

北京大学学报(自然科学版)

Acta Scientiarum Naturalium Universitatis Pekinensis

doi: 10.13209/j.0479-8023.2018.059

基于改进孪生网络结构的相似法律案例检索研究

李兰君 周俊生[†] 顾颜慧 曲维光

南京师范大学计算机科学与技术学院, 南京 210023; [†] 通信作者, E-mail: zhoujs@njnu.edu.cn

摘要 针对现有的基于孪生网络结构的文档相似度计算方法大多将整个文档看成模型的输入序列, 易导致数据稀疏的问题, 提出利用层级注意力机制来改进孪生网络结构中的文档表示。针对基于层级注意力机制的孪生网络计算模型在输入时有可能忽略文档中重要句子的问题, 进一步提出一种引入文档内容压缩的两步骤文档相似度计算方法。利用开发的法律案例文档相似度标注数据集进行实验, 结果表明所提方法明显优于基于长短期记忆模型的孪生网络计算模型。

关键词 文档相似度计算; 孪生网络; 注意力机制; 文档内容压缩

Similar Legal Case Retrieval Based on Improved Siamese Network

LI Lanjun, ZHOU Junsheng[†], GU Yanhui, QÜ Weiguang

School of Computer Science and Technology, Nanjing Normal University, Nanjing 210023;

[†] Corresponding author, E-mail: zhoujs@njnu.edu.cn

Abstract In view of the existing research about document similarity calculation methods based on Siamese network, the entire document is regarded as the input sequence of model that may lead to sparse data. Hierarchical attention mechanism is used to improve the document representation in the Siamese network. For the Siamese network computing model based on hierarchical attention mechanism may ignore the important sentence in the document when inputting, a two-step document similarity calculation method that introduces the compression of document content is further proposed. The experimental results show that the proposed method is obviously superior to the Siamese network computing model based on the Long Short-Term Memory.

Key words document similarity calculation; siamese network; attention mechanism; document content compression

各种法律数据库已经存储了大量电子格式数据。由于数据库仅能做简单的案件分类, 所以通过数据库查询相似案例显得费时费力。如何从海量法律案例中更快、更方便地查询相似案例, 是一项值得探讨的工作。随着互联网技术的迅速发展, 一些学者已经在研究与机器学习相关的法律案例检索技术。这些法律案例检索技术可以更快地查询相似案例, 降低

时间成本和人力成本, 提高效率。

文档相似度计算是法律案例检索的重要环节, 旨在比较文档对的相似程度。文档相似度计算的研究成果可以应用到很多自然语言处理任务中, 例如信息检索、机器翻译、自动问答、复述问题以及对话系统等。在一定程度上, 这些自然语言处理任务都可以抽象为文档相似度计算问题。例如, 信息检索可

国家自然科学基金(61472191, 61772278, 41571382)、福建省信息处理与智能控制重点实验室开放基金(MJUKF201705)、江苏省高校哲学社会科学基金项目(2016SJB740004)和江苏省高校自然科学研究重大项目(15KJA420001)资助

收稿日期: 2018-04-15; 修回日期: 2018-08-13; 网络出版时间: 2018-08-22 16:33:53

以归结为查询项与数据库中文档的相似度计算问题。

目前,有一些与法律相关的信息检索技术^[1-3]也是基于文档相似度计算。Lau 等^[1]开发了一个运用信息检索和结构信息匹配进行法规相关性分析的系统,其中结构信息匹配采用的是向量空间模型。Ashley 等^[2]研究纠纷判决的信息检索方法,并提供在线纠纷解决平台,案例间的文档相似度计算方法采用基于关键特征的最近邻算法。Carneiro 等^[3]研究从法律案例文档中检索出与论据有关的信息,其中涉及案例文档相似度的计算,采用基于词频的贝叶斯统计方法。

上述方法均基于关键词(特征)和统计信息进行文档相似度计算,由于特征是根据特定任务人工设计的,因此在很大程度上限制了模型的泛化能力。深度学习方法可以自动地从原始数据中抽取特征^[4],根据训练数据的不同,可以方便地适配到其他文档相似度的相关任务中。有较多学者提出基于深度学习方法的文档相似度计算方法,其中一些与孪生网络相关的计算模型已取得较好的实验结果。根据采用的网络结构不同,这些孪生网络计算模型可以分为基于全连接的方法、基于卷积网络的方法以及基于循环神经网络的方法。这些方法将整个文档看成模型的输入序列,由于不同文档的长度可能相差较大,容易导致数据稀疏问题。此外,目前还没有公开的中文法律案例文档相似度数据集或其他相关任务的中文文档相似度数据集。

针对上述问题,我们开发了法律案例文档相似度标注数据集,并提出一种能有效地避免数据稀疏问题的文档相似度计算方法。本文的主要贡献有 3 个方面: 1) 开发法律案例文档相似度标注数据集,这是我们实验的基础; 2) 提出层级注意力机制的孪生网络计算模型; 3) 提出一种引入文档内容压缩的两步骤文档相似度计算方法。

1 相关工作比较

孪生网络^[5]是一类包含两个或更多相同子网络的神经网络架构。这里的“相同”是指它们具有相同的配置,即具有相同的参数和权重,参数更新在两个子网上同时进行。孪生网络利用同质的网络得到两个文档的向量表达,然后通过向量表达的相似度来衡量两个文档的相似度。

目前,已经有一些基于孪生网络结构的文档相似度算法的研究。深度语义结构模型(Deep Structured Semantic Model, DSSM)^[6]是最早将孪生网络应

用在文档相似度计算的工作之一,该模型主要对查询内容和文档的相似度进行建模,每个文档对象都是由 5 层的网络单独进行向量化,最后通过计算两个文档向量的余弦相似程度来确定两个文档的相似度。DSSM 的子网络结构可以分成两个主要部分,第一部分所做的处理是将查询内容和文档中的单词通过哈希方式映射到单词级别的向量,第二部分是在哈希层之后连接 3 层的全连接神经网络来表达整个文档的主题向量。该方法输入数据利用的是词袋模型,忽略文档中的词序关系,对于文档相似度计算这种任务,无法将一些学习到的局部相似度信息应用到全局。此外,全连接神经网络的参数太多,不利于优化。Shen 等^[7-8]改进基于全连接神经网络的深度语义结构模型,提出基于单词序列的卷积深度语义结构模型(Convolutional Deep Semantic Model, CDSSM)。CDSSM 与 DSSM 的区别在于: 1) 将中间生成主题向量的全连接层换成卷积神经网络的卷积层和池化层; 2) 将查询内容和文档中的每个单词都表示为一个词向量。虽然卷积神经网络减少了训练参数,但是无法捕捉句子长距离的依存关系。考虑到长短时记忆网络^[9]能够保持并利用长距离信息的特性,Palangi 等^[10]提出基于长短时记忆网络的文档相似度算法(Long Short-Term Memory, LSTM)。LSTM 的查询内容与文档分别经由长短时记忆网络得到其向量表达。Neculoiu 等^[11]提出基于多层双向长短期记忆网络的文档相似度算法(Bi-Directional Long Short-Term Memory, BiLSTM),孪生网络的每个网络分支是多层双向的长短期记忆网络。

用 DSSM, CDSSM 或 LSTM 等模型进行文档相似度计算时,都是将整个文档看成一个序列,并将其作为模型的输入。由于不同文档的长度可能相差很大,这样的简单处理容易导致数据稀疏问题。针对这些模型的不足,我们提出一种改进的基于孪生网络结构的文档相似度计算方法。

2 法律案例文档相似度标注数据集的开发

实验数据集是我们验证文档相似度计算模型有效性的基础。但是,目前尚没有公开的法律案例文档相似度标注数据集或者其他相关任务的中文文档相似度数据集,因此,我们开发了一个中文法律案例文档相似度标注数据集。

2.1 数据集来源及标注过程

中文法律案例文档相似度标注数据集来源于中

国裁判文书网(<http://wenshu.court.gov.cn/>)上的案例。我们通过自动爬虫程序,下载裁判文书网上的部分案例文档。文档覆盖 3 个较常见的法律案例类别:借贷案、离婚案、劳务纠纷案。该数据集的文档长度为 126~344 字。我们标注的法律案例文档相似度数据集共有 2500 对文档,去掉自身匹配对文档后,还有 1225 对文档。

我们借鉴 Lee^[12]的 50 标准文档相似度测试数据集(该数据集集合来自澳大利亚广播公司文档)的标注方法。为了提高标注速度和质量,首先设计一个文档相似度标注辅助工具,用于展示待标注文档的全文。评分者被要求阅读并判断成对展示的文档的相似性,并在系统提供的文本框中输入相似度值。我们设定的文档间的相似度值是 1~5,其中 1 表示两个文档最不相似,5 表示两个文档最相似。标注系统界面中左右侧同时展示的文档随机出现。对标注结果中置信度低的文档对,重新进行标注。最后,对结果相差较大(例如相似度差值大于 1)的文档对,采取投票法标注。

2.2 标注数据集的统计分析

所有实验的评级分布显示,标注结果严重倾向低相似度值,文档标注的相似度值 1, 2, 3, 4 和 5 出现的频率分别为 0.4, 0.2, 0.2, 0.1 和 0.1。

为了测试相似性等级的个体差异,计算一个主题下的每对文档对的评分与该文档对的评分总平均值间的差异。在 5 点量表上,显示平均绝对差约为 0.44,超过 92% 的差异小于 1。我们还通过对每一对文档随机选择一个评级,并测量与其余评分者判断的平均值的相关性,从而产生评分者之间的相关性。1000 对文档相关系数的平均值为 0.615,说明个体差异产生的影响可以忽略。

为了测试标注系统界面中文档所处位置不同是否影响文档对的相似性判断,我们计算不同位置平均相似度的差异。在 5 点量表上的平均差异是 0.4,差异在 1 以内的占比 93% 以上,说明可以忽略文档所处位置的影响。

以上结果表明,在不同的主题之间,相似性判断没有显著的差异,也不受标注系统中文档出现的位置的影响。将所得的 5 点相似性评分标准化为 0~1 尺度,即 0.2, 0.4, 0.6, 0.8 和 1,以便与各种相似性模型进行比较。

3 改进的基于孪生网络结构的法律案例文档相似度计算方法

在现有的研究中,大多将整个文档看成模型输入序列,易导致数据稀疏的问题。针对这些模型的不足,我们提出利用层级注意力机制改进文档的向量表示。考虑到文档的层级特性(单词组成句子,句子构成一个文档),我们在构建一个文档表达的时候,首先构建句子的向量表达,然后将它们聚合成一个文档表达。其次,由于单词和句子的重要性高度依赖于上下文,即同一个词或句子在不同的上下文中重要性不同,因此在单词层级和句子层级,我们的模型分别采用注意力机制^[13]。由于基于层级注意力机制的文档相似度计算方法是选取文档中的部分句子作为模型的输入,如果采用随机或其他简单方式,将导致文档中部分关键性句子被忽略的问题。我们进一步提出引入文档内容压缩的两步骤文档相似度计算方法。使用文档内容压缩方法,选取文档中一些重要句子,将这些重要句子作为层级注意力机制模型的输入。为验证我们改进的基于孪生网络结构的法律案例文档相似度计算方法的有效性,选取基于多层双向长短期记忆网络的文档相似度计算方法^[11]作为基线(baseline)系统。

3.1 基于孪生网络结构的法律案例文档相似度计算的基线系统

全连接神经网络参数过多,导致训练时间长;卷积神经网络无法捕捉句子长距离的依存关系;循环神经网络能够保持并利用长距离信息的特性。因此,我们选取长短期记忆模型作为孪生结构网络的一部分,设计并实现基于多层双向长短期记忆模型的文档相似度计算^[11]的基线系统。

3.1.1 孪生网络结构

在文档相似度计算任务中,将文档表示成一个向量,称为文档的向量表达。如图 1 所示,基于孪生网络结构的文档相似度计算方法利用相同子网络得到两个对象的向量表达,然后利用向量表达的相似度来计算文档间的相似度。

3.1.2 基于多层双向长短期记忆网络的文档相似度计算

Neculoiu 等^[11]提出基于多层的双向长短期记忆网络孪生结构计算文档相似度的方法,其模型结构

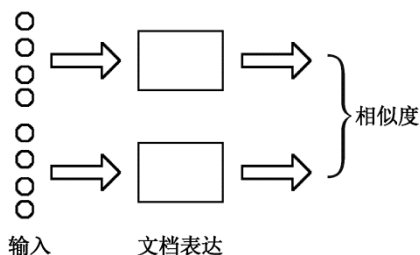


图1 基于孪生网络结构的文档相似度计算框架
Fig.1 Framework of document similarity calculation based on Siamese network structure

分为两部分: 1) 由两个平行的双向长短期记忆网络构成的特征抽取部分; 2) 由全连接词构成的分类/拟合层。最终, 通过能量函数组合这两个网络分支。选取的能量函数为余弦相似度函数如式(1)所示。将最终输出的实数作为文档间的相似度值。

$$E_W(x_1, x_2) = \frac{\langle f_W(x_1), f_W(x_2) \rangle}{\|f_W(x_1)\| \|f_W(x_2)\|} \quad (1)$$

其中, $f_w(x_1)$ 和 $f_w(x_2)$ 分别代表文档对的特征表达向量。模型的输入是中文文档序列

$$x = (x_0, x_1, x_2 \dots x_n), y = (y_0, y_1, y_2 \dots y_n)。$$

通过词嵌入矩阵 W_e , 将词映射为向量

$$(I_i = W_e x_i, J_i = W_e y_i)。$$

假设数据集 $X = \{x_1^{(i)}, x_2^{(i)}, y^{(i)}\}$, 损失函数可以

简写为

$$\mathcal{L}_W(X) = \sum_{i=1}^N L_W^{(i)}(x_1^{(i)}, x_2^{(i)}, y^{(i)}), \quad (2)$$

损失函数 $L_W^{(i)}$ 是交叉熵损失函数。模型优化算法采用自适应矩估计算法(adaptive moment estimation), 简称 Adam 算法, 模型的正则化方法采用 dropout 技术。

3.2 利用层级注意力机制的孪生网络模型计算法律案例文档相似度

目前已有一些基于孪生网络结构的文档相似度算法研究, 这些方法的研究主体是整个文本。从内容来看, 文本可能是短语、句子或文档, 但训练模型时都是将整个文本看成一个序列, 将其作为同质网络的输入。这种做法虽然简单, 但存在明显的问题, 即对于长度相差较大的文档, 由于是两个相同的网络分支, 它们具有相同的参数规模, 从而会出现数据稀疏问题。根据经验可知, 文档中不同的单词和句子, 其重要程度是不同的; 相同的单词和句子, 在不同

的上下文中重要程度也不同。

基于以上事实, 我们提出层级注意力机制的孪生网络模型。在构建一个文档表达的时候, 首先构建句子的向量表达, 然后将它们聚合成一个文档表达。在单词级别和句子级别, 采用注意力机制。孪生网络结构中每个分支如图2所示。

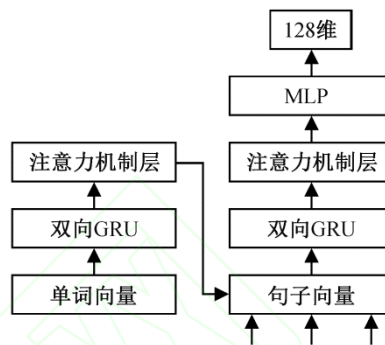


图2 层级注意力机制结构

Fig.2 Hierarchical attention mechanism structure

该孪生网络是一个4层的结构, 先是通过双向GRU (gated recurrent unit)层^[14]得到单词的编码, 连接一个注意力机制层, 得到一个包含重要信息的句子向量表示。将文档中句子的向量表示连接起来, 作为下一层(双向GRU层)的输入, 双向GRU层的输出可以看成句子的编码。与单词编码层相同, 再连接一个注意力机制层。权值共享的孪生神经网络结构包含上述提到的单词编码层、单词级别的注意力机制层、句子编码层和句子级别的注意力机制层。基于句子级别的注意力机制层得到的向量连接一个前馈神经网络是整个文档的向量表达, 这个向量有128个维度。

3.2.1 编码层

单词编码层和句子编码层都利用双向GRU模型, 它们的不同之处在于单词编码层的输入是句子中每个词通过词嵌入矩阵映射的二维向量, 而句子编码层的输入是由单词级别注意力机制向量连接而来。

给定一个句子, 由一个序列组成, 即 $w_{it}, t \in [0, T], T$ 为句子长度。通过词嵌入矩阵 W_e , 将词映射为向量, $x_{ij} = W_e w_{ij}$ 。

通过双向GRU模型, 获取单词或句子的向量表达, 可以提供给输出层过去时刻和未来时刻两个方向的信息。双向GRU模型包含一个向前隐藏层GRU \vec{f} 和向后隐藏层GRU \overleftarrow{f} 。

$$\vec{h}_{it} = \overline{\text{GRU}(x_{it})}, t \in [1, T/L], \quad (3)$$

$$\vec{h}_{it} = \overline{\text{GRU}(x_{it})}, t \in [T/L, 1], \quad (4)$$

其中, L 表示句子数量。

3.2.2 注意力机制层

句子中每个词项的重要程度不同。同样地, 不同的句子, 由于其在文档中所处的位置和-content不同, 其重要程度也不同。此外, 相同的单词和句子在不同的上下文中重要程度也不同。在进行单词编码和句子编码后), 分别采用注意力机制获取文档的重要信息。本文采用的注意力机制可以用公式表示为

$$u_{it} = \tanh(W_w h_{it} + b_w), \quad (5)$$

$$\alpha_{it} = \frac{\exp(u_{it}^T u_w)}{\sum_t \exp(u_{it}^T u_w)}, \quad (6)$$

$$s_i = \sum_t \alpha_{it} h_{it} \circ \quad (7)$$

式(5)~(7)的含义是, 先将输入向量 h_{it} 送入一层的MLP (multi-layer perceptron), 获得 h_{it} 的隐层表示 u_{it} , 词的重要程度用 u_{it} 和 u_w 的相似度来衡量; 然后利用Softmax函数做归一化处理, 获得attention矩阵 α_{it} ; 最后输入向量 h_{it} , 通过attention矩阵加权求和。

3.3 引入文档内容压缩的两步骤文档相似度计算

基于层级注意力机制的孪生网络计算模型在输入时可能产生忽略文档中重要句子的问题, 为此我们提出引入文档内容压缩的两步骤文档相似度计算方法, 计算框架如图 3 所示。首先将文档内容压缩, 从案例文档中抽取部分重要的句子, 然后将压缩后的内容分别送入相同的层级注意力机制网络中。

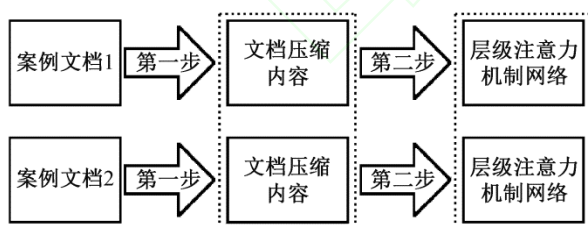


图 3 基于两步骤的文档相似度计算框架

Fig. 3 Two-step document similarity calculation framework

我们采用的文档内容压缩方法是一种用于文本处理的基于图的排序算法——TextRank 算法^[15], 其基本思想来源于谷歌的 PageRank 算法, 通过将法律案例文档分割成若干组成单元(单词、句子)来建立图

模型, 法律案例文档中的重要成分通过投票机制进行排序, 仅利用单篇法律案例文档的信息, 就可以实现关键词提取。TextRank 算法与 LDA 和 HMM 等模型不同, 不需要事先对多篇文档进行学习训练, 因其简洁有效而得到广泛应用。对案例文档使用 TextRank 算法进行文档内容压缩, 可以提取重要句子, 删除冗余内容。通过文档内容压缩, 可以解决层级注意力机制孪生网络模型容易忽略文档中关键句子的问题。

TextRank 算法将每个句子看成图中的一个节点, 若两个句子之间有相似性, 则认为对应的两个节点之间有一个无向有权边, 权值是相似度。两个句子相似度的计算公式如下:

$$\text{Similarity}(S_i, S_j) = \frac{|\{w_k | w_k \in S_i, w_k \in S_j\}|}{\log(|S_i|) + \log(|S_j|)}, \quad (8)$$

其中, 分子表示两个句子中都出现的单词的数量, $|S_i|$ 和 $|S_j|$ 分别表示两个句子的单词数量。

TextRank 算法提取重要句子的计算公式如下:

$$\text{WS}(V_i) = (1 - d) + d \times \sum_{V_j \in \text{In}(V_i)} \frac{w_{ji}}{\sum_{V_k \in \text{Out}(V_j)} w_{jk}} \text{WS}(V_j), \quad (9)$$

其中, $\text{WS}(V_i)$ 表示句子 V_i 的重要程度, d 是阻尼系数, $\text{In}(V_i)$ 是与句子 V_i 存在连接边的节点。

4 实验

4.1 评价方法

为了判断不同文档相似度计算方法的优劣, 需要对比实验结果与评分者提供的文档对相似度量值的相关性和差异性, 实验中分别采用皮尔逊相关系数(r)和均方误差(MSE)来衡量相关性和差异性。 r 取值范围为 $[-1, 1]$, $r > 0$ 表示实验结果与标准观察者所提供文档对的相似度量值正相关。 $\text{MSE} > 0$, 值越小, 表示预测模型描述实验数据具有更好的精确度。实验中, 通过将标注数据集按照 7:3 的比例分割, 获得训练集和测试集。训练集和测试集的文档相似度分布与标注数据集相同。实验中设置一些超参数, 取值如表 1 所示。超参数值的设定依据经验和实验验证。

4.2 评价结果

通过实验, 验证层级注意力机制孪生网络(Hierarchical Attention Semantic Model, HASM)结构的可用性以及文档内容压缩方法的有效性。

4.2.1 验证层级注意力机制孪生网络结构的可用性

为了验证层级注意力机制孪生网络结构的可用

表 1 超参数设置

Table 1 Super parameter settings

超参数名称	取值
embedding 维数	50
Bi-LSTM 输出向量维度	100
Bi-GRU 输出向量维数	100
学习率	0.01
单句字数	15
选取句子数	10

性, 将层级注意力机制孪生结构网络与基线系统基于长短时记忆网络的方法进行实验对比, 结果表明, 前者可以学习到文档中的重要信息。我们适当修改基线系统的方法, 在长短时记忆网络层之后连接注意力机制层(与 3.2.2 节的做法相同), 最后与我们的 THSM 方法进行对比, 实验结果如表 2 所示。实验中, 基于长短期记忆模型的方法取所有案例文档长度均值 180 作为模型输入序列的长度。

表 2 层级注意力机制孪生网络与基线系统的实验结果对比

Table 2 Experimental comparison results of hierarchical attention mechanism and baseline system

方法	r	MSE
baseline	0.86	0.020
baseline+attention	0.95	0.007
THSM	0.96	0.007

表 2 显示, 与原有方法相比, 结合注意力机制的基线方法皮尔逊相关系数和均方误差结果分别提升 0.09 和 0.013。与结合注意力机制的基线方法相比, 层级注意力机制的孪生网络模型皮尔逊相关系数提升 0.01。实验结果表明, 结合注意力机制的基线方法明显优于原有方法, 层级注意力机制的孪生网络模型优于结合注意力机制的基线方法。

4.2.2 验证文档内容压缩方法的有效性

对于本文提出的层级注意力机制的孪生网络结构, 我们分别设置不包含文档内容压缩方法和包含文档内容压缩方法的两组实验, 每个文档选取的句子数为 10, 单句字数为 15, 实验结果如表 3 所示。可以看出, 结合文档内容压缩方法的层级注意力机制网络的文档相似度计算方法在皮尔逊相关系数和均方误差指标上均优于层级注意力机制网络, 分别提升 0.01 和 0.001, 说明在进行层级注意力机制孪生网络训练前先进行文档内容压缩是必要的。

表 3 验证文档内容压缩方法有效性的实验结果对比

Table 3 Experimental comparison results of validating the effectiveness of document content compression methods

方法	r	MSE
THSM	0.96	0.007
THSM+TextRank	0.97	0.006

5 结论及下一步工作

本文采用基于孪生网络结构的文档相似度计算方法, 对相似法律案例的检索问题进行探索性的研究。首先通过搜集实际法律案例文书, 开发一个法律案例文档相似度的标注数据集; 然后将层级注意力机制的孪生网络结构与文档内容压缩相结合, 提出一种改进的基于孪生网络结构的文档相似度计算方法。首先对案例文档进行文档内容压缩处理, 同质网络采用层级注意力机制网络。通过压缩文档内容, 可以获得文档中关键性句子, 解决了基于层级注意力机制的孪生网络计算模型在输入时可能忽略文档中重要句子的问题。层级注意力机制模型较好地解决了已有研究中将整个文档看成模型输入序列, 易导致数据稀疏的问题。在法律案例文档相似度标注数据集上的实验结果显示, 该方法可以提高文档相似度计算的准确率。

目前, 我们的法律案例文档相似度标注数据集规模相对较小。在下一步工作中, 我们计划扩展该数据集。另外, 后续工作中将进一步对神经网络结构进行调整(例如在循环网络层之外再设置卷积层), 以期更好地提高文档相似度计算的准确率。

参考文献

- [1] Lau G T, Law K H, Wiederhold G. Legal information retrieval and application on E-rulemaking // Proceedings of the 10th International Conference on Artificial Intelligence and Law. Bologna, 2008: 146-154
- [2] Ashley K D, Walker V R. From information retrieval (IR) to argument retrieval (AR) for legal cases: report on a baseline study. *Frontiers in Artificial Intelligence & Applications*, 2013, 259: 29-38
- [3] Carneiro D, Novais P, Andrade F. Retrieving information in online dispute resolution platforms: a hybrid method // Proceedings of the 13th International Conference on Artificial Intelligence and Law. Sardinia, 2011: 224-228

- [4] Robertson S, Zaragoza H. The Probabilistic relevance framework: BM25 and beyond. *Foundations & Trends in Information Retrieval*, 2009, 3(4): 333–389
- [5] Chopra S, Hadsell R, Lecun Y. Learning a similarity metric discriminatively, with application to face verification // *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. San Diego, 2005:539–546
- [6] Huang P S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data // *Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management*. San Francisco, 2013: 2333–2338
- [7] Shen Y, He X, Gao J, et al. A latent semantic model with convolutional-pooling structure for information retrieval // *Proceedings of the 23rd ACM international Conference on Conference on Information and Knowledge Management*. Washington, 2014: 101–110
- [8] Shen Y, He X, Gao J, et al. Learning semantic representations using convolutional neural networks for web search // *Proceedings of the 23rd International Conference on WWW*. Seoul, 2014:373–374.
- [9] Hochreiter S, Schmidhuber J. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*. Berlin: Springer, 1997: 1735–1780
- [10] Palangi H, Deng L, Shen Y, et al. Deep sentence embedding using long short-term memory networks: analysis and application to information retrieval. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 2015, 24(4): 694–707
- [11] Neculoiu P, Versteegh M, Rotaru M. Learning text similarity with siamese recurrent networks // *Proceedings of the 1st Workshop on Representation Learning for NLP*. Berlin, 2016: 148–157
- [12] Lee M D, Pincombe B, Welsh M. An empirical evaluation models of text document similarity // *Proceedings of the 27th Annual Conference of the Cognitive Science Society*. Stresa, Italy, 2005: 1254–1259
- [13] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need // *Proceedings of the 31st Conference Neural Information Processing Systems*. Long Beach, 2017: 6000–6010
- [14] Chorowski J, Bahdanau D, Cho K, et al. End-to-end continuous speech recognition using attention-based recurrent NN: first results [EB/OL]. (2014–12–04) [2017–08–10]. <https://arxiv.org/abs/1412.1602>
- [15] Mihalcea R, Tarau P. TextRank: bringing order into texts // *Proceedings of International Conference on Empirical Methods in Natural Language Processing*. Barcelona, 2004: 404–411