANN*, 1999.

[14] Peters, Matthew E. and Neumann, Mark and Iyyer, Mohit and Gardner, Matt and Clark, Christopher and Lee, Kenton and Zettlemoyer, Luke, "Deep contextualized word representations," in *NAACL*, 2018.

[15] Christopher Clark, Matt Gardner, "Simple and Effective Multi-Paragraph Reading Comprehension," 29 October 2017. [Online]. Available: https://arxiv.org/abs/1710.10723. [Accessed 31 January 2020].

[16] Minghao Hu, Furu Wei, Yuxing Peng, Zhen Huang, Nan Yang, Dongsheng Li, "Read + Verify: Machine Reading Comprehension with Unanswerable Questions," in *AAAI*, 2019.

[17] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer, "Deep contextualized word representations," in *NAACL*, 2018.