# Legal Judgment Prediction with Multi-Stage Case Representation Learning in the Real Court Setting

Luyao Ma[1,2], Yating Zhang[2], Tianyi Wang[2], Xiaozhong Liu[3†], Wei Ye[1†]
Changlong Sun[2], Shikun Zhang[1]
[1]National Engineering Research Center for Software Engineering, Peking University
[2]Alibaba Group
[3]Indiana University Bloomington
1701210338@pku.edu.cn;ranran.zyt@alibaba-inc.com;will.wty@alibaba-inc.com
liu237@indiana.edu;wye@pku.edu.cn;changlong.scl@taobao.com;zhangsk@pku.edu.cn

## ABSTRACT

Legal judgment prediction(LJP) is an essential task for legal AI. While prior methods studied on this topic in a pseudo setting by employing the judge-summarized case narrative as the input to predict the judgment, neglecting critical case life-cycle information in real court setting could threaten the case logic representation quality and prediction correctness. In this paper, we introduce a novel challenging dataset[1] from real courtrooms to predict the legal judgment in a reasonably encyclopedic manner by leveraging the genuine input of the case – plaintiff's claims and court debate data, from which the case's facts are automatically recognized by comprehensively understanding the multi-role dialogues of the court debate, and then learnt to discriminate the claims so as to reach the final judgment through multi-task learning. An extensive set of experiments with a large civil trial data set shows that the proposed model can more accurately characterize the interactions among claims, fact and debate for legal judgment prediction, achieving significant improvements over strong state-of-the-art baselines. Moreover, the user study conducted with real judges and law school students shows the neural predictions can also be interpretable and easily observed, and thus enhancing the trial efficiency and judgment quality.

## CCS CONCEPTS

• **Computing methodologies → Discourse, dialogue and pragmatics**; **Multi-task learning**; • **Applied computing → Law**.

## KEYWORDS

judgment prediction, case life-cycle, multi-task learning

**ACM Reference Format:**
Luyao Ma[1,2], Yating Zhang[2], Tianyi Wang[2], Xiaozhong Liu[3†], Wei Ye[1†] and Changlong Sun[2], Shikun Zhang[1]. 2021. Legal Judgment Prediction with Multi-Stage Case Representation Learning in the Real Court Setting . In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3404835.3462945

## 1 INTRODUCTION

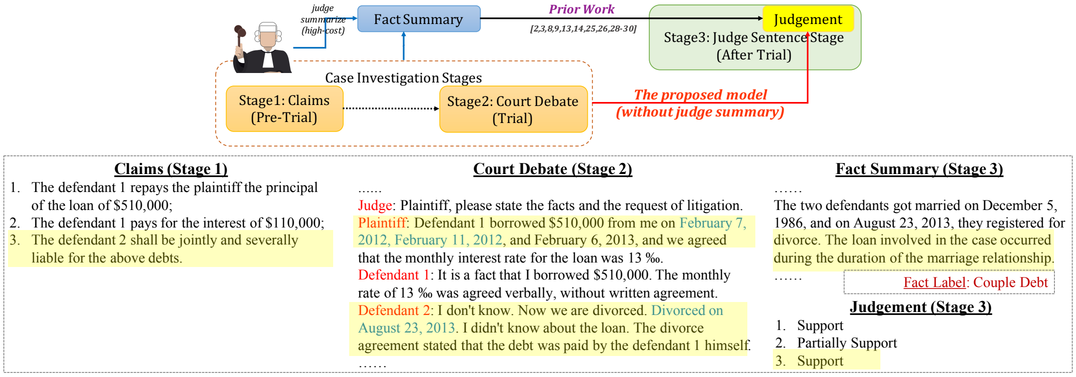
Argus, the Greek mythological giant with hundred eyes, could be an appropriate metaphor to characterize an ideal legal judgment predictor, which is expected to comprehensively examine the case from different views and across various stages of the case life-cycle. Neglecting any detailed information, e.g., admissibility of evidence, narrative from witnesses, and response from plaintiff/defendant, could mislead the prediction outcome. While legal judgment prediction has been originally proposed in 1960s (entitled "Using Simple Calculations Predict Judicial Decisions [15]"), unfortunately, prior studies [2, 3, 6–8, 11–13, 22, 26, 27, 29–31] ignored the multi-stage nature of the legal case, instead researching in a pseudo setting which regards the judge-summarized case narrative as input to predict the judgment results. The neglect of critical case life-cycle information in real court setting, however, could threaten the case logic representation quality and prediction correctness.

In this study, in order to recover the case jigsaw puzzle, we introduce a novel challenging dataset collected from real courtrooms as well as propose an innovative neural model to integrate pretrial claims and court debate. More importantly, the case's facts are automatically recognized by comprehensively understanding the colloquial and lengthy court debate, and then learning to discriminate each claim so as to reach the final judicial conclusion.

For a litigation process, a case life-cycle often experiences three critical stages (depicted in Fig. 1): pre-trial claim collection stage (e.g., plaintiff provides narrative to judge for the target case), trial court debate stage (e.g., plaintiff, defendant, witness, lawyer and judge debate on the court focusing on the claims), and after trial judge sentence stage (judge generates verdict, often including case fact summary and judgement). In the first and second stages, the judge usually spends high cost in identify legal facts from court debate transcript and evidence materials so as to further make final judgement in stage 3. Thus a comprehensive case life-cycle representation learning can be nontrivial for legal prediction.

[1] https://github.com/mly-nlp/LJP-MSJudge
† Corresponding authors

*SIGIR '21, July 11–15, 2021, Virtual Event, Canada*
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8037-9/21/07...$15.00
https://doi.org/10.1145/3404835.3462945


judge summarize (high-cost)
Fact Summary
Prior Work
[2,3,8,9,13,14,25,26,28-30]
Judgement
Stage3: Judge Sentence Stage (After Trial)
Case Investigation Stages
Stage1: Claims (Pre-Trial)
Stage2: Court Debate (Trial)
The proposed model (without judge summary)

**Claims (Stage 1)**

1. The defendant 1 repays the plaintiff the principal of the loan of $510,000;
2. The defendant 1 pays for the interest of $110,000;
3. The defendant 2 shall be jointly and severally liable for the above debts.

**Court Debate (Stage 2)**

......

Judge: Plaintiff, please state the facts and the request of litigation.

Plaintiff: Defendant 1 borrowed $510,000 from me on February 7, 2012, February 11, 2012, and February 6, 2013, and we agreed that the monthly interest rate for the loan was 13 ‰.

Defendant 1: It is a fact that I borrowed $510,000. The monthly rate of 13 ‰ was agreed verbally, without written agreement.

Defendant 2: I don't know. Now we are divorced. Divorced on August 23, 2013. I didn't know about the loan. The divorce agreement stated that the debt was paid by the defendant 1 himself.

......

**Fact Summary (Stage 3)**

......

The two defendants got married on December 5, 1986, and on August 23, 2013, they registered for divorce. The loan involved in the case occurred during the duration of the marriage relationship.

......

Fact Label: Couple Debt

**Judgement (Stage 3)**

1. Support
2. Partially Support
3. Support

**Figure 1: Conceptual overview of the difference between previous work and MSJudge.**

To achieve this goal, the biggest challenge lies in the difficulty of accurately representing the multi-role court debate, where different camps may not necessarily share the same vocabulary space, and classical NLP algorithms can hardly consume this variation. For instance, the judge can be more responsible for investigating the facts and reading the court rules while the other litigants answer the questions from the judge. Moreover, with opposite position, plaintiff and defendant's attitudes, sentiments and descriptions to the same topic can be quite different. The second barrier comes from the gap between the litigants' statements during the court debate and the facts to be recognized by the judge. There exist much non-judicial content and noisy logic in court debate. As the exemplar case shown in Fig. 1, whether the third claim should be supported or not is highly rely on whether the loan is occurred during the defendants' marriage (as described in fact summary), thus the model should learn to identify such fact from the court debate even though much noisy information (e.g., *divorce*, *paid by the defendant 1 himself*) may distract such recognition. Only the statements convincing enough with thorough evidences to be supported can be regarded as identified facts for further making judgment prediction. Therefore, discoverability of the truth/facts from colloquial dialogue content is an essential factor in the task of legal judgment prediction. The last but not the least is the challenge of representing the relations/interactions among the debate, facts, claims and judgment. In civil cases, the judgment can be generalized as the answer to the claims while it is common to have multiple claims in one case and whether they are established or not is not relatively independent.

Motivated by such observations, in this paper, we propose a novel neural judgment prediction model by addressing multi-stage representation learning called MSJudge (**M**ulti-**S**tage **Judge**ment Predictor). It jointly learns the identification of the legal facts in the court debate and simultaneously predicts the judgment result of each claim. As aforementioned example, due to the content and logic gap existing between court debate and fact summary, an auxiliary supervision by the fact enables to filter the noisy and non-judicial content from stage 1 and 2 for further better judgment prediction. Fig. 2 depicts the architecture of MSJudge. In a joint learning process, various kinds of information collected from different stages, e.g., court debate, facts, claims, are encapsulated to regularize the judgment prediction of the civil case through the fine-tuning of the court debate representation. To make the prediction results interpretable and being easily observed, we provide visualization for accessing the mutual influence between different components.

In summary, this paper makes the following key contributions: (1) We take a concrete step towards augmenting judicial decision making by investigating a novel task of judgment prediction through the case life-cycle data (Section 3), which is essential for the practical use of AI techniques in a real court setting. (2) We propose an end-to-end framework called MSJudge (Section 4) that operates in a manner of multi-task supervision with multi-stage representation learning. The proposed model enables to address the judgment prediction by exploring and visualizing the interactions/mutual effect between "debate and fact", "fact and claim" and "across claims". It summons a practical scenario for legal judgement automation. (3) MSJudge is trained in a supervised manner where the training data is extracted from over 70$k$ court records of civil trials along with their judgment documents. Our experimental study (Section 6) demonstrates superiority of MSJudge over competitive baselines. The proposed method achieves 86.5% in micro $F\_1$ score on the multi-stage trial datase, and significantly outperforms the best performing baselines (3.6% increase in performance). Meanwhile, in the experiment, the proposed approach challenged upper bound idealization (pseudo setting), i.e., using judge summarized case fact as input. The equivalent prediction performance (86.6% in micro $F_1$ score) grants us confidence that the proposed model is able to optimize the case representation across different stages for more practical use in real court setting. Importantly, we also show its effectiveness in judicial decision making with a user study involving real law school students and judges (Section 6.5).

## 2 RELATED WORK

### 2.1 Legal Judgment Prediction

Recently, legal judgment prediction has addressed much attention and achieved great progress. Thanks to the accessibility of the large amount of legal judgment data[2], a rising number of work has been dedicated to this research topic. However, they simplified the task to predict the court's outcome given a text describing the facts of a legal case manually summarized by the judge based on the case materials [2, 3, 7, 8, 12, 13, 26, 27, 29–31]. Xiao et al. [23] proposed to take the fact description as input and predict

[2] European Court of Human Rights (ECHR): https://echr.coe.int/Pages/home.aspx?p=home; China Judgements Online: http://wenshu.court.gov.cn/

**Table 1: Comparison among several state-of-the-art work. Note that the meaning of the colors is the same as in Table 2. 🟨 🟩 🟦 stand for employing the data at stage 1 (claim), stage 2 (court debate) and stage 3 (fact summary), respectively. ⬛ represents to employ the fact labels at stage 3 for supervision in training phase.**

| Study | Task | Case Type | Methodology | Stage | | |
|---|---|---|---|---|---|---|
| Few-Shot[7] | Charge Prediction | Criminal Case | an attribute-attentive charge prediction model, which employs discriminative attributes to learn attribute-aware fact representation. | | | 🟦 |
| TOPJUDGE[30] | Charge Prediction<br>Law Prediction<br>Term Prediction | Criminal Case | a topological multi-task learning framework for LJP, which formalize the dependencies among subtasks as a Directed Acyclic Graph. | | | 🟦 |
| LJP-QA[31] | Charge Prediction | Criminal Case | a reinforcement learning method by iteratively questioning and answering to provide interpretable results for legal judgment prediction. | | | 🟦 |
| LADAN[26] | Charge Prediction<br>Law Prediction<br>Term Prediction | Criminal Case | use a novel graph distillation operator (GDO) to extract discriminative features for effectively distinguishing confusing law articles. | | | 🟦 |
| Autojudge[12] | Judgment Prediction | Civil Case | formalize the task as Legal Reading Comprehension according to the legal scenario. | 🟨 | | 🟦 |
| **MSJudge** | Judgment Prediction<br>Judicial Fact Recognition | Civil Case | release a case life-cycle dataset and propose a multi-task learning framework by leveraging multi-stage judicial data with consideration of the interactions among claims, court debate and fact recognition. | 🟨 | 🟩 | ⬛ |

**Table 2: Definition of Notations**

| | |
|---|---|
| $D$ | a debate dialogue containing $n$ utterances |
| $U_i$ | the $i$-th utterance in $D$ |
| $S_i$ | the text content of $U_i$ |
| $r_i$ | the role of the speaker in $U_i$(i.e. judge, plaintiff, defendant and witness) |
| $w_{it}^{u}$ | the $t$-th word in $S_i$ |
| $C$ | a set of claims of the case |
| $c_j$ | the $j$-th claim in $C$ |
| $w_{jv}^{c}$ | the $v$-th word in $c_j$ |
| $y_p^f$ | the predicted probability of $p$-th fact |
| $y_j^c$ | the predicted probability distribution of $j$-th claim |

charges by using a criminal case dataset named CAIL in Chinese. Zhong et al. [30] constructed a topological network to capture the dependencies among subtasks via multi-task learning relying on the fact description as well. Chalkidis et al. [2] released the first large-scale English legal judgment prediction dataset. Compared to the previous work, we are the first to perform judicial decision making through court debate and pre-trial claim data where we comprehensively examine the case across various stages of the case life-cycle.

Table 1 compares the state-of-the-art works related to judgment prediction over different dimensions, including the task to be solved, case type, methodology (highlight) as well as the phases of dataset used in the study. It is clear that most of the previous work are conducted on stage 3 where the judge-summarized fact summary is the algorithm input. To prove the effectiveness of our model in optimizing the case representation across different stages to challenge the idealization scenario, we compare our proposed model with four representative methods [7], [26], [2] and [12]. [7] represents the faction of using discriminative legal attributes for judgment prediction which emphasizes more on the judicial fairness during prediction. [26] stands for the research works based on distinguishing law articles for judgment prediction which has been proved to be effective especially for criminal cases. [2] leverages BERT to focus only on learning good representation of the pure input fact text for judgment prediction. [12] works also on the civil cases as ours and despite the fact summary, it also employs the claims of the case and the related law articles in the verdict as input.

## 2.2 Multi-Task Learning

The usage of multi-task learning models has become ubiquitous for many natural language processing tasks. For example, Cao et al. [1] leverages text classification to improve the performance of multi-document summarization via joint learning. Guo et al. [5] improved abstractive text summarization with the auxiliary tasks of question generation and entailment generation. Nishida et al. [16] used the QA model for answer selection and extractive summarization for evidence extraction in a neural multi-task learning framework. In the field of legal judgment prediction, multi-task framework has been widely adopted for learning several target objectives simultaneously [2, 8, 13, 26, 27, 30]. In this work, we introduce a framework with a "bionic design" to conduct judicial admissibility inspection (main task) across different stages via the supervision of the recognized facts by the judge (auxiliary task).

# 3 PRELIMINARIES

In this section, we first introduce the problem addressed in this paper. Then we provide an overview of the MSJudge framework to address it. The set of key notations used in this paper is given in Table 2. Note that $U_i$, $r_i$, $S_i$, $w$ and $c$ represent the embedding representations of the corresponding variables in the table.

## 3.1 Problem Formulation

Let $D = \{U_1, U_2, \cdots, U_n\}$ denote a court debate with $n$ utterances where each utterance $U_i$ consists of a sequence of $l$ words $S_i = \{w_{i1}, w_{i2}, \cdots, w_{il}\}$ and the role of its speaker $r_i$. Each case has a set of $k$ claims $C = \{c_1, c_2, \cdots, c_k\}$ where each claim is composed of a sequence of $q$ words $c_j = \left\{w_{j1}, w_{j2}, \cdots, w_{jq}\right\}$. Based on the debate and claims of a case, the main task aims to predict judgment results $Y^c = \left\{y_1^c, y_2^c, \cdots, y_k^c\right\}$ of all claims. In order to imitate the judge's decision process, the model predicts $z$ fact labels $Y^f = \left\{y_1^f, y_2^f, \cdots, y_z^f\right\}$ simultaneously as an auxiliary task to improve the results of the main task.

## 3.2 Overview

As formulated above, in civil litigation, the judgment prediction is conducted on each claim, thus the proposed case representation is rooted in those claims (from stage 1) and intensified with the recognized facts from court debate (from stage 2). Modeling the interactions among claims, debate and recognized facts becomes essential in judgment prediction especially in case life-cycle scenario.

Fig. 2 depicts the architecture of MSJudge. It learns multi-stage context representation and capturing the mutual effect across different stages of data to further make judgment with respect to the claims. Specifically, it is composed of the following three major components.

- *Multi-Stage Context Encoding* simulates a judge to parse and understand the court debate and its pre-trial claims.
- *Multi-Stage Content Interaction* simulates a judge's need to find clues and diagnose the correlations of the key information in the multi-stage content (e.g., claims, court debate). We model the interaction between utterances and claims, interaction between facts and claims, as well as the interaction across claims to enhance the claim representations.
- *Fact Recognition and Judgment Prediction* identifies the legal fact by considering all the information mined in the previous step and then make final judgment with respect to each claim.

# 4 THE MSJUDGE FRAMEWORK

## 4.1 Multi-Stage Context Encoding

*4.1.1 Debate Utterance Encoder.* Given an utterance $U_i$ with $l$ words $S_i = \left\{w^u_{i1}, w^u_{i2}, \cdots, w^u_{il}\right\}$ and the role of its speaker $r_i \in R$, we first embed the words to vectors to obtain $\hat{S_i} = \left\{\mathrm{w}^u_{i1}, \mathrm{w}^u_{i2}, \cdots, \mathrm{w}^u_{il}\right\}$ where $\mathrm{w}^u \in \mathbb{R}^d$ and employ role embedding to encode the role. The role embedding $\mathrm{e}^r_i \in \mathbb{R}^r$ is randomly initialized and jointly learnt during the training process.

To involve the role information into the utterance, we concatenate the role information with each word in the utterance, which is able to project the same word into different dimensional spaces w.r.t.the target role. We hypothesize that the same word may need differentiate when different speakers use it.

$$\mathrm{e}^u_{it} = \mathrm{w}^u_{it} \oplus \mathrm{e}^r_i,\ t \in [1, l] \tag{1}$$

where $\oplus$ denotes a concatenation operation and then the dimention of $\mathrm{e}^u_{it}$ is $(d + r)$.

Then we utilize a bidirectional-LSTM to encode the semantics of the utterance while maintaining its syntactic.

$$\mathrm{h}^u_{it} = [\overrightarrow{\mathrm{LSTM}^U(\mathrm{e}^u_{it})}, \overleftarrow{\mathrm{LSTM}^U(\mathrm{e}^u_{it})}],\ t \in [1, l] \tag{2}$$

where $\mathrm{h}^u_{it}$ is the $t$-th word in $i$-th utterance's representation.

To strengthen the relevance between words in an utterance, we employ the attention mechanism to obtain $\mathrm{U}_i$, which can be interpreted as a local representation of an utterance:

$$\begin{aligned} \mathrm{U}_i &= \sum_{t=1}^{l} \alpha^u_{it}\mathrm{h}^u_{it} \\ \alpha^u_{it} &= \frac{\exp(\mathrm{Q}^u\mathrm{h}^u_{it})}{\sum_{t=1}^{l}\exp(\mathrm{Q}^u\mathrm{h}^u_{it})} \end{aligned} \tag{3}$$

where $\mathrm{Q}^u$ are learnable parameters and all parameters in utterance encoder are shared across utterances.

*4.1.2 Debate Dialogue encoder.* To represent the global context in a dialogue, we use another bidirectional-LSTM to encode the dependency between utterances to obtain a global representation of an utterance, denoted as $\overline{\mathrm{U}_i}$.

$$\overline{\mathrm{U}_i} = [\overrightarrow{\mathrm{LSTM}^D(\mathrm{U}_i)}, \overleftarrow{\mathrm{LSTM}^D(\mathrm{U}_i)}],\ i \in [1, n] \tag{4}$$

where $\overline{\mathrm{U}_i}$ is the $i$-th utterance's global representation.

*4.1.3 Pre-trial Claim Encoder.* Similar to the utterances, we encode the claims via bidirectional-LSTM and use attention mechanism to obtain the local representations of claims. We share word embedding matrix across the utterance encoder and the claim encoder.

$$\begin{aligned} \mathrm{h}^c_{jv} &= [\overrightarrow{\mathrm{LSTM}^C(\mathrm{w}^c_{jv})}, \overleftarrow{\mathrm{LSTM}^C(\mathrm{w}^c_{jv})}],\ v \in [1, q] \\ \mathrm{C}_j &= \sum_{v=1}^{q} \alpha^c_{jv}\mathrm{h}^c_{jv} \\ \alpha^c_{jv} &= \frac{\exp(\mathrm{Q}^c\mathrm{h}^c_{jv})}{\sum_{v=1}^{q}\exp(\mathrm{Q}^c\mathrm{h}^c_{jv})} \end{aligned} \tag{5}$$

where $\mathrm{C}_j$ is the $j$-th claim's representation and $\mathrm{Q}^c$ are learnable parameters and the parameters are shared across claims.

## 4.2 Multi-Stage Content Interaction

*4.2.1 Debate-to-Claim.* Utterance vectors are stacked and regarded as an utterance memory $\mathrm{m}^u = \left\{\overline{\mathrm{U}_1}, \overline{\mathrm{U}_2}, \cdots, \overline{\mathrm{U}_n}\right\}$. We compute attention weights where each weight indicates the correlation between a claim vector $\mathrm{C}_j$ and an utterance memory unit $\mathrm{m}^u_i$.

$$\begin{aligned} \mathrm{O}^u_j &= \sum_{i=1}^{n} \alpha^d_{ji}\overline{\mathrm{U}_i} \\ \alpha^d_{ji} &= \frac{\exp(\mathrm{C}_j\overline{\mathrm{U}_i})}{\sum_{i=1}^{n}\exp(\mathrm{C}_j\overline{\mathrm{U}_i})} \end{aligned} \tag{6}$$

where $\mathrm{O}^u_j$ is the output vector of the interaction between utterance memory and a claim.

*4.2.2 Debate-to-Fact.* As introduced above, the identified judicial facts are the key factors for the judge to make decisions of the target case, thus we conduct judicial fact recognition as auxiliary task (see Sec. 4.3.2). To discover the projection of different facts on the certain dialogue fragments in a debate, we employ attention mechanism to obtain fact representation $\mathrm{f}_p$ for the $p$-th fact.

$$\begin{aligned} \mathrm{f}_p &= \sum_{i=1}^{n} \alpha^r_{pi}\overline{\mathrm{U}_i} \\ \alpha^r_{pi} &= \frac{\exp(\mathrm{Q}^r_p\overline{\mathrm{U}_i})}{\sum_{i=1}^{n}\exp(\mathrm{Q}^r_p\overline{\mathrm{U}_i})} \end{aligned} \tag{7}$$

where $\mathrm{Q}^f_p$ are learnable parameters.

*4.2.3 Fact-to-Claim.* Similar to utterance memory, the fact vectors are multiplied by the prediction probability and then we stack the fact vectors of the facts as a fact memory $\mathrm{m}^f = \left\{\overline{\mathrm{f}_1}, \overline{\mathrm{f}_2}, \cdots, \overline{\mathrm{f}_z}\right\}$. Moreover, we attentively sum the fact memory to produce output representation.

$$\begin{aligned} \mathrm{O}^f_j &= \sum_{p=1}^{z} \alpha^f_{jp}\overline{\mathrm{f}_p} \\ \alpha^f_{jp} &= \frac{\exp(\mathrm{C}_j\overline{\mathrm{f}_p})}{\sum_{p=1}^{z}\exp(\mathrm{C}_j\overline{\mathrm{f}_p})} \end{aligned} \tag{8}$$

where $\mathrm{O}^f_j$ is the output vector of the interaction between fact memory and a claim.

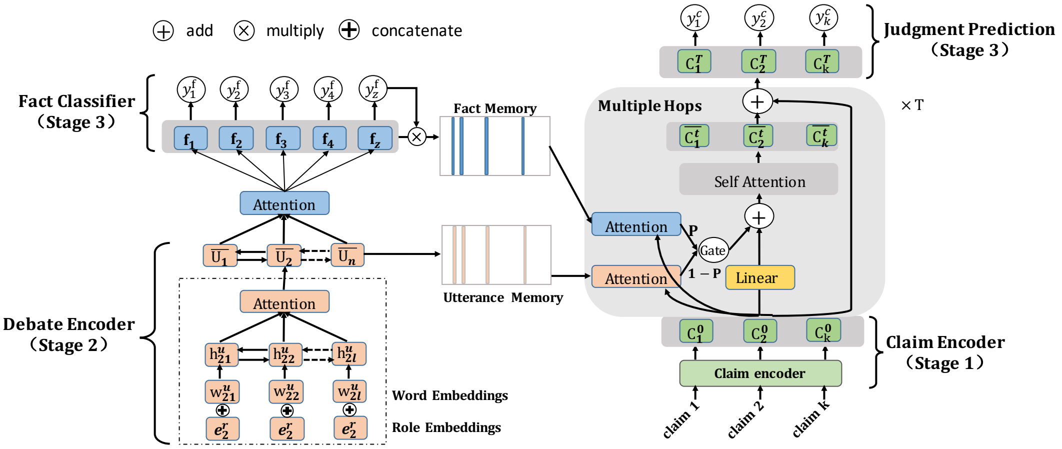


**Figure 2: Architecture of MSJudge. The model is divided into three modules: (1) Multi-Stage Context Encoding: encoding claim and debate. (2) Multi-stage Content Interaction: capturing the correlations among claim, debate and fact. (3) Fact Recognition and Judgment Prediction: identifying the legal facts and then make judgment with respect to each claim.**

#### 4.2.4 *Fusion.*

For each claim, we obtain the output $O^u_j$ from utterance memory and the output $O^f_j$ from fact memory. We use gate mechanism to control the weight of the two vectors to pass to the next layer. We further apply a linear layer with Rectifier Liner Unit (ReLU) to obtain $\hat{C}_j$. After the addition, we get $\overline{C_j}$ as the claim representation via memory blocks.

$$\begin{aligned} g_j &= \sigma(W^u O^u_j + W^f O^f_j + b^g) \\ \hat{C}_j &= \mathrm{ReLU}(W^l C_j + b^l) \\ \overline{C_j} &= \hat{C}_j + g_j * O^u_j + (1 - g_j) * O^f_j \end{aligned} \tag{9}$$

where $\sigma$ is sigmoid activation function, $W^u$, $W^f$, $W^l$, $b^g$ and $b^l$ are trainable parameters shared across claims.

#### 4.2.5 *Across-Claim.*

As aforementioned it is common to have multiple claims in one case and whether they are established or not is not relatively independent, it is necessary to model the dependency across claims. Technically, we employ self attention mechanism to capture the relationships across claims.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{10}$$

where $Q \in \mathbb{R}^{d_k \times k}, K \in \mathbb{R}^{d_k \times k}, V \in \mathbb{R}^{d_k \times k}$ are query, key, value which are the same vector.

We take the stack of claim vectors $\overline{C^c} = \left\{\overline{C_1}, \overline{C_2}, \cdots, \overline{C_k}\right\}$ as input to a self-attention layer with residual connections.

Moreover, we employ multiple ($T$) hops (denoted as the grey block in Fig. 2) in our model where the output of the previous hop is considered as the input of next hop. Previous works [18, 20] have proved the usage of multiple hops in memory network could yield learn the deep abstraction of text.

## 4.3 Task-specific Decoders

#### 4.3.1 *Main Task: Judgment Prediction.*

After $T$ hops updates, we obtain the final claim representation $C^T_j$ for $j$-th claim and feed it to the softmax layer for judgment prediction.

$$y^c_j = \mathrm{softmax}(W^c C^T_j + b^c) \tag{11}$$

We train our model in an end-to-end manner by minimizing the cross-entropy loss.

$$\mathcal{L}_c = -\frac{1}{k}\sum_{j=1}^{k}\sum_{d=1}^{|Y_c|} g^c_{jd}\log(y^c_{jd}) \tag{12}$$

where $g^c_{jd}$, $y^c_{jd}$ are the ground truth and the predicted probability of $d$-th class for $j$-th claim for each training instance, respectively.

#### 4.3.2 *Auxiliary Task: Judicial Fact Recognition.*

As for the judicial fact recognition task, we feed fact representation $f_p$ to the sigmoid layer to predict the probability of a fact should be recognized.

$$y^f_p = \mathrm{sigmoid}(W^f_p f_p + b^f_p) \tag{13}$$

Similarly, the training loss is constructed as:

$$\mathcal{L}_f = -\frac{1}{z}\sum_{p=1}^{z}\left[g^f_p\log(y^f_p) + (1 - g^f_p)\log(1 - y^f_p)\right] \tag{14}$$

where $g^f_p$, $y^f_p$ are the ground truth and the predicted fact recognition probability of the $p$-th fact for each training instance.

Finally the total loss is the sum of these two losses:

$$\mathcal{L}_{total} = \mathcal{L}_c + \mathcal{L}_f \tag{15}$$

# 5 EXPERIMENT SETTINGS

## 5.1 Dataset Construction

In the experiment, we collected 70, 482 cases of Private Lending category. Each case includes plaintiff's claims, court debate records and judgment verdict. In total, it contains more than 4.1 million utterances and 133, 209 claims. On average, each case contains 58.17 utterances and 1.89 claims and the average number of words of debates is 950 compared to the average number of words of fact summary is 143. The ratio of category labels[3] in main task is 1 : 2.6 : 10.9. We define 10 fact labels in auxiliary task[4]. The labeling of the fact label is based on the fact finding section in the verdict, and the judgment result is annotated based on the judgment

[3] The labels of main task classifier: *reject*, *partially support* and *support*

[4] The labels of auxiliary task classifier: *Agreed Loan Period*, *Couple Debt*, *Limitation of Action* , *Liquidated Damages*, *Repayment Behavior*, *Term of Guarantee*, *Guarantee Liability*, *Term of Repayment*, *Interest Dispute*, *Loan Established*

paragraph in the verdict. All the annotations are conducted by legal experts with three rounds of training led by civil judges. In the end, the experts achieved Kappa coefficient = 0.8 (substantial agreement). To the best of our knowledge, this is the very first large aligned civil trial court debate and judgment document (case life-cycle) dataset. We show two complete and real life-cycle materials of the cases after removing sensitive information[5] (claims, court records, and their verdicts) to expose how the judges hear the case and make judgment in practical situation. We will release all the experiment data to motivate other scholars to further investigate this problem.

## 5.2 Training Details

The dimensions of word embeddings and role embeddings are set to 300. Word embeddings are trained using the Skip-Gram model[14] on the debate dialogues and role embeddings are randomly initialized. We tune the performance by following the grid search tuning method and cross-validation. The size of hidden states of bidirectional-LSTM is 256. The neural networks are trained using Adam Optimization[10] with a learning rate set to 0.001, and perform the mini-batch gradient descent with a batch size of 16. The dropout is set to 0.8.

To minimize judicial discrimination, we preprocess the data and replace all the personal information (e.g., person name, ID number, address, and gender) with special characters before the model training. During the experiment, we adopted cross-validation to ensure the rationality of the models. We will release our code as well as the dataset for reproducibility.

## 5.3 Evaluation Metrics

We use Macro $F_1$ and Micro $F_1$ (Mac.$F_1$ and Mic.$F_1$ for short) as the main metrics for algorithm evaluation. In a multi-class classification setup, macro-average reflects the robustness of the model if there exists class imbalance. Note that as for all the baselines, we set the debate content concatenated with each claim of the case as input[6] and the judgment result for each claim as output. As for the proposed methods, we are capable of predicting the judgment results of all the claims of a case at once, thus in each sample we conduct $k$ multi-class classification tasks where $k$ is the number of claims in a case.

## 5.4 Tested Methods

### 5.4.1 Baselines.

To extensively validate the effectiveness of the proposed model, the following baselines under different scenarios are employed for comparison.

We first testify several variants of encoders by setting same input type as MSJudge using claim and court debate from stage 1 and 2:

- *Traditional machine learning based method*
  **TFIDF+SVM** is a robust multi-class classification by means of TFIDF plus SVM [19].
- *Deep learning based methods*
  **TextCNN** is a convolutional neural networks trained on top of pre-trained word vectors[7] for sentence-level classification tasks [9]. **BiGRU-ATT** employs Bi-directional GRU with attention mechanism [24] to capture context semantics and automatically selects important features through attention during training. For **TextCNN** and **BiGRU-ATT**, we use entire debate content as input. **HAN** stands for Hierarchical Attention Network [28] which is a hierarchical text classification model with two levels of attention mechanisms for aggregating words to utterance and utterances to dialogue. **Transformer**[21] is based solely on attention mechanisms. We use the encoder part of the transformer encoding entire debate content, followed by classifiers.
- *Multi-task learning methods*
  To validate the superiority of joint learning with auxiliary task (judicial fact recognition), we conduct experiments for the above models with the multi-task framework. These models are denoted as **TextCNN-MTL**, **BiGRU-ATT-MTL**, **Transformer-MTL** and **HAN-MTL**.

We then compare several state-of-the-art (SOTA) methods on judgment prediction.

- *Discriminative fact attributes based method*
  **Few-Shot(fact)**[8] [7] proposes to discriminate confusing charge, which extracts predefined attributes from fact descriptions to enforce semantic information. In the experiment, we use the ten fact labels in the auxiliary task of a case.
- *Law articles based method*
  **LADAN(fact)**[9] [26] introduces a novel graph distillation operator (GDO) to extract discriminative features for distinguishing confusing law articles. The original work is in a multi-task framework in criminal cases. In our civil case settings, it is trained by the two objectives: judgment prediction and law prediction.
- *Content based method*
  **BERT(fact)**[10] [4] is a fine-tuning representation model which has been applied to learn good representation of the input fact summary for judgment prediction [2]. In the experiment, we take the representation of "[CLS]" as aggregated representation and add a softmax layer on the top of BERT for judgment prediction.
  **Autojudge(fact)** [12] formalizes the task of judgment prediction as a reading comprehension task where the claims, fact and the related law articles are injected as input. For fair comparison, the input of the referenced law articles is removed to avoid information leakage during judgment prediction.

We also evaluate their models by replacing the fact summary with the case life-cycle data (claims, debate and fact labels) as conducted in our settings (the rows **Few-Shot-MTL**, **LADAN-MTL** and **Autojudge-MTL** in Table 3). The results, then, can comprehensively validate the methodological and data hypotheses.

Note that all the baselines tested under the real court setting (see Table 3) shared the same claim and debate encoder as MSJudge to make fair comparison.

### 5.4.2 Variants of our proposed method.

To comprehensively evaluate each component of the proposed method, we make several variants to address the corresponding issue.

**MSJudge-MTL** is our proposed method in multi-task framework. **MSJudge** is the single task version by removing the entire fact related parts (auxiliary task and fact memory). **MSJudge(fact)** takes the judge-summarized fact at stage 3 as input.

[5] https://github.com/mly-nlp/LJP-MSJudge

[6] If a case contains $k$ claims, then such case forms to $k$ samples in which each sample input is the combination of the debate content and one of the $k$ claims.

[7] Skip-gram model[14] is utilized for pretraining word representations.

[8] https://github.com/thunlp/attribute_charge

[9] https://github.com/prometheusXN/LADAN

[10] https://github.com/google-research/bert

**Table 3: Main Results of All Tested Methods for the Main Task. Note that the average scores shown at rows MSJudge(fact), MSJudge and MSJudge-MTL are statistically significant different from the corresponding value of all the baseline models (* denotes the $p$-value<0.05, † denotes the $p$-value<0.01). 🟨 🟩 🟦 stand for employing the data at stage 1 (claim), stage 2 (court debate) and stage 3 (fact summary), respectively. ▨ represents to employ the fact labels at stage 3 for supervision in training phase.**

| | Method | Average | | | | Stage | | |
|---|---|---|---|---|---|---|---|---|
| | | Mac.P | Mac.R | Mac.F1 | Mic.F1 | 1 | 2 | 3 |
| Pseudo setting | Few-Shot(fact)[7] | 76.3 | 69.6 | 72.5 | 84.2 | 🟨 | | 🟦 |
| | LADAN(fact)[26] | 76.8 | 67.4 | 71.2 | 83.7 | | | |
| | Autojudge(fact)[12] | 76.1 | 69.4 | 72.3 | 83.9 | | | |
| | BERT(fact) | 76.5 | 72.9 | 74.6 | 84.2 | | | |
| | **MSJudge (fact)** | **77.7*** | **73.4*** | **75.8*** | **86.6*** | | | |
| Real court setting | TFIDF+SVM | 72.6 | 53.6 | 58.7 | 79.9 | 🟨 | 🟩 | |
| | TextCNN | 70.9 | 62.0 | 65.5 | 81.1 | | | |
| | Transformer | 73.9 | 63.8 | 67.7 | 82.0 | | | |
| | BiGRU-ATT | 75.5 | 66.7 | 70.3 | 83.1 | | | |
| | HAN | 75.8 | 67.0 | 70.4 | 83.2 | | | |
| | **MSJudge** | **76.3*** | **71.4*** | **73.6*** | **84.6*** | | | |
| Real court setting-MTL | TextCNN-MTL | 70.8 | 62.8 | 66.1 | 81.4 | 🟨 | 🟩 | ▨ |
| | BiGRU-ATT-MTL | 75.5 | 68.0 | 71.2 | 83.3 | | | |
| | Transformer-MTL | 74.8 | 68.9 | 71.5 | 82.2 | | | |
| | HAN-MTL | 75.7 | 69.0 | 71.8 | 83.1 | | | |
| | Few-Shot-MTL[7] | 76.1 | 69.6 | 72.3 | 83.5 | | | |
| | LADAN-MTL[26] | 74.5 | 66.6 | 69.8 | 83.1 | | | |
| | Autojudge-MTL[12] | 72.5 | 71.8 | 72.2 | 83.2 | | | |
| | **MSJudge-MTL** | **77.5†** | **72.5†** | **74.8†** | **86.5†** | | | |

# 6 RESULT DISCUSSION

## 6.1 Overall Performance

To evaluate the performance of the proposed model, we export the results from the following perspectives:

**Comparison against the baselines in real court setting.** Table 3 summarizes the performance of all the tested methods for the main task. The column "*Stage*" indicates the case life cycle segment(s) used for prediction. Based on the results, the following observations are recorded: (1) In real court setting group, it is not surprising to see that the traditional machine learning based methods didn't perform well in terms of $F_1$ score. It indicates the importance of legal case representation learning for better judgment prediction. Among the deep learning based baselines, **HAN** outperforms the other "single-level" models for both macro $F_1$ and micro $F_1$ scores which indicates the necessity of using hierarchical context representation to capture the dependency within words, utterance, and dialogue in the court debate scenario. (2) With the real court settings, we also employ multi-task training process (judgment prediction and judicial fact recognition) (see group Real court setting-MTL) for **TextCNN-MTL**, **BiGRU-ATT-MTL**, **Transformer-MTL** and **HAN-MTL**, all of them show positive effects compared to their single task mode, which proofs the validity of employing multi-task training process in the judgment prediction task. Compared with SOTA methods (**Few-Shot-MTL**, **LADAN-MTL**, **Autojudge-MTL**), Since MSJudge-MTL is purposely designed to jointly learn the tasks of judgment prediction and fact recognition together. It outperforms all the other tested methods under case life-cycle scenario. Another finding worth to mention is that **LADAN-MTL** does not perform well in civil case scenario in that the law articles applied in criminal cases indicate the corresponding criminal charge but in civil cases the law articles are usually for reference use.

**Table 4: The comparison on the performance of judicial fact recognition with different input types.**

| Input Type / Fact Label | Court Debate | | Fact Summary | |
|---|---|---|---|---|
| | *Mic.F*$_1$ | *Mac.F*$_1$ | *Mic.F*$_1$ | *Mac.F*$_1$ |
| Agreed Loan Period | 73.1 | 70.6 | 83 | 81.9 |
| Couple Debt | 96.8 | 59.7 | 92.3 | 75.1 |
| Limitation of Action | 82.2 | 70.1 | 95.3 | 93.7 |
| Liquidated Damages | 90.1 | 75.3 | 95.9 | 91.3 |
| Repayment Behavior | 76.4 | 71.5 | 87.5 | 85.6 |
| Term of Guarantee | 93.2 | 67.7 | 98.5 | 94.6 |
| Guarantee Liability | 90.3 | 77.6 | 94.9 | 88.1 |
| Term of Repayment | 84.2 | 55.6 | 83 | 81.9 |
| Interest Dispute | 79.1 | 79.1 | 93.3 | 93.3 |
| Loan Established | 90.5 | 79.8 | 96.2 | 92.4 |
| Averaged | 85.6 | 83.7 | 93 | 92.3 |

**Comparison with upper bound in pseudo setting** In the experiment, we also let the proposed approach challenge the upper bound conducted in pseudo setting, i.e., using judge summarized case fact as input (see Group pseudo setting as shown in Table 3). The proposed model with fact summary as input outperforms the state-of-the-art baselines which indicates the interactions between claims and fact summary as well as the interactions across the claims can be also effective in the idealization scenario. Meanwhile, the equivalent prediction performance between **MSJudge(fact)** and **MSJudge-MTL** grants us confidence that the proposed model is able to optimize the case representation across different stages.

Various types of inputs, either court debate data or fact summary (see Table 4), are compared by leveraging the results of judicial fact recognition (the auxiliary task). As aforementioned, because of content/logic gaps between the two stages' data, the judicial fact recognition results tell the similar pattern. By exploring the bad cases, one can, sometimes, highlight misalignments between court debate and judge summarized case fact, e.g., some evidence and material (from fact) are not mentioned in the court debate.

**Component Assessment via Ablation Test.** To assess the contribution of different components in the proposed method, we conduct ablation tests in Table 5. To validate the influence of the interaction between fact and claim as illustrated in Sec. 4.2.3, we remove the attention layer between fact memory and claim (see the row "w/o fact memory")[11]. Similarly, the influence of utterance on representing claims and the impact across the claims are demonstrated in the rows "w/o utterance memory" and "w/o self attention" respectively. Table 5 clearly tells that all the components contribute positively to the results. To be specific, the *utterance memory* feature has largest impact - their removal causes 22.6% relative increase in error (RIE) for macro $F_1$ score. The interaction across claims shows significant impact for judgment prediction which demonstrates the claims are highly correlated. In addition, the *role* information also has great influence on the judgment prediction tasks over dialogues. Experimental results prove that multi-stage case representation learning can be critical for legal AI system.

## 6.2 Convergence Analysis

To further validate the performance of the proposed model, we conduct convergence analysis to monitor the changes and trends during training for different variants of the proposed model. As Fig. 4 depicts, one can observe that the performance of the model with training on all components is consistently superior than the models removing a particular component.

[11] Note that in this setting, the fact classifier still exist for regularizing the debate encoder during supervision.

Table 5: Ablation Test.

| Models | Mic.F1 | Mac.F1 | RIE(%) |
|---|---|---|---|
| MSJudge-MTL | 86.5 | 74.8 | – |
| w/o role embeddings | 85.2 | 74.0 | 3.2 |
| w/o utterance memory | 82.8 | 69.1 | 22.6 |
| w/o fact memory | 84.9 | 74.6 | 0.79 |
| w/o self attention | 83.6 | 72.5 | 9.1 |

Table 6: Effects of Hops

| #Hops | Micro F1 | Macro F1 |
|---|---|---|
| hops(1) | 85.2 | 74.3 |
| hops(2) | 85.4 | 74.5 |
| **hops(3)** | **86.5** | **74.8** |
| hops(4) | 85.0 | 73.8 |
| hops(5) | 84.8 | 73.3 |
| hops(6) | 84.3 | 73.1 |

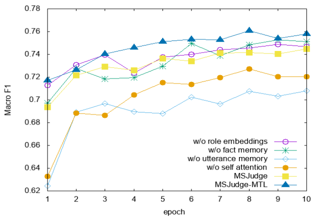


**Figure (4) The performance of tested methods over epochs.**

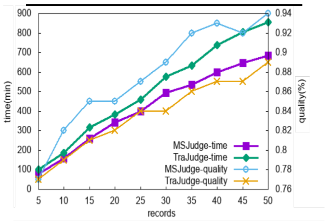
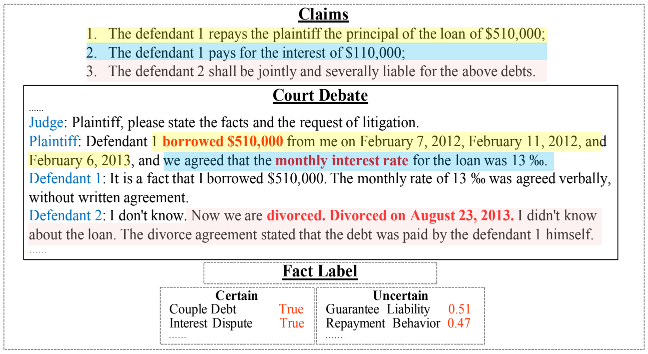


**Figure (5) User Study: Judgment Efficiency and Quality.**



Claims

1. The defendant 1 repays the plaintiff the principal of the loan of $510,000;
2. The defendant 1 pays for the interest of $110,000;
3. The defendant 2 shall be jointly and severally liable for the above debts.

Court Debate

......

Judge: Plaintiff, please state the facts and the request of litigation.

Plaintiff: Defendant 1 **borrowed $510,000** from me on February 7, 2012, February 11, 2012, and February 6, 2013, and we agreed that the **monthly interest rate** for the loan was 13 ‰.

Defendant 1: It is a fact that I borrowed $510,000. The monthly rate of 13 ‰ was agreed verbally, without written agreement.

Defendant 2: I don't know. Now we are **divorced. Divorced on August 23, 2013.** I didn't know about the loan. The divorce agreement stated that the debt was paid by the defendant 1 himself.

......

Fact Label

| Certain | | Uncertain | |
|---|---|---|---|
| Couple Debt | True | Guarantee Liability | 0.51 |
| Interest Dispute | True | Repayment Behavior | 0.47 |
| ...... | | ...... | |

**Figure 5: User Study: example of context enhanced by MSJudge.**

## 6.3 Effect of Multiple Hops

Table 6 shows the performances of our model with 1 to 6 memory hops where MSJudge-MTL(t) means our model using $t$ memory hops. The results show that our model with 3 memory hops achieves the best result and then starts decreasing as the number of hops increases, which might be due to the loss of generality with the increase of model complexity[20, 25, 32]. Note that the parameters are shared over hops, thus the amount of parameters will not increase as the number of hops increases. We use the parameters at best hop for experimental results and visualization in Fig. 6.

## 6.4 Model Interpretability

To help readers better consume our algorithm outcomes, Fig. 6 depicts a case study by visualizing all intermediate results at multiple stages for interpretations. In this case, the plaintiff raised in total 6 claims to initialize the case where the claims 3, 4, 5 get judicial support and the rest are partially supported. Our predictions for these claims match the ground truth perfectly. Fig. 6(a) depicts the dependency across claims which is plotted by the weights on the self-attention layer as described in Section 4.2.5. In this case we can observe claims 3, 4, 5 have similar patterns due to their common petition of joint and several liability for the three guarantors. Similarly, Fig. 6(c) visualizes the correlation between claims and fact labels[12], where claims 3, 4, 5 were demonstrated to be highly correlated to almost the same set of fact labels about guarantee. Fig. 6(b) shows the importance of the utterances to the corresponding claims[13] as well as the key phrases highlighted within each utterance[14].

Table 7: User Study:open survey on MSJudge based on age

| Questions | 18-28 | 29-40 | 41-65 |
|---|---|---|---|
| MSJudge can be helpful in fact prediction. | 4.3 | 4.1 | 3.9 |
| MSJudge can capture the key information. | 3.8 | 3.8 | 4.1 |
| MSJudge can be used in real court setting. | 3.8 | 4.1 | 3.3 |
| MSJudge can help reduce the workload of trial judges. | 3.7 | 4.0 | 3.4 |

## 6.5 User Study

To justify the role of MSJudge in assisting the judges to make judicial decisions in real court setting, we conduct a user study by exploring the judgment effect with/without MSJudge. We select 12 judges as subjects who were either judges or law school students. They are from three age groups (18-28, 29-40, 41-65) with four people in each group, where half of them use MSJudge (denoted as *MSJudge Judge*), and the other half do not (denoted as *Traditional Judge*).

We randomly selected 50 real cases[15] and let the subjects to make judicial decisions. For each case, the subject should provide the legal fact recognized in this case and the final judgment to each claim. The judgment effect was evaluated after they finish every 5 cases. The MSJudge facilitates the decision making by automatically highlighting the most relevant keyword or sentence in the court debate with respect to each claim according to debate-to-claim attention as described in Section 4.2.1. The *MSJudge judge* can then quickly capture the key information in the court debate for making judgment, rather than go over all the context. In addition, The MSJudge displays the predicted legal facts with their certainty/uncertainty as illustrated in Section 4.3.2. We define those facts with a probability greater than 0.7 as certain facts, and those with a probability between 0.45 and 0.55 as uncertain facts. Thus those uncertain ones warn the subjects to focus on the potential controversies before making judgment. An example of enhanced context with predicted facts is shown in Fig.5. The *Traditional judge* instead need to read through the entire court debate and analyze the logic by themselves. Note that all the subjects enabled to check the correct answers (the ground truth of legal facts and judgments) once they finish every 5 cases.

We evaluate the judgment effect of the two approaches from two perspectives: (a) **Judgment Efficiency** measures the time the subject consuming on analyzing the case till making judicial decisions. The lesser the time cost, the more efficient). (b) **Judgement quality** estimates whether the subject can correctly identify legal facts, and at the same time make correct judgments based on the identified legal facts. Thus the judgment quality was formalized as the weighted average of the accuracy of legal fact recognition and the accuracy of judgment to each claim.

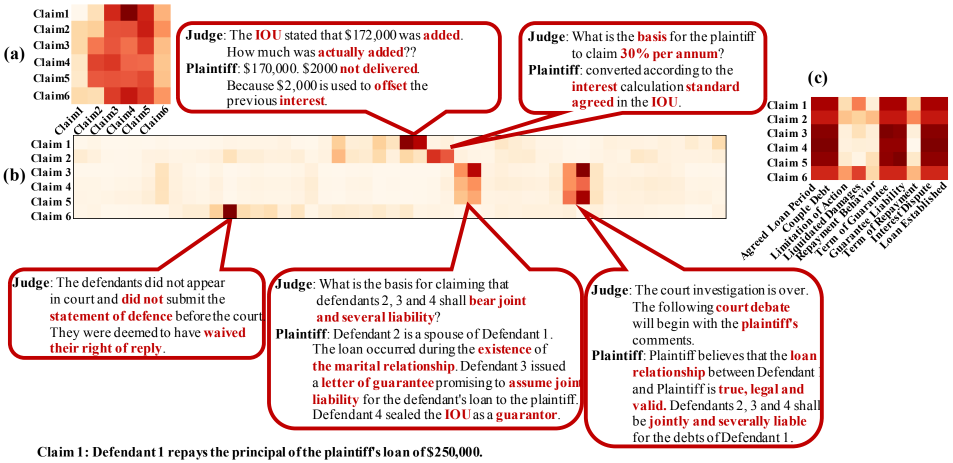
Fig. 5 depicts the change curves of two groups in terms of consuming time and judgment quality, with respect to the increase of the testing cases. It is worth mentioning that as the number of case increases, the growth rate of the consuming time of MSJudge

[12] The weights on fact memory block.

[13] The weights on utterance memory block where the darker the color, the more important a certain utterance is to a target claim.

[14] The value on the attention layer of debate encoder.

[15] Each case contains the claims from plaintiff and the corresponding court debate transcript, which performs the same setting with the input of the proposed MSJudge model.



**Claim 1: Defendant 1 repays the principal of the plaintiff's loan of $250,000.**
**Claim 2: Defendant 1 pays the plaintiff interest of $37,500 from July 16, 2015 to January 16, 2016 at the rate of 30% per annum until the actual repayment of the loan.**
**Claim 3: Defendant 2 shall be jointly and severally liable for the debts of Defendant 1 in the above-mentioned claim.**
**Claim 4: Defendant 3 shall be jointly and severally liable for the debts of Defendant 1 in the above-mentioned claim.**
**Claim 5. Defendant 4 shall be jointly and severally liable for the debts of Defendant 1 in the above-mentioned claim.**
**Claim 6. The litigation expenses in this case shall be borne by the four defendants.**

**Figure 6: Model Interpretability. (a) depicts the dependency across claims; (b) describes the importance of the utterances/their key phrases to the claims; and (c) shows the correlation between facts and claims.**

(the curve MSJudge-time) tends to decline in contrast to the curve TraJudge-time. In other words, the learning efficiency gradually increases for MSJudge. Moreover, the *MSJudge subjects* are found to achieve higher judgment quality (see curve MSJudge-quality) compared to the *traditional subjects* (see curve TraJudge-quality).

In addition, we surveyed the *MSJudge subjects* with several open questions as shown in Table 7, to assess the opinions on the usage of MSJudge in real court setting. For each question, Likert scale of 1 to 5 needs to be selected; the higher the score, the more agreement to a statement. Based on the survey results, we found that most people think MSJudge can be helpful in identifying legal facts from colloquial and lengthy court debate, and it accelerates the process of locating key information for making final judgment. As for the practical use of MSJudge in real court to reduce the workload of judges, the age range in the 18-28 and 29-40 responded positively, while the 41-65 group had some reservations. But they still thought it is possible to involve the active supervision of a judge while using MSJudge as assistance in the real court.

## 6.6 Error analysis

For the bad cases, 12% of the errors come from the fact identification when the court debate does not contain the utterances focusing on the corresponding fact which might be discovered only in the evidences (those data are not available). In such context, the evidence analysis can be another promising future work for the case life-cycle learning. Meanwhile, according to the confusion matrix, we also find that the specificity of the semantics in the court debate impairs the model performance when discriminating the labels of "partially support" and "support".

## 6.7 Ethical Statement

Lastly, we would like to discuss ethical concerns of our work. We are aware of potential risks of structured social biases that can be repeated or even enhanced in any machine learning system trained on large-scale uncontrolled datasets [17]. The need to assure equality, judicial impartiality, and judicial diversity must be properly addressed, and is a topic of great significance in judicial decision making. To combat these concerns, we anonymized the data by removing sensitive information (e.g., gender, race, etc.). We also applied the technique of oversampling to cope with infrequent questions to be generated to assure well-balanced training dataset. Despite the potential bias, the potential system error would be as follows: a) recognizing a wrong legal fact and b) generating a wrong judgment result. As for these concerns, in the user study, we managed to provide the users with the certainty/uncertainty of the identified facts to show some warnings for those uncertain facts. In addition, we suggest the future automatic judgment prediction system should be under Man-machine inclusive mode, which allows users to make corrections at the key points (e.g., legal fact identification) before the last step (e.g., judgment prediction) to ensure the correctness of the final conclusion. And indeed, when we tried to manually correct the answer at fact identification step, the performance judgment prediction increases significantly.

# 7 CONCLUSION

Performing case life-cycle admissibility inspection over court debate can be practically useful to assist the judges to adjudicate cases. In this work, we introduce a novel and challenging dataset addressing the life-cycle case representation technique. An end-to-end framework MSJudge is proposed which is in a manner of multi-task learning process. The empirical findings validate our hypothesis that joint learning with auxiliary task can improve the performance over state-of-the-art approaches. Additionally, MSJudge can more accurately characterize the interactions among claims, fact and debate for judgment prediction, achieving significant improvements over strong state-of-the-art baselines. Moreover, the user study conducted with judges and law school students shows the neural predictions can also be interpretable and easily observed, and thus enhancing the trial efficiency and judgment quality.

# 8 ACKNOWLEDGMENTS

This research was supported by the National Key Research And Development Program of China (2018YFC0830200,2018YFC0830206, 2019YFB1405802,2020YFC0832505).

## REFERENCES

[1] Ziqiang Cao, Wenjie Li, Sujian Li, and Furu Wei. 2017. Improving multi-document summarization via text classification. In *Thirty-First AAAI Conference on Artificial Intelligence.*

[2] Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. 2019. Neural Legal Judgment Prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 4317–4323.

[3] Huajie Chen, Deng Cai, Wei Dai, Zehui Dai, and Yadong Ding. 2019. Charge-Based Prison Term Prediction with Deep Gating Network. *arXiv preprint arXiv:1908.11521* (2019).

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 4171–4186.

[5] Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2018. Soft Layer-Specific Multi-Task Summarization with Entailment and Question Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 687–697.

[6] Charles M Haar, John P Sawyer Jr, and Stephen J Cummings. 1977. Computer power and legal reasoning: A case study of judicial decision prediction in zoning amendment cases. *Law & Social Inquiry* 2, 3 (1977), 651–768.

[7] Zikun Hu, Xiang Li, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2018. Few-shot charge prediction with discriminative legal attributes. In *Proceedings of the 27th International Conference on Computational Linguistics*. 487–498.

[8] Xin Jiang, Hai Ye, Zhunchen Luo, WenHan Chao, and Wenjia Ma. 2018. Interpretable rationale augmented charge prediction system. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*. 146–151.

[9] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1746–1751.

[10] Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

[11] Reed C Lawlor. 1963. What computers can do: Analysis and prediction of judicial decisions. *American Bar Association Journal* (1963), 337–344.

[12] Shangbang Long, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2019. Automatic judgment prediction via legal reading comprehension. In *China National Conference on Chinese Computational Linguistics*. Springer, 558–572.

[13] Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. 2017. Learning to Predict Charges for Criminal Cases with Legal Basis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2727–2736.

[14] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[15] Stuart Nagel. 1960. Using simple calculations to predict judicial decisions. *American Behavioral Scientist* 4, 4 (1960), 24–28.

[16] Kosuke Nishida, Kyosuke Nishida, Masaaki Nagata, Atsushi Otsuka, Itsumi Saito, Hisako Asano, and Junji Tomita. 2019. Answering while Summarizing: Multi-task Learning for Multi-hop QA with Evidence Extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2335–2345.

[17] Evaggelia Pitoura, Panayiotis Tsaparas, Giorgos Flouris, Irini Fundulaki, Panagiotis Papadakos, Serge Abiteboul, and Gerhard Weikum. 2017. On Measuring Bias in Online Information. *SIGMOD Rec.* 46, 4 (2017), 16–21.

[18] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*. 2440–2448.

[19] Johan AK Suykens and Joos Vandewalle. 1999. Least squares support vector machine classifiers. *Neural processing letters* 9, 3 (1999), 293–300.

[20] Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect Level Sentiment Classification with Deep Memory Network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 214–224.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. *Advances in Neural Information Processing Systems* 30 (2017), 5998–6008.

[22] Frederick Bernays Wiener. 1962. Decision prediction by computers: Nonsense cubed—and worse. *American Bar Association Journal* (1962), 1023–1028.

[23] Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, et al. 2018. Cail2018: A large-scale legal dataset for judgment prediction. *arXiv preprint arXiv:1807.02478* (2018).

[24] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*. 2048–2057.

[25] Nan Xu, Wenji Mao, and Guandan Chen. 2019. Multi-Interactive Memory Network for Aspect Based Multimodal Sentiment Analysis. In *Thirty-Third AAAI Conference on Artificial Intelligence*.

[26] Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. 2020. Distinguish Confusing Law Articles for Legal Judgment Prediction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 3086–3095.

[27] Wenmian Yang, Weijia Jia, XIaojie Zhou, and Yutao Luo. 2019. Legal judgment prediction via multi-perspective bi-feedback network. *arXiv preprint arXiv:1905.03969* (2019).

[28] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*. 1480–1489.

[29] Hai Ye, Xin Jiang, Zhunchen Luo, and Wenhan Chao. 2018. Interpretable Charge Predictions for Criminal Cases: Learning to Generate Court Views from Fact Descriptions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 1854–1864.

[30] Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal judgment prediction via topological learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 3540–3549.

[31] Haoxi Zhong, Yuzhong Wang, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. Iteratively Questioning and Answering for Interpretable Legal Judgment Prediction. *Association for the Advancement of Artificial Intelligence* (2020).

[32] Peisong Zhu and Tieyun Qian. 2018. Enhanced aspect level sentiment classification with auxiliary memory. In *Proceedings of the 27th International Conference on Computational Linguistics*. 1077–1087.