软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(9):4365−4376 [doi: 10.13328/j.cnki.jos.006956]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

Pobe: 一种基于生成式模型的分布外文本检测方法*

欧阳亚文[1,2], 高 源[1,2], 宗 石[2], 鲍 宇[1,2], 戴新宇[1,2]

[1](计算机软件新技术国家重点实验室 (南京大学), 江苏 南京 210023)

[2](南京大学 计算机科学与技术系, 江苏 南京 210023)

通信作者: 戴新宇, E-mail: daixinyu@nju.edu.cn

摘　要: 对于安全可靠的机器学习系统, 具备检测训练集分布外 (out-of-distribution, OOD) 样本的能力十分必要. 基于似然的生成式模型由于训练时不需要样本标签, 是一类非常受欢迎的 OOD 检测方法. 然而, 近期研究表明通过似然来检测 OOD 样本往往会失效, 并且失效原因与解决方案的探究仍较少, 尤其是对于文本数据. 从模型层面和数据层面分析文本上失效的原因: 生成式模型的泛化性不足和文本先验概率的偏差. 在此基础上, 提出一种新的 OOD 文本检测方法 Pobe. 针对生成式模型泛化性不足的问题, 引入 KNN 检索的方式, 来提升模型的泛化性. 针对文本先验概率偏差的问题, 设计一种偏差校准策略, 借助预训练语言模型改善概率偏差对 OOD 检测的影响, 并通过贝叶斯定理证明策略的合理性. 通过在广泛的数据集上进行实验, 证明所提方法的有效性, 其中, 在 8 个数据集上的平均 AUROC 值超过 99%, FPR95 值低于 1%.

关键词: 机器学习; 分布外检测; 生成式模型; 文本检索; 预训练语言模型

中图法分类号: TP18

中文引用格式: 欧阳亚文, 高源, 宗石, 鲍宇, 戴新宇. Pobe: 一种基于生成式模型的分布外文本检测方法. 软件学报, 2024, 35(9): 4365–4376. http://www.jos.org.cn/1000-9825/6956.htm

英文引用格式: Ouyang YW, Gao Y, Zong S, Bao Y, Dai XY. Pobe: Generative Model-based Out-of-distribution Text Detection Method. Ruan Jian Xue Bao/Journal of Software, 2024, 35(9): 4365–4376 (in Chinese). http://www.jos.org.cn/1000-9825/6956.htm

Pobe: Generative Model-based Out-of-distribution Text Detection Method

OUYANG Ya-Wen[1,2], GAO Yuan[1,2], ZONG Shi[2], BAO Yu[1,2], DAI Xin-Yu[1,2]

[1](State Key Laboratory for Novel Software Technology (Nanjing University), Nanjing 210023, China)

[2](Department of Computer Science and Technology, Nanjing University, Nanjing 210023, China)

Abstract: It is essential to detect out-of-distribution (OOD) training set samples for a safe and reliable machine learning system. Likelihood-based generative models are popular methods to detect OOD samples because they do not require sample labels during training. However, recent studies show that likelihoods sometimes fail to detect OOD samples, and the failure reason and solutions are under explored, especially for text data. Therefore, this study investigates the text failure reason from the views of the model and data: insufficient generalization of the generative model and prior probability bias of the text. To tackle the above problems, the study proposes a new OOD text detection method, namely Pobe. To address insufficient generalization of the generative model, the study increases the model generalization via KNN retrieval. Next, to address the prior probability bias of the text, the study designs a strategy to calibrate the bias and improve the influence of probability bias on OOD detection by a pre-trained language model and demonstrates the effectiveness of the strategy according to Bayes' theorem. Experimental results over a wide range of datasets show the effectiveness of the proposed method. Specifically, the average AUROC is over 99%, and FPR95 is below 1% under eight datasets.

Key words: machine learning; out-of-distribution detection; generative model; text retrieval; pre-trained language model

* 基金项目: 国家自然科学基金 (61936012, 61976114)
收稿时间: 2022-06-02; 修改时间: 2022-09-20, 2022-12-22; 采用时间: 2023-04-13; jos 在线出版时间: 2023-09-20
CNKI 网络首发时间: 2023-09-21

对开放世界的机器学习系统而言, 检测样本是否与训练数据来自相同分布至关重要, 例如, 当一个订餐功能的对话系统开发完成并上线后, 面临的用户输入可以是各式各样的, 用户也许会问: 奥密克戎的症状是什么? 由于该对话系统专为订餐设计, 所以训练文本均与订餐领域相关, 当遇到像这样的训练集分布外 (out-of-distribution, OOD) 的文本后[1], 系统没有能力提供相关服务, 所以此时系统应当检测出该类文本, 并给予预设的安全回复.

随着深度学习的发展, 各种新颖的 OOD 检测方法被提出, 主流方法可基于生成式模型[2−4], 分类器[1,5]等, 这些方法都旨在设计一个打分函数: 把输入样本映射为一个分数 (score), 并给予分布外 (OOD) 和分布内 (in-distribution, ID) 样本不同的分数, 如对分布内样本打分较高, 对分布外样本打分较低.

相较于基于分类器的方法, 基于生成式模型的 OOD 检测方法近年来愈发受到研究人员的青睐. 基于分类器的 OOD 检测方法在训练时需要给定样本的类别标签[1], 但对医药, 法律等领域而言, 标签标注往往是昂贵且困难的, 而基于生成式的模型则无需使用样本的类别标签, 所以更加容易实现, 有着更广泛的应用场景. 具体而言, 训练时, 基于生成式模型的 OOD 检测方法通过优化生成式模型来拟合训练样本的分布[4], 测试时, 利用训练完成的生成式模型计算样本的似然作为分数, 似然较低 (小于预设的阈值) 的样本被视为 OOD 样本, 似然较高的被视为 ID 样本.

然而, 有研究者发现, 完全基于生成式模型所计算出的似然在用于 OOD 检测存在失效的风险, 即 OOD 样本在测试阶段的似然可能较高, 而 ID 样本的似然可能较低[4]. 在计算机视觉领域, 不少工作给出了关于 OOD 较高似然的原因[3,6,7], 在自然语言处理领域, 模型失效的原因探究较少, Gangal 等人[2]虽指出生成式模型失效的原因是受到了文本“表层特征 (surface-level)”的影响, 但对“表层特征”并没有进行进一步解释. 因此生成式模型在文本的 OOD 检测为何失效, 仍是一个亟待解决的问题.

针对基于似然的 OOD 检测方案失效的现象, 本文通过分析生成式模型在检测失效的样例 (似然较低的 ID 句子和似然较高的 OOD 句子), 将失败原因归结为以下两个方面: 1) 模型层面, 生成式模型的泛化性受限: 测试阶段, 对于一些分布内的稀有的语言样式 (pattern), 生成式模型仍可能输出较低似然; 2) 数据层面, 输入文本先验概率的偏差: 在对文本先验概率量化后, 我们发现模型输出的似然与文本的先验概率有着强相关性——对于先验概率较高的样本, 生成式模型同样会输出较高的似然, 即使它来自 OOD; 对于先验概率较低的样本, 生成式模型同样会输出较低的似然, 即使它来自 ID.

为解决上述问题, 本文提出一种新的基于生成式模型的 OOD 文本检测方法 (记作 Pobe). 具体来说, Pobe 引入了基于 KNN 的检索技术来检索训练集中的相似样本, 并将由生成式模型对文本的“预测任务”拓展成“预测任务”和“检索任务”的结合, 以提升模型的泛化性. 随后, 为了消除文本先验概率偏差对检测 OOD 的影响, 本文设计一种校准策略: 先使用预训练语言模型 GPT-2[8]来估计文本的先验概率, 接着使用检索增强后的文本似然和文本先验概率的比值作为打分函数来检测 OOD. 理论上, 本文通过贝叶斯定理证明了 Pobe 的合理性; 实验上, 在多个基准数据集上的实验结果表明, Pobe 超过了其他的基线模型, 实现了当前最佳性能, 代码已开源在 https://github.com/Gy915/Pobe.

综上所述, 本文的贡献如下: 1) 从模型层面和数据层面分析, 给出基于似然的生成式模型在检测 OOD 文本时失败的原因: 模型自身的泛化性受限和输入文本先验概率的偏差; 2) 提出一种新的 OOD 文本检测方法来提升生成式模型的泛化性并消除文本先验概率的偏差, 并通过实验验证了方法的有效性.

1 相关工作

1.1 基于生成式模型的分布外检测

该类方法通过生成式模型估计训练样本的分布, 以似然作为打分函数[3,4]. 训练时, 该类方法对训练集样本进行极大似然估计, 过程中样本标签是非必需的. 测试时, 由于生成式模型刻画的是训练数据的分布, 因此直觉上会对与训练集同分布的, 即 ID 样本赋予较高的似然, 对与训练集不同分布的, 即 OOD 样本赋予较低的似然, 所以可以借助似然区分 ID 和 OOD 样本. 然而, Nalisnick 等人[4,9] 发现了基于似然的 OOD 检测方法在图像数据上往往会

失效, 即模型会对部分 OOD 图片赋予较高的似然, 对部分 ID 图片赋予较低的似然. 不少研究者对失败的原因进行了探索, 并发现图片的背景像素[4] , 复杂度[6]等都会导致基于似然的检测方法失效.

研究人员在文本数据上也发现了相似的问题, Gangal 等人[2]认为 OOD 文本中存在的一些“表层特征”会影响模型对似然的估计, 并且用随机扰动后的训练数据额外训练一个背景模型来消除这些特征的影响, 然而, Gangal 等人[2]并未指出“表层特征”的具体内容, 通过扰动的方法来消除这些特征也存在一定的不确定性: 扰动方法和扰动比例的不同都可能会对结果造成较大影响. 而本文为生成式 OOD 检测失败的原因提供了清晰解释, 并设计了一种新的, 不需扰动的解决方法.

1.2 基于分类器的分布外检测

该类方法基于训练样本和样本标签得到的分类器[1,10−12]. 训练时, 该类方法往往使用交叉熵作为损失函数; 测试时, 不同方法依赖分类器推导出的不同打分函数. Hendrycks 等人[1]的工作使用 Softmax 值作为打分函数, 由于 Softmax 值可衡量样本的标签在训练标签集合上的分布, ID 样本的标签属于训练标签集合, 所以 Softmax 值较高, 相反地, OOD 样本的标签不属于训练标签集合, 所以 Softmax 值较低. Softmax 值虽然计算简单, 但 Liu 等人[12] 的工作证明了 OOD 样本的 Softmax 值也有可能较高, 所以 Softmax 值不适合区分 ID 和 OOD 样本; Podolskiy 等人[10] 和 Lee 等人[13]使用样本在特征空间上的向量表示与训练标签高斯分布的马氏距离作为打分函数, 由于 ID 样本的标签属于训练标签集合, 所以其向量表示更加服从训练标签的高斯分布, 马氏距离更小, 相反地, OOD 样本的马氏距离更大, Lee 等人[13]通过实验证明了马氏距离相比 Softmax 值的优越性, 但理论层面, 训练标签是否服从高斯分布仍需要进一步论证. Liu 等人[12]和 Ouyang 等人[11]将分类器推导出的能量值作为打分函数, 尽管能量值的计算简单, 又有理论保证, 但训练时, 在原有的交叉熵损失基础上, 需要借助额外的 OOD 样本对分类器进行联合训练, 这无疑增添了收集数据的成本. 以上介绍的方法虽有各自的优缺点, 但都需要依赖大量的标注数据去训练分类器, 这在许多应用场景下是比较困难且昂贵的, 而本文提出的方法则无需标注数据.

1.3 检索增强的生成式模型

随着神经网络技术的发展, 自然语言生成任务往往通过生成式模型来学习记忆训练集中的语言知识. 然而, 近期研究人员注意到, 生成式模型往往受限于自身的学习能力, 难以学习记忆训练集中的所有语言知识, 特别是复杂训练集 (比如句子较长) 中稀有的语言样式[14−17]. 以语言模型任务为例, 对于训练集中出现次数较少的专有名词, 如“糖醋里脊”, 模型在预测阶段, 难以通过“糖醋里”这一前缀预测出下个字“脊”. 为了解决这一挑战, 一个主流的方案是将训练集中的语言知识向量化, 并存储至数据库中, 在预测阶段从数据库中检索出与预测相关的语言知识用以辅助预测, 这种技术被称为检索增强的生成式技术.

该技术在语言模型[14]、机器翻译[15,18,19]、问答[20]领域都有应用. 以机器翻译为例, Khandelwal 等人[15] 提出的 KNN-MT 模型可以将训练集中的翻译知识以向量的形式存储至数据库中, 在测试阶段, 模型通过向量检索的方式从数据库中获取与当前解码位置最相关的 K 个单词, 并使用距离加权融合的方式获取这些单词的分布, 最终联合生成式模型的输出作为解码结果. 实验显示, 检索增强的生成式模型能够有效提升翻译结果. 本文受到这些领域工作的启发, 首次将检索增强应用在 OOD 检测任务上, 并设计启发式方法来选择生成结果和检索结果, 提升对 ID 文本的建模效果, 进而更好地区分 ID 和 OOD 文本.

2 基于 Pobe 的生成式 OOD 检测方法

本文提出了 Pobe 来增强生成式模型的泛化性和消除文本先验概率偏差. 具体而言, 首先通过引入 KNN 检索来计算泛化性增强后的似然 $p_{\max}(x)$ (见第 2.1 节), 其次通过预训练语言模型估计文本先验概率 $p_{\text{GPT2}}(x)$ (见第 2.2 节), 最终使用两者的比值进行 OOD 检测 (见第 2.3 节), 整体流程可参考图 1.

需要注意的是, 这些过程都只发生于测试阶段, 即检测 OOD 阶段, 在训练阶段不会引入额外的开销, 与一般的生成式方法相同, Pobe 的生成式模型的网络结构可基于流行的 GRU 或 Transformer[4,5], 训练目标可通过对训练集 $D_{\text{train}}=\{x^i\}_{i=1}^N$ 的文本进行极大似然估计, 从而得到生成式模型的参数 θ:

$$\theta=\arg\max_{\theta^*}\sum_{x^i\in D_{\text{train}}}\log p(x^i;\theta^*) \tag{1}$$

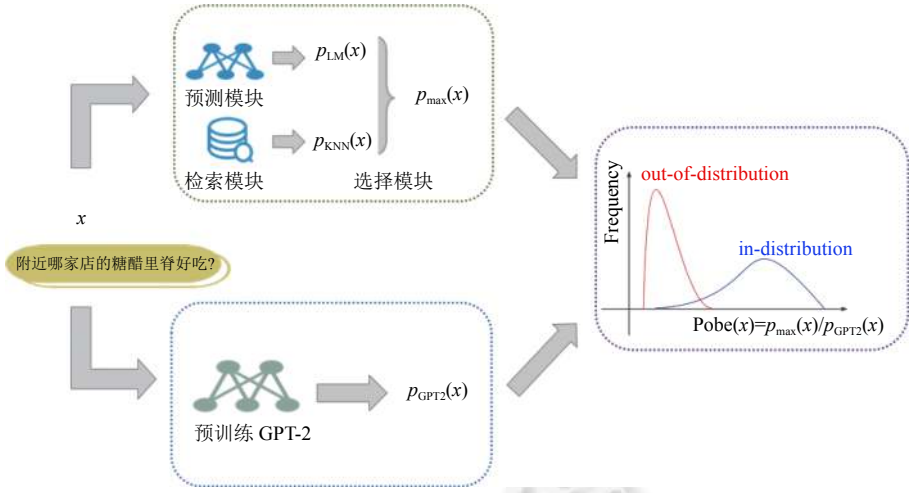


图 1 基于 Pobe 的 OOD 检测模型图

2.1 基于 KNN 的增强

为了更好地建模 ID 样本的分布, 在测试阶段, 本文在生成式模型预测的基础上, 引入基于 KNN 的检索方法, 并启发式地选择模型预测结果和检索结果作为文本似然的最终估值. 引入检索的动机是, 一般的生成式模型通过神经网络参数对训练数据集进行隐式化的记忆, 这种方式对 ID 样本的泛化能力有限, 尤其在一些训练集中的稀有的语言样式上[14,21]. 而检索式方法对训练数据集进行显示化存储, 能够改善隐式化记忆导致的建模能力受限的问题, 进而提升对 ID 样本的泛化能力.

整个流程可归纳为 3 个模块, 给定需要估计似然的测试文本 $x=\{w_1,w_2,\ldots,w_n\}$, 其中 w_i 可表示 x 中的第 i 个字 (也可以表示词), 预测模块通过生成式模型来估计似然, 记作 $p_{LM}(x)$; 检索模块通过 KNN 来检索训练集中相似前缀来估计似然, 记作 $p_{KNN}(x)$; 选择模块从以上两个模块中选择对本文建模较好的模块作为文本似然的最终估值 $p_{max}(x)$.

● 预测模块: 通过生成式模型 θ 来直接估计 x 的似然:

$$\log p_{LM}(x)=\sum_{t=1}^{n}\log p_{LM}(w_t|w_{<t}) \tag{2}$$

其中, $w_{<t}=\{w_1,\ldots,w_{t-1}\}$.

● 检索模块: 通过检索的方式估计 x 似然, 该模块需要对训练文本的每个字与其前缀表示进行存储. 具体而言, 在训练结束后, 先对训练集中的每个文本 $x^i=\{w_1^i,w_2^i,\ldots,w_m^i\}$, 使用 θ 对其编码, 获取每个字 w_{t-1}^i 的隐层向量作为下个字 w_t^i 的前缀表示 (由于 w_{t-1}^i 的隐层向量编码了 $\{w_1^i,w_r^i-w_{t-1}^i\}$ 的信息, 可作为 w_t^i 前缀的表示), 接着将表示作为键 (key), w_t^i 作为值 (value), 以键值对的形式存储至集合 D 中. 例如, 对于“这家店的糖醋里脊很美味”的训练文本, 图 2 绿色模块的第 1 行以“脊”字为例, 在获取其前缀“这家店的糖醋里”的表示, 即“里”字的隐层向量后, 将该向量和“脊”字以键值对的形式进行存储, 用于测试阶段检索.

在检索时, 先通过 θ 测试文本 x 进行编码, 获得 x 中每个字 w_t 的前缀表示, 即 w_{t-1} 的隐层向量, 接着将该向量用作查询 (query), 计算其与 D 中键的欧氏距离, 并挑出最近的 K 个键, 将这些键对应的值 $\{v_1,\ldots,v_K\}$, 连同各自与查询的距离 $\{d_1,\ldots,d_K\}$, 构成集合 $N=\{(v_1,d_1),\ldots,(v_K,d_K)\}$, 最后对集合 N 中的值进行加权得到 p_{KNN}.

$$p_{KNN}(w_t|w_{<t})=\frac{\sum_{(v_i,d_i)\in N}I_{w_t=v_i}\exp(-d_i)}{Z} \tag{3}$$

这里 I 是指示函数, 当 $w_t=v_i$ 时返回 1, $w_t\neq v_i$ 时返回 0, $Z=\sum_{(v_i,d_i)\in N}\exp(-d_i)$ 为归一化系数. 例如, 针对“附近哪家店的糖醋里脊好吃”的测试文本, 图 2 的蓝色模块展示了以“脊”字前缀“附近哪家店的糖醋里”为例的查询, 先

计算其表示与集合 D 中键的欧氏距离, 接着挑选出最近的 3 个键, 将这些键对应的值 (脊, 脊, 骨) 与距离 (1, 2, 4) 存入集合 N 中, 最后通过公式 (3) 计算得到“脊”字的似然.

● 选择模块: 在得到 $p_{\mathrm{LM}}(x)$ 与 $p_{\mathrm{KNN}}(x)$ 后, 该模块启发式地选择对 x 建模较好, 即似然较高的模块作为最终对文本的似然估计 $p_{\max}(x)$:

$$\log p_{\max}(x)=\sum_{t=1}^{n}\max\{\log p_{\mathrm{LM}}(w_t|w_{<t}),\log p_{\mathrm{KNN}}(w_t|w_{<t})\} \tag{4}$$

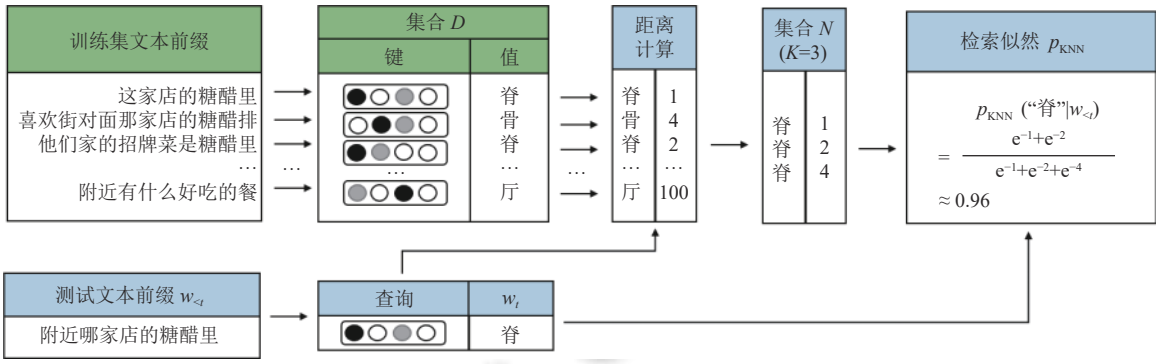


图 2 检索模块示例图

2.2 基于预训练 GPT-2 的先验概率

即使 $p_{\max}(x)$ 对 ID 样本有更好的建模效果, 本文认为用其作为打分函数来检测 OOD 样本仍有不足, 因为会受到的 x 先验概率 $p(x)$ 偏差的影响. 由于 $p_{\mathrm{LM}}(x)$ 用来估计样本的条件概率 $p(x|\mathrm{ID})$ [22], 检索增强后的 $p_{\max}(x)$ 也只是更好地估计 $p(x|\mathrm{ID})$, 而判定 x 属于 OOD 的标准是当且仅当 $p(\mathrm{ID}|x)<0.5$ (当 x 属于 OOD 时, $p(\mathrm{ID}|x)<p(\mathrm{OOD}|x)$, 结合 $p(\mathrm{ID}|x)+p(\mathrm{OOD}|x)=1$, 推出 $p(\mathrm{ID}|x)<0.5$). 根据贝叶斯定理, 有如下公式成立:

$$p(\mathrm{ID}|x)\propto p(x|\mathrm{ID})/p(x) \tag{5}$$

由于 $p(x)$ 项的存在, $p(x|\mathrm{ID})$ 并不正比于 $p(x|\mathrm{ID})$, 例如给定 ID 样本 x_1 和 OOD 样本 x_2, 由前文可知 $p(\mathrm{ID}|x_2)<p(\mathrm{ID}|x_1)$, 假如 $p(x_1)<<p(x_2)$, $p(x_2|\mathrm{ID})$ 反而可能高于 $p(x_1|\mathrm{ID})$. 综上所述, 如果用 $p(x|\mathrm{ID})$ 检测 OOD, 由于 $p(x)$ 的影响, ID 样本的 $p(x|\mathrm{ID})$ 可能低于 OOD 样本, 因此直接用 $p(x|\mathrm{ID})$ 作为打分函数不是最优.

本文为消除 $p(x)$ 的影响, 先对其进行估计, 具体选用由预训练语言模型 GPT-2 得到的概率 (如公式 (6)) 来作为 $p(x)$.

$$\log p_{\mathrm{GPT2}}(x)=\sum_{t=1}^{n}\log p_{\mathrm{GPT2}}(w_t|w_{<t}) \tag{6}$$

由于 GPT-2 是通过数百万的文档进行预训练, 包含了广泛的领域, 本文假设这些文档中既包括了 ID 文本还有 OOD 文本, 因此估计 $p(x|\mathrm{ID},\mathrm{OOD})$ 可等价于 $p(x)$.

2.3 基于 Pobe 的 OOD 检测

为消除先验概率偏差的影响, 本文设计了一种新的校准策略: 使用 $p_{\max}(x)$ 和 $p_{\mathrm{GPT2}}(x)$ 的比值 Pobe(x) 作为打分函数来检测 OOD, 即:

$$\mathrm{Pobe}(x)=p_{\max}(x)/p_{\mathrm{GPT2}}(x) \tag{7}$$

由第 2.2 节可知, $p_{\max}(x)$用于估计 $p(x|\mathrm{ID})$, $p_{\mathrm{GPT2}}(x)$用于估计 $p(x)$, 由公式 (5) 可知, 两项的比值 Pobe(x)$\propto p(\mathrm{ID}|x)$, 所以当 Pobe(x) 小于给定阈值 τ, 可判定 x 是 OOD, 否则判定是 ID. 阈值可根据用户需求选取, 例如实验中用到的 FPR95_IN 评价指标是要求阈值召回不低于 95% 的 ID 样本. 具体检测过程如算法 1 所示.

算法 1. 基于 Pobe 的 OOD 检测.

输入: 测试文本 x, 生成式模型 θ, 预训练 GPT-2, 集合 D, 阈值 τ;
输出: 输出文本 x 是否为 OOD.

1. 根据公式 (4) 计算 x 的检索增强似然 $p_{\max}(x)$
2. 根据公式 (6) 计算 x 的先验概率 $p_{\mathrm{GPT2}}(x)$
3. 根据公式 (7) 计算 Pobe(x)
4. If Pobe(x) < τ
5. Return OOD
6. Else
7. Return ID

3 实验与结果

3.1 实验设置

● 数据集设置: 为了分析方法的性能, 本文参考 Arora 等人[5]的设置, 选取 IMDB[23]数据集作 ID, CLINC150[24], SST2[25], Yelp[26], WOS[27]分别作 OOD; WOS 作 ID, CLINC150, SST2, Yelp, IMDB 分别作 OOD. 一共 8 组 ID-OOD 组合. 其中 IMDB 为长影评情感分类数据集, WOS 为科学类文章类别分类数据集, SST2 为短影评情感分析数据集, CLINC150 为对话系统意图识别的数据集, Yelp 为商户点评情感分析数据集. 选用 IMDB 作 ID, SST2 作 OOD 时, 此时训练集来自 IMDB 训练集, 验证集来自 IMDB 验证集, 测试集来自 IMDB 和 SST2 测试集, 即训练集和验证集都只来自 ID 数据集, 其他 ID-OOD 组合亦同. 关于数据集的统计结果可查看表 1, 需要注意的是, 由于 IMDB 没有提供验证集, 这里选用该数据集提供的无标签数据的前 10 000 条数据作为验证集. 对于 WOS, 参考 Lingkai 等人[28]的设置, 这里选取前 100 个类的数据, 按照 8∶1∶1 划分训练集、验证集、测试集, 表中数据按照“样本个数/平均长度”格式显示.

表 1 实验用到的数据集

数据集	训练集	验证集	测试集
IMDB	25 000/233.8	10 000/236.9	25 000/228.5
WOS	27 940/193.3	3 492/193.3	3 493/192.5
CLINC150	—	—	3 492/193.3
SST2	—	—	1 821/19.2
Yelp	—	—	38 000/135.2

● 基线系统: 为了验证 Pobe 的有效性, 实验选取对数似然估计方法[4], 困惑度方法[5], 对数似然比值[2]方法作为基线系统进行比较, 其中对数似然比值是目前最佳的生成式方法.

(1) 对数似然估计方法 (log-likelihood estimation, LLE): 通过 ID 样本训练生成式模型, 在测试阶段使用生成式模型计算样本的对数似然作为打分函数.

(2) 困惑度方法 (perplexity, PPL): 通过 ID 样本训练生成式模型, 在测试阶段使用生成式模型计算样本的困惑度作为打分函数.

(3) 对数似然比值方法 (log-likelihood ratio, LLR): 通过 ID 样本训练生成式模型, 同时通过扰动后的 ID 样本训练背景模型, 在测试阶段使用两者似然比值的对数作为打分函数.

● 超参数设置及实现: 对于 Pobe 中的生成式模型, 参考 Arora 等人[5]的设置, 采用 12 层的预训练 GPT-2[8]作为生成式模型并进行调优, 早停 (early stop) 设置为 5, 设置初始化学习率为 5×10^{-5}. 对于 Pobe 的检索模块, 采用

Meta (原 Facebook) 公司开发的 Faiss 工具构造索引. 为了兼顾检索的效率和准确率, 参考 Jiang 等人 [18]的设置, 实验采用 IVFFLAT 索引, 设置簇类个数为 100, 搜寻簇类个数为 10, 检索过程中设置近邻个数 k 的范围为 [5, 20, 50, 100, 200, 300, 500, 1024, 2048] 并通过 ID 验证集选择最优超参数: 计算不同 k 取值下 ID 验证集的似然, 并选用使得平均对数似然最高的 k 作为对应的超参数. 对于先验概率估计部分, 采用 12 层的预训练的未调优 GPT-2 来估计 $p_{\mathrm{GPT2}}(x)$.

对于 LLE 方法和 PPL 方法, 为了保证实验的公平性, 生成式模型同样采用 12 层的调优后的 GPT-2, 调优过程与 Pobe 相同; 对于 LLR 的方法, 本文参考 Gangal 等人[2]的设置, 采用隐藏层为 1 层, 隐藏层大小为 300 的自左向右的 LSTM [29] 模型作为语言模型, 使用 100 维的 GloVe [30] 对词表示进行初始化, 每次扰动 50% 的文本, 使其等概率替换成词表里的随机词.

● 评价指标: 参考现有的工作[1,5,10,31]选用 OOD 检测任务常见的评价指标, 具体如下.

(1) 受试者工作特征曲线下面积 (AUROC): 即真阳率-假阳率曲线下面积, 反映模型对正样本和负样本的排序质量. 得分越高, 模型越能区分正负样本, 即性能越好. 不论 ID 还是 OOD 样本作为正样本, AUROC 值都相同. 此评价指标作为主要评价指标.

(2) 精确率-召回率曲线下面积 (AUPR): 反映了模型在查准率和查全率上取得“双高”的比例. 得分越高, 模型性能越好. 在测试过程中, 分别将 ID 和 OOD 视为正样本, 计算 AUPR_IN 和 AUPR_OUT 的值.

(3) FPR95: 当真阳率达到 95% 时, 假阳率的值. 反映了模型在具有较高召回率时误召回的比例. 得分越低, 模型性能越好. 在测试过程中, 同样分别将 ID 和 OOD 视为正样本, 计算 FPR95_IN 和 FPR95_OUT 的值.

3.2 实验结果

各个模型在不同数据集上的评测结果如表 2 所示, ↑ 表示数值越大越好, ↓ 表示数值越小越好. 从表 2 中数据可以得出以下结论.

表 2 OOD 检测性能在不同数据集上结果对比 (%)

ID	OOD	Methods	AUROC ↑	FPR95_OUT ↓	FPR95_IN ↓	AUPR_OUT ↑	AUPR_IN ↑
IMDB	CLINC150	LLE	0.027	99.996	100	8.049	66.154
		PPL	98.490	6.300	5.756	95.813	99.655
		LLR	99.947	0.136	0	**99.381**	99.991
		Ours	**99.988**	**0.032**	**0**	98.849	**99.998**
	SST2	LLE	0.506	99.988	100	3.480	80.458
		PPL	96.314	21.428	14.333	82.824	99.628
		LLR	**99.909**	**0.348**	**0**	**99.671**	**99.993**
		Ours	99.725	0.976	0.165	95.053	99.980
	Yelp	LLE	32.272	99.796	98.505	50.000	29.2060
		PPL	81.964	61.928	62.534	87.025	75.801
		LLR	90.823	40.676	41.516	93.641	87.947
		Ours	**99.866**	**0.388**	**0.474**	**99.912**	**99.804**
	WOS	LLE	54.711	91.212	99.828	12.595	89.548
		PPL	73.181	85.280	72.402	35.370	93.920
		LLR	95.588	23.412	22.674	83.994	99.288
		Ours	**99.998**	**0**	**0**	**99.987**	**100**
WOS	CLINC150	LLE	0.001	100	100	35.744	25.989
		PPL	99.983	0.086	0.033	99.987	99.980
		LLR	**99.994**	**0.057**	**0**	**99.995**	**99.992**
		Ours	99.889	0.172	0	99.736	99.927
	SST2	LLE	0.220	100	100	19.529	44.170
		PPL	99.945	0.086	0.110	99.901	99.971
		LLR	**99.959**	**0.143**	**0**	**99.912**	**99.981**
		Ours	99.919	0.172	0	99.720	99.968

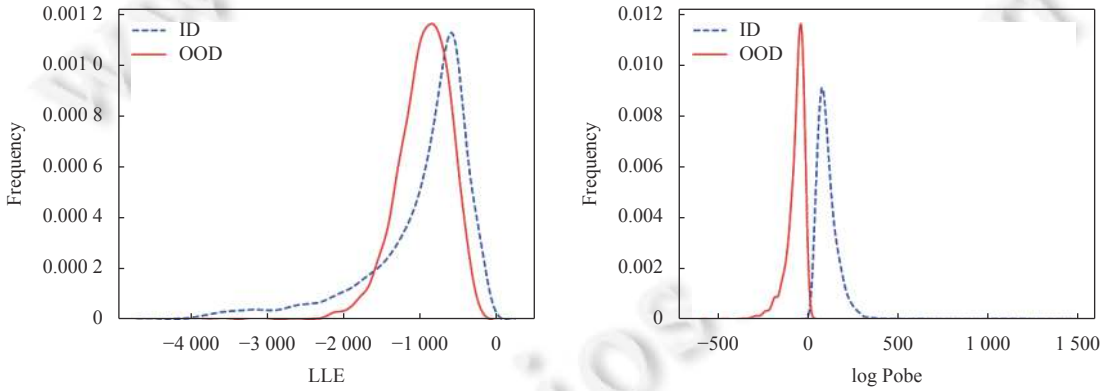
表 2 OOD 检测性能在不同数据集上结果对比 (%) (续)

ID	OOD	Methods	AUROC ↑	FPR95_OUT ↓	FPR95_IN ↓	AUPR_OUT ↑	AUPR_IN ↑
WOS	Yelp	LLE	37.141	99.914	86.732	90.903	6.101
		PPL	97.225	12.740	16.479	99.693	86.712
		LLR	85.013	63.441	46.753	98.382	33.157
		Ours	**99.983**	**0.086**	**0**	**99.998**	**99.919**
	IMDB	LLE	63.876	96.078	71.196	93.464	15.953
		PPL	95.213	18.523	27.812	99.104	82.539
		LLR	91.277	31.492	47.716	98.403	76.092
		Ours	**99.992**	**0.086**	**0**	**99.999**	**99.961**

(1) 本文提出的方法在几乎所有数据集上都达到了评价指标的最佳性能, 例如在 AUROC 上的得分均超过 99.5%, FPR 值均小于 1%, 这不仅表明本文的方法可以很好地检测 OOD, 同样具有很强的通用性. 图 3 更直观地表现出本文的方法相比基线方法的优越性 (图中的 ID 来自 IMDB 数据集, OOD 来自 WOS 数据集), 对于 LLE 方法, ID 和 OOD 重叠面积较大, 表示 ID 和 OOD 分数比较接近, 不适合用来检测 OOD (AUROC 为 54.71%), 而对于 Pobe 方法, ID 和 OOD 重叠面积较小, 表示 ID 和 OOD 分数差异较大, 适合用来检测 OOD (AUROC 为 99.99%).

(2) 基线方法 LLE 方法在 ID 文本长度长, OOD 文本长度短的数据集组合上结果较差, 例如在 IMDB 作为 ID 数据集 (平均文本长度约为 228), CLINC150 作为 OOD 数据集时 (平均文本长度约为 8), AUROC 值小于 1%, FPR 值接近于 100%, 这说明即使模型建模的是 ID 数据集的分布, 但测试时也不会赋予 ID 较高的似然, OOD 较低的似然. 如前文所述, 这是由于: ① 对于较长的 ID 本文, 模型受限于自身容量, 泛化性不足, 导致 ID 文本的似然较低; ② 受到文本先验概率偏差的影响, 此时, 由于 OOD 文本较短, 往往具有较高的先验概率, 所以模型同样会赋予其较高的似然. 本文将在第 3.3 节中具体论证.

(3) 最好的基线方法 LLR 在稳定性上低于 Pobe. 尽管 LLR 在一些 ID-OOD 组合上同样可以达到最佳性能, 但在部分 ID-OOD 组合上性能却远逊于 Pobe (如 IMDB-Yelp 组合), 这可能是由于 LLR 需要对数据集进行随机扰动, 这就为最终 OOD 检测性能引入了不确定性, 而 Pobe 没有扰动, 所以保持了稳定性.



(a) LLE方法 (b) Pobe方法

图 3 LLE 方法和 Pobe 方法的样本分数分布直方图

3.3 讨论和分析

● 消融实验: 为了探究 KNN 检索增强和预训练 GPT-2 先验概率校准两个改进分别对性能的影响, 本文对 Pobe 方法进行消融实验, 结果如表 3 所示, 两个改进单独以及结合起来使用对 OOD 检测的效果均有提升. 其中, GPT-2 的先验概率校准方法对检测效果有着较大的提升, 而检索式的增强方法能够在此基础上进一步提升检测性能, 逼近 OOD 检测的最优效果. 以下是对两个改进的进一步分析.

(1) 为了验证 KNN 检索增强对泛化性的提升, 本文计算了在两个 ID 测试集上引入 KNN 检索增强方法前后

的平均对数似然, 如表 4 所示, 在使用 KNN 检索增强方法后, 两个 ID 测试集的平均对数似然得到明显的提高, 证明了使用 KNN 检索方法能够提升模型的泛化性.

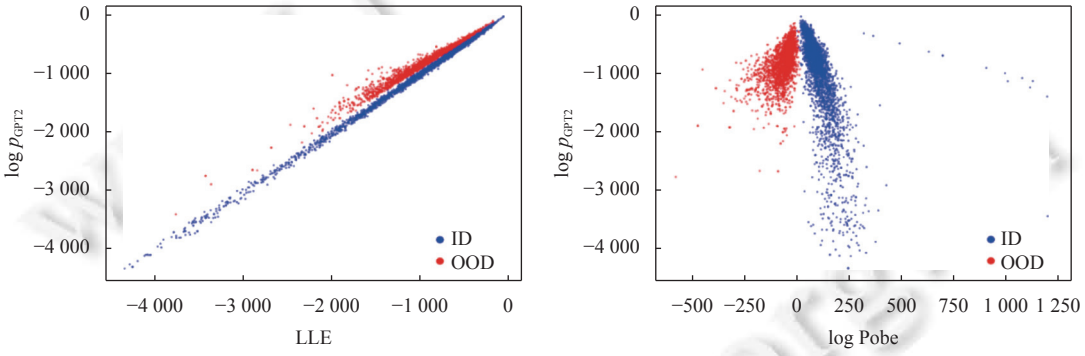
(2) 为了验证 GPT-2 对先验概率的校准效果, 本文随机选取一组 ID-OOD 组合, 分别可视化了 LLE 与 $\log p_{\text{GPT2}}$, $\log \text{Pobe}$ 与 $\log p_{\text{GPT2}}$ 的散点图, 如图 4 所示 (图中的 ID 来自 IMDB 数据集, OOD 来自 WOS 数据集. 为了方便可视化, 每幅图只随机选取了 3 000 个样本点), LLE 与 $\log p_{\text{GPT2}}$ 有着强相关性 (皮尔森系数接近 1), 即对于 $\log p_{\text{GPT2}}$ 较高的样本, LLE 同样较高, 即使它来自 OOD, 这一现象的理论原因也在方法章节中通过贝叶斯定理进行了论证. 而 $\log \text{Pobe}$ 则可以消除文本先验概率的偏差, 即 OOD 样本的值总是较低, ID 样本的值总是较高, 与它对应的 $\log p_{\text{GPT2}}$ 无关, 说明先验概率偏差问题得到了解决.

表 3 Pobe 方法在数据集上的消融研究 (AUROC 作为评测指标) (%)

设置	IMDB				WOS			
	CLINC150	SST2	Yelp	WOS	CLINC150	SST2	Yelp	IMDB
去除GPT-2和KNN	0.03	0.51	32.27	54.71	0	0.22	37.14	63.88
去除GPT-2	0.36	0.89	33.21	57.97	0.07	0.47	38.98	65.84
去除KNN	98.24	95.97	99.23	99.91	99.26	99.52	99.89	99.97
Pobe	99.99	99.73	99.87	100	99.89	99.92	99.98	99.99

表 4 KNN 检索增强方法对 ID 测试集平均对数似然的影响

数据集	原始	KNN检索增强
IMDB	−1 001.92	−945.42
WOS	−784.23	−728.81



(a) LLE方法 (b) Pobe方法

图 4 LLE 与 $\log p_{\text{GPT2}}$ 和 log Pobe 与 $\log p_{\text{GPT2}}$ 之间的关系

● 超参实验: 为了验证不同 k 取值对 OOD 检测性能的影响, 本文选取了两组 ID-OOD 组合, 并可视化了不同 k 取值下的 AUROC 值, 由图 5 所示, 对于不同的 k 值, AUROC 变化幅度不大, 且都好于不用 KNN 的情况, 证明了 k 选择的稳健性.

● KNN 增强似然样例分析: 为了更直接地说明 KNN 模块的作用, 表 5 从 IMDB 测试集中筛选了一些样例 (IMDB 作为 ID 数据集), 对于这些样例, 生成式模型, 即使用 IMDB 训练集调优后的 GPT-2 仍无法很好地建模, 赋予这些词较低的似然: $p_{\text{LM}}(x)$ 都小于 0.01, 而使用 KNN 模块后, 模型可以直接检索训练集的数据, 得到具有相似前缀的近邻, 利用近邻的 next word 计算得到的 $p_{\text{KNN}}(x)$ 都大于 0.5, 实现对 ID 样本更好的建模效果.

● 消除先验概率偏差样例分析: 为了更直观地说明预训练 GPT-2 对先验概率偏差的消除, 表 6 采样了一些来自分布内和分布外的样例, 可以发现即使样例来自分布内 (样例 1), 其对数似然 LLE 仍可能低于其他分布外的样本 (样例 2, 3, 4). 同时可以发现 LLE 和先验概率 $\log p_{\text{GPT2}}$ 有较强的相关性, 而 Pobe 则缓解了和 $\log p_{\text{GPT2}}$ 的强相关性问题, 消除了先验概率的偏差, 达到很好地区分 ID 和 OOD 样本的效果.

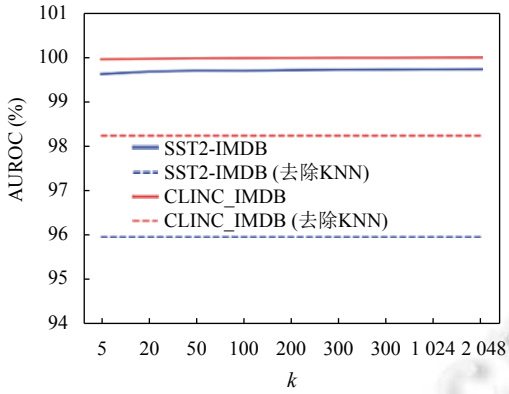


图 5 检索模块中近邻个数 k 与 AUROC 之间的关系

表 5 KNN 模块增强似然样例

示例	内容	next word	Likelihood
Case1	Son in Law is a good comedy worth your **<word>** …	time	$p_{\text{LM}}(w_i = \text{time}\|w_{<i}) = 0.0019$
Neighbors from datastore	Paper House is worth your **<word>** …	time	$p_{\text{KNN}}(w_i = \text{time}\|w_{<i}) = 0.6018$
	It's definitely worth your **<word>** …	time	
	I guarantee it's worth your **<word>** …	money	
	Trust me this is worth your **<word>** …	6	
Case2	… It's a feel-**<word>** film and …	good	$p_{\text{LM}}(w_i = \text{good}\|w_{<i}) = 0.008$
Neighbors from datastore	This is a so called 'feel-**<word>**' …	good	$p_{\text{KNN}}(w_i = \text{good}\|w_{<i}) = 1$
	In the end, a **<word>** …	good	
	This is simply a nice, feel-**<word>** …	good	
	… and enjoys a feel-**<word>** …	good	
	Even the only supposed-**<word>** …	good	

表 6 消除先验概率偏差样例

No.	Utterance	Dataset	Distribution	LLE	$\log p_{\text{GPT2}}$	log Pobe
1	I thought this movie was horrible. I was bored and had to use all the self control I have to not scream at the screen. Mod Squad was beyond cheesy, beyond cliche, and utterly predictable.	IMDB	In	−134.42	−155.53	27.09
2	Great place! Good pizza that tastes great and is very authentic! We had a mushroom and sausage and we were very impressed. You won't be disappointed!	Yelp	Out	−112.00	−105.28	−2.66
3	an experience so engrossing it is like being buried in a new environment.	SST2	Out	−69.60	−66.00	−1.39
4	please tell me how traffic from the new jersey turnpike into the lincoln tunnel looks currently	CLINC150	Out	−121.80	−111.89	−2.08

4 总 结

分布外检测的目标是为了检测来自训练分布外的样本, 为达这一目标, 本文提出 Pobe 方法. Pobe 基于生成式模型, 并通过检索的方式来提升模型的泛化性, 使用预训练语言模型来消除样本先验概率偏差. 实验表明, Pobe 方法可以极大提升基线模型的性能, 并在多个数据集上达到当前最佳性能.

本文未来会在不同领域、模态的数据集上对 Pobe 进行测试. 在检测出 OOD 文本后, 如何对这些文本进一步挖掘, 如聚类, 生成关键信息等, 从而减轻数据的人工标注成本, 提升系统迭代效率, 将成为本文的未来工作.

References:

[1] Hendrycks D, Gimpel K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: Proc. of the 5th Int'l Conf. on Learning Representations. Toulon: OpenReview.net, 2017.

[2] Gangal V, Arora A, Einolghozati A, Gupta S. Likelihood ratios and generative classifiers for unsupervised out-of-domain detection in task oriented dialog. In: Proc. of the 34th AAAI Conf. on Artificial Intelligence. New York: AAAI Press, 2020. 7764–7771. [doi: 10.1609/aaai.v34i05.6280]

[3] Ren J, Liu PJ, Fertig E, Snoek J, Poplin R, DePristo MA, Dillon JV, Lakshminarayanan B. Likelihood ratios for out-of-distribution detection. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2019. 1317. [doi: 10.5555/3454287.3455604]

[4] Nalisnick ET, Matsukawa A, Teh YW, Görür D, Lakshminarayanan B. Do deep generative models know what they don't know? In: Proc. of the 7th Int'l Conf. on Learning Representations. New Orleans: OpenReview.net, 2019.

[5] Arora U, Huang W, He H. Types of out-of-distribution texts and how to detect them. In: Proc. of the 2021 Conf. on Empirical Methods in Natural Language Processing. Punta Cana: Association for Computational Linguistics, 2021. 10687–10701. [doi: 10.18653/v1/2021.emnlp-main.835]

[6] Serrà J, Álvarez D, Gómez V, Slizovskaia O, Núñez JF, Luque J. Input complexity and out-of-distribution detection with likelihood-based generative models. In: Proc. of the 8th Int'l Conf. on Learning Representations. Addis Ababa: OpenReview.net, 2020.

[7] Schirrmeister RT, Zhou YX, Ball T, Zhang D. Understanding anomaly detection with deep invertible networks through hierarchies of distributions and features. In: Proc. of the 34th Conf. on Neural Information Processing Systems. Vancouver: NeurIPS, 2020. 21038–21049.

[8] Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I. Language models are unsupervised multitask learners. OpenAI Blog, 2019, 1(8): 9.

[9] Nalisnick E, Matsukawa A, Teh YW, *et al*. Detecting out-of-distribution inputs to deep generative models using typicality. In: Proc. of the 8th Int'l Conf. on Learning Representations. 2020.

[10] Podolskiy A, Lipin D, Bout A, Artemova E, Piontkovskaya I. Revisiting Mahalanobis distance for Transformer-based out-of-domain detection. In: Proc. of the 35th AAAI Conf. on Artificial Intelligence. AAAI Press, 2021. 13675–13682. [doi: 10.1609/aaai.v35i15.17612]

[11] Ouyang YW, Ye JS, Chen Y, Dai XY, Huang SJ, Chen JJ. Energy-based unknown intent detection with data manipulation. In: Proc. of the 2021 Findings of the Association for Computational Linguistics. Association for Computational Linguistics, 2021. 2852–2861. [doi: 10.18653/v1/2021.findings-acl.252]

[12] Liu WT, Wang XY, Owens JD, Li YX. Energy-based out-of-distribution detection. In: Proc. of the 34th Conf. on Neural Information Processing Systems. Vancouver: NeurIPS, 2020. 21464–21475.

[13] Lee K, Lee K, Lee H, Shin J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In: Proc. of the 32nd Conf. on Neural Information Processing Systems. Montreal: NeurIPS, 2018. 7167–7177.

[14] Khandelwal U, Levy O, Jurafsky D, Zettlemoyer L, Lewis M. Generalization through memorization: Nearest neighbor language models. In: Proc. of the 8th Int'l Conf. on Learning Representations. Addis Ababa: OpenReview.net, 2020.

[15] Khandelwal U, Fan A, Jurafsky D, Zettlemoyer L, Lewis M. Nearest neighbor machine translation. In: Proc. of the 9th Int'l Conf. on Learning Representations. OpenReview.net, 2021.

[16] Gu JT, Wang Y, Cho K, Li VOK. Search engine guided neural machine translation. In: Proc. of the 32nd AAAI Conf. on Artificial Intelligence. New Orleans: AAAI Press, 2018. 5133–5140. [doi: 10.1609/aaai.v32i1.12013]

[17] Borgeaud S, Mensch A, Hoffmann J, *et al*. Improving language models by retrieving from trillions of tokens. In: Proc. of the 39th Int'l Conf on Machine Learning. Baltimore: PMLR, 2022. 2206–2240.

[18] Jiang QN, Wang MX, Cao J, Cheng SB, Huang SJ, Li L. Learning kernel-smoothed machine translation with retrieved examples. In: Proc. of the 2021 Conf. on Empirical Methods in Natural Language Processing. Punta Cana: Association for Computational Linguistics, 2021. 7280–7290. [doi: 10.18653/v1/2021.emnlp-main.579]

[19] Feng Y, Zhang SY, Zhang AD, Wang D, Abel A. Memory-augmented neural machine translation. In: Proc. of the 2017 Conf. on Empirical Methods in Natural Language Processing. Copenhagen: Association for Computational Linguistics, 2017. 1390–1399. [doi: 10.18653/v1/D17-1146]

[20] Kassner N, Schütze H. BERT-KNN: Adding a KNN search component to pretrained language models for better QA. In: Proc. of the 2020 Findings of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 3424–3430. [doi: 10.18653/v1/2020.findings-emnlp.307]

[21] He JX, Neubig G, Berg-Kirkpatrick T. Efficient nearest neighbor language models. In: Proc. of the 2021 Conf. on Empirical Methods in Natural Language Processing. Punta Cana: Association for Computational Linguistics, 2021. 5703–5714. [doi: 10.18653/v1/2021.emnlp-main.461]

[22] Bishop CM. Novelty detection and neural network validation. IEE Proc.—Vision, Image and Signal Processing, 1994, 141(4): 217–222. [doi: 10.1049/ip-vis:19941330]

[23] Maas AL, Daly RE, Pham PT, Huang D, Ng AY, Potts C. Learning word vectors for sentiment analysis. In: Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland: Association for Computational Linguistics, 2011. 142–150.

[24] Larson S, Mahendran A, Peper JJ, Clarke C, Lee A, Hill P, Kummerfeld JK, Leach K, Laurenzano MA, Tang LJ, Mars J. An evaluation dataset for intent classification and out-of-scope prediction. In: Proc. of the 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int'l Joint Conf. on Natural Language Processing. Hong Kong: Association for Computational Linguistics, 2019. 1311–1316. [doi: 10.18653/v1/D19-1131]

[25] Socher R, Perelygin A, Wu J, Chuang J, Manning CD, Ng AY, Potts C. Recursive deep models for semantic compositionality over a sentiment treebank. In: Proc. of the 2013 Conf. on Empirical Methods in Natural Language Processing. Seattle: Association for Computational Linguistics, 2013. 1631–1642.

[26] Zhang X, Zhao JB, LeCun Y. Character-level convolutional networks for text classification. In: Proc. of the 28th Int'l Conf. on Neural Information Processing Systems. Montreal: NIPS, 2015. 649–657. [doi: 10.5555/2969239.2969312]

[27] Kowsari K, Brown DE, Heidarysafa M, Meimandi KJ, Gerber MS, Barnes LE. HDLtex: Hierarchical deep learning for text classification. In: Proc. of the 16th IEEE Int'l Conf. on Machine Learning and Applications. Cancun: IEEE, 2017. 364–371. [doi: 10.1109/ICMLA.2017.0-134]

[28] Kong LK, Jiang HM, Zhuang YC, Lyu J, Zhao T, Zhang C. Calibrated language model fine-tuning for in- and out-of-distribution data. In: Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2021. 1326–1340. [doi: 10.18653/v1/2020.emnlp-main.102]

[29] Sundermeyer M, Schlüter R, Ney H. LSTM neural networks for language modeling. In: Proc. of the 13th Annual Conf. of the Int'l Speech Communication Association. Portland: ISCA, 2012. 194–197.

[30] Pennington J, Socher R, Manning C. GloVe: Global vectors for word representation. In: Proc. of the 2014 Conf. on Empirical Methods in Natural Language Processing. Doha: Association for Computational Linguistics, 2014. 1532–1543. [doi: 10.3115/v1/D14-1162]

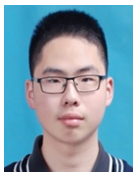
[31] Zhu PF, Zhang WY, Wang Y, Hu QH. Multi-granularity inter-class correlation based contrastive learning for open set recognition. Ruan Jian Xue Bao/Journal of Software, 2022, 33(4): 1156−1169 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6468.htm [doi: 10.13328/j.cnki.jos.006468]

附中文参考文献:

[31] 朱鹏飞, 张琬迎, 王煜, 胡清华. 考虑多粒度类相关性的对比式开放集识别方法. 软件学报, 2022, 33(4): 1156–1169. http://www.jos.org.cn/1000-9825/6468.htm [doi: 10.13328/j.cnki.jos.006468]

欧阳亚文(1996－), 男, 博士生, 主要研究领域为自然语言理解, 开放环境下的机器学习.

鲍宇(1993－), 男, 博士, 主要研究领域为自然语言处理, 科学智能.



高源(1998－), 男, 硕士生, 主要研究领域为自然语言处理, 机器学习.

戴新宇(1979－), 男, 博士, 教授, 博士生导师, CCF 专业会员, 主要研究领域为自然语言处理, 知识工程.

宗石(1992－), 男, 博士, 主要研究领域为计算语言学.