Automated Screening of Antibacterial Nanoparticle Literature: Dataset Curation and Model Evaluation

M. Alperen Öztürk[1,*], Şaziye Betül Özateş[1,*], Sophia Bahar Root[2], Angela Violi[5,7], Nicholas A. Kotov[3,4,5], J. Scott VanEpps[3,6], Emine Sümeyra Turalı Emre[2]

[1]Institute for Data Science and Artificial Intelligence, [2]Biomedical Engineering Institute, Boğaziçi University, Türkiye

[3]Biointerfaces Institute, [4]Center of Complex Particle Systems, [5]Chemical Engineering Department,

[6]Department of Emergency Medicine, [7]Department of Mechanical Engineering, University of Michigan, USA

[*]These authors contributed equally to this work.

Correspondence: {eminesumeyra.turali,saziye.ozates}@bogazici.edu.tr

Abstract

Antimicrobial resistance is a growing global health threat, driving interest in nanoparticle-based alternatives to conventional antibiotics. Inorganic nanoparticles (NPs) with intrinsic antibacterial properties show significant promise; however, efficiently identifying relevant studies from the rapidly expanding literature remains a major challenge. This step is crucial for enabling computational approaches that aim to model and predict NP efficacy based on physicochemical and structural features. In this study, we explore the effectiveness of traditional machine learning and deep learning methods in classifying scientific abstracts in the domain of NP-based antimicrobial research. We introduce the "**A**ntibacterial **I**norganic **NA**noparticles Dataset" **AINA** of 7,910 articles, curated to distinguish intrinsic antibacterial NPs from studies focusing on drug carriers or surface-bound applications. Our comparative evaluation shows that a fine-tuned BioBERT classifier achieved the highest macro F1 (0.82), while a lightweight SVM model with TF-IDF features remained competitive (0.78), highlighting their utility in low-resource settings. AINA enables reproducible, large-scale identification of intrinsically bactericidal inorganic NPs. By reducing noise from non-intrinsic contexts, this work provides a foundation for mechanism-aware screening, database construction, and predictive modeling in antimicrobial NP research.

1 Introduction

Infectious diseases remain one of the leading causes of global mortality (Ng and Shamsi, 2022; Al-Awsi et al., 2023), and the increase in antimicrobial resistance poses an urgent public health crisis due to its association with widespread bacterial infections and treatment failure (Gómez-Núñez et al., 2020). Inorganic nanoparticles (NPs)—especially those composed of metals and metal oxides—have emerged as promising alternatives to conventional antibiotics. These materials exhibit tunable physicochemical properties and the potential to act through multiple antimicrobial mechanisms, reducing the likelihood of resistance development (Franco et al., 2022; Correa et al., 2020; Al-Awsi et al., 2023).

Many studies attribute NP efficacy to canonical mechanisms such as reactive oxygen species (ROS) generation, ion release, and membrane disruption (Hajipour et al., 2012; Kamat and Kumari, 2023). However, these explanations often fall short of capturing the full scope of NP-bio interactions. For example, zinc oxide (ZnO) NPs have demonstrated selective inhibition of methicillin-resistant *Staphylococcus aureus* (MRSA) through transcriptomic interference, affecting bacterial metabolism and gene expression (Kadiyala et al., 2018). Shape-dependent enzyme inhibition has also been reported, suggesting that shape and surface characteristics play a critical role in NP behavior (Cha et al., 2015). Other studies show that graphene quantum dots can inhibit biofilms by disrupting amyloid protein assembly, indicating physical interference mechanisms (Wang et al., 2019). These findings underscore the complexity of NP's mechanism of action and the need for systematic approaches to identify and analyze them.

Recent successes in AI-driven drug design (Jiménez-Luna et al., 2021) and protein structure prediction (Jumper et al., 2021) demonstrate the potential of machine learning (ML) for biological discovery. However, despite rapid growth in the nanomedicine literature, efforts to apply computational tools and ML to NP modeling are hindered by the lack of high-quality, domain-specific corpora to support natural language processing (NLP) applications. Biomedical NLP models such as BioBERT have shown strong performance in clinical domains (Lee et al., 2020), but general-purpose approaches struggle with NP literature due to in-

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 454–465
March 24-29, 2026

consistent terminology and unstructured reporting (Koutsomitropoulos and Andriopoulos, 2022).

Biomedical text classification has become essential for navigating the complexity and scale of scientific publications. Techniques using contextual embeddings, domain ontologies, and hybrid deep learning architectures have improved performance across tasks such as disease tagging and clinical note categorization (Ibrahim et al., 2021; Parwez et al., 2023). Unlike general biomedical classification tasks, identifying studies that investigate NPs with intrinsic antibacterial activity requires a nuanced interpretation of mechanism-specific language and exclusion of drug-conjugated or surface-bound systems. Addressing this gap is critical for downstream applications such as: (i) building structured datasets that capture the compositions, mechanisms, and outcomes of NPs; (ii) supporting predictive modeling based on physicochemical descriptors; (iii) facilitating systematic reviews in nanomedicine; and (iv) guiding the rational design of new antimicrobial materials informed by published literature.

To fill this gap, we develop a supervised text classification framework that performs binary classification of scientific articles based on their titles and abstracts, labeling them as either included or excluded for further investigation. Articles labeled as included are expected to discuss NPs with intrinsic antibacterial activity, making them suitable for downstream literature mining and data-driven research in antimicrobial nanomedicine.

Our contributions are as follows:

- **The AINA (Antibacterial Inorganic NAnoparticles) Dataset:** A domain-specific annotated dataset curated from NP literature with a mechanism-aware rubric designed to distinguish intrinsic antimicrobial agents from drug carriers or surface-bound applications. This dataset will not only support classification tasks but also lay the groundwork for future extraction of mechanistic insights and predictive modeling.

- **Comparative evaluation of different classifier architectures:** We systematically compare the performance of traditional ML models with advanced deep learning architectures to assess their effectiveness in identifying the articles. The findings highlight the relative strengths of each model type in handling domain-specific textual data.

- **Public resources:** We make the AINA dataset and the best performing models publicly available for academic use at `https://github.com/alperenmo/AINA_Dataset` to accelerate literature mining, ML-based materials discovery, and systematic reviews in antimicrobial nanomedicine.

This work bridges biomedical NLP and nanoscience by enabling high-fidelity classification of complex, interdisciplinary literature—laying the foundation for data-driven discovery and computational modeling of next-generation antibacterial NPs.

2 Related Work

Kebede et al. (2023) conducted a comprehensive evaluation of supervised ML algorithms to predict the inclusion of relevant articles for full-text review within the context of a systematic review. Utilizing a dataset comprising 16,430 manually screened records, the authors examined various text representation techniques, including term frequency–inverse document frequency (TF-IDF) and n-gram models. They implemented multiple classifiers on these features, such as Naïve Bayes, SVM, regularized Logistic Regression, Neural Networks, Random Forest, LogitBoost, and XGBoost. Their findings indicate that the combination of TF-IDF features with a linear SVM yielded the highest performance.

Wilson et al. (2023) conducted a comparative analysis of screening methods to identify *in vitro* biomedical studies that typically assess cellular responses to stress or the effects of interventions aimed at enhancing survival. They compared human screening of titles/abstracts and full texts with automated screening using text mining and ML. Text mining relied on RegEx patterns, while an SVM classifier was trained using trigram bag-of-words representations. Their findings showed that full-text human screening achieved the highest sensitivity and specificity, whereas title and abstract screening showed that automated text mining outperformed human screening. These results support using automated title and abstract screening as an effective first step in systematic reviews, aligning with our approach to automatically identifying articles discussing NPs as antimicrobial agents.

Pilz et al. (2024) presented a semi-automated title and abstract screening pipeline for systematic reviews using NLP and ML methods. Their goal was to develop a simple, transparent, customizable, and generalizable framework for article inclusion/exclusion across different research domains. They demonstrated the framework on two datasets and evaluated models including SVM, Logistic Regression, Random Forest, LightGBM, BERT, and GPT-2. However, despite using models such as BERT and GPT-2, their approach remains general-purpose and does not assess performance in highly specialized literature, such as NP research.

While these studies demonstrate the utility of text mining and classification in systematic reviews, they are largely focused on clinical or *in vitro* biomedical domains. To date, no prior work has focused on the automated identification of literature within the field of nanomedicine, particularly studies investigating the intrinsic antibacterial activity of inorganic NPs. This domain presents unique classification challenges due to inconsistent terminology, interdisciplinary reporting styles, and the technical complexity of materials science publications. To our knowledge, this is the first study to automate the identification of publications on the intrinsic antibacterial activity of NPs using a domain-specific rubric and curated dataset.

3 Methodology

In this section, we describe the AINA dataset, including the curation process, the preprocessing steps applied to the text data, dataset statistics, and the range of ML and deep learning models used for binary classification of the articles.

3.1 Dataset

The dataset was curated to identify research articles where the NP itself—rather than a drug-loaded system or surface-bound material—acts as the primary antibacterial agent. Over 300,000 nanoparticle-related publications were initially retrieved from MEDLINE (via OVID) using Boolean queries. A filtered set of approximately 10,000 papers was screened, and 8,000 were manually annotated using a structured rubric (see Appendix A). Inclusion required synthesis, characterization, and antibacterial testing of inorganic NP. Papers were excluded if they involved photocatalysis, nanofibers, liposomes, engineered peptides, or particles exceeding 1 μm in any dimension. Annotators reviewed abstracts and consulted full texts when needed. Although the original plan was for all papers to be reviewed by two independent annotators, the process proved significantly time-consuming. To balance coverage and quality, the final dataset consists of:

- **Training set:** 7,336 papers annotated by a single reviewer, including:
  - 2,233 included papers (nanoparticles with intrinsic antibacterial activity). These papers are given the label **include**.
  - 5,103 excluded papers (e.g., drug carriers, surface-bound materials). These papers are given the label **exclude**.
- **Test set:** 574 papers annotated by two independent reviewers who reached consensus on the labels. Of these, 204 were labeled as **include**, while the remaining 370 papers were labeled as **exclude**. **Table 1** summarizes the statistics about the dataset.

Class	Training Set	Test Set	Entire Dataset
Exclude	5,103	370	5,473
Include	2,233	204	2,437
Total	7,336	574	7,910

Table 1: Summary of class distribution and training-test split in the dataset.

The labels are imbalanced, with the **exclude** class outnumbering **include** by a ratio of 2.25:1, making it an imbalanced dataset for classification.

Dataset Preprocessing: Each entry in the final dataset used for model training corresponds to a scientific abstract and a ground-truth label indicating whether the study met the predefined inclusion criteria for NP-based antimicrobial research. Prior to training, all abstracts were preprocessed to ensure textual consistency. This included removing non-alphanumeric characters, normalizing whitespace, and converting to lowercase. These steps helped reduce textual noise and improve model robustness and interpretability.

Our dataset[1] is fully anonymized and publicly available for research purposes under https://github.com/alperenmo/AINA_Dataset.

3.2 Models

To classify scientific articles as **include** or **exclude** based on whether they include NPs with intrinsic

[1]To comply with licensing restrictions, the released dataset includes the annotated labels and the corresponding DOI numbers of the articles, not the full abstract texts. However, we provide an accompanying script that allows users to automatically retrieve the titles and abstracts of these articles via publicly accessible APIs where permitted.

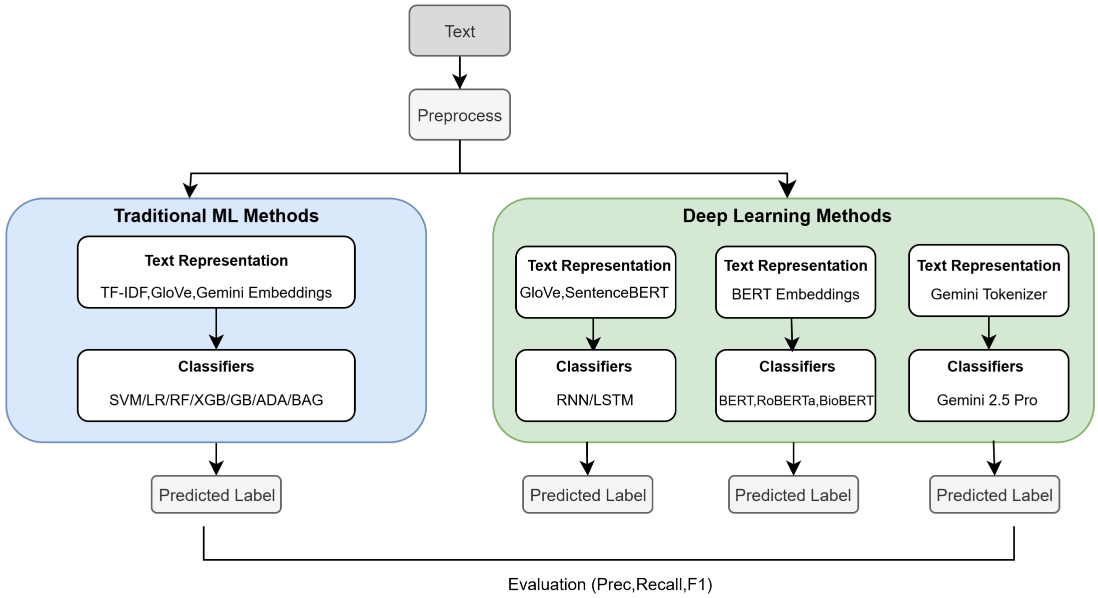

Figure 1: Flowchart of the classification methodology used for the various models employed in this study.

antibacterial activity, we used a range of models. Following Li et al. (2022), we grouped the models into two primary categories: traditional ML methods and deep learning-based methods **(Figure 1)**.

The motivation to evaluate models under these categories arose from the necessity to work with a smaller subset of data during the initial stages of our study. The limited dataset, which was heavily skewed toward the **exclude** class, led to poor performance by deep-learning-based methods in the classification task. To improve performance, we experimented with traditional classifiers, which are reported to perform more effectively on smaller datasets in several studies (Gasparetto et al., 2022; Grinsztajn et al., 2022). In our case, the performance of traditional classifiers trained on the smaller dataset was notably higher than that of the deep-learning models. Consequently, we compared the performance of both traditional and deep learning-based methods in our text classification task.

3.2.1 Traditional Machine Learning Models

The traditional ML classifiers we employed include Linear Support Vector Machines (SVMs), Logistic Regression (LR), Random Forest (RF), Gradient Boosting (GB), XGBoost (XGB), AdaBoost (ADA), and Bagging (BAG).[2] The ensemble methods combine decision trees based on the CART algorithm (Breiman et al., 2017) to achieve better classification performance. They aggregate the results of multiple classifiers using the bootstrap aggregation method (Li et al., 2022). These models were trained separately using TF-IDF and GloVe (Pennington et al., 2014) based embeddings.

We performed hyperparameter tuning on the maximum number of TF-IDF features. For this purpose, we used a 90/10 train-development split on the training dataset, and found that using 1,000 features yielded the best overall performance across models. Additionally, we set the number of estimators to 100 for the ensemble models (i.e., AdaBoost, Random Forest, and Bagging).

3.2.2 Deep Learning Models

As deep learning-based models, we experimented with recurrent architectures, pre-trained language models (PLMs), and large language models (LLMs). The use of recurrent architectures stems from their effectiveness at capturing sequential dependencies in text. We utilized RNN and LSTM architectures with both sentence-level and word-level representations. For sentence-level representations, we relied on SBERT (Reimers and Gurevych, 2019), while word-level embeddings were generated using GloVe.

PLMs have demonstrated state-of-the-art performance across a wide range of NLP tasks, including text classification. In this study, we employed three BERT-based models that span from general-

[2]We used implementations of Linear SVM, Logistic Regression, Random Forest, Gradient Boosting, AdaBoost, and Bagging from scikit-learn, which is licensed under the BSD 3-Clause License, and XGBoost from the official XGBoost library, licensed under the Apache License 2.0.

purpose to domain-specific usage: BERT (Devlin et al., 2019), the original baseline model; RoBERTa (Liu et al., 2019), which refines BERT through optimized training strategies and larger-scale data; and BioBERT (Lee et al., 2020), which is pre-trained on extensive biomedical corpora to address domain-specific challenges. To adapt these models for our classification task, we appended a classification head to the [CLS] token representation, enabling effective downstream fine-tuning.

In addition to recurrent architectures and pre-trained language models, we incorporated Gemini 2.5 Pro (Comanici et al., 2025), a large language model with strong reasoning capabilities, using both zero-shot prompting and few-shot prompting via a Retrieval-Augmented Generation (RAG) pipeline. The choice of Gemini 2.5 Pro arises from its large token context window (approximately 1 million tokens) and its high performance in complex reasoning tasks.

The architectural and training details of the utilized deep models are summarized below:

RNN and LSTM classifiers: These models were trained separately using SBERT and GloVe embeddings. Training was conducted for up to 50 epochs with a batch size of 8. Early stopping was employed with a patience of 5 epochs. Both RNN- and LSTM-based classifiers consist of a single-layer RNN/LSTM with a hidden size of 128 and a sigmoid output layer, and use the Adam optimizer with a learning rate of 1e-3.

BERT-based classifiers: All BERT-based models were fine-tuned on the labeled training dataset for a maximum of 20 epochs with a batch size of 8. Early stopping was applied based on validation loss, with training terminated if no improvement was observed for three consecutive epochs. We utilized base models for BERT, RoBERTa, and BioBERT from Hugging Face for the classification task. Tokenization was handled using their respective tokenizers, with all text truncated or padded to a maximum sequence length of 512 tokens. For training, we adopted the Hugging Face `Trainer` API. The AdamW optimizer was used with a learning rate of 5e-5, and a warm-up phase was applied over the first 500 steps. A batch size of 8 was used for both training and evaluation, and the best-performing model was retained. The training and testing were conducted on a single NVIDIA A100 GPU with 32 GB of RAM and took approximately 1.5 hours for BERT and RoBERTa, and about 1 hour for BioBERT.

Gemini 2.5 Pro (zero-shot): Google's state-of-the-art LLM Gemini 2.5 Pro was accessed via the Google API. The model was prompted to output results in JSON format. A fallback mechanism automatically defaulted to the **exclude** label if the model failed to produce a valid response containing either **include** or **exclude**. No fallback cases were observed. The process took approximately 2.5 hours. The prompt template used for this task is provided in Appendix B.1.

Gemini 2.5 Pro (few-shot learning): Gemini 2.5 Pro was also evaluated in a few-shot setting to provide the model with in-context information. The few-shot examples were selected using the RAG method. RAG facilitates example selection by first indexing a set of documents and then retrieving the most relevant ones. During the indexing step, examples are transformed into embeddings and stored as vectors. In the retrieval step, the k most similar documents in the vector database to a given query are retrieved. We used `all-mpnet-base-v2` from the Hugging Face Sentence-Transformers library—a sentence- and paragraph-level transformer—to convert the training data into embeddings. FAISS (Douze et al., 2025) was employed to construct the vector database from the embedded training data and to retrieve the most semantically similar documents. For each query, we retrieved the five most similar documents after transforming it into a vector using the same transformer. These documents were then provided as few-shot examples in the prompt. The same fallback mechanism was implemented, but no fallback cases were observed. The process took approximately 3.5 hours. The prompt template used for this task is provided in Appendix B.2.

Figure 1 shows the general flow diagram of the classification methodology.

4 Experiments

4.1 Experimental Settings

For the experiments, we conducted five independent trials for each model to ensure reliability and robustness, as well as to mitigate the effects of randomness in training. The performance metrics, consisting of precision, recall, and F1 score, are reported as the average values across these five runs. We used macro-averaged metrics to better reflect per-class performance and to mitigate majority-class bias in imbalanced datasets, as opposed to weighted averages.

	Category	Model	Representation	Precision	Recall	F1
Traditional	Single	**SVM**	GloVe	0.76	0.66	0.67
			TF-IDF	<u>0.81</u>	<u>0.76</u>	<u>0.78</u>
		LR	GloVe	0.77	0.66	0.67
			TF-IDF	0.81	0.74	0.75
	Ensemble	**RF**	GloVe	0.75	0.60	0.59
			TF-IDF	0.80	0.70	0.71
		XGB	GloVe	0.72	0.64	0.65
			TF-IDF	0.80	0.72	0.74
		GB	GloVe	0.74	0.63	0.63
			TF-IDF	0.80	0.74	0.75
		ADA	GloVe	0.69	0.63	0.64
			TF-IDF	0.79	0.72	0.74
		BAG	GloVe	0.71	0.61	0.61
			TF-IDF	0.78	0.72	0.73
Deep	Recurrent NN	**RNN**	GloVe	0.75	0.66	0.67
			SBERT	0.79	0.74	0.75
		LSTM	GloVe	0.75	0.67	0.67
			SBERT	0.78	0.74	0.75
	PLM	**BERT**	Contextual	<u>0.81</u>	<u>0.78</u>	<u>0.79</u>
		BioBERT	Contextual	**0.84**	**0.81**	**0.82**
		RoBERTa	Contextual	0.80	0.78	0.78
	LLM	**Gemini 2.5 Pro (Zero-Shot)**	Contextual	0.70	0.65	0.55
		Gemini 2.5 Pro (Few-Shot)	Contextual	0.76	0.76	0.76

Table 2: Performance of different classifier models on the test set of the AINA dataset. The best scores are in bold, and the second-best scores are underlined.

For tokenization, we relied on the tokenizers of the embedding methods, except for GloVe embeddings. The TF-IDF vectorizer employs a regex-based tokenization strategy. SBERT uses its own subword tokenizer based on WordPiece, while GloVe-based representations typically require manual word-level tokenization, which we handled by utilizing NLTK's word tokenizer.

4.2 Experimental Results

Best-performing model. The experimental results in **Table 2** provide a comparative analysis of various models using different text representations. Overall, the best-performing model across all evaluation metrics is BioBERT, achieving a macro-averaged precision of 0.84, recall of 0.81, and F1 score of 0.82. This strong performance is expected, as BioBERT builds upon the BERT architecture, which effectively captures contextual information through its transformer-based design. In addition, BioBERT's pre-training on large-scale biomedical corpora further enhances its ability to model domain-specific language, leading to superior results compared with more general-purpose models. The baseline BERT model achieved the second-best performance, with an F1 score of 0.79—approximately three percentage points lower than BioBERT.

The effect of text representation methods. Among the traditional classifiers, TF-IDF consistently yielded better results than GloVe across all models. This is likely because TF-IDF captures the importance of words relative to the dataset by emphasizing distinctive terms that help separate classes. In contrast, GloVe embeddings are pre-trained on general corpora and may not capture domain-specific nuances or discriminative terminology relevant to antibacterial nanoparticle research.

SVM with TF-IDF achieved the highest F1 score (0.78) among traditional models. LR and GB showed comparable performance, each with an F1 score of 0.75 using TF-IDF.

Despite being a simpler model, SVM achieved competitive results with transformer-based models. To further investigate, we assessed whether improved text representations could enhance SVM performance by integrating Gemini Embeddings (Lee et al., 2025), a recently introduced state-of-the-art deep text representation method. Gemini Embeddings were used as input features for SVM to test whether better contextualization improves classification. **Table 3** presents the results.

SVM	Class Label	Prec.	Recall	F1
TF-IDF	include	0.81	0.60	0.69
	exclude	0.81	0.92	0.86
	macro avg	0.81	0.76	0.78
GloVe	include	0.79	0.37	0.51
	exclude	0.73	0.95	0.83
	macro avg	0.76	0.66	0.75
Gemini Embd.	include	0.72	0.74	0.73
	exclude	0.85	0.84	0.85
	macro avg	0.78	0.79	0.79

Table 3: Class-wise and overall performance of the SVM classifier with different text representations.

We observed that Gemini Embeddings improved SVM's performance for the **include** class, but overall gains were modest. SVM with Gemini Embeddings achieved higher recall but slightly lower precision compared with TF-IDF. The contextualized Gemini Embeddings resulted in only a 1% increase in F1 compared with SVM using TF-IDF.

Performance of ensemble models. Among ensemble methods, Gradient Boosting with TF-IDF achieved the highest F1 score (0.75), followed by XGBoost and AdaBoost (0.74 each). Bagging and Random Forest with TF-IDF yielded the lowest performance among the ensemble group. Since SVM with TF-IDF attained a higher F1 score of 0.78, ensemble methods using TF-IDF representations appear less suitable for this task.

Performance of recurrent architectures. The two recurrent neural network architectures, RNN and LSTM, exhibited relatively poor performance with GloVe embeddings. Combined with the other results, this finding suggests that pre-trained GloVe embeddings are not well suited to this domain-specific binary classification task. When SBERT was used for input representation, the F1 scores of both RNN and LSTM improved to 0.75, underscoring the value of context-aware embeddings. Nevertheless, their performance remained below that of the simpler SVM model with TF-IDF input.

Zero- and few-shot prompting with Gemini 2.5 Pro. The final rows in Table 2 show the classification performance of Gemini 2.5 Pro under zero- and few-shot prompting. As expected, Gemini 2.5 Pro in the zero-shot setting (classifying without example demonstrations) performed substantially worse than its few-shot counterpart. The few-shot version outperformed zero-shot across all metrics: by 6% in precision, 11% in recall, and 21% in F1 score. In fact, Gemini 2.5 Pro in the zero-shot setting was the weakest model overall (F1 = 0.55).

The considerable gap between settings can be attributed to the limited contextual information in abstracts. In the zero-shot mode, the model likely relied on surface-level cues, such as domain-specific keywords (*inhibits*, *kills*, nanoparticle types, or microorganism names), leading to false positives. Conversely, the few-shot setting, enhanced with RAG, provided concrete labeled examples, allowing stricter adherence to inclusion criteria rather than keyword reliance. This interpretation aligns with the class-based precision and recall scores shown in **Table 4**.

Model	Label	Prec.	Recall	F1
Gemini 2.5 Pro (Zero-Shot)	include	0.45	0.98	0.61
	exclude	0.96	0.33	0.49
	macro avg	0.70	0.65	0.55
Gemini 2.5 Pro (Few-Shot)	include	0.68	0.70	0.69
	exclude	0.83	0.82	0.83
	macro avg	0.76	0.76	0.76

Table 4: Class-wise and overall performance of Gemini 2.5 Pro under zero- and few-shot settings.

Class-wise performance of the best models from each category: **Table 5** presents the class-wise performance of the best-performing models in each category: SVM with TF-IDF, GB with TF-IDF, BioBERT, and Gemini 2.5 Pro (few-shot). All models exhibited stronger performance on the **exclude** class, likely due to class imbalance or more distinct features in that category.

Model	Label	Prec.	Recall	F1
SVM (TF-IDF)	include	0.81	0.60	0.69
	exclude	0.81	0.92	0.86
GB (TF-IDF)	include	0.82	0.54	0.65
	exclude	0.79	0.94	0.85
BioBERT	include	0.82	0.73	0.77
	exclude	0.86	0.91	0.88
Gemini 2.5 Pro (Few-Shot)	include	0.68	0.70	0.69
	exclude	0.83	0.82	0.83

Table 5: Class-wise performance of the best classifier models in each category on the test set.

For the **include** class, precision exceeded recall in SVM, GB, and BioBERT, whereas the reverse was observed for **exclude**. This suggests that when models classify an article as related to antibacterial nanoparticle activity, they are usually correct

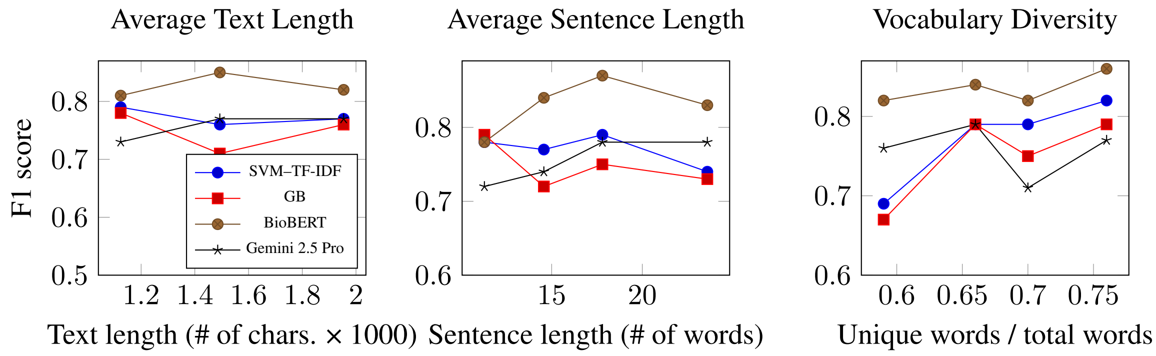

Figure 2: Variation of F1 scores across average text length, average sentence length, and vocabulary diversity for the best models in each category: SVM with TF-IDF, GB with TF-IDF, BioBERT, and Gemini 2.5 Pro (few-shot).

but may miss relevant studies due to lower recall. BioBERT achieved an F1 score of 0.77 for the **include** class, surpassing SVM by 8% points. This highlights BioBERT's superior capability to detect relevant literature on antibacterial nanoparticles, which is essential for advancing this research field.

In the few-shot setting, Gemini 2.5 Pro showed balanced precision and recall across both classes, indicating stable performance. However, BioBERT still outperformed it on all metrics, reaffirming the effectiveness of fine-tuned transformer models for domain-specific classification.

4.3 Discussion

Figure 2 compares the variation in F1 score for the best-performing models from each category—SVM with TF-IDF,[3] GB with TF-IDF, BioBERT, and Gemini 2.5 Pro (few-shot)—across key textual properties. Because titles and abstracts often omit mechanistic detail and share overlapping terminology across subfields, model behaviour is tightly linked to the degree of contextual signal available in each abstract. The following trends thus reflect both model capacity and the linguistic and structural properties of the input texts.

When examining average input text length, BioBERT and Gemini 2.5 Pro display an upward trend in performance, improving as text length increases. This suggests that contextual models benefit from longer inputs containing richer semantic cues and interrelated entities. In contrast, SVM and GB—both relying on sparse lexical representations such as TF-IDF—perform best on shorter texts where salient keywords are concentrated and shallow features suffice for discrimination.

As average sentence length grows, SVM and GB performance declines, whereas BioBERT and Gemini 2.5 Pro improve. Longer sentences increase syntactic complexity and introduce multiple clauses, diluting the signal captured by TF-IDF and other bag-of-words features. Contextual models, by contrast, can exploit extended dependencies through transformer attention mechanisms. This observation helps explain why distinguishing intrinsically antibacterial nanoparticles from carriers, coatings, or trigger-dependent systems is challenging at the abstract level: key semantic cues are subtle, distributed, and often implied rather than explicit.

Vocabulary diversity exhibits a positive correlation with model performance for all models except Gemini 2.5 Pro. A higher ratio of unique to total words enhances TF-IDF representations by amplifying distinctive terms and improving class separability. For BioBERT, diverse lexical input enriches contextual embeddings by providing a broader semantic landscape. Gemini 2.5 Pro in the few-shot setting, however, depends heavily on the in-context examples within the prompt; its reliance on demonstration-driven reasoning limits the benefit of lexical variety.

Our qualitative error analysis revealed two dominant patterns. False positives were typically triggered by surface cues—organism names or phrases such as *bactericidal*, *nanostructured*, or *biofilm inhibition*—in abstracts describing systems that were not intrinsically antibacterial (e.g., nanoparticles serving as antibiotic carriers, surface coatings, or photoactivated agents). False negatives often occurred when intrinsic activity was obscured by contextual framing (e.g., environmental microbiol-

[3] Although SVM with Gemini Embeddings achieved a 1% improvement in macro-averaged F1 score over SVM with TF-IDF, the latter demonstrated superior performance (3% higher) in accurately classifying abstracts belonging to the **include** class. Considering its lightweight architecture and reliability for this critical class, we selected SVM with TF-IDF for further analysis.

ogy studies), discipline-specific terminology (e.g., dental ceramics), or misleading usage (e.g., "gentamicin" mentioned as a precursor rather than a payload). These patterns emphasize the need for stricter rubric rules distinguishing the nanoparticle's mechanistic role (active agent vs. carrier or coating) and for more balanced sampling from underrepresented application domains.

The AINA rubric's explicit separation of intrinsically bactericidal nanoparticles from carriers, coatings, and trigger-dependent systems reduces label noise and yields more interpretable comparisons across studies. Overall, our best-performing model, BioBERT, achieves an F1 score of 0.82 on this challenging binary classification task. This result is encouraging given the high semantic overlap between **include** and **exclude** abstracts. Excluded papers are not irrelevant: many describe nanoparticles as drug delivery systems or structural supports, including peptide- and lipid-based nanomaterials. Surface-bound particles, in particular, inhibit bacterial adhesion rather than directly killing bacteria—mechanistically distinct yet linguistically similar phenomena. Consequently, keywords such as "nanoparticle," "antibacterial," or "quantum dots" appear in both categories, confounding models reliant on lexical features.

BioBERT's advantage likely stems from its capacity to capture these nuanced contextual differences—discerning whether the nanoparticle functions as the antimicrobial agent itself or merely as a carrier or barrier. This underscores the necessity of deep contextual understanding for biomedical abstract classification, where keyword- or frequency-based methods alone are insufficient. By clarifying semantic scope at the abstract level, AINA enables cleaner literature mapping and supports reliable downstream analyses focused on nanoparticles whose antibacterial action originates intrinsically from the material itself. This foundation facilitates structured data extraction, mechanism-aware summarisation, and rational screening of nanoparticle chemistries and morphologies with true antimicrobial potential.

5 Conclusion

We framed the identification of articles on antibacterial nanoparticle use as a binary classification task and introduced AINA, a novel manually curated dataset. Evaluating both traditional ML and deep learning models, we found that the BioBERT-based classifier achieved the best performance in identifying relevant articles.

What distinguishes this work is its focus on a narrowly defined yet highly impactful domain: nanoparticles with intrinsic antibacterial activity. The domain's inconsistent terminology, unstructured abstracts, and mechanistic complexity pose challenges beyond general biomedical classification. By developing a specialized classifier, this study enables automated literature triaging, supports the creation of structured databases of nanoparticle properties and efficacy, and lays the groundwork for computational discovery of new antimicrobial materials.

Future work may expand the AINA dataset, refine classification methods, and integrate the approach into automated review pipelines. Beyond binary classification, the framework introduced here holds potential for diverse downstream applications. It can support the construction of structured datasets that capture nanoparticle compositions, mechanisms of action, and experimental outcomes—providing a foundation for data-driven modeling. Such models could be used to predict antibacterial efficacy based on physicochemical and morphological descriptors, accelerating hypothesis generation and materials screening. Furthermore, this approach can facilitate systematic reviews and evidence synthesis in nanomedicine by enabling rapid identification of relevant studies. Embedding literature insights into computational pipelines will guide the rational design of next-generation antimicrobial materials informed by prior research.

6 Limitations

Despite the careful curation and novel focus of the dataset, several limitations should be noted. First, a substantial portion of the training data was annotated by a single annotator, increasing the potential for label noise, especially in ambiguous cases. Additionally, the binary classification framework oversimplifies a complex inclusion spectrum, which may obscure nuanced distinctions in nanoparticle mechanisms. Reliance solely on titles and abstracts, without access to full-text details, further restricts the model's ability to make informed classifications, particularly in cases where methodological details are crucial. Lastly, class imbalance and the relatively small dataset size hinder the performance of deep learning models—despite BioBERT's strong results—and underscore the continued relevance of traditional models in low-resource settings.

Acknowledgements

This work was supported by TÜBİTAK BİLGEM under the project *Development of a deep learning–based data augmentation tool to predict the efficacy of antibacterial nanoparticles used in nanomedicine*. Computational resources were provided by the National Center for High Performance Computing of Turkey (UHeM) at Istanbul Technical University (ITU) under project allocation #1022642025. Portions of the dataset were developed under the BlueSky Initiative at the University of Michigan College of Engineering, with project leadership by Angela Violi, Nicholas A. Kotov, and J. Scott VanEpps, and supervised by Emine Sümeyra Turali-Emre and J. Scott VanEpps.

References

GRL Al-Awsi, AA Alameri, AMB Al-Dhalimy, GA Gabr, and E Kianfar. 2023. Application of nano-antibiotics in the diagnosis and treatment of infectious diseases. *Brazilian Journal of Biology*, 84:e264946.

Leo Breiman, Jerome Friedman, Richard A Olshen, and Charles J Stone. 2017. *Classification and regression trees*. Routledge.

Sang-Ho Cha, Jin Hong, Matt McGuffie, Bongjun Yeom, J Scott VanEpps, and Nicholas A Kotov. 2015. Shape-dependent biomimetic inhibition of enzyme by nanoparticles and their antibacterial activity. *ACS Nano*, 9(9):9097–9105.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3416 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

Matías Guerrero Correa, Fernanda B. Martínez, Cristian Patiño Vidal, Camilo Streitt, Juan Escrig, and Carol Lopez de Dicastillo. 2020. Antimicrobial metal-based nanoparticles: a review on their synthesis, types and antimicrobial action. *Beilstein Journal of Nanotechnology*, 11:1450–1469.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2025. The FAISS library. *Preprint*, arXiv:2401.08281.

Domenico Franco, Giovanna Calabrese, Salvatore Pietro Paolo Guglielmino, and Sabrina Conoci. 2022. Metal-based nanoparticles: Antibacterial mechanisms and biomedical application. *Microorganisms*, 10(9).

Andrea Gasparetto, Matteo Marcuzzo, Alessandro Zangari, and Andrea Albarelli. 2022. A survey on text classification algorithms: From text to predictions. *Information*, 13(2).

Mario F Gómez-Núñez, Mariel Castillo-López, Fernando Sevilla-Castillo, Oscar J Roque-Reyes, Fernanda Romero-Lechuga, Diana I Medina-Santos, Ricardo Martínez-Daniel, and Alberto N Peón. 2020. Nanoparticle-based devices in the control of antibiotic resistant bacteria. *Frontiers in microbiology*, 11:563821.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. 2022. Why do tree-based models still outperform deep learning on tabular data? *Preprint*, arXiv:2207.08815.

Mohammad J. Hajipour, Katharina M. Fromm, Ali Akbar Ashkarran, Dorleta Jimenez de Aberasturi, Idoia Ruiz de Larramendi, Teofilo Rojo, Vahid Serpooshan, Wolfgang J. Parak, and Morteza Mahmoudi. 2012. Antibacterial properties of nanoparticles. *Trends in Biotechnology*, 30(10):499–511.

Muhammad Ali Ibrahim, Muhammad Usman Ghani Khan, Faiza Mehmood, Muhammad Nabeel Asim, and Waqar Mahmood. 2021. GHS-NET a generic hybridized shallow neural network for multi-label biomedical text classification. *Journal of Biomedical Informatics*, 116:103699.

José Jiménez-Luna, Francesca Grisoni, Nils Weskamp, and Gisbert Schneider. 2021. Artificial intelligence in drug discovery: recent advances and future perspectives. *Expert opinion on drug discovery*, 16(9):949–959.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, and 1 others. 2021. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589.

Usha Kadiyala, Emine Sumeyra Turali-Emre, Joong Hwan Bahng, Nicholas A Kotov, and J Scott

Vanepps. 2018. Unexpected insights into antibacterial activity of zinc oxide nanoparticles against methicillin resistant Staphylococcus aureus (MRSA). *Nanoscale*, 10(10):4927–4939.

Siya Kamat and Madhuree Kumari. 2023. Emergence of microbial resistance against nanoparticles: Mechanisms and strategies. *Frontiers in Microbiology*, 14:1102615.

Mihiretu M Kebede, Charlotte Le Cornet, and Renée Turzanski Fortner. 2023. In-depth evaluation of machine learning methods for semi-automating article screening in a systematic review of mechanistic literature. *Research Synthesis Methods*, 14(2):156–172.

Dimitrios A Koutsomitropoulos and Andreas D Andriopoulos. 2022. Thesaurus-based word embeddings for automated biomedical literature classification. *Neural Computing and Applications*, 34(2):937–950.

Jinhyuk Lee, Feiyang Chen, Sahil Dua, Daniel Cer, Madhuri Shanbhogue, Iftekhar Naim, Gustavo Hernández Ábrego, Zhe Li, Kaifeng Chen, Henrique Schechter Vera, Xiaoqi Ren, Shanfeng Zhang, Daniel Salz, Michael Boratko, Jay Han, Blair Chen, Shuo Huang, Vikram Rao, Paul Suganthan, and 28 others. 2025. Gemini embedding: Generalizable embeddings from gemini. *Preprint*, arXiv:2503.07891.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S Yu, and Lifang He. 2022. A survey on text classification: From traditional to deep learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(2):1–41.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *Preprint*, arXiv:1907.11692.

Ida M. J. Ng and Suhaili Shamsi. 2022. Graphene oxide (GO): A promising nanomaterial against infectious diseases caused by multidrug-resistant bacteria. *International Journal of Molecular Sciences*, 23(16).

Md Aslam Parwez, Mohd Fazil, Muhammad Arif, Md Tabrez Nafis, and Md Rabiul Auwul. 2023. Biomedical text classification using augmented word representation based on distributional and relational contexts. *Computational Intelligence and Neuroscience*, 2023(1):2989791.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Maximilian Pilz, Samuel Zimmermann, Juliane Friedrichs, Enrica Wördehoff, Ulrich Ronellenfitsch, Meinhard Kieser, and Johannes A Vey. 2024. Semi-automated title-abstract screening using natural language processing and machine learning. *Systematic Reviews*, 13(1):274.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Yichun Wang, Usha Kadiyala, Zhibei Qu, Paolo Elvati, Christopher Altheim, Nicholas A Kotov, Angela Violi, and J Scott VanEpps. 2019. Anti-biofilm activity of graphene quantum dots via self-assembly with bacterial amyloid proteins. *ACS Nano*, 13(4):4278–4289.

Emma Wilson, Florenz Cruz, Duncan Maclean, Joly Ghanawi, Sarah K McCann, Paul M Brennan, Jing Liao, Emily S Sena, and Malcolm Macleod. 2023. Screening for in vitro systematic reviews: a comparison of screening methods and training of a machine learning classifier. *Clinical Science*, 137(2):181–193.

A Prepared Rubric for the Manual Annotation of Scientific Articles

Question Set 1: If all answers are yes, then: Include

- Does the article describe the synthesis/manufacture of a nanoparticle (NP)?
- Does the article characterize the synthesized NP?
- Does the article evaluate the antimicrobial properties of the NP?
- Does the article report any antimicrobial effect?

Question Set 2: If any answer is yes, then: Exclude

- Is the NP a nanofiber, a film, or a particle with any dimension greater than 1 micron?
- Is the NP used in combination with known antimicrobials such as antibiotics, nisin, curcumin, cyclodextrin, or chitosan?
- Is the NP an engineered peptide (e.g., an antimicrobial peptide)?
- Does the article describe an antimicrobial mechanism that involves photocatalysis?
- Is the material described in the article a liposome?

B Prompts Used with Gemini 2.5 Pro

B.1 Zero-Shot Prompt:

```
"""
You are an expert in materials science, nanotechnology, and microbiology.

Task: Decide whether the following abstract is about antimicrobial
properties of nanoparticles.

Rules:
1)If yes -> respond strictly as JSON:{{'answer':'include'}}
2)If no -> responds strictly as JSON:{{'answer':'exclude'}}
Abstract:
{query}       # A query is an abstract to be labelled
Answer:
"""
```

B.2 Few-Shot Prompt:

```
"""
You are an expert in materials science, nanotechnology, and microbiology.

Task: Decide whether the following abstract is about antimicrobial
properties of nanoparticles.

Rules:
1)If yes -> respond strictly as JSON:{{'answer':'include'}}
2)If no -> responds strictly as JSON:{{'answer':'exclude'}}

Here are some examples:
{examples_text}     # k examples retrieved using RAG; k=5

Now classify this abstract:
{query}             # A query is an abstract to be labelled
Answer:
"""
```