Towards Automatic Curation of Antibiotic Resistance Genes via Statement Extraction from Scientific Papers: A Benchmark Dataset and Models

Sidhant Chandak[♣], **Liqing Zhang**[♠], **Connor Brown**[♠], **Lifu Huang**[♠]
[♣]Indian Institute of Technology Kanpur, [♠]Virginia Tech
[♣]sidhant@iitk.ac.in, [♠]{lqzhang,clb21565,lifuh}@vt.edu

Abstract

Antibiotic resistance has become a growing worldwide concern as new resistance mechanisms are emerging and spreading globally, and thus detecting and collecting the cause – Antibiotic Resistance Genes (ARGs), have been more critical than ever. In this work, we aim to automate the curation of ARGs by extracting ARG-related assertive statements from scientific papers. To support the research towards this direction, we build SciARG, a new benchmark dataset containing 2,000 manually annotated statements as the evaluation set and 12,516 silver-standard training statements that are automatically created from scientific papers by a set of rules. To set up the baseline performance on SciARG, we exploit three state-of-the-art neural architectures based on pre-trained language models and prompt tuning, and further ensemble them to attain the highest 77.0% F-score. To the best of our knowledge, we are the first to leverage natural language processing techniques to curate all validated ARGs from scientific papers. Both the code and data are publicly available at https://github.com/VT-NLP/SciARG.

1 Introduction

Antibiotic resistance (AR), the ability of bacteria to survive and propagate in the presence of antibiotics, is a prevalent phenomenon worldwide and poses a serious health threat to humans and animals. Automatically detecting the antibiotic resistance genes (ARGs- the root cause of AR) in clinical and natural environments has been critical for mitigating the spread of AR. However, though the research on ARGs has grown exponentially over the past 10-15 years, existing ARG databases, such as CARD (Alcock et al., 2020), ARDB (Liu and Pop, 2009), ARGO (Scaria et al., 2005), and ARGMiner (Arango-Argoty et al., 2020), only contain a fraction of ARGs that have been discovered and validated by researchers, making it difficult to fully keep track of the research on ARGs.

Statement 1: Gram-negative Enterobacteriaceae with *resistance* to *carbapenem* conferred by New Delhi metallo-*beta-lactamase* 1 (**blaNDM-1**) are potentially a major global health problem.

Statement 2: The **NDM 1** producing Gram-negative bacteria are mainly Enterobacteriaceae, which can cause colonization or fatal infections, with worrying antimicrobial susceptibility profiles: some isolates have developed *resistance* to practically all available *antibiotics*.

Figure 1: Example of assertive statements for ARGs. The red color shows the target genes while blue background indicates the contextual features.

To automate the process of collecting validated ARGs to enrich the ARG databases, we propose a literature mining approach to automatically extract the assertive statements that indicate the antibiotic resistance property of genes from scientific papers with computational approaches. Based on these assertive statements, we can easily collect all the validated ARGs in the literature. Taking the two statements extracted from (Kumarasamy et al., 2010) in Figure 1 as examples, we can confidently infer the antibiotic resistance of *NDM-1* based on the highlighted contextual words as *beta-lactamases are enzymes produced by bacteria that provide multi-resistance to beta-lactam antibiotics*.

In this paper, we introduce SciARG, the first benchmark dataset for extracting statements that indicate antibiotic resistance of genes from scientific publications. SciARG contains 2,000 and 286 statements with target genes that are manually annotated by domain experts as the test and dev dataset, and about 12,516 silver-standard training statements which are automatically created by a set of rules. The rules are carefully designed by two experts in ARG research. Each statement is a natural language sentence containing a target gene, and is labeled as *Positive* or *Negative*, indicating whether the statement implies antibiotic resistance of the target gene or not. To establish the baseline performance on SciARG, we design three approaches by leveraging the state-of-the-art

Proceedings of the BioNLP 2022 workshop, Dublin, Ireland, pages 402–411

pre-trained language model and prompt tuning. As the training statements are created based on rules, the approaches are very easily overfitting to the keywords from the rules. To mitigate overfitting, we employ a mask language model pre-training strategy which is shown effective in improving the generalization of the baseline approaches. The ensemble of the three supervised approaches attain the highest 77.0% F-score on SCIARG. In summary, we make the following contributions:

- To the best of our knowledge, we are the first to curate ARGs from scientific papers by leveraging natural language processing techniques.
- We build the first benchmark dataset to support the research on ARG-related assertive statement prediction and establish baseline performance based on state-of-the-art pre-trained language models and prompt tuning techniques.

2 Dataset Design

2.1 Statement Collection

To collect the positive statements, we need to first get a collection of validated ARGs as the target genes. To do so, we leverage the CARD (Alcock et al., 2020) database which is a rigorously curated collection of characterized, peer-reviewed resistance determinants and associated antibiotics. CARD contains 3,100 ARGs while for 2,207 of them, CARD provides related PMIDs or PMCIDs from PubMed as reference. To collect the statements about these target ARGs, we leverage the Pubtator API[1] to get the full-text articles based on the PMCIDs of each target gene. As many articles are not freely available, we finally crawl 102 full-text articles for 91 ARGs.

For each of the 102 full-text articles, we segment them into sentences and extract the sentences that contain the target ARG as candidate statements. To enrich the context of each statement, we also prepend the preceding sentence and append the following sentence. In this way, we collect 2,286 statements for 91 confirmed ARGs. We then ask a senior student majoring in Biomedical Sciences to verify the statement in terms of whether they indicate the antibiotic resistance property of the target ARG, and an expert PhD student who has done extensive research on ARG to verify 100 samples randomly selected from the annotations. The inter-annotator-agreement is 88%, indicating that the annotations are mostly correct. In cases of disagreement between the two annotators, we ask them to discuss and achieve an agreement in terms of the label. We take 286 manually annotated statements as the dev set and the remaining 2,000 as the test set. The dev set is carefully chosen such that it has perfectly balanced classes, while the test set contains 1,083 positive and 917 negative statements.

2.2 Silver Training Set Creation

To create the training dataset, we take the remaining 2,105 PMIDs/PMCIDs for which we cannot successfully collect any full-text articles as seeds, and apply the Entrez API[2] to retrieve the papers that cite or are being cited by these seed papers. The assumption is that, if a paper cites or is cited by the paper about a particular ARG, it's more likely about ARGs as well. Based on this assumption, we follow the same procedures as Section 2.1 to collect additional 24,733 statements for 1,133 target ARGs. As it's very expensive and time consuming for a human to manually annotate these statements, we design the following rules based on the antibiotic resistance mechanisms to automatically create the positive training statements:

Rule 1: *If a statement mentions a particular antibiotic, together with "resistan" (the stem of resistance) or "efflux", it will be labeled as positive.*

The rule is based on the fact that the *efflux* of the drug from the bacterial cell is a key antibiotic resistance mechanism generally found in gram-negative bacteria. To apply this rule, we collect 604 antibiotics from the CARD database which cover the synonyms, abbreviations and common names of antibiotics. Two examples are shown in Table 1.

Rule 2: *If a statement mentions any of the enzymes produced by bacteria that catalyzes antibiotic hydrolysis, it will be labeled as positive.*

The enzystome[4] is a community of thousands of enzymes and its mutants, responsible for antibiotic resistance. These enzymes act by modifying the

[1] https://www.ncbi.nlm.nih.gov/research/pubtator/api.html

[2] https://www.ncbi.nlm.nih.gov/pmc/tools/cites-citedby/

[3] mdtEF is a multidrug transport class of efflux pump that confers resistance to a variety of drugs

[4] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6351036/

Rule	Examples
Rule 1	*Example 1*: Detection of rpsI-associated integrases in Bacillus and S . aureus reveals a potential for broad-host range dissemination of the novel methicillin resistance gene mecD. Macrococcus is evolutionarily closely related to the genus Staphylococcus, but possesses a distinctly smaller genome with a size of 2
	Example 2: Deletion of mdtEF [3] completely suppressed GadX-mediated multidrug resistance. Our results indicate that the GadX regulator, in addition to its role in acid resistance, increases multidrug resistance in E . coli by activating the MdtEF multidrug efflux pump .
Rule 2	*Example*: The emergence of one of the most recently described carbapenemases, namely, the New Delhi metallo-lactamase (NDM-1), constitutes a critical and growingly important medical issue . This resistance trait compromises the efficacy of almost all lactams (except aztreonam), including the last resort carbapenems
Rule 3	*Example*: the bla NDM-type genes are found to be either plasmid- or chromosome-located, and in the rare NDM-1-producing P . aeruginosa, the bla NDM-1 gene was found to be chromosomally located . Investigations on the immediate genetic environment of bla NDM genes revealed the presence of a conserved structure that always associated the complete or truncated insertion sequence ISAba125 at the 5'-end and the ble MBL gene (encoding resistance to the anticancer drug bleomycin) at the 3'-end of the bla NDM genes
Rule 4	*Example*: MIC values of beta-lactams for the E . coli TOP10 strain, which harbours recombinant plasmid pTOPO-MUS-2, showed that the bacteria was resistant to amoxicillin and ticarcillin and had a reduced susceptibility to piperacillin, in addition it showed an increased resistance to extended-spectrum cephalosporins and carbapenems by at least four-fold of MIC (Table 2) . Finally, PFGE analysis showed the three strains of M.

Table 1: Example statements of each rule. Green colour indicates the target ARG while Red colour highlights the keywords from the corresponding rule.

cellular targets of various antimicrobial drugs, or by modifying the antimicrobial drug itself. If a statement contains any of the enzymes, we will label it as positive. An example is shown Table 1.

Rule 3: *If the prefix of the target gene is an ARG indicator, we will label the statement as positive.*

The prefix of the target gene sometimes provides clues about whether the gene confers antibiotic resistance or not. The indicator can be either "bla" or "mec": *bla* genes are resistant to beta-lactam antibiotics and *mec* genes are resistant to methicillin antibiotic. An example is shown Table 1.

Rule 4: *If the statement mentions "MIC" and "increase" or "fold" within a context window of 10 words, we will label it as positive.*

Minimum inhibitory concentration (MIC) is the lowest concentration of an antibiotic that inhibits visible growth of the microorganism. Antimicrobial susceptibility tests (ASTs) measures the ability of an antibiotic or other antimicrobial agent to inhibit the in vitro microbial growth. The results of the test (e.g., *increase in MIC*, *MIC becoming multiple fold*) tells us whether the organism is susceptible to the antibiotic or resistant. An example is shown in Table 1.

Statistics	Train	Dev	Eval
# of Target Genes	1,886	56	91
# of Statements	12,516	286	2,000
Average Length of Statement	61.2	55.0	56.3
Minimum Length of Statement	3	9	8
Maximum Length of Statement	502	121	232

Table 2: Statistics of SciARG

Based on the above rules, we collect 6,258 positive statements out of 24,733 candidates. To collect negative statements, we first get a list of human genes from HGNC[5]. If a gene is not included in CARD (Alcock et al., 2020) or ARGMiner (Arango-Argoty et al., 2020), we consider it as a non-ARG and further collected papers

[5]The resource for approved human gene nomenclature https://www.genenames.org/

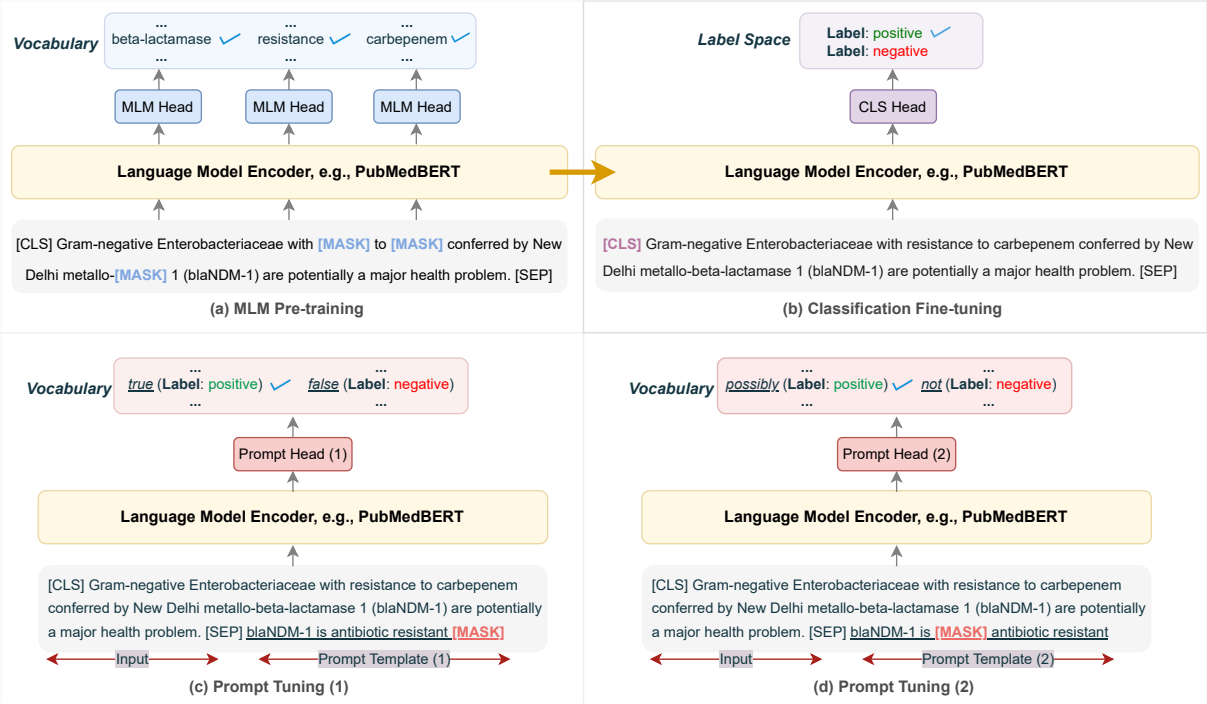

Figure 2: Overview of the three approaches for ARG statement prediction.

about it and statements from the papers. If the statement does not satisfy any of the above rules, we take it as a negative training statement. Finally, we randomly sample 6,258 negative training statements and obtain 12,516 statements in total as the training set. We name the dataset as SCIARG and show the statistics of SCIARG in Table 2.

3 Approach

To set up the baseline performance on SCIARG, we exploit three supervised approaches.

3.1 Supervised Classification

As Figure 2 (b) shows, given an input statement $X = [x_0, x_1, ..., x_n]$ for a target gene $g = [x_i, ..., x_j]$, we first apply the tokenizer of PubMedBERT (Shin et al., 2020a), a state-of-the-art pre-trained language model from PubMed papers, and concatenate all tokens to form a new sequence $[\texttt{[CLS]}, X, \texttt{[SEP]}]$, where `[CLS]` is a special token used for classification and `[SEP]` is a delimiter. We use a position label 1 to indicate the tokens from the target gene and 0 for all the remaining tokens from the statement. Then each token is initialized with a vector by summing the corresponding token, segment and position embeddings from the pre-trained PubMedBERT, and encoded into a hidden state. We use $[\boldsymbol{H}_{cls}, \boldsymbol{H}_{x_0}, ..., \boldsymbol{H}_{x_n}, \boldsymbol{H}_{sep}]$ to denote the encoding outputs. Finally, we predict a label for the statement based on $\boldsymbol{H}_{cls}$, and use the negative log likelihood as the training objective:

$$L = -\log(\text{softmax}(W_1 \boldsymbol{H}_{cls})) \quad (1)$$

where W_1 is a learnable parameter matrix.

It turns out that the model easily overfits to the keywords of the rules (e.g., *efflux*, *resistance*, *MIC*) that are used to create the training samples. To overcome this issue, we further add a mask languge modeling (MLM) pre-training strategy to encourage the model to learn more features from context. As Figure 2 (a) shows, given an input statement, we find all the keywords that are from the rules, and randomly replace $m \in \{0\%, 25\%, 50\%, 75\%, 100\%\}$ of such tokens with `[MASK]`. Then, we apply the same MLM objective as PubMedBERT to ask the model to recover the original token for each `[MASK]`. The training objective of MLM is also based on the negative log likelihood:

$$L = -\log(\text{softmax}(W_2 \boldsymbol{H}_{mask})) \quad (2)$$

where W_2 is another learnable parameter matrix.

We explore two training strategies: optimizing the MLM objective (Equation 2) and the supervised classification objective (Equation 1) simultaneously or sequentially. The sequential training strategy shows better performance.

3.2 Prompt Tuning

To predict the antibiotic resistance property of genes, it also requires extensive domain specific knowledge, e.g., *interpreting results of ASTs, knowledge of the enzymes or efflux pumps that are responsible for antibiotic resistance*, which is likely to have been captured by the large-scale pre-trained language models. To better induce such knowledge, we further exploit prompt tuning based approaches.

Specifically, we design two prompts: (P_1) "The <*target gene*> is antibiotic resistant `[MASK]`", and (P_2) "The <*target gene*> is `[MASK]` antibiotic resistant", where <*target gene*> refers to the gene of interest in each input statement. As shown in Figure 2 (c) and (d), we concatenate each input statement with each prompt as $[\texttt{[CLS]}, X, \texttt{[SEP]}, P, \texttt{[SEP]}]$, and get a contextual representation for each token within the sequence based on PubMedBERT. Based on the contextual representation of `[MASK]` in the prompt, we apply a linear function with softmax to predict a probability for each token in the target vocabulary.

$$L = -\log(\text{softmax}(W_P \boldsymbol{H}_{mask|P})) \quad (3)$$

where $P \in \{P_1, P_2\}$. W_P are learnable parameters for each prompt learning approach. $\boldsymbol{H}_{mask|P}$ denotes the contextual representation of `[MASK]` from the corresponding prompt.

For prompt P_1, we use *true* and *false* as the label of *positive* and *negative* category respectively, and compare their probabilities to get the final label. Similarly, for prompt P_2, we use *possibly* and *not* to predict the label of each statement. Similar as the supervised classification approach, we first pre-train the PubMedBERT with the MLM objective (Equation 2) and then fine-tune it with the prompts based on the negative log likelihood objective (Equation 3).

4 Experiments

4.1 Experiment Setup

We compare our approaches with baseline methods that are based on the rules illustrated in Section 2.2. We use the classification F-score on the positive statements as the evaluation metric, and use grid search to tune the parameters: training epochs 10, learning rate $\in$ {2e-5, 3e-5, 5e-5}, training batch size $\in$ {8, 12, 16, 20}.

4.2 Results and Analysis

Table 3 shows the performance of varying approaches on SCIARG. We can see that, (1) the precision, recall and F-score of different rules vary a lot across the development and evaluation sets. For example, Rule 2 results in the highest precision and recall among the four rules on the development set while the Rule 3 yields the highest recall on the evaluation set. We ascribe it to the sampling of the evaluation instances - though we carefully select the evaluation subset to make sure it's balanced in terms of the target positive/negative labels, we cannot guarantee that the rationals for the target labels are also balanced. (2) the supervised approaches perform significantly better than the rule based methods, especially on recall, demonstrating that the rules are not enough to retrieve most of the positive statements; (3) the classification based approach with MLM pre-training outperforms the two prompt-tuning based methods, due to the possible reason that the prompts were hand engineered and could be sub-optimal. However, by analyzing the errors of the three supervised approaches, we also notice that the prompt-tuning based methods tend to make more positive predictions, and perform better on the statements with complex or ambiguous context. Taking the following sentence as an example:

"*aeruginosa, is underway . The combined effects of various signals mediated by multiple regulators, including CpxR and MexR, on MexAB-OprM expression will be understood in a broader physiological context in the near future . For the determination of putative orthologous proteins, a primary BLASTP search in a given genome was conducted for the gene with the highest similarity.*"

The classification approach mistakenly predicted it as negative. However, *MexAB-OprM* is a major *efflux pump of P. aeruginosa*, a common disease causing gram-negative bacteria, that contributes to clinical antibiotic resistance. Such knowledge is possibly captured by the pre-trained language model and the prompt-tuning methods can better induce such knowledge from PubMedBERT and thus make correct predictions.

Based on the above observation, we further ensemble the three supervised approaches based on their predicted label for each statement. [6] Specifically, for each statement, we label it as positive if

[6] We tried several ensembling strategies and the one we discussed provides the best performance.

Model	Dev (%)			Eval (%)		
	Precision	Recall	F-Score	Precision	Recall	F-Score
Baseline w/ Rule 1	40.0	2.8	5.3	85.9	4.5	8.6
Baseline w/ Rule 2	72.2	9.1	16.2	96.6	10.5	19.0
Baseline w/ Rule 3	20.6	4.9	7.9	89.9	20.5	33.4
Baseline w/ Rule 4	50.0	2.1	4.0	84.8	2.6	5.0
Baseline w/ All Rules	37.1	16.1	22.4	89.3	32.2	47.4
Classification w/ MLM	57.0	88.1	69.2	63.3	92.7	75.2
Prompt 1 w/ MLM	52.4	97.9	68.3	57.7	97.2	72.4
Prompt 2 w/ MLM	53.8	95.1	68.7	58.6	94.9	72.4
Ensemble	61.3	85.3	71.3	67.8	89.0	77.0

Table 3: Comparison of varying approaches

and only if all the three individual models predict a positive label, otherwise, it will be labeled as negative. As Table 3 shows, the ensembling approach further provides significant improvement over each individual method.

4.3 Impact of MLM pre-training

Figure 3 shows the effect of MLM pre-training strategy based on different percentages of masked keywords for each supervised approach. As we can see, it provides improvement to all the three supervised approaches, demonstrating that it can encourage the language model encoder to better capture contextual features and generalize to other clues and indicators that are not from the rules.

4.4 Limitation of the Rule-based Methods

It's not surprising that rule-based methods show very low recall on the manually annotated test dataset as (1) there are a large number of resistance mechanisms while most of them also facilitate the biological processes that are not related to antibiotics. For instance, the *tolC-hlyD-hlyB* and related systems are nearly ubiquitous type 1 secretion systems that facilitate secretion of a very broad range of substrates, such as *virulence factors*, *bacteriocins*; (2) there are a lot of other terms that could have been included in the rules but their mere mentions are not enough to indicate the antibiotic resistance of genes. For instance, *plasmids* frequently but not always carry antibiotic resistance genes, and similar terms also include *transposon*, *integron*, *genomic island* and so on.

5 Remaining Challenges

To understand the remaining challenges of SciARG, we randomly sample 100 prediction errors of the ensembling approach from the development set, and summarize the following three key remaining challenges.

Challenge 1: Lack of Domain Specific Knowledge The ARG statement prediction requires extensive domain specific knowledge to help the models better understand the text and disambiguate the meanings. For example, in the following statement:

"*Minimal inhibitory concentrations (MICs) of ciprofloxacin, ofloxacin, ceftazidime, cefsulodin, and aztreonam, but not amikacin, were increased at least 4-fold by ectopically expressed CpxR in PA14 and PA14DeltacpxR strains (Table 2) in a manner dependent on MexA, but not MuxA . In this case, ectopically expressed CpxR failed to increase the MICs of the tested antibiotics in a mexA null-mutant PA14DeltamexA strain . In contrast, the MIC increases caused by the ectopically expressed CpxR were not altered in a muxA null-mutant PA14DeltamuxA strain (Table 2)*"

The term "*ectopic expression*" either refers to "*heterologous expression*" or "*a specific experimental condition*", which lead to distinct predictions. The model cannot correctly interpret the meaning and thus made a wrong prediction.

Challenge 2: Limited Contextual Cues about the Target Gene For many statements, the context is not enough to confidently infer whether the target gene is an ARG or not. For instance, in the following statement:

"*MphI shares high sequence identity (94%) to*

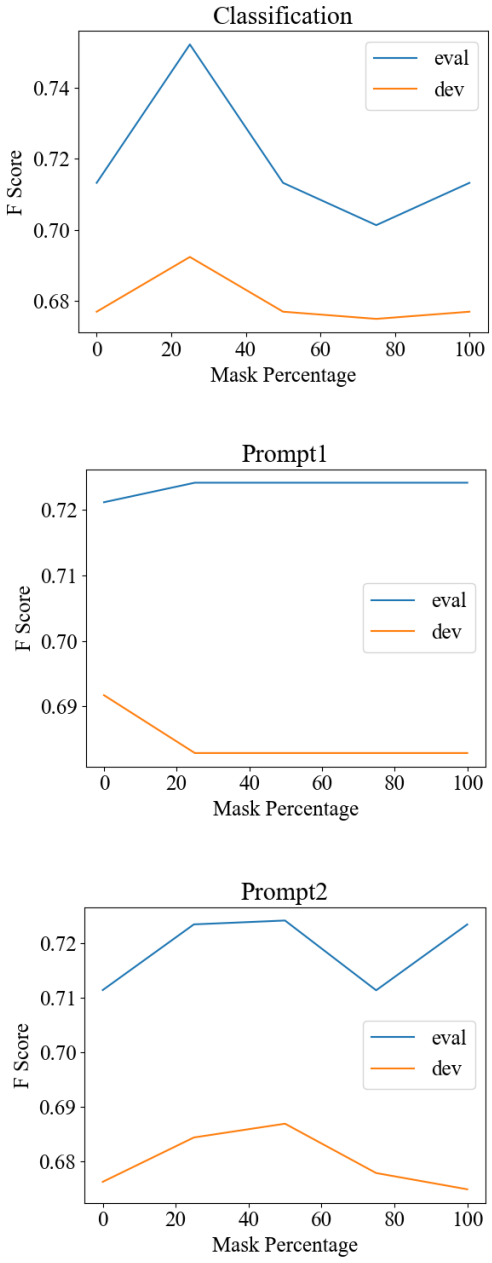

Figure 3: Impact of MLM pre-training with different percentages of masked keywords.

homologs found in related surface Paenibacillus sp . , indicating the functional divergence of MphI is not recent . The Bacillus cereus group have two genetically and functionally distinct Mph enzymes; one that modifies a broad range of macrolides and another that cannot modify macrolides with 16-membered rings"

Our ensembling approach mistakenly predicts it as positive while the source article concluded that *mphI does NOT encode an ARG*. The description of "*MphI modifies macrolides*" does not necessarily imply that *it neutralizes or inactivates macrolides, a class of antibiotics, thereby causing resistance*. This is a special case that will happen occasionally - where the statements are characterizing an ARG homolog, but not an ARG.

Challenge 3: Noisy and Insufficient Training Data The training is created based on a set of rules, which leads to two major problems: (1) It introduces noise since the rules are not 100% precise. As Table 3 shows, the precision ranges from 20.6% for Rule-3 to 72.2% for Rule-2; (2) The positive ARG statements covered in the training data is not diverse enough as they are constrained by the 4 rules. Though the MLM strategy helps the model generalize to more broad contextual features, it still suffers from the low recall. Many types of ARG statements in the development and test sets are not covered in the training set. For instance, for the following statement from the development set:

"*The nature of the activating ligand for VanSA has not been identified, therefore this work sought to identify and characterise ligand(s) for VanSA. In vitro approaches were used to screen the structural and activity effects of a range of potential ligands with purified VanSA protein. Of the screened ligands (glycopeptide antibiotics vancomycin and teicoplanin, and peptidoglycan components N-acetylmuramic acid, D-Ala-D-Ala and Ala-D-y-Glu-Lys-D-Ala-D-Ala) only glycopeptide antibiotics vancomycin and teicoplanin were found to bind VanSA with different affinities (vancomycin 70 muM; teicoplanin 30 and 170 muM), and were proposed to bind via exposed aromatic residues tryptophan and tyrosine.*"

The reason that "*VanSA*" is labeled as an ARG is that "*the ligand interaction of VanSA with glycopeptide antibiotics (GPA).*" implies that VanSA is an ARG since it inactivates the antibiotic vancomycin by binding to it, while such rules are not covered in the current training dataset.

6 Related Work

Machine Learning for Antibiotic Resistance Prediction Traditional antimicrobial susceptibility testing (AST) is time-consuming, low throughput and viable only for cultivable bacteria, thus rapid and accurate AMR diagnostic methods are very urgently needed. Recent years, machine learning based methods have been widely explored as clinical decision support tools for the prediction of antimicrobial resistance (AMR) (Feretzakis et al., 2021, 2020; Martínez-Agüero et al., 2019; Oon-

sivilai et al., 2018). Ren et al. (2021) compared four different machine learning methods (Random Forests, Logistic Regression, Support Vector Machines and Convolutional Neural Networks) for the prediction of AMR based on different encodings and whole-genome sequencing data without previously known knowledge. Deep learning algorithms have also shown significant potential for predicting new antibiotic drugs, AMR genes and AMR peptides (Kumaresan et al., 2018; Stokes et al., 2020; Veltri et al., 2018). However, these studies focused on genome variants (such as single-nucleotide polymorphisms, SNPs) or other features only related to resistant genes identified in previous studies or resistant databases, while in this work, we focus on curating antimicrobial susceptibility data by leveraging computational approaches and large-scale scientific papers. In addition, we approach the ARG curation as a entity classification task instead of recognition as genes are easily detected based on the existing knowledge bases and it's more challenging to infer the antibiotic resistance the genes based on the context. The curated ARG database can provide clinicians useful information regarding possible antibiotic resistance and aid clinicians in selecting appropriate empirical antibiotic therapy by taking into consideration the local antimicrobial resistance ecosystem.

Prompt Learning Prompt learning aims to learn a task-specific prompt while keeping most of the parameters of the model freezed (Li and Liang, 2021; Hambardzumyan et al., 2021; Brown et al., 2020). It has shown competitive performance in a wide variety of applications in natural language processing (Raffel et al., 2020; Brown et al., 2020; Shin et al., 2020b; Jiang et al., 2020; Lester et al., 2021; Schick and Schütze, 2021b). Previous work either use a manual (Petroni et al., 2019; Brown et al., 2020; Schick and Schütze, 2021a) or automated approach (Jiang et al., 2020; Yuan et al., 2021; Li and Liang, 2021) to create prompts. In this work, we mainly explore two manually defined prompts for ARG statement extraction task. The reason of applying prompt learning for ARG statement classification lies that though the training dataset size is not small, the clues of indicating antibiotic resistance covered in the training set is limited to the manually defined rules, thus applying prompt learning can to some extent leverage the knowledge, especially antibiotic resistance related knowledge captured by the large-scale language models during pre-training. Based on the experimental results of the ensembling approach, we see that although the prompt learning based approaches do not perform as well as the supervised classification based method, they are still complimentary to each other.

7 Conclusion and Future Work

In this work, we present the first computational framework that aims to automatically curate ARGs by extracting ARG-related assertative statements from scientific papers in PubMed. To support the research, we introduce SCIARG, a dataset that contains 2,000 manually annotated statements as the test set and 12,516 silver-standard training statements that are automatically created from scientific papers by a set of rules. We also present extensive empirical results by comparing various state-of-the-art neural architectures based on pre-trained language models for statement classification, and demonstrate that there is still a large room to improve based on the current highest 77% F-score on SCIARG.

Considering the remaining challenges that we have discussed, there are multiple future directions: (1) developing more advanced frameworks that incorporate domain-specific knowledge from external resources or knowledge bases to better interpret the statements; (2) learning contextual features of target genes from more broad context, such as the paragraph, chapter or the whole document; (3) leveraging self-training or co-training framework to take advantage of the large-scale unlabeled corpus from PubMed to enrich the training samples.

Acknowledgements

We thank the anonymous reviewers for their valuable time and constructive comments, and the helpful discussions with Huimin Han.

References

Brian P Alcock, Amogelang R Raphenya, Tammy TY Lau, Kara K Tsang, Mégane Bouchard, Arman Edalatmand, William Huynh, Anna-Lisa V Nguyen, Annie A Cheng, Sihan Liu, et al. 2020. Card 2020: antibiotic resistome surveillance with the comprehensive antibiotic resistance database. *Nucleic acids research*, 48(D1):D517–D525.

GA Arango-Argoty, GKP Guron, Emily Garner, Maria V Riquelme, Lenwood S Heath, Amy Pruden, Peter J Vikesland, and Liqing Zhang. 2020.

Argminer: a web platform for the crowdsourcing-based curation of antibiotic resistance genes. *Bioinformatics*, 36(9):2966–2973.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Georgios Feretzakis, Evangelos Loupelis, Aikaterini Sakagianni, Dimitris Kalles, Maria Martsoukou, Malvina Lada, Nikoletta Skarmoutsou, Constantinos Christopoulos, Konstantinos Valakis, Aikaterini Velentza, Stavroula Petropoulou, Sophia Michelidou, and Konstantinos Alexiou. 2020. Using machine learning techniques to aid empirical antibiotic therapy decisions in the intensive care unit of a general hospital in greece. *Antibiotics (Basel)*, 9(2):50.

Georgios Feretzakis, Aikaterini Sakagianni, Evangelos Loupelis, Dimitris Kalles, Nikoletta Skarmoutsou, Maria Martsoukou, Constantinos Christopoulos, Malvina Lada, Stavroula Petropoulou, Aikaterini Velentza, Sophia Michelidou, Rea Chatzikyriakou, and Evangelos Dimitrellos. 2021. Machine learning for antibiotic resistance prediction: A prototype using off-the-shelf techniques and entry-level data to guide empiric antimicrobial therapy. *Healthc. Inform. Res.*, 27(3):214–221.

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. 2021. WARP: Word-level Adversarial ReProgramming. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4921–4933, Online. Association for Computational Linguistics.

Zhengbao Jiang, Frank F. Xu, J. Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Karthikeyan K Kumarasamy, Mark A Toleman, Timothy R Walsh, Jay Bagaria, Fafhana Butt, Ravikumar Balakrishnan, Uma Chaudhary, Michel Doumith, Christian G Giske, Seema Irfan, et al. 2010. Emergence of a new antibiotic resistance mechanism in india, pakistan, and the uk: a molecular, biological, and epidemiological study. *The Lancet infectious diseases*, 10(9):597–602.

Deepak Kumaresan, Jason Stephenson, Andrew C Doxey, Hina Bandukwala, Elliot Brooks, Alexandra Hillebrand-Voiculescu, Andrew S Whiteley, and J Colin Murrell. 2018. Aerobic proteobacterial methylotrophs in movile cave: genomic and metagenomic analyses. *Microbiome*, 6(1):1.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *EMNLP*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, abs/2101.00190.

Bo Liu and Mihai Pop. 2009. Ardb—antibiotic resistance genes database. *Nucleic acids research*, 37(suppl_1):D443–D447.

Sergio Martínez-Agüero, Inmaculada Mora-Jiménez, Jon Lérida-García, Joaquín Álvarez-Rodríguez, and Cristina Soguero-Ruiz. 2019. Machine learning techniques to identify antimicrobial resistance in the intensive care unit. *Entropy (Basel)*, 21(6):603.

Mathupanee Oonsivilai, Yin Mo, Nantasit Luangasanatip, Yoel Lubell, Thyl Miliya, Pisey Tan, Lorn Loeuk, Paul Turner, and Ben S Cooper. 2018. Using machine learning to guide targeted and locally-tailored empiric antibiotic prescribing in a children's hospital in cambodia. *Wellcome Open Res.*, 3:131.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*.

Yunxiao Ren, Trinad Chakraborty, Swapnil Doijad, Linda Falgenhauer, Jane Falgenhauer, Alexander Goesmann, Anne-Christin Hauschild, Oliver Schwengers, and Dominik Heider. 2021. Prediction of antimicrobial resistance based on whole-genome sequencing and machine learning. *Bioinformatics*, 38(2):325–334.

Joy Scaria, Umamaheswaran Chandramouli, and Sanjay Kumar Verma. 2005. Antibiotic resistance genes online (argo): A database on vancomycin and βlactam resistance genes. *Bioinformation*, 1(1):5.

Timo Schick and Hinrich Schütze. 2021a. Few-shot text generation with pattern-exploiting training.

Timo Schick and Hinrich Schütze. 2021b. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 2339–2352.

Hoo-Chang Shin, Yang Zhang, Evelina Bakhturina, Raul Puri, Mostofa Patwary, Mohammad Shoeybi, and Raghav Mani. 2020a. Biomegatron: Larger biomedical domain language model. *arXiv preprint arXiv:2010.06060*.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020b. AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Jonathan M Stokes, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackermann, Victoria M Tran, Anush Chiappino-Pepe, Ahmed H Badran, Ian W Andrews, Emma J Chory, George M Church, Eric D Brown, Tommi S Jaakkola, Regina Barzilay, and James J Collins. 2020. A deep learning approach to antibiotic discovery. *Cell*, 180(4):688–702.e13.

Daniel Veltri, Uday Kamath, and Amarda Shehu. 2018. Deep learning improves antimicrobial peptide recognition. *Bioinformatics*, 34(16):2740–2747.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. BARTScore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems*.