基于深度学习的多模型智慧政务系统—智政

摘要—由于政务信息不断在增长，由于传统的完全人工处理政务信息的方法的局限性，给日常处理政务信息带来了许多问题。本文通过结合现有的自然语言处理技术和传统算法，着手解决了在处理政务信息过程中的分类，热点问题发掘以及政务回复质量评估这三个问题。

对于任务一，分类任务是监督学习领域的典型问题，我们首先分析数据，对数据进行了一定的预处理，是对每一折中的数据分布大致相同。之后我们分别尝试值使用主题预测、只使用留言详情预测以及结合两者指标一起预测的三种模型，最终使用集成学习模型。

对于任务二，首先需要对留言进行分类，我们分别使用 **TF-IDF**、**LDA**、**BERT**、**AutoEncoders** 来表示文本。之后分别使用改进的 **SOP** 算法与 **K-means** 算法对文本进行分类，并使用来搜狗新闻数据集验证分类的效果。热度评价方面我们综合考虑了问题的关注度，突发性与延续性，并建立了一个统一的热度模型来衡量其热度。

对于任务三，我们在题目给定三个因素的基础上提出了及时性的指标来评测回复是否及时。对于相关性，使用隐含狄利克雷分布（**Latent Dirichlet Allocation，LDA**）模型来对留言以及回复中的关键词进行提取，然后通过计算关键词的相似程度来计算相关性；对于完整性和可解释性，由于这两个指标都属于某种形式规范，而数据集又没有被标记。所以人工标定部分数据集，采用半监督学习方法来完成规范的检测；对于及时性，结合已有数据的分布，标定各个时间区域的及时性得分。在求得各个指标后，使用层次分析法计算各个指标的评价权重，以对答复进行评估。

Abstract - Due to the continuous growth of government information and the limitations of traditional methods of completely manual processing of government information, many problems are brought to the daily processing of government information. In this paper, by combining the existing natural language processing technology and traditional algorithm, we start to solve the three problems in the process of processing government information: classification, hot issue discovery and government reply quality evaluation. For problem one, the classification task is a typical problem in the field of supervised learning. First, we analyze the data, preprocess the data to a certain extent, and roughly distribute the data in each group. After that, we try to use topic prediction, only message details prediction and combine the two indicators to predict three methods, and choose the best method as the first topic model. For the second problem, first of all, we need to classify the messages. We use TF-IDF, LDA, Bert and autoencoders to represent the text. Then we use the improved SOP algorithm and K-means algorithm to classify the text, and use them to search the dog news data set to verify the classification effect. In the aspect of heat evaluation, we comprehensively consider the concern, suddenness and continuity of the problem, and establish a unified model to measure its heat. For last question, we put forward the timeliness index to evaluate whether the reply is timely or not based on three factors. For relevance, we use the model of implicit Dirichlet allocation (LDA) to extract keywords from messages and replies, and then calculate the relevance by calculating the similarity degree of keywords; for completeness and interpretability, because these two indicators belong to a formal specification, and the dataset is not marked, thus, part of the data set is calibrated manually, and the semi-supervised learning method is used to complete the standard test; for timeliness, combining with the distribution of existing data, the timeliness scores of each time area are calibrated. After each index is obtained, the evaluation weight of each index is calculated by AHP to evaluate the response.

关键词: Machine learning, RNN, BERT, Attention, Auto encoder, SOP, K-means, LDA, AHP, Semi-surpervised learning

I. 问题描述

A. 问题重述与分析

随着互联网的发展，每天产生的文本信息量不断攀升，但传统的人工处理也越来约困难，也给网络问政留言平台上的工作人员带来了巨大的压力，人力成本不断上升的同时但效率反而在不断降低。

在这种情况下，我们希望通过算法建模以及数据发掘，使得一些工作可以准确的自动化完成。这将会节约巨大的资源，显著提高问政平台的工作效率，使的民众和相关部门沟通的效率大大增加。

任务一需要对群众的留言进行各个类型的分类，使得群众留言可以交给相关的部门处理。在描述中已经说明留言的具体类别已经给定，需要预测留言所属的问题类别。可以发现这是一个传统的监督学习问题，下文将给出几种方法的探索。

任务二需要将留言进行归类，再给出每个相似类别的概括以及对应的热度。

任务三需要从三个维度来对留言的回复进行评估，再将三个维度的评估结果整合为总体的分数。

B. 论文结构安排

论文共分为五章，各章的安排如下：

I 对论文的问题进行简单的描述与分析，并介绍整篇论文的整体结构

$II \sim II$ 每一章对应着一个问题的求解部分

V 对于三个问题的总结与讨论

II. 任务一

A. 问题分析

群众留言的分类问题是一个监督学习问题，附件 1 中给出了内容分类的三级标签体系，附件 2 中给出了待分类的数据。

待分类的数据中包含的属性有：留言编号、留言用户、留言主题、留言时间、留言详情、一级标签。一级标签即为需要预测的标签。这其中留言主题和留言详情是文本数据，留言用户、留言编号、留言时间可以转换为数值数据。该问题需要预测留言所

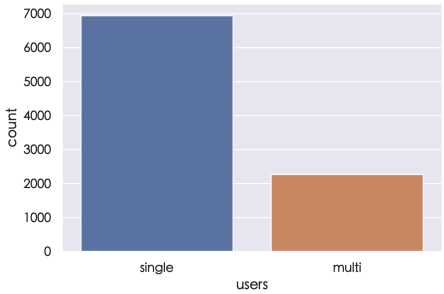

图 1: 用户评论次数

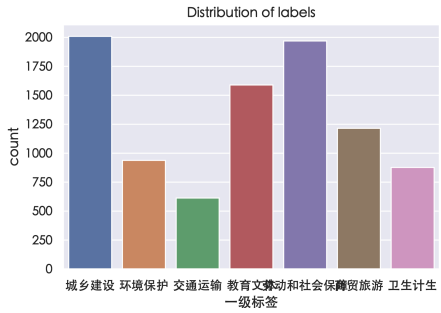

图 2: 一级标签分布

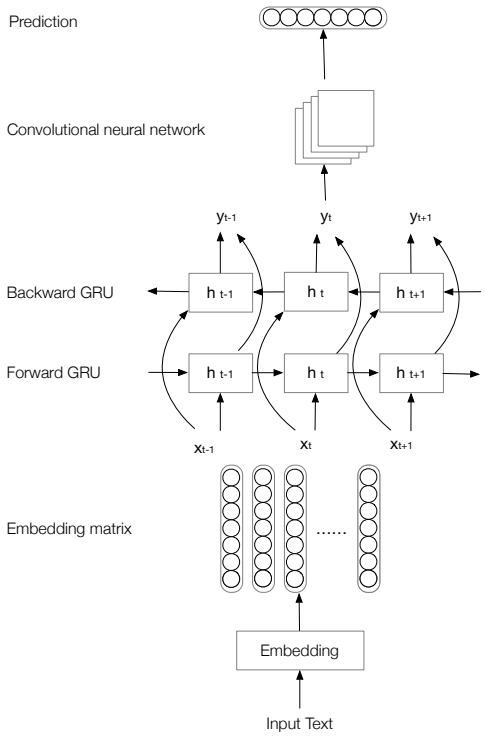

图 3: BiGRU-CNN model

属的问题类别，与留言用户没有直接联系，但从数据分析中可以发现，同一个用户可能留下多条评论，因此这个特征可以作为元数据 (meta data) 在训练时使用，使模型可以更好地学习到数据的分布特征。考虑到一段时间内可能有多个留言反映相似的问题，所以将时间作为模型的一个输入特征。

文本数据包含留言主题和留言详情，留言详情中包含大量的无效信息，可能会干扰模型的预测结果，故需对其作处理，提取其对预测作用较大的特征。我们分别尝试了只使用主题预测，只使用留言详情预测以及结合这两者进行预测的方法，并取其中结果最好的作为最终使用的模型。

B. 数据预处理

用户的评论数如图 1，可见一部分用户评论了多次，该信息可以作为元数据在训练时使用，即将同一用户的评论都放在同一折中训练模型。

数据集中标签的分布如图 2。为了保证数据分布的一致性，我们采用了 K-fold 交叉验证的方式进行训练，并使每一折中的数据分布大致相同。另外通过计算发现，留言主题平均长度为 20 个字，最多 48 个字，留言详情的平均长度为 428 个字，最多为 6237 个字。留言详情的文本过长，若是使用其所有字符建立模型，训练效率会很低，并且其中很多信息对分类是没有作用，反而会降低分类的效果。

C. XGBoost

我们首先尝试了传统的机器学习方法，使用 XGBoost 作为分类器，使用 TF-IDF 提取文本特征。XGBoost 的输入为基于单词的特征与基于字符的特征。提取单词特征时，首先使用 Jieba 对输入文本分词，之后使用 Hit 的 stopwords 去除文本中的停用词，然后提取多个连续的单词 (n-grams) 来保证连续单词组成的特征不丢失，之后使用 TF-IDF 获得基于单词的特征向量。字符级别的特征提取步骤与其类似，提取 n-grams 之后使用 TF-IDF 获得基于字符的特征向量，之后将词向量与字符向量拼接，将拼接后的向量送入 XGBoost 进行训练。

D. BiGRU-CNN Model

相比于机器学习的方法，使用深度学习的模型可以更好地学习到句子的语义。我们使用了 Bidirectional GRU 和 CNN 模型处理文本数据，然后输入到全连接层得到各个标签的分数，最后使用 Softmax 函数计算预测结果，其流程图如图 3。

1) Bidirectional GRU Layer: 当输入为包含 n 个词的文本 $[w_c^1, w_c^2, \ldots, w_c^n]$，对应标签为 y，其中 w 代表一个单词。每个单词可以用一个低维的向量表示，这样可以得到 $w^k \in R^d$，输入文本可以表示为 $I^{n\times d}$，其中 k 表示单词的顺序，d 为嵌入层的维度。嵌入层可以通过模型学习得到，也可以使用预训练的词向量作为权重。我们的模型中我们使用了 Qiu et al. (2018) 提出的词向量作为嵌入层权重。

之后，我们使用双向的 GRU 单元来学习单词的语义。由于单词在句子中与上下文有紧密的联系，RNN 可以学习到单词的上下文联系，GRU 作为 RNN 的变体，可以更好地学习到距当前单词较远的前文的语义并一定程度上缓解梯度爆炸与梯度消失问题，并且相对于 LSTM 更易与之后的 CNN 结合。其公式化描述如式 (1)：

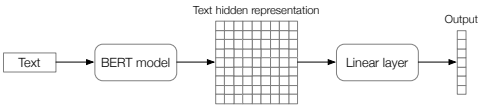

图 4: BERT Model

$$
\begin{aligned}
r_k &= \sigma(W_{ir}w_k + b_{ir} + W_{hr}h_{(k-1)} + b_{hr}) \\
z_k &= \sigma(W_{iz}w_k + b_{iz} + W_{hz}h_{(k-1)} + b_{hz}) \\
n_k &= \tanh(W_{in}w_k + b_{in} + r_k * (W_{hn}h_{(k-1)} + b_{hn})) \\
h_k &= (1 - z_k) * n_k + z_k * h_{(k-1)}
\end{aligned} \tag{1}
$$

当输入为 w^k 时，$h_{(k-1)}$ 是上一步的隐层表示，r_k, z_k, n_k 分别为重置门，更新门和新门。σ 是 sigmoid 函数，$*$ 为 Hadamard 积。通过 GRU 层，我们可以得到输入文本的隐层表示为 $[\overrightarrow{h_1}, \overrightarrow{h_2}, \ldots, \overrightarrow{h_n}]$，对其以逆向顺序输入 GRU，可以得到每个单词的反向的隐层表示 $[\overleftarrow{h_1}, \overleftarrow{h_2}, \ldots, \overleftarrow{h_n}]$，$\overrightarrow{h_k} \in R^h$，$h$ 为 GRU 的隐层维度，将其拼接得到每个单词的隐层表示 $h_k \in R^{2h}$。最后 BiGRU 的输出为 $[h_1, h_2, \ldots, h_n]$。

2) Convolutional Layer: 我们的模型中卷积层设置在 BiGRU 层之后，因为这样可以更好地提取到 n-gram 特征。BiGRU 的输出为 $[h_1, h_2, \ldots, h_n]$，其组成了一个输入矩阵 $r \in R^{n \times d}$，我们使用了 L 个大小为 $w \times d$ 卷积核来学习其 n-gram 特征，多个卷积核是为了确保不同的 n-gram 语义都可以学习到，第 l 个卷积核的输出如式 (2)：

$$
y_k^l = f(W \circ \mathbf{x}_{k:k+w} + b^l) \tag{2}
$$

其中 $\circ$ 是卷积操作，$W \in R^{w \times d}$ 和 b 分别为卷积核的权重和偏移量，w 是卷积核的长度，即 n-gram 的长度，d 是 BiGRU 输入的维度，f 是激活函数，我们这里选择使用 ReLU。当卷积核 l 从 $\mathbf{x}_{1:w-1}$ 遍历到 $\mathbf{x}_{n+w-1:n}$ 时的输出 $y^l = y_1^l, y_2^l, \ldots, y_{n-w+1}^l$。

3) Pooling Layer: 将卷积层输出分别输入平均池化层 (Average Pooling Layer) 和最大池化层 (Max Pooling Layer)，再将两个池化层的输出拼接，最为池化层的输出，其公式化描述如式 (3)：

$$
r_{avg} = \sum_{i=1}^{L} y^i / L r_{max} = \max y^i \tag{3}
$$

4) Linear Layer: 最后将池化层的输出输入到全连接层得到各个标签对应的得分。然后对全连接层的输出使用 softmax 函数得到分类到各个标签的概率，作为模型的预测结果。

我们选择使用交叉熵 (categorical cross entropy) 作为模型的目标函数，使用 Adam 方法优化梯度，使用准确率作为验证指标。

III. BERT

我们尝试的第三种分类模型为 BERT 模型，BERT 是一个双向训练的 Transformer 模型，不同于基于 RNN 的模型，RNN 模型单向地有序读取文本数据，而 Transformer 模型一次性读入整个文本，这样 Transfomer 模型可以更好地理解单词和文本之间的上下文关系。BERT 模型的另一个特点是其具有迁移学习能力，BERT 的嵌入层在经过大量语料库的训练后，可以更好地理解单词与句子的语义。因此 BERT 可以作为后继模型的嵌入层，为后继模型提供文本的语义表示。如图 4 所示，对于输入 $\mathbf{x} = [x_1, \ldots, x_n]$，其中 x_i 代表输入文本的一个词，n 为文本的最大长度，以及共有 m 个类别的标签 $\mathbf{y}$，使用 BERT 得到输入文本 $\mathbf{x}$ 的低维向量表示，如下公式所示：

$$
E_x = Bert(\mathbf{x})
$$

之后使用全连接层计算各个类别的得分，最后使用 Softmax 函数得到分类到的标签。其公式如下：

$$
y_{pred} = f(W \cdot E_x + b)
$$

其中 W 和 b 分别为全连接层的权重和偏移量，f 为 Softmax 函数。我们使用交叉熵作为目标函数，使用 BertAdam 作为优化方法。

IV. 任务二

A. 问题分析

任务 2 的热点问题挖掘旨在挖掘出用户的留言中所包含的关键信息，便于政府工作人员更好地了解民意，并有针对性地处理，提升服务效率。近年来，自然语言处理技术发展迅速，使用自然语言处理以及文本挖掘技术可以实现对用户留言的自动化，智能化分析与处理，以此提升政府服务效率。

任务二中的热点问题挖掘的子任务包括将某一特定时间段内反映特定地点或特定人群问题的留言进行归类、定义热度评价指标、提取热点反映的地点或人群和生成热点问题的描述。

B. 相关工作

热点问题是在特定时期内发生的活动或事件，热点问题挖掘旨在发现并追踪热点问题。话题识别与追踪 (Topic Detection and Tracking, TDT) 的研究在 1997 年（Allen et al.) 首次提出，随后的研究人员在这个问题上也作了大量工作。

TDT 问题的主要由两部分组成：将文本根据话题进行分类，对不断加入的新文本分类。文本分类需要理解文本的语义并计算文本间的相似度，在 Allen et.al [**?**] 的论文中，使用了基于 TF-IDF 的方法来将文本映射到向量空间，之后通过计算不同文本对应向量的相似度来得到文本相似度。为了解决新加入文本的分类问题，他们提出了一种称为 Single one pass (SOP) 的分类方法，该方法从空的话题开始，不断加入新的文本，若是新文本与任意话题的相似度不小于 α，则将其分类到与其相似度最高的一个话题中，并重新计算该话题的相似度，若是与任意话题的相似度均小于 α，则新建一个话题，将这个新文本作为该话题的第一个文本，之后不断重复该过程。不过这种方法存在以下一些问题：

1) 数据过小时，模型误差大

   模型初期文本不足时，模型的判断的误差较大，模型的结果很容易被输入顺序所影响。这时候文本的来源，初识参数的选择很容易影响模型的分类结果。

2) 话题中心偏移 (topic center shift)
虽然相似的文本描述的大多是一个问题，但不同文本的侧重点可能不同。随着时间的推移，每个话题内的文本数量增多，话题的中心可能会发生偏移，这时候早期的文本与后期的文本间的差异可能会越来越大。

为了解决这些问题，后来的研究人员也作了许多的工作。Wang (2011) [8] 提出了多中心的话题模型，Deng et al. (2019) [**?**] 提出了使用基于滑动时间窗口的可增长 k-means 模型，Yi et al. 与问题热度评价方法 (2013) [**?**]

除了 SOP 算法，K-Means 等聚类算法也在 TDT 问题上有广泛应用，传统的 K-Means 算法必须预先设置好聚类的簇数 k，但我们预先并不知道需要分类的簇数，为了解决该问题 Pham et al. (2004) 提出了可以自适应的 Incremental K-Means 算法。

对于文本语义的表示，传统的方式是使用 TF-IDF 等向量空间模型 (VSM) [7] 来将文本映射到向量空间。然而 VSM 只考虑单词的频率，而没有考虑到单词间的语义联系，所以这样的表示方式可能会带来文本噪声。为了加强模型对文本语义的理解能力，语言模型 [6]，后缀树模型 (suffix tree) [9]，本体模型 (ontology) [4] 等模型相继被提出。

由于文本内容可能包含很多无关元素，所以在处理文本时需要提取其关键特征，对文本表示进行降维。Latent Dirichlet Allocation (LDA) [1] 、Latent Semantic Analysis (LSA)、Probabilistic Latent Semantic Analysis (pLSA) 等话题模型常用来提取文本的关键特征。相比于 VSM 方法，这类模型可以提取文本的主要话题，进而完成对文本表示的降维。另外 Bahdanau et al. (2015) 提出的基于 Attention 机制的 Encoder-Decoder 模型也可以用于获得文本的降维表示。

在文本内容表示上，我们分别测试了使用改进版的 TF-IDF 算法、LDA、LSA 模型、BERT 模型、Annotated encoder-decoder 作为话题模型。分类算法上我们分别测试了使用带滑动窗口的 SOP 算法和 Incremental K-Means 算法。

C. 文本表示方式

我们分别测试了 TF-IDF，LDA，BERT 与 Autoencoder 这几种文本表示方式。

1) Weighted TF-IDF: TF-IDF 是词频和逆文本频率指数的积，本任务中需要根据特定人群与特定地点将留言进行分类，因此表示地点/人群的词更为重要。另外由于留言详情中的文本噪声过大，包含大量无效信息，实验结果表明直接对留言详情使用 TF-IDF 方法构建文本表示反而会使分类效果下降。所以在使用 TF-IDF 方法时，我们给予标题中的词更大的权重，我们提出的 TF-IDF 计算方法如式 (5)：

$$
\begin{aligned}
W(t,d) &= \alpha \cdot \beta \cdot \gamma \cdot TF(t,d) \cdot IDF(t) \\
\alpha &= \begin{cases} 4,\ t\ represent\ person \\ 1,\ others \end{cases} \\
\beta &= \begin{cases} 4,\ t\ represent\ place \\ 1,\ others \end{cases} \\
\gamma &= \begin{cases} 8,\ t\ is\ in\ title \\ 1,\ others \end{cases}
\end{aligned} \tag{4}
$$

其中 $W(t,d)$ 为单词 t 在文本 d 中的权重，$TF(t,d)$ 为 d 文本中 t 的词频，$IDF(t)$ 为 t 的逆文本频率指数，α, β, γ 是当 t 作为人物，地点，标题中的词时的权重。该算法的流程如图 **[?]**，我们使用 jieba 对文本进行分词，之后使用哈工大提供的中文停用词列表去除停用词。另外为了得到 α, β, γ 的值，我们使用了 HanLP [3] 中提供的 MSRA_NER_BERT_BASE_ZH 模型，该模型是基于 BERT 的最准确的中文 NER 模型。

TF-IDF 是词频和逆文本频率指数的积，本任务中需要根据特定人群与特定地点将留言进行分类，因此表示地点/人群的词更为重要。另外由于留言详情中的文本噪声过大，包含大量无效信息，实验结果表明直接对留言详情使用 TF-IDF 方法构建文本表示反而会使分类效果下降。所以在使用 TF-IDF 方法时，我们给予标题中的词更大的权重，我们提出的 TF-IDF 计算方法如式 (5)：

$$
\begin{aligned}
W(t,d) &= \alpha \cdot \beta \cdot \gamma \cdot TF(t,d) \cdot IDF(t) \\
\alpha &= \begin{cases} 4,\ t\ represent\ person \\ 1,\ others \end{cases} \\
\beta &= \begin{cases} 4,\ t\ represent\ place \\ 1,\ others \end{cases} \\
\gamma &= \begin{cases} 8,\ t\ is\ in\ title \\ 1,\ others \end{cases}
\end{aligned} \tag{5}
$$

其中 $W(t,d)$ 为单词 t 在文本 d 中的权重，$TF(t,d)$ 为 d 文本中 t 的词频，$IDF(t)$ 为 t 的逆文本频率指数，α, β, γ 是当 t 作为人物，地点，标题中的词时的权重。该算法的流程如图 **[?]**，我们使用 jieba 对文本进行分词，之后使用哈工大提供的中文停用词列表去除停用词。另外为了得到 α, β, γ 的值，我们使用了 HanLP [3] 中提供的 MSRA_NER_BERT_BASE_ZH 模型，该模型是基于 BERT 的最准确的中文 NER 模型。

2) LDA: Latent Dirichlet Allocation (LDA) [5] 是处理离散数据语料库的概率模型。它基于 bag-of-words (BOW) 对单词建模，忽略单词的顺序。在这种“可交换性”之后，单词的分布将是独立的，并且在某些给定的参数条件下会具有相同的分布。这种条件独立性使我们能够为文档和单词的语料库建立分层的贝叶斯模型。

LDA 是优秀的话题模型，官方给定的数据中留言详情包含很多噪声，使用 LDA 方法可以提取到留言详情的主题，相比于

TF-IDF 方法，该方法得到的文本的维度更少，特征更明显，分类速度也更快。但是数据中的留言主题的表达已经足够简洁，使用 LDA 方法可能会导致信息的丢失，所以对于留言主题，我们仍旧使用 TF-IDF 方法计算文本表示。

3) BERT: TF-IDF 模型与 LDA 模型都忽视了单词的顺序，并且都是基于单词出现的频率来计算文本的表示，忽视了单词本身的语义。BERT 模型在大量的语料训练下可以学习到文本的语义，因此我们尝试了使用 Cui et al. [2] 提供的中文 BERT-wwm-base 作为文本的表示模型。

4) Attention based autoencoders: 自编码器 (Autoencoders) 是一个无监督的模型，常用于表示学习任务。该模型的目标是学习到输入数据的表示，该表示的维度常小于输入数据的维度，以此达到对输入数据降维，并去除输入数据中的噪声的作用。

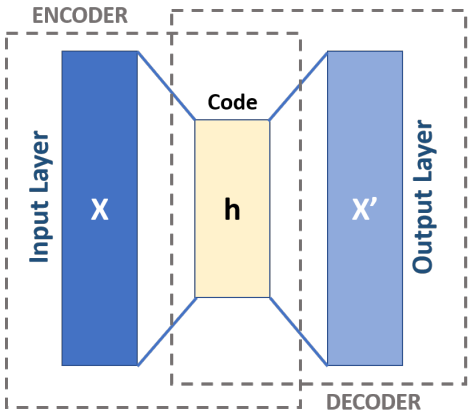

图 5: Autoencoder

传统的自编码器结构如图 5 所示，其由两部分组成：Encoder 与 Decoder。对于输入的文本 $\mathbf{X} = [x_1, x_2, \ldots, x_n]$，我们 Encoder 的模型为 Bidirectional LSTM，前向的 LSTM 读取 $\mathbf{X}$ 并计算得到其隐层表示 $[\overrightarrow{h_1}, \ldots, \overrightarrow{h_n}]$，反向的 LSTM 反向读取 $\mathbf{X}$ (从 x_n 读到 x_1)，得到隐层表示为 $[\overleftarrow{h_1}, \ldots, \overleftarrow{h_n}]$。将这两种隐层表示拼接得到 $h_j = [\overleftarrow{h_j}, \overrightarrow{h_j}]$，$h_j$ 即可看作输入文本的总结。解码的模型也使用了 LSTM，解码时使用了 Attention 机制，将注意力集中在更重要的词上，解码器每一步都输出单词 y_i 的概率，其公式表示如下：

$$p(y_i|y_1, \ldots, y_{i-1}, \mathbf{x}) = g(y_{i-1}, s_i, c_i)$$

其中 s_i 是 LSTM 模型在第 i 步时的隐层状态，可以通过如下方法计算：

$$s_i = f(s_{i-1}, y_{i-1}, c_i)$$

其中 c_i 是内容向量，其计算方式如下：

$$c_i = \sum_{j=1}^{n} a_{ij} h_j$$

每个单词的隐层表示 h_j 对应的 a_{ij} 通过如下方式计算：

$$a_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{n} exp(e_{ik})} e_{ij} = a(s_{i-1}, h_j)$$

其中 e_{ij} 是 alignment model，其值反映了位置 i 的输入和位置为 j 的输出之间的匹配程度。

Autoencoder 是一个无监督模型，其输入与输出一致，均为 $\mathbf{x}$，我们使用 cross entropy 作为目标函数，使用 Adam 作为优化方法。对于输入 $\mathbf{x}$ 其降维后的表示即为 Encoder 的输出 $h = [h_1, \ldots, h_n]$。

D. 分类方法

1) Improved single one pass: 我们使用基于滑动时间窗口的 Single one pass (SOP) 算法来解决传统 SOP 算法前期受输入顺序影响以及后期计算量大的问题。具体算法如下：

1. 给定新输入的文本 d
2. 计算 d 与各个主题 T_i 间的相似度 $Sim(d, T_i)$，l 为相似度最高的话题，$l = argmax_i(Sim(d, T_i))$
3. 设置两个阈值 θ_{class} 和 $\theta_{candidate}$，如果 $Sim(d, T_l) > \theta_{class}$，则将 d 分类到 l 类中
   若是 $\theta_{candidate} < Sim(d, T_i) < \theta_{class}$，则将 d 作为 l 的候选，作为候选的 d 不参与 T_i 的更新并且在之后的 k 个时间段内，每次都需重新计算 $Sim(d, T_i)$，k 为滑动时间窗口的大小。若在这 k 个时间段内，其 $Sim(d, T_i) > \theta_{class}$ 则将其分到第 i 类中
4. 每次有新的文本 d 加入主题 T_i 时，均需要更新 T_i ，其更新如下：
   $$T_i = \sum_{i=1}^{m} d_i / m$$
5. 不断重复以上过程直到所有文本处理完毕

其中 $Sim(d, T_1)$ 是计算文本 d 与话题 T_l 相似度的函数。任务中需要将一段时间内的相似留言分类在一起，因此 $Sim(d, T)$ 函数的计算式如下：

$$Sim(d, T) = \exp(-\Delta t) \cdot cos(d, T)$$

$cos(d, T)$ 是计算文本 d 与话题 T 的余弦相似度，Δt 表示 T 中最先出现的留言与 d 之间的时间差。

2) K-means: 我们也尝试了使用 K-means 方法进行分类。传统的 K-means 算法需要提前设置 k 的值，但是我们预先不知道需要分成多少类，Pham et al. (2004) 提出了可以自动调整 k 值的 Incremental K-means [11] 方法。其算法描述如图 6

对于分类后的文本计算最早出现的留言与最后出现的留言，作为主题的时间范围；使用 MSRA_BERT_BASE_ZH 来提取名称实体，作为特定地点/人群；对每个主题的所有包含的留言标题计算其与主题的相似度，选择相似度最高的标题作为该主题的问题描述。

3) 评价标准: 我们使用了 TDT2003 [**?**] 中的评价方法，该方法根据缺失分类的比例以及错误分类的比例来计算分类的效果，其公式描述如式 (6)：

$$(C_{det})_{norm} = \frac{C_{miss} \cdot P_{miss} \cdot P_{target} + C_{fa} \cdot P_{fa} \cdot P_{non-target}}{\min(C_{miss} \cdot P_target, C_{fa} \cdot P_{non-target})} \quad (6)$$

Listing 1 K-means algorithm

```
Assign K = 1.
 Phase 1. Normal training
      Step 1. If K = 1, choose an arbitrary point
              for a cluster centre.
              If K > 1, insert the centre of the
              new cluster in the cluster with the
              greatest distortion.
      Step 2. Assign each object in the training
              set to the closest cluster and update
              its centre.
      Step 3. If the cluster centre does not move,
              go to phase 2.
      Else, go to phase 1, step 2.

 Phase 2. Increasing the number of clusters
      If K is smaller than a specified value,
      increase K by 1 and go to phase 1, step 1.
      Else, stop.
```

图 6: Incremental K-means algorithm

其中 $P_{target} = 1 - P_{non-target}$ 是一个文本属于对于主题的先验概率，C_{miss} 和 C_{fa} 分别为文本缺失和误分类的置信系数，我们设置 $P_{target} = 0.02, C_{false} = 0.1, C_{miss} = 1$。$P_{miss}$ 是话题中缺失文本的比例，P_{fa} 是话题中被错误分类文本的比例。

我们使用了搜狗新闻提供的话题追踪数据集 [8]，由于任务二只需要提取最热的 5 个主题，所以我们只选取了 [13] 中所含文本数最多的 10 个话题的来计算分类方法的 C_{det} 。

E. 热度评价算法

对于数据的分析，我们首先判断数据的维度，若维度众多，则应首先采用例如主成分分析法，因子分析法等降维方法简化数据维度，接下来再做进一度的操作。

首先，我们从数据入手，表中直观的可以用来进行热度计算的因素包括：发帖次数、点赞次数和发帖时间。

显然，由于维度较少，数据的维度并无降维必要。因此主成分分析及因子分析法这类降维分析法无需使用。接下来便是对数据进行具体的分析，对于计算主题热度，Wang, M Zhang, L Ru 在文章中对于新闻热度提出了一种具体的算法，在此我们对其进行部分修改，使其更适用于我们的场景。本题中我们采用主客观的方法为评价指标赋予权重，形成兼顾专家意见与客观数据因素的评价指标权重。

建立评价模型之前，首先我们给出评价指标。根据上问所述，从直观数据中，我们进一步推导出指标包括：发帖的次数，对于某一类主题的总赞数、帖子的时间属性（发帖的频率/发贴的时间跨度）。根据这些推出的指标，我们进一步来建立更加细致的指标如下。

1) 关注度 *O:* 对于关注的定义，一定时间内居民对于事件的关心程度即为关注度，Wang, M Zhang, L Ru 在文中采用 user attention 进行表示。在此，结合文章我们给出：某事件的点赞数与反对数是和该事件的关注度正相关，与该主题在所有主题中的所被关注的程度正相关，由此，我们引入变量：

$like$: 点赞数

$unlike$：反对数

α：比例因子

即

$$O_n = \alpha(like + unlike) \tag{7}$$

其中， 应能表示该主题在所有主题中的被关注程度，因此，我们可以得到比例因子 的表达式：

$$\alpha = \frac{1}{tot\_like + tot\_unlike} \tag{8}$$

以此完成对该主题关注度的计算。

2) 突发性 *B:* 对于突发性，突发意味着短时间内的大量出现。很直观的，突发性与一定时间内发帖的数量成正比，与该时间内所有帖子的数量成反比。在此我们引入变量：

NUM_n: 某事件段内，该主题贴的总数量。

$totNUM_{time\_span}$: 该时间段内的所有帖子的数量。

得到

$$B_n = \frac{NUM_n}{totNUM_{time\_span}} \tag{9}$$

3) 延续性 *D:* 延续性即为该主题所持续的时间，C Wang, M Zhang, L Ru 在文中提出了 media focus 来表示该主题的延续性。直观上，一个主题若长时间被提及，那么其问题热度为较高，因此我们采用该主题从第一次发表到最后一次发表的时间跨度与第一次发表距今的总跨度的比值作为延续性的主体，并且，在该期间的发帖数也是影响因子之一，在此，我们引入变量：

$time\_span_n$: 该主题所延续的时间长度。

$last\_time\_to\_today$: 该主题最后一次被提及距今的时间。

β：修正因子。

对于修正因子 β 的确定，本文中，由于因子数较少，我们对文中给出的 media focus 进行了简化，采用该时间段内，所有的帖子的数量 tot_NUMtime_span 表示 media focus， 即于该值成反比：

$$\beta = \frac{1}{totNUM_{time\_span}} \tag{10}$$

综上所述我们给出延续性的计算公式：

$$D_n = \beta(NUM_n \times \frac{time\_span_n}{last\_time\_to\_today}) \tag{11}$$

4) 时效性 *H:* 时效性，即为该主题是否过时的评价指标，对于该指标，其影响因素较多，根据 C Wang, M Zhang, L Ru, 文中对于时效性方法 Aging Theory，我们进行了部分修改，依据贴子数、时间等因素，引入变量：

$After\_post\_num$: 在该主题最后一次被提及之后所发出的总帖子数。

在此给出评价公式：

$$H_n = (\frac{after\_post\_num}{after\_post\_num + NUM_n})^{\frac{time\_span_n}{last\_time\_to\_today}} \tag{12}$$

综上，根据所得到的评价指标，我们给出热度评价标准：

$$e^{(n)} = O_n + B_n + D_n - H_n \tag{13}$$

V. 任务三

A. 问题分析

在本问中，要求对回复从相关性、完整性和可解释性三个维度进行评价。接下来，我们将根据这三个维度，结合所给出的数据，分别给出评价方式。首先对三个维度进行解释：

1) 相关性，定义上相关性指的是两个变量的相关程度，在此应为回复与问题是否相关；
2) 完整性，通过对数据的观察，我们发现每一条回复可以被分为 3 部分：回复开头、回复正文、回复结尾。结合不同的回复, 我们观察到对于每一部分都有完整的格式，因此对于完整性的判断就从该问题的回复是否满足这些格式入手；
3) 可解释性，可解释即是否有根据。通过观察数据，我们发现，部分的回复引用了问题相关文献，因此判断完整性需要观察每一条回复在正文阶段是否有相关规章制度的引用或该引用是否合理，以此给出对应的分数

依据上述解释，可以发现，对文本的理解是本问的重点。经过调查和学习，结合本问的文本特点，我们最终采用 Shubhanshu Mishra，Jana Diesner 在 Semi-supervised Named Entity Recognition in noisy-text 中提出的方法作来解决问题

B. 模型

我们使用了线性 CRF 模型作为我们的分类器。CRF 即条件随机场（Conditional Random Fields），是在给定一组输入随机变量条件下另外一组输出随机变量的条件概率分布模型，它是一种判别式的概率无向图模型，既然是判别式，那就是对条件概率分布建模。CRF 较多用在自然语言处理和图像处理领域，在 NLP 中，它是用于标注和划分序列数据的概率化模型，根据 CRF 的定义，相对序列就是给定观测序列 X 和输出序列 Y，然后通过定义条件概率 P(Y|X) 来描述模型。

CRF 的输出随机变量假设是一个无向图模型或者马尔科夫随机场,而输入随机变量作为条件不假设为马尔科夫随机场,CRF 的图模型结构理论上可以任意给定，但我们常见的是定义在线性链上的特殊的条件随机场，称为线性链条件随机场。

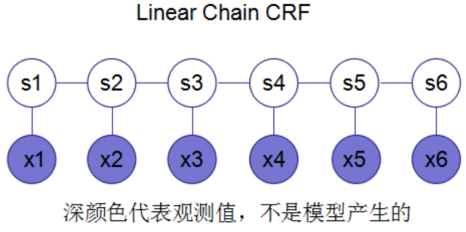

图 7: 线性链条件随机场

表 I: 半监督学习算法流程

Given	
L	a small set of labeled training data
U	unlabeled data
Loop for k iterations:	
Step 1	Train a classifier Ck based on L
Step 2	Extract new data D based on Ck
Step 3	Add D to L

CRF 转移概率全局进行归一化如式 (14)

$$p(\vec{s}|\vec{x}) = \frac{\exp\left(\vec{w}^T \Phi(\vec{s}, \vec{x})\right)}{\sum_{\vec{s'} \in S^n} \exp\left(\vec{w}^T \Phi(\vec{s'}, \vec{x})\right)} \tag{14}$$

CRF 特征函数如式 (15)

$$\Phi(\vec{s}, \vec{x}) = \sum_i \phi\left(s_{i-1}, s_i, \vec{x}\right) \tag{15}$$

C. Semi-surpervised learning

由于官方提供的数据并没有给标签，所以我们采用半监督学习的训练方法来解决可解释性和相关性的训练问题。我们手动标注了 500 组数据作为我们的种子数据，标注的数据有两个标签：完整性和可解释性，每个标签对应一个 0-5 之间的实数。之后用半监督学习的方法进行训练，其算法描述大致如表 I：

我们首先在一小部分已经标注了的数据集 L 上训练，之后在未标注的数据集 U 上进行预测，取其中预测的置信度较高的一部分数据 D，将 D 加入到 L 中，重复该过程。

D. 相关性

之前的部分已经给出了完整性和可解释性的评估方法，在这一节中将介绍相关性的评估方法

首先给出本节的符号说明如表 II 所示。

表 II: 符号说明表

符号	解释说明
S_m	留言中的关键词集合
S_r	回复中的关键词集合
$Similarity$	相关性

对于相关性的评估本质上其实就是留言和回复的相似性的评估。通过任务二中的 LDA 模型，可以提取文本中的关键词来代表留言以及回复中的文本特征。对于留言提取出的关键词集合 S_m 与回复中的关键词集合 S_r，留言与回复之间的共同特征即为两个集合的交集 $S_m \cap S_r$

那么以留言为基准，评估回复相对于留言的相似性 $Similarity$ 就可以表示为式 (16)

$$Similarity = \frac{|\ S_m \cap S_r\ |}{|\ S_m\ |} \tag{16}$$

E. 层次分析法

在??中，已经给出了相关性、完整性、可解释性和及时性几个指标的计算的方法和流程，在本节中将通过层次分析法来计算出每个指标在最后评价体系中的权重。

本节的内容一共分为三个部分：

1) 层次结构的建立
2) 构造成对比较矩阵
3) 计算权向量并做一致性检验

由于层次结构中准则层只有一层，所以层次总排序部分没有必要，所以在本次层次分析法过程中略去不做。

表 III 给出本节出现的符号定义。

表 III: 符号说明表

符号	解释说明
$C1$	相关性
$C2$	完整性
$C3$	可解释性
A	成对比较矩阵
a_{ij}	成对比较矩阵中的元素，代表准则 i 与准则 j 相比的重要程度
λ_{max}	成对比较矩阵的最大特征根
CI	一致性指标
RI	随机一致性指标
CR	检验系数

1) 层次结构的建立: 根据已有的三个指标可以建立层次结构，如图 8 所示。

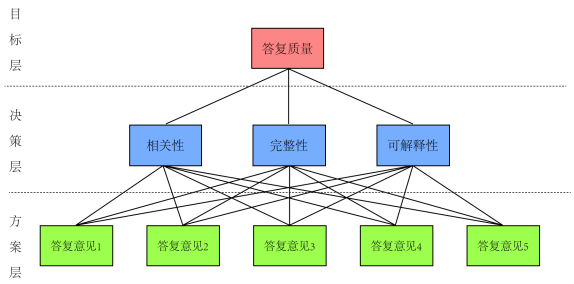

图 8: 层次结构图

2) 构造成对比较矩阵: 在层次结构构建好之后，为了计算出不同准则对于最后答复质量影响的权重，层次分析法需要构造成对比较矩阵。

为了比较两个准则对于最后答复质量的重要程度，使用数量化的相对权重来描述其重要性。根据表 IV 构造 $4*4$ 的成对比较矩阵。对于矩阵中的每个元素 a_{ij}，若 $a_{ij} > 1$，代表着准则 C_i 与准则 C_j 相比的重要程度对应在表 IV 中的值，相应的 $a_{ji} = 1/a_{ij}$。

最后得到成对比较矩阵如式 (17)：

$$A = \begin{pmatrix} 1 & 5 & 7 \\ 1/5 & 1 & 3 \\ 1/5 & 1/7 & 1 \end{pmatrix} \tag{17}$$

表 IV: i 与 j 相比重要性的强烈程度的对应分数

强烈程度	数值分数
极度重要	9
介于极度重要和非常重要之间	8
非常重要	7
介于非常重要和重要之间	6
重要	5
介于重要和略微重要之间	4
略微重要	3
介于略微重要和重要性相同	2
重要性相同	1

3) 计算权向量并做一致性检验: 通过成对比较矩阵 A 可以求得最大特征根 $\lambda_{max} = 4.1504$ 以及最大特征根的特征向量，将特征向量归一化后可以得到对于的权向量为 $W = (0.7306, 0.1884, 0.081)^T$。

由于客观事物的复杂性以及人们对事物判断比较时的模糊性，很难构造出完全一致的判断矩阵 [10]。所以需要对于已经构建好的成对比较矩阵作一致性检验。通过式 (18) 计算出该矩阵的一致性指标, 可以得到一致性指标 $CI = 0.0325$。

$$CI = \frac{\lambda_{max} - n}{n - 1} \tag{18}$$

由于一致性的偏离可能是随机原因造成的，所以需要引入随机一致性指标 RI 衡量随机因素所造成的一致性偏离的大小。可以通过查表 V 得 $RI = 0.58$。

表 V: 随机一致性指标

矩阵阶数	1	2	3	4	5	6	7	8
RI	0	0	0.58	0.90	1.12	1.24	1.32	1.41

最终通过式 (19) 计算检验系数 $CR = 0.056 < 0.1$，所以比较矩阵 A 通过一致性检验。

$$CR = \frac{CI}{RI} \tag{19}$$

VI. 总结与讨论

总结综上所述，对于第一问，我们利用主题预测、只使用留言详情预测以及结合两者指标一起预测的三种方法分别进行测试，最后利用集成学习的方式，将三种模型进行融合，得到最好的效果；对于第二问，我们在将文本进行分类后，综合考虑问题的关注度，突发性与延续性，建立统一热度模型来衡量其热度，并在搜狗新闻数据集上进行验证；对于第三问，由于缺乏数据集，我们利用半监督学习方式，针对不同的指标采用不同的方式进行验证，最终采用层次分析法计算各个指标的评价权重，以对答复进行评估。

参考文献

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[2] Y. Cui, W. Che, T. Liu, B. Qin, Z. Yang, S. Wang, and G. Hu. Pre-training with whole word masking for chinese bert. *arXiv preprint arXiv:1906.08101*, 2019.

[3] H. He. HanLP: Han Language Processing, 2020.

[4] A. Hotho, A. Maedche, and S. Staab. Ontology-based text document clustering. *KI*, 16(4):48–54, 2002.

[5] B. Huang, Y. Yang, A. Mahmood, and H. Wang. Microblog topic detection based on lda model and single-pass clustering. In *International Conference on Rough Sets and Current Trends in Computing*, pages 166–171. Springer, 2012.

[6] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, 1998.

[7] G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[8] C. Wang, M. Zhang, L. Ru, and S. Ma. Automatic online news topic ranking using media focus and user attention based on aging theory. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1033–1042, 2008.

[9] O. Zamir and O. Etzioni. Web document clustering: A feasibility demonstration. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 46–54, 1998.

[10] M. 智库百科. 层次分析法. https://wiki.mbalib.com/wiki/%E5%B1%82%E6%AC%A1%E5%88%86%E6%9E%90%E6%B3%95.