CLASSIFICATION AND PREDICTION OF ANTIMICROBIAL PEPTIDES USING N-GRAM REPRESENTATION AND MACHINE LEARNINIG

by
Manal Othman
A Dissertation
Submitted to the
Graduate Faculty
of
George Mason University
in Partial Fulfillment of
The Requirements for the Degree
of
Doctor of Philosophy
Bioinformatics and Computational Biology

Committee:

IVaisman ________________ Dr. Iosif Vaisman, Committee Chair

________________ Dr. Saleet Jafri, Committee Member

M. Emelianenko ________________ Dr. Maria Emelianenko, Committee Member

IVaisman ________________ Dr. Iosif Vaisman, Director, School of Systems Biology

Donna M. Fox ________________ Dr. Donna Fox, Associate Dean, Office of Student Affairs & Special Programs, College of Science

________________ Dr. Fernando Miralles-Wilhelm, Dean, College of Science

Date: 07/ 30/2020 ________________

Summer Semester 2020
George Mason University
Fairfax, VA

Classification and Prediction of Antimicrobial Peptides Using N-gram Representation and Machine Learning

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at George Mason University

By

Manal Othman
Master of Science
Nova Southeastern University, 2014
Bachelor of Science
King Abdulaziz University, 2010

Director: Iosif Vaisman, Professor
Bioinformatics and Computational Biology

Summer Semester 2020
George Mason University
Manassas, VA

Dedication

To almighty Allah,
To the soul of my lovely father
To my lovely mother,
To my lovely husband,
To my lovely son and daughter,
To my lovely friends
To my lovely country,
I dedicate this work.

Acknowledgments

I want to extend special gratitude and appreciation to my esteemed professor, Iosif Vaisman, who devoted his knowledge, time, efforts, and patience to guiding me through this journey. In addition, I would like to thank my committee members, Dr. Saleet Jafri and Dr. Maria Emelianenko for their guidance, hard work, and support.

I want to thank my class-mate Krista Jastrzembski for sharing the negative dataset.

I want to thank Bader Alharbi and Fahad Almuqhim for helping me in creating the program.

I am also indebted to Prince Norah Bint Abdulrahman University, Riyadh, Saudi Arabia, and the government of the Kingdom of Saudi Arabia for providing me the scholarship to pursue the Ph.D. certificate of Bioinformatics and Computational Biology.

Thanks, must also go to George Mason University, School of Systems Biology for allowing me the precious opportunity to be one of their students.

I am also grateful to Dr. Dmitri Klimov, the program director for his unceasing support, which has ensured my smooth progress through the program.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ABPs	Antibacterial Peptides
ACC	Accuracy
AdBo	Adaboost
AFPs	Antifungal Peptides
AMPs	Antimicrobial Peptides
AMSDb	Antimicrobial Sequences Database
ANN	Artificial Neural Networks
APD	Antimicrobial Peptide Database
APPs	Antiparasitic Peptides
auROC	Area Under the ROC Curve
AVPs	Antiviral Peptides
BER	Balanced Error Rate
BLOSUM	Blocks Substitution Matrix
CAMP	Collection of Anti-Microbial Peptides
CDC	Centers for Disease Control
DNA	Deoxyribonucleic Acid
EFC	Evolutionary Feature Construction Algorithm
FCBF	Fast Correlation-Based Filter Selection
FN	False Negatives
FP	False Positives
GUI	Graphical User Interfaces
HDG	Hydrophobic Dominated Grouping
HIV	Human Immunodeficiency Virus
J48	Decision Tree
KRR	Kernel Ridge Regression
LZ	Lempel-Ziv
MCC	Mathew's Correlation Coefficient
MICs	Minimum Inhibitory Concentrations
MJ	Miyazawa–Jernigan Interaction
ML	Machine Learning
MLP	Multilayer Perceptron
MRSA	Methicillin-Resistant Staphylococcus Aureus
NB	Naïve Bayes
Non-AMPs	Non-Antimicrobial Peptides

PDG	Polar Dominated Grouping
PPV	Positive Predictive Value
PR	Protease
PRC	Precision Recall Curve
PSSM	Position-Specific Scoring Matrix
QSAR	Quantitative Structure-Activity Relationships
RA	Reduced Alphabet
RF	Random Forest
RNA	Ribonucleic Acid
ROC	Receiver Operating Characteristic
RT	Reverse Transcriptase
SAPD	Synthetic Antibiotic Peptide Database
SCOP	Structural Classification of Proteins
SDG	Singlet Dominated Grouping
SVM	Support Vector Machines
TN	True Negatives
TP	True Negatives
WEKA	Waikato Environment for Knowledge Analysis

Abstract

CLASSIFICATION AND PREDICTION OF ANTIMICROBIAL PEPTIDES USING N-GRAM REPRESENTATION AND MACHINE LEARNINIG

Manal Othman, Ph.D.

George Mason University, 2020

Dissertation Director: Dr. Iosif Vaisman

Current antibiotic treatments for infectious diseases are rapidly losing effectiveness, as the organisms they target are developing drug resistance over time. In the United States alone antibiotic-resistant bacterial infections annually result in more than 35,000 deaths, and much higher morbidity rates A promising alternative to the current antibiotic treatments is antimicrobial peptides (AMPs), short strings of amino acid residues that are able to inhibit the propagation of pathogens. A problem of correctly identifying AMPs based on their sequence features remains a subject of active investigations. In this dissertation, we successfully explored many features of AMP sequences using reduced amino acid alphabets and machine learning algorithms. Sequence patterns and sequence composition were represented by vectors of N-gram frequencies, where N-grams are substrings of length N. Machine learning (ML) models were used to differentiate between AMPs and non-AMPs, and to classify AMPs based on target pathogen class. These models demonstrated performance comparable or exceeding many states of the art models based on more complex peptide descriptors. Peptide representation based on reduced alphabets and N-gram frequencies can be used for design of novel AMP for targeting specific pathogens, which may provide a potential pathway for alternatives to antibiotic treatments. This work opens opportunities for collaboration with the wet lab researchers who can test the designed AMPs in experimental setting. N-gram a new publicly available application created for the peptide representation using N-grams and reduced amino acid alphabets is available at http://www.binf.gmu.edu/mothman/N-gram-Classification-Application/

Chapter 1: Introduction

1.1 The Antibiotic Resistance Problem

Antibacterial resistance is one of the most serious public health threats which is increasing with the proliferation of drug-resistant bacteria. A related and even more serious problem of multidrug-resistant pathogens is linked to microorganisms which developed resistance to different types of antibiotics. The fast appearance of resistance to antibiotics is occurring worldwide, jeopardizing the therapeutic ability of antibiotics that have transformed modern medicine and protected millions of lives (Ventola, 2015).

The antibiotics age started in 1928 by Alexander Fleming's discovery of penicillin. . Penicillin was successfully used to control the bacterial infections among the soldiers during the World War II (Ventola, 2015). However, soon thereafter penicillin resistance became a huge medical problem. Lately, other front-line drugs have begun to be ineffective against bacteria such as methicillin-resistant staphylococcus aureus (MRSA) and carbapenem-resistant enterobacteriaceae (CRE).

Many common medical procedures such as cancer chemotherapy and different types of surgery become significantly riskier without effective antimicrobial medications. According to the US Centers for Disease Control (CDC), the mortality rates due to multidrug-resistant bacterial infections are growing. Each year, more than 63,000 patients in the US die from hospital-acquired bacterial infections, including 35,000 deaths from antibiotic-resistant infections. More than 2.8 million people in the U.S. are diagnosed with antibiotic-resistant infections every year (CDC, 2019.; Magana et al., 2020).

The antibiotic resistance crisis has been exacerbated by the misuse and overuse of antibiotics, as well as by lack of new drug development by the pharmaceutical industry (Aminov, 2010). Due to this problem, only a few drugs show the effectiveness of treating some of the opportunistic infections. However, some of these drugs have the disadvantage of toxicity, such as amphotericin B, that could limit patients from receiving other treatments with toxic medications (Amaral et al., 2012). Coordinated efforts to reinvigorate medical research, implement new strategies for drug discovery, and pursue steps to control this crisis are badly needed.

Nowadays, with the increase of bacterial resistance and spreading of infectious diseases becoming potential threats to humans, the discovery of more antimicrobial peptides (AMPs) or designing peptide from scratch (de novo) have emerged as a promising interest area in antibiotic research to overcome this crisis. For this reason, AMPs show the potential to be used as bactericidal and antifungal drugs and how they can be successfully used to fight multi-drug resistant bacteria (Magana et al., 2020).

The development of microbial drug resistance is challenging for the researchers who were working on the design of new anti-pathogenic drugs. This kind of drug is presented in almost all living organisms as a part of their innate non-specific immune system; AMPs are much valued as lead compounds for the invention of human therapeutics to stop the development of antibiotic drug resistance. As drugs, AMPs display unique characteristics like, low toxicity, high biological activity, and specificity, which makes them attractive therapeutic agents (Phoenix et al., 2013a). This dissertation supports these efforts by using computational study for classification and prediction of these naturally occurring AMPs. The work described in this thesis has a potential to assist in discovering novel treatments and to help in AMP design or modification in wet lab research against multi-drug-resistant bacteria (Magana et al., 2020).

1.2 The History of Antimicrobial Peptides

A peptide is a short chain of amino acids. There are 20 naturally occurring amino acids, and they can be combined into an enormous variety of different molecules. The amino acids are connected in a sequence by peptide bonds. Proteins are long molecules made up of multiple peptide subunits (polypeptides). Historically AMPs have been referred to as cationic host defense peptides, anionic antimicrobial peptides/proteins, cationic amphipathic peptides, cationic AMPs, host defense peptides, and α-helical AMP. Unlikely antibiotics that target particular cellular activities (e.g., synthesis of protein, cell wall, or DNA), the AMPs target is the lipopolysaccharide layer of the cell membrane, which is universal in the microorganisms. Having low anionic charge and a high level of cholesterol puts eukaryotic cells out of the target range of many AMPs (Bahar & Ren, 2013).

AMPs first became a research interest in the middle decades of the twentieth century. It starts with the examination of cecropins from magainins from frogs and moths.

At the end of the 1920s, the first AMP lysozyme was discovered by Alexander Fleming as mentioned above. In 1928, he discovered penicillin. Then, in 1939, Dubos had been identified the AMPs in prokaryotic cells. This extract was showed to protect mice from pneumonia infection. A year after, Hotchkiss and Dubos identified gramicidins that isolated from Bacillus Brevis and were seen to have activity against a wide range of gram-positive bacteria in vitro and in vivo (Bahar & Ren, 2013). Despite some reported toxicity related to an intraperitoneal application, gramicidins were later proved to use successfully to heal infected wounds on the guinea-pig skin, showing their therapeutic potential for clinical use. Gramicidins were the first AMPs to be commercially manufactured as antibiotics.

In 1940, along with Fleming, Ernst Chain and Howard Florey started a successful therapeutic use of penicillin, and for this discovery, these three men shared Nobel prize in medicine (Gaynes, 2017). In the following year, tyrocidine was discovered to be effective against both gram-positive and gram-negative bacteria and exhibit to be toxic to human blood cells. At the same time, purothionin was isolated from a plant Triticumaestivum and found effective against several pathogenic bacteria and fungi (Bahar & Ren, 2013).

The appearance of penicillin and streptomycin started the “Golden Age of Antibiotics”, which led to a quick loss of interest in the therapeutic of natural antibiotics as lysozyme and the consequence of this immune defense strategy (Gaynes, 2017). Though, in 1942, the antimicrobial substance had been found in wheat flour that was isolated from Triticum aestivum (wheat endosperm) and worked as a growth inhibitor of specific phytopathogens, such as Xanthomonas campestris and Pseudomonas solanacearum (Gaynes, 2017). By 1956, defensin appeared as first the AMP originated from an animal which isolated from rabbit leukocytes. After that, lactoferrin from cow milk, bombinin from epithelia, and AMPs in human leukocytes in their lysosomes were reported.

These types of natural peptides play vital roles in the modulation of both the innate and adaptive immune systems, and the ability to promote and suppress the inflammation response to microbial infection. A few years later, in the 1960s, the scientist starts realizing that the "Golden Age of Antibiotics" had finished and with the appearance of multidrug-resistant microbial pathogens, and an awakened interest in host defense molecules was prompted (Phoenix et al., 2013b). In the late 1980s, numerous researchers described several AMPs from leukocytes as (α-defensins) from humans and rabbits (Gaynes, 2017).

Conversely, after the 1980s, the number of AMPs has burgeoned to over 2000 with representatives in almost all eukaryotic organisms (Bahar & Ren, 2013). In previous days when the number of AMPs was limited, these AMPs were writing in review articles. With a fast increase in the number of such AMPs, it became unreasonable to continue to do them manually. As a result, several databases have been established to categorize these peptides. Recently, these peptides were sequenced, characterized, and renamed as the more familiar "cecropins", thus representing the first major α-helical AMPs to be reported.

Since the 1980s, computational quantitative structure-activity relationships (QSARs) models for peptides have been used as prediction and sequence optimization for some biological activities. By 1990s, the artificial neural networks (ANN) as machine learning (ML) methods substituted the traditional QSAR models. Nowadays, a computational design approach that is joining a sophisticated activity estimator with a stochastic optimization technique (Fjell et al., 2011). Evolutionary algorithms, evolution strategies, and genetic algorithms have been used to search for peptides with improved activity in silico through consecutive generations of mutations, deletions and sequence shuffling.

1.3 Antimicrobial Peptides (AMPs)

Many studies indicated that AMPs exist in nearly all multicellular organisms, see figure 1.1. These peptides have been known at most sites of the human body, usually exposed to microbes like the skin mucosae, and are produced by some blood cell types, involving platelets, neutrophils, and eosinophils.

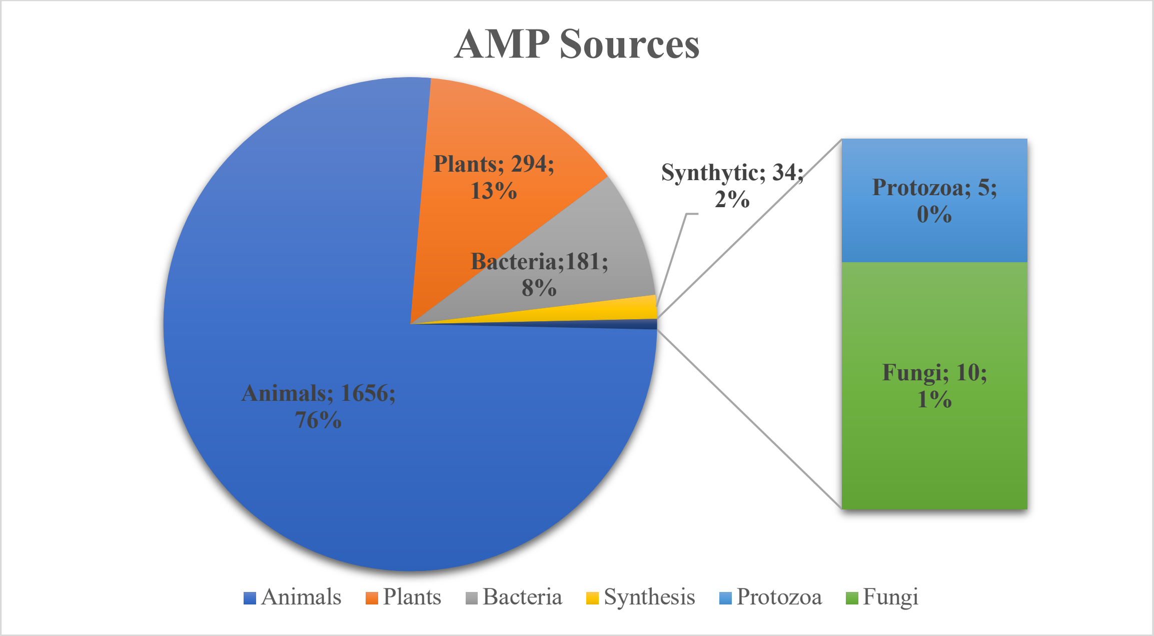


Figure 1.1: Sources of antimicrobial peptides (Wang, 2013)

AMPs demonstrate the potent killing of a wide range of microorganisms such as gram-positive, gram-negative bacteria, viruses, and fungi. These AMPs serve as a first-line defense system that is existing constitutively, but it may increase with injury and inflammation. Besides, AMPs act beyond the first-line defense system and have vital interactions with host adaptive immune responses and repair.

AMPs are a group of molecules that form a significant part of the innate immune system. AMPs are small oligopeptides, cationic or anionic, amphipathic molecules (hydrophobic and hydrophilic regions) of variable amino acid composition that enables the particle to be soluble in aqueous environments and enter lipid-rich membranes (Peters et al., 2010). The length of AMPs ranges from (five to over a hundred amino acid residues) and can be found through all classes of life, including bacteria, fungi, plants, vertebrates, and invertebrates. These peptides have a broad spectrum of targeted organisms, such as viruses and parasites. New synthetic peptides are created in silico using alternating variation selection operators and ML model that guides the design of sequence space that include residue sequences with a higher biological activity prediction (Gaynes, 2017).

1.4 Structure of AMPs:

Most AMPs can be characterized based on their secondary structures as one of the following four types:

1. β-sheet peptides are composed of at least two β-strands with disulfide bonds between these strands.
2. α-helix peptides: In α-helix structures, the distance between two adjacent amino acids is around 0.15 nm.
3. Extended peptides.
4. Loop peptides.

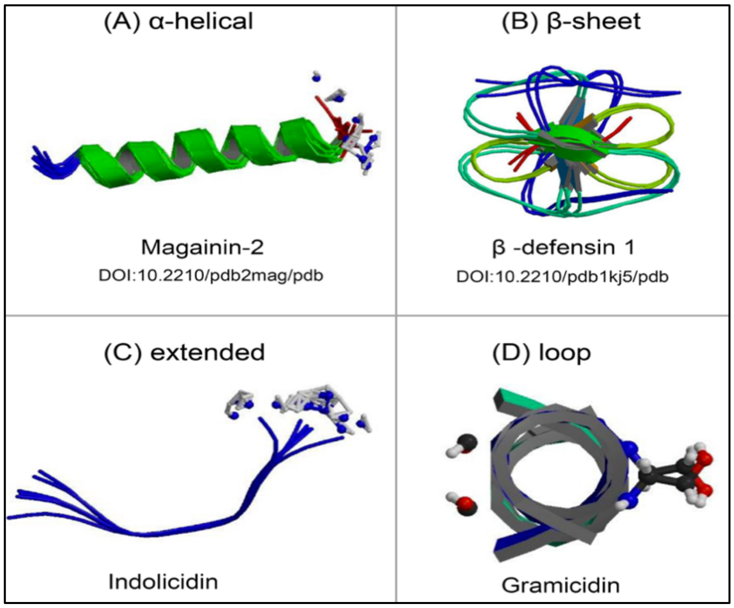


Figure 1.2 Example of structural differences of the four classes of antimicrobial peptides.
(A) α-helical peptides, (B) peptides composed of a series of β -sheets, (C) extended helices peptides, and (D) loops peptides (Peters, Shirtliff, & Jabra-Rizk, 2010).

α-helix and β-sheet structures are more common among these structural groups, while α-helical structure AMPs are the most studied to date. Most AMPs relate to one of the above four classes conversely some AMPs do not relate to any of these groups. Some AMPs include two different structural segments. Also, many peptides make their active structure only when they interact with the membranes of target cells. Similarly, they alter its conformation site during interaction with DNA (Bahar & Ren, 2013).

1.5 Major Categories of AMPs:

1. **Antibacterial Peptides (ABPs):** ABPs are the most studied AMPs to date, and most of them are cationic, which target bacterial cell membranes and cause a

breakdown of the lipid bilayer structure or by inhibiting some vital pathways inside the cell such as protein synthesis and DNA replication. Most of these AMPs are net positive charge (to enhance interaction with anionic lipids and other bacterial targets), hydrophobicity (for membrane insertion), and flexibility (to allow the peptide to switch from rest conformation to membrane-interacting conformation). There are many ABPs that do not fit into the simplified four structural classifications, as mentioned above. For instance, many bacterial peptides have two domains, one of which is α-helical, while the other has a β-structure, e.g., bovine neutrophil indolicidin (Jenssen et al., 2006). Some of the certain AMPs have been shown the ability to kill antibiotic-resistant bacteria. For instance, a methicillin resistant staphylococcus aureus (MRSA) strain was described to be sensitive to nisin (an AMP), while it is resistant to vancomycin (an antibiotic). Buforin, drosocin, pyrrhocoricin, and apidaecin are examples of ABPs (Bahar & Ren, 2013).

2. **Antiviral Peptides (AVPs):** AVPs from all four structural classes of the cationic host defense peptides have displayed the ability to inhibit viral infection. The spectra of viruses that are affected mainly enveloped DNA and RNA viruses, except for non-enveloped adenovirus, feline calici virus, and echovirus (Jenssen et al., 2006). AVPs are often highly cationic and amphiphilic. The antiviral activity works by neutralizing viruses by integrating into the viral envelope and cause membrane weakness, rendering the viruses unable to infect the host cell, and reduce the binding of viruses to the membrane. Also, some of the antiviral AMPs can stop viral particles from entering host cells by inhibiting specific receptors on

mammalian cells. Also, these AVPs able to cross the cell membrane, and locate in the cytoplasm and organelles and change the gene expression profile of the host cells to support the host defense system fighting viruses or block their gene expression. Example of AVPs: α-helical AVPs magainins, dermaseptin, and melittin, β-sheet peptides such as defensins, tachyplesin and protegrins, β-turn peptide as lactoferricin. However, it seems to be impossible to predict antiviral activity based on peptide's secondary structures (Bahar & Ren, 2013).

3. **Antifungal Peptides (AFPs):** To our knowledge of AFPs has accelerated in recent years, and the numbers of known AFPs increase. Peptides with mainly antifungal activity, tend to be rich in neutral and polar amino acids, such as many of those isolated from plants (Jenssen et al., 2006). AFPs can kill fungi by targeting either the cell walls or intracellular components. This binding ability helps AFPs to target fungal cells efficiently. Cell wall targeting-antifungal kill the target cells by interrupting the fungal membranes, by rising permeability of the plasma membrane, or by creating pores directly. These types of peptides have members from different sequence and structure classes such as α-helical (D-V13K and P18), β-sheet (defensins and a coleopteran), and extended (indolicin) (Bahar & Ren, 2013).
4. **Antiparasitic Peptides (APPs):** APPs are a smaller group compared to the other three previous classes. APPs kill cells by directly interacting with the cell membrane. The first antiprotozoan peptide is magainin, which can kill by swelling and eventual bursting of Paramecium caudatum. Recently, a synthetic peptide was developed for treating naturally acquired canine leishmaniasis, and it is shown to

be safe and effective (Jenssen et al., 2006). It looks likely that antiprotozoan activity it may be dependent on peptide motifs basically different from those that needed for bacterial, viral, and fungal activities (Bahar & Ren, 2013).

1.6 Mechanism of Antimicrobial Activity

AMPs are a common defense system of almost all forms of life. The importance of microbial activities in contributing to host defense may change between different sites within a distinct organism and among various kinds of organisms (Jenssen et al., 2006).

The molecular mechanism of membrane permeation different from peptide to other depending on several factors, such as the amino acid sequence, peptide concentration, and membrane lipid composition (Jenssen et al., 2006). Despite their vast diversity, most AMPs work straight against microbes through a mechanism involving:

1- Membrane integrity disruption by interaction with the negative charge of the cell membrane.
2- Pore formation, which permitting the efflux of vital nutrients and ions.
3- Inhibiting proteins, RNA, and DNA synthesis.
4- Inhibition of cell wall biosynthesis.
5- Interacting with specific intracellular targets (Bahar & Ren, 2013).

Naturally, an AMP is only effective against one type of microorganism, for example, bacteria or viruses. Some AMPs are recognized to have different mechanisms against various types of microorganisms. For instance, indolicidin can kill; 1) fungi by destructive of the cell membrane, 2) bacteria, e.g., E. coli by inhibiting DNA synthesis and, 3) displays anti-HIV activities by inhibiting HIV-integrase. In contrast, some AMPs have the same way of killing action for different types of cells. For instance, PMAP-23 can kill parasites and fungi by creating pores in their cell walls (Bahar & Ren, 2013).

Most membrane-active AMPs are amphipathic (cationic and hydrophobic faces) to ensure the electrostatic interaction with the negatively charged cell membrane, while the hydrophobic face helps insertion of AMP molecule into the cell membrane. These interactions with cell membrane typically depend on cationic state and hydrophobicity of the AMP (Bahar & Ren, 2013).

Recent studies were found that some of the AMPs start mechanism of killing by membrane permeabilization at concentrations lower than their minimum inhibitory concentrations (MICs), while others could start at concentrations higher than their MICs. However, some AMPs can kill their target cells without affecting membrane permeabilization by interacting with targets inside the cells. Also, some AMPs can also stop the proteases of microbes. Interestingly, there are some AMPs can only kill cells at specific growth stages, while others have multiple targets. Instead, the same AMP can trigger an autolysin protein inside the target cells, causing the autolysis of the cell (Bahar & Ren, 2013).

These features, merged with the broad spectrum of activity and the short interaction time required to stimulate killing, have led to the consideration of AMPs as exceptional candidates for development as novel therapeutic agents. Consequently, insights into the mechanisms used by AMPs will enable new methods to discover and develop pharmaceutic agents.

1.7 AMPs Database History

AMPs characteristics a remarkable diversity of structural motifs, resulting in a wide variety of the primary sequence, from the amino acid conformation to the total length. Because of this diversity, a complete dataset of active and inactive peptides is hard to obtain without initiating biases. For these points, in the last few years, different bioinformatics approaches were used to gather as much as possible data of natural and synthetic AMPs from literature. This method has the benefit to reduce the bias of current sequence selection and give detailed and uniform of peptide activity information. Likewise, complex prediction models frequently require many measured values of AMPs activity to fit the broad set of parameters. While the process of information collecting can be automated, because of the sensitivity and difficulty of the data process, manually collected datasets are more appreciated.

In 1998, Antimicrobial Sequences Database (AMSDb) appeared to be the first database of AMPs available online in an intensive manner, covering the sequences of a gene-encoded AMP and proteins from animal and plant sequences. The information format of this database is identical to the SWISS-Prot (UniProt), and it includes 895 antimicrobial peptides (Maccari et al., 2013).

Synthetic Antibiotic Peptide Database (SAPD) was developed in 2002. This SAPD is formed on two pre-existing computer databases for naturally occurring peptide antibiotics, the Peptaibol Database, and the AMSDb, which contains both biological and chemical information on all published synthetic antibiotic peptides (Wade & Englund, 2002).

Unfortunately, AMSDb is not updated, and the fast increase in natural AMPs makes it a challenging task to manage such data manually. As a consequence, three databases were released in 2004. The first one is ANTIMIC reported more than 1700 entries, while the last version of ANTIMIC called DAMPD contains 1232 entries (Maccari et al., 2013). The second database was Peptaibol database, which contains 307 peptides isolated from soil fungi (Wang, 2010). The third is APD. The first version of the APD reported 525 peptide entries. These peptides were manually collected from the literature with the assistance of public search engines such as Pub-Med, Swiss-Prot, and PDB. By 2009, the peptide number reached 1228 entries in the second version of the APD (APD2). As of Jan 7, 2017, the latest AMP database (APD3) contains 2767 AMPs and proteins, 98% of them are less than 100 amino acids and from living eukaryotic and prokaryotic organisms (Wang, 2016). These comprehensive APD databases provide helpful information on amino acidic frequency, the presence of conserved motifs, chemophysical properties, peptide discovery timeline, nomenclature, classification, glossary, calculation tools, and statistics (Maccari et al., 2013).

Consequently, several other databases for particular types of AMPs were constructed. Cybase is a database for cyclic polypeptides from animals, plants, and bacteria. PenBase is devoted to shrimp AMPs, defensins and bactibase knowledgebase were also founded. AMPer is an AMPs prediction tool based on the peptides collected from SwissProt and AMSDb. In 2008, a specialized database for recombinant AMPs (RAPD) was also created to document peptide expression, carrier, host, cleavage method and, by 2009, PhytAMP database was established that specialized for plant AMPs, which contains 271 peptides (Wang, 2010). In January 2010, collection of AMPs (CAMP) appeared. The CAMP database is the manually curated collection of AMPs, which includes further tools as a local BLAST query system and prediction tools that based on amino acid frequencies, Random Forest (RF) algorithm, biological activity against different strains and chemophysical and taxonomical characteristics (Maccari et al., 2013).

These databases can be used to enable effective search, prediction, and design of peptides with antimicrobial activities, chemotactic, immune modulation, or anti-oxidative properties. Also, it makes predictions based on the database-defined parameter space and offers a list of the sequences most similar to natural AMPs. These comprehensive AMP databases (as table 1.1 below) are a useful tool for both research and education (Wang, 2015).

Table 1.1: Summary of the major databases of AMPs (Wang, 2015)

Year	Database	Web site	Content
2002	AMSDb	http://www.bbcm.univ.trieste.it/~tossi/amsdb.html	Plant/ animal AMPs
2002	SAPD	http://oma.terkko.helsinki.fi:8080/~SAPD	Synthetic AMPs

2004	Peptaibol	http://www.cryst.bbk.ac.uk/peptaibol/home.shtml	Fungal AMPs
2004	APD	http://aps.unmc.edu/AP	Natural AMPs
2004	ANTIMIC	Not Active	Natural AMPs
2006	PenBase	http://penbase.immunaqua.com	Shrimp AMPs
2006	Cybase	http://research1t.imb.uq.edu.au/cybase	Cyclotides
2007	BACTIBASE	http://bactibase.pfba-lab-tun.org/main.php	Bacteriocins
2007	Defensins	http://defensins.bii.a-star.edu.sg	Defensins
2007	AMPer	http://marray.cmdr.ubc.ca/cgi-bin/amp.pl	Plant/ animal AMPs
2008	RAPD	http://faculty.ist.unomaha.edu/chen/rapd/index.php	Recombinant AMPs
2009	PhytAMP	http://phytamp.pfba-lab-tun.org/main.php	Plant AMPs
2010	CAMP	http://www.bicnirrh.res.in/antimicrobial	All AMPs

1.8 Role of Computation in AMP Classification and Predication

The field of bioinformatics has impressively strengthened the ability to understand biological procedures and their mechanisms. Bioinformatics develops and uses data, computational tool, and algorithms to conduct biological research. In this dissertation, bioinformatics approach has been utilized specifically to classify and predict antimicrobial peptides (AMPs).

While more and more pathogens continue to develop resistance toward previously effective antimicrobial drugs, it decreases the number of medical treatment options available, creating a potential to trigger a global health security emergency. New variations of antimicrobial-resistant infections are growing, especially in conditions with overused and misused antimicrobial drugs, poor sanitary conditions, unsatisfactory infection control or inappropriate food-handling. The failure of antimicrobial-resistant infections to counter to previously effective drug treatments leads to prolonged infection as well as a higher threat of death.

Furthermore, antibiotic resistance places a heavy load on the world economy. By 2050, due to antibiotic resistance, the world population is estimated to be between 11 million and 444 million lower than it could have been otherwise, and the economy will lose between $2.1 trillion and $124.5 trillion ("Antibiotic / Antimicrobial Resistance | CDC," n.d.). Research on substitutes for antibiotic drugs is still in the early stages, and only few of them approved for clinical use.

Nevertheless, AMPs have been identified as promising candidates to combat drug-resistant pathogens. Because microbes do not tend to modify their external membrane, it results in a decreased likelihood of AMPs targets developing resistance. Many of eukaryotic cells are not targeted by AMPs, due to the eukaryotic cells' high level of cholesterol and low anionic charge. AMPs are extremely efficient killers, as it takes only seconds for them to kill the certain microbe after initial contact with the cell membrane.

This study integrates alphabet reduction technique, N-gram analysis, and ML, three different approaches, to develop a computational model, which is able to accurately classify AMPs. This sequence-based method allowed to analyze how well N-gram frequencies can be used to train ML algorithms.

Current predictors of AMP use multiple sequence alignments, PSI-BLAST sequence profiles, distinctive residue compositions, or secondary structure analyses. These predictors require comparing and analyzing entire sequences and take longer time compared to N-grams, which decompose sequences into smaller parts, each of these parts can be readily analyzed quantitatively. Likewise, some of computational techniques used for predictions are generally “black-box” models like Artificial Neural Networks (ANN) and Support Vector Machines (SVM), and the features that these models utilize are not fully well-defined. To help with feature selection, N-gram frequency analysis can also be used to train decision trees, which can provide more vision into how the training dataset is actually used to create the decision-making process.

Our goals in this project, were to uncover specific patterns within the sequence of AMPs, and effectively classify between AMPs and Non-antimicrobial peptides (Non-AMPs) as well as the subclasses of AMPs. In addition, we aimed to help researcher to create new AMPs sequences based on the classification features discovered. These goals were accomplished using a novel method of analyzing the frequencies of N-grams combinations with ML algorithms. The frequencies of every N-gram were calculated, the complexity of data was reduced by using alphabet reduction to create clusters of specific amino acids with similar properties. This decreased the number of possible N-gram combinations and lowered the number of frequencies that were computed. The results from this thesis could be particularly interesting as applying the knowledge toward synthesizing pathogen-specific AMPs in the wet laboratory, focusing in a clearer direction when searching for replacements to antibiotic treatments.

1.9 Dissertation Structure

We were having introduced AMPs and their primary biological activity and how these AMP work above, we begin in chapter two by providing related background information on AMP and overview of the current computational AMP classification using alphabet reduction technique. While chapter three displayed our sequence-based methodology using N-gram, reduced alphabet, and machine learning classifiers. In addition to the datasets that were used repeatedly throughout this work. In chapter four, we performed different experiments using our novel methodology to classify AMPs, uncover specific arrangements and interesting features throughout AMPs sequences, and how these features affect the accuracy of the models. After showing relations to be beneficial, Chapter 5 leverages the power of our proposed algorithm by validating it using different evaluation metrics considered for model performance for prediction and classification of AMP. Chapter six details an application constructed to make this novel sequence-based method freely accessible to the AMP research community. This application contains many features that assist in model and dataset preparation to be ready for ML classifiers. This dissertation concludes with chapter seven, which presents a final overview and discussion of the work demonstrated and possible future research directions for this line of AMPs research.

Chapter 2: Background Information and Related Work

2.1 Literature Review of AMPs

Nowadays, most researchers in the bioinformatics field are focused on computational methods for screening and in silico modeling of novel AMPs, to accelerate the development of antimicrobial drug discovery and design (Hammami & Fliss, 2010). Many methods have been developed for predicting new AMPs with the potential therapeutic application. Some algorithms take benefit of data mining and high-throughput screening techniques to scan peptide and protein sequences (Lata et al., 2010). Most of the researches have been used QSAR descriptors together with ANN (Cherkasov & Artem, 2005; Fjell et al., 2009), SVM (Taboureau et al., 2006), or linear discriminant (Wang et al., 2011). In addition to K-mer, genetic programming, and sequence alignment for prediction of peptide's activity.

A study was done in 2009 (Fjell et al., 2009), demonstrated that the QSAR descriptors and ML techniques have successfully utilized in silico screening for potent antibiotic peptides. On the basis of over 1400 random peptides and an independent test set of 100,000 virtual peptides, the artificial network models predict and rank the relative activities of novel AMPs with 94% accuracy in identifying highly active peptides. In another study, Cherkasov & Artem showed the QSAR descriptors had also successfully recognized the antibacterial activities with up to 93% accuracy of correct separation of compounds with- and without antibacterial activity from a large set of 657 chemical structures. With a limited number of AMPs datasets, a common obstacle of using QSAR descriptors is the high dimensionality of the input space that interrupts the estimation of internal parameters classifier.

ML methodologies can significantly develop the progression and even relatively replace expensive wet laboratory trials by learning a predictor with an existing dataset or with a smaller quantity of data generation. In 2015, ML and kernel methods were used to assist the design of highly active peptides for drug discovery. Kernels methods are symmetric positive semi-definite similarity functions between strings (amino acids). These algorithms are exceptionally effective at offering accurate models for a broad range of biological and chemical problems such as anti-cancer activity and antimicrobial activity. A study was conducted using this method, two different datasets of 132 peptides were used for testing and validation. The highest predicted biological activity was generated by using the K-longest path algorithm and the predictors learned by Kernel Ridge Regression (KRR). As a result, this approach demonstrates the ability to predict peptides with the highest biological activity for ML predictors and potential functional motifs (Giguère et al., 2015).

Integrating sequence alignment and SVM could also be used to predict AMPs. (Ng et al., 2015) proposed the new algorithm that was analyzed using jackknife test and independent test. The Jackknife test is used to measure the performance of a different version of the sequence alignment method (BLASTP). This algorithm is divided into two main stages; by combining it with the sequence alignment method to predict AMPs sequences, and with SVMs Lempel-Ziv (LZ) pairwise algorithm. The positive training set consists of 2752 sequences and 10014 in the negative training set. The proposed algorithm obtained 95.28% and 87.59% of sensitivity in the jackknife test and in the independent test respectively. This declares that the pairwise similarity scores have significant development in the sensitivity measurement and helps to increase the prediction accuracy.

Another study by using ML and genetic programming was done by (Veltri et al., 2015). They explored a novel method for feature construction and selection to improve AMP recognition. They used k-mer or motif as a foundational building block (a construct similar to N-gram). The presence of such features allowed them to use an Evolutionary Feature Construction algorithm (EFC) based on genetic programming with the fast correlation-based filter selection (FCBF) algorithm, for discovering the potentially vast area in search of those that differentiate between AMPs and Non-AMPs classification setting. The EFC-FCBF features offer substantial developments in AMP recognition over state of the art. The FCBF provides a set of highly relevant features with low redundancy. A comparative analysis in two different experimental settings was conducted. The first displays the advantage of how wet laboratory researchers could combine their sequence-based features with additional knowledge of AMPs to create better predictive models. In the second experiment, they used different datasets and compared the EFCFCBF method to several AMPs recognition methods, which validate the ability to identify target specific classes of AMPs. These outcomes suggest that the quality of the features found by EFC-FCBF is much higher than that of k-mer features with more than 14%.

On the other hand, N-gram approach was successfully utilized for prediction of Human Immunodeficiency Virus (HIV) drug resistance. In 2011, Masso used the N-grams approaches for representing as feature vectors, two large datasets of V3 loop peptide sequences of HIV-1 viruses, and the RF algorithm is applied for classification. These datasets of gp120 V3 loop sequences, taken from patient HIV-1 viruses with known co-receptor usage. The method starts by using a sliding window of size n on every V3 loop sequence to identify all subsequences of n consecutive amino acids residues. The RF algorithm creates multiple bootstrap datasets of size n from the original set, and each bootstrap dataset is used to train an unpruned classification tree that available with the WEKA suite. A comparison of the accuracy reported for those ML classifiers with the performance accomplished using relatively easier and more computationally efficient N-grams. This reveals significant advantages for the prediction of HIV drug resistance (Masso, 2011).

In the following year, Masso and Vaisman used N-gram for sequence and structure-based models of HIV-1 protease (PR) and reverse transcriptase (RT) drug resistance. Statistical learning algorithms were implemented to develop structure and sequence-based models for predicting the effects of mutations in the PR and RT proteins. Relative frequencies or counts of N-grams were applied for developing a sequence-based model and generate vectors for representing mutant proteins. All algorithms were implemented using the WEKA for classifications. These models provided orthogonal and complementary prediction methodologies and were used to classify all pairs of RT inhibitors as part of an antiretroviral cocktail, or a combination that is to be avoided (Masso & Vaisman, 2013).

2.2 Reduced Alphabet Literature Review

Reducing the alphabet without losing vital biochemical data unlocks the door to potentially faster data mining, ML, and optimization applications in the bioinformatics field. A cell requires many different proteins to regulate and perform cellular processes. On the atomic level, the structures of these proteins are highly diverse and complex. The fundamental building blocks of the proteins are the 20 naturally occurring amino acids. An amino acid contains both an amino group and a carboxylic group. Amino acids that have an amino group attached directly to the alpha-carbon are stated as alpha amino acids. From a combinatorial outlook, there is an almost infinite variety of sequences that can be created from a 20-letter code, for instance, for a polypeptide chain of length 100 can make 20^{100} possible combinations (Melo & Marti-Renom, 2006).

A large number of reduced amino acid alphabets have been proposed to simplify compositional representation of proteins. The resulting reduced amino acids alphabets have been applied to protein folding, protein structure prediction, generation of consensus sequences from multiple alignments and pattern recognition. Therefore, reducing the amino acids alphabet would permit a more detailed examination of other properties in protein structures that could become important but have not yet been studied due to the outlined limitations.

Thus, the complex sequence of amino acids of a protein encodes for its specificity and diversity. There have been several attempts to reduce the naturally occurring amino acids alphabet because they share similar physicochemical properties and can be naturally replaced by protein sequences of the same family. The problem lies in finding the proper grouping of amino acids that holds most of the information required for the integrity of the structure and function of proteins (Solis, 2015).

Previously, some theoretical works have proposed that the minimum number of amino acids types required to encode for native proteins is less than 20. The usual way to design a reduced amino acids alphabet consists of bundling amino acids into groups according to certain features. These features include size, flexibility, hydrophobicity scale and common chemical groups at the side chains (Melo & Marti-Renom, 2006).

In 1999, Wang and Wang derived clusters from the Miyazawa–Jernigan (MJ) interaction 20×20 matrix. They present a reduction method based on an analysis of the statistical contact potentials of the MJ matrix. By minimizing the mismatches score between a reduced matrix and the MJ matrix versus the number of residue types, they find three regions: (1) a polar dominated grouping (PDG) (2) a hydrophobic dominated grouping (HDG) and (3) a singlet dominated grouping (SDG) (Wang & Wang, 1999). In this work, the minimized mismatch finds a theoretical way to understand the process of reduction to schemes with different sets of residues. The plateaus in three regions provide suitable representations of proteins related to different interactions, such as polarity or hydrophobicity. Also, the comparison of results from sequences with 20 residue types and their reduced alphabet representations shows that the reduction by mismatch minimization is successful, e.g. sequences with five residues types have a good folding ability and kinetic accessibility in model studies. Briefly, the five-letter scheme may be a form of simplified representation of natural proteins (Wang & Wang, 1999).

Another study done by (Murphy et al., 2000), they make an analogy between sequence patterns that create foldable sequences and those which make it possible to find structural homologs by aligning sequences and use it to recommend the possible size of a reduced alphabet. This estimate that 10–12 reduced alphabets letters can be used to design foldable sequences of protein families. The estimation is based on the observation of a slight loss of the information required to pick out structural homologs in a clustered sequence of protein database when appropriate reduction from 20 to 10 of the amino acids alphabet letters is made. However, this information is ruined when additional reductions in the alphabet are made. The amino acids reduction scheme is formed on the analysis of correlations indicated by the Blocks Substitution Matrix (BLOSUM) 50 similarity matrix that used for sequence alignments. They find when the alphabet size is reduced, the information of the amino acid's sequences responsible for protein fold recognition is degraded. Furthermore, they conclude that a minimum of three different amino acids types is necessary for protein folding, and the sequences constructed from 10-letter alphabets obtained by grouping amino acids appropriately hold approximately as much information as the natural sequences have (Murphy et al. , 2000).

In 2002, Liu et al., propose an algorithm for amino acids alphabet reduction based on random background deviation of conditional probability and to compare results found from other schemes of reduction. They detected a sequence homology in SCOP database with the derived coarse-grained BLOSUM similarity matrices and the clustering using residue counts of either BLOSUM or MJ is not completely hierarchical. Their results of homology recognition with reduced alphabets show that the percentage coverage retained is reduced by only 10% for 9-letters. Hence, there is no significant drop in the coverage if the number of letters is not smaller than 9. The 5-letters correlation coefficient and covariance are still reasonable even if the number of clusters is as small as 5. In the end, they conclude that the 9-letter reduced alphabet preserves most information of the original 20-letter alphabet, and the 5-letter alphabet reduced is still a reasonable choice (Liu et al., 2002).

On the other hand, Li et al. demonstrated that ten types of residues may be the minimum number of letters needed to construct a rational folding model. They used a simplified BLOSUM62 matrix to perform a global sequence alignment and create coverage detection on the remotely related homologous proteins throughout the Structural Classification of Proteins (SCOP) 40 database for several levels of reduction. With these reduced alphabets, they achieve recognition of the protein folding based on the sequence alignment similarity score, which can reserve the maximal information on the original sequence. They found that groups more than N = 10 will not increase the efficiency of the description of the protein's complexity from the feature of the sequence alignment. Consequently, 10-letters of amino acids may be the degree of freedom for characterizing the complexity in proteins and the maximum information that could make the protein closer to that consisting of 20 amino acids (Li et al., 2003).

Melo and Renom derived an amino acids substitution matrices and statistical potentials for the prediction of remote homologs of protein structure. These substitution matrices were based on several reduced amino acids alphabets, their sequence alignment and fold evaluation of protein structure models, which use as a reference frame the 20 amino acids of standard alphabet. The results of this work showed that a large reduction in the total number of residue types does not indeed translate into an important loss of discriminative power for sequence alignment and fold assessment. However, few residue types can encode most of the significant sequence and structure information which is present in the 20-standard alphabet amino acids. In other words, reduced alphabets display a similar performance as the standard alphabet in the tasks of sequence alignment of remote homologs and fold assessment of protein structure models (Melo & Marti-Renom, 2006).

Lately, automated methods to reduce the dimensionality of protein structure prediction datasets have been used. These methods are easier and faster learning process and generation of more compact and human-readable solutions. This simplification contains an alphabet reduction technique to map the 20 naturally occurring amino acids into the lower cardinality alphabet by grouping similar amino acids types. In 2007, a researcher performed experiments to reduce the amino acids alphabet into two, three, four, and five groups. They tested the performance of the reduction criteria found by this optimization procedure by learning the reduced dataset and comparing the predictive performance to the one obtained by learning the original 20 letters amino acids alphabet. Genetic algorithm was used, and the results were validated by learning the reduced dataset with a genetics-based ML algorithm (Bacardit et al., 2007).

This study indicated that it is possible to make a reduction into a new alphabet with only three letters, which lead to faster computation and more compact rules. Although, these three letters alphabet is not significantly different when compared to the performance obtained from the full 20 amino acids alphabet based on the protein-wise accuracy metric. Therefore, this automated alphabet reduction method showed some promising performance, and it has the potential to be a very valuable tool to simplify the learning procedure of several datasets associated with protein structure prediction (Bacardit et al., 2007).

Peterson et al. demonstrated that a reduced alphabet approach to building up protein profiles may advance the ability to detect proteins with structural homology by expanding the knowledge of the chemical prosperities of the amino acids to build up a physical image of a fold. In this study, over 150 of the amino acids clustering schemes were tested with all-versus-all pairwise sequence alignments of sequences in the matrix alignment database and combined it with several metrics as mean precision and area under the Receiver Operating Characteristic (auROC) curve. They also examined the statistical significance of the best matrices to determine whether the differences in performance are significant or not (Peterson et al., 2009).

As a result, they found that reduced alphabets can perform at a level comparable to full 20 alphabets in correct pairwise alignment and can display increased sensitivity to pairs of sequences with structural similarity but low-sequence identity. The consensus from these methods is that performance enhancements can be made by correctly grouping the amino acids into 9–12 clusters. In addition, they found that reduced alphabets can return more remotely related pairs of proteins comparable to full alphabets (Peterson et al., 2009).This contrast with some earlier studies (Murphy et al., 2000; Liu et al., 2002; Li et al., 2003) which stated that reduced alphabets could only produce losses in performance relative to a full alphabet.

In 2009, Bacardit et al., investigated automated and generic alphabet reduction procedures for protein structure prediction datasets by using a primary sequence representation of proteins. Two protein structural features were applied that are contact number and relative solvent accessibility. For both features, they generated alphabets of two, three, four, and five letters. The five-letter alphabet obtained to reduce a protein representation using evolutionary information and a position-specific scoring matrix (PSSM) representation. Besides, they compared the automatically designed alphabets - quantitatively analyzed- against other reduced alphabets taken from the human-designed or literature, outperforming them. Their results indicate that the five-letter alphabets provided prediction accuracies within 1% of that obtained by full amino acids alphabet, and higher accuracy than the other reduced alphabets involved in the comparison. In this experiment, they found a reduced alphabet with a performance that is statistically equal to the performance obtained with the full amino acids type representation is possible, and this does not compromise accuracy and enhances interpretability (Bacardit et al., 2009).

Recently, Solis has designed a fully automatic amino acids alphabet reduction algorithm. This algorithm has generated an optimal clustering of the 20 amino acids into smaller cluster groups (from 2 to 19). The clustering design recovers amino acids properties such as hydrophobicity, charge, polarity, size, and aromaticity. 114 reduced alphabets were assembled (alphabet sizes ten and below) from the literature. Then, they classified them into four general categories based on physicochemical properties, multiple sequence alignments of structural homologs, similarities in local structure or backbone coding, and long-range interaction considerations (Solis, 2015).

In this experiment, Solis found that a significant amount of mutual information around 75% is preserved by 2-letter reduction. Correspondingly, reduced alphabets less than10 can capture almost all the information residing in native contacts and may be sufficient for fold recognition, as demonstrated by extensive tests. He also found that much of the mutual information is preserved at significantly small alphabet sizes, e.g., 5-letter alphabet captures around 90%, whereas a 9-letter alphabet nearly 96%, and further information is achieved when expanding to higher alphabet sizes (Solis, 2015).

Based on these studies, we figured out that reduced alphabets could derived from varying methods, including those derived from local structure considerations, physicochemical intuition, genetic code, and sequence alignments of remote homologs. All these different methods often led to highly divergent final results. Indicating that the definition of a reduced amino acids alphabet is very dependent on the clustering method and the information used. The most simplistic reduction alphabet of the amino acids explained to date consists of the definition of two residue types, which is identified as the hydrophobic-polar or hydrophilic model. Lastly, reduced alphabets may also show performance gains with more sophisticated methods such as profile and pattern searches.

Chapter 3: Material and Methods

3.1 Reduced Alphabets

The 20-letter amino acid alphabets were reduced to significantly fewer letters of an alphabet to quicken and simplify the ML process, as the model would have to analyze fewer N-grams (attributes) during training and testing the model. Another issue with using the original 20-letter alphabet was that the total number of features (N-grams) would exceed the number of sequences, 7984 at most, which is not ideal considering that each instance (sequence) only contains a maximum of 120 amino acid residues. By decreasing the number of letters in the alphabet, and thus decreasing the number of possible N-grams, the number of sequences would be significantly larger than the total number of features, so it would be highly unlikely for the model to overfit.

Residues can be clustered based on several properties, including chemical and genetic properties, as the table 3.1 below. Reduced alphabets cluster the residues in ways that prevent the loss of critical biochemical information.

Table 3.1: Amino acids can be clustered according to hydrophobicity and polarity.

1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
C	M	F	I	L	V	W	Y	A	G	T	S	N	Q	D	E	H	R	K	P
Hydrophobic								Polar											
CMFILVWY								AGTSNQDEHRKP											

In this study, over 45 reduced alphabets were collected from outside sources as shown in tables 3.2 to table 3.8 below. The reduced alphabets in the table named according to:

RA-NAME:

1- RA: Reduced Alphabet.
2- First letter of Author last name.
3- Year of publication.
4- Number of groups of letters.
5- Counting letter.

Table 3.2: Two, three, and four-letters reduced alphabets and their sources.

RA-Name	**1.**	**2.**	**3**	**4**
RA-E07-2A	STQNGPAHRED	LIFVMYWCK		
RA-M00-2A	LVIMCAGSTPFYW	EDNQKRH		
RA-B07-2A	ACFGHILMVWY	DEKNPQRST		
RA-L03-2A	CMFILVWY	AGTSNQDEHRKP		
RA-B09-2A	CLVIMAFYWGH	TSNRKDEPQ		
RA-B09-2B	AMWLYCFIV	PGHTSDEKNQR		
RA-B09-2C	LYMFIVCAWGHTS	DEKNPQR		
RA-B09-2D	HCILMVFWYAGSTNR	DEKQP		
RA-L02-2A	MFILVAW	CYQHPGTSNRKDE		
RA-L02-2B	IMVLFWY	GPCASTNHQEDRK		
RA-M00-3A	LASGVTIPMC	EKRDNQH	FYW	
RA-V98-3A	MHVYNDI	QLEKF	WPRGSATC	
RA-B07-3A	ACFILMVWY	DEKNPQR	GHST	
RA-L03-3A	CMFILVWY	AGTSP	NQDEHRK	
RA-B09-3A	CLVIMAFYW	GHTS	NRKDEPQ	
RA-B09-3B	AMWLYCFIV	PGHTS	DEKNQR	
RA-B09-3C	LYMFIV	CAWGHTS	DEKNPQR	
RA-B09-3D	CIMFLVWY	AGHTNSP	RDEKQ	
RA-B09-3E	CILMVFWY	AGHSTP	DEKNQR	
RA-B09-3F	CILMVFWY	AGHST	EKDNRPQ	
RA-B09-3G	HCILMVFWY	AGSTNR	DEKQP	
RA-W99-3A	CMFILVWY	ATHGPR	DESNQK	
RA-S15-3A	ACGPSTWY	DEHKNQR	FILMV	
RA-S15-3B	AFILMVWY	C	DEGHKNPQRST	

RA-S15-3C	CFILMVWY	DEGKNQS	AHPRT	
RA-L02-3A	MFILVAW	CYQHPGTSNRK	DE	
RA-L02-3B	IMVLFWY	GPCAST	NHQEDRK	
RA-M06-4A	AGPST	CILMV	DEHKNQR	FYW
RA-P09-4A	ADKERNTSQ	YFLIVMCWH	G	P
RA-M00-4A	LVIMC	AGSTP	FYW	EDNQKRH
RA-B07-4A	AFHTY	CILMV	DEKPQ	GNRSW
RA-L03-4A	CFYW	MLIV	GPATS	NHQEDRK
RA-L03-4B	CMFWY	ILV	AGTS	NQDEHRKP
RA-B09-4A	CLVIM	AFYHT	WGSNR	KDEPQ
RA-B09-4B	CALM	VIFW	YGH	TSNRKDEPQ
RA-B09-4C	AMW	LYC	FIV	GHTSDEKNQRP
RA-B09-4D	CALY	MFIV	WGHT	SDEKNPQR
RA-B09-4E	ACIMHT	FLVWY	GNSR	PDEKQ
RA-B09-4F	CILMVFWY	AGHST	EKD	NRPQ
RA-S15-4A	FWY	CILMV	DEGKNQS	AHPRT
RA-S15-4B	AFILMPVW	CGNQSTY	DE	HKR
RA-S15-4C	AFILMVWY	C	DEKR	GHNPQST
RA-L02-4A	MFILV	ACW	YQHPGTSNRK	DE
RA-L02-4B	IMVLFWY	G	PCAST	NHQEDRK

Table 3.3: Five-letters reduced alphabets and their sources

RA-Name	**1.**	**2.**	**3.**	**4.**	**5.**
RA-M06-5A	AHT	CFILMVWY	DE	GP	KNQRS
RA-M06-5B	AEHKQRST	CFILMVWY	DN	G	P
RA-M06-5C	AG	C	DEKNPQRST	FILMVWY	H
RA-E07-5A	G	P	IVFYW	ALMEQRK	NDHSTC
RA-M00-5A	LVIMC	ASGTP	FYW	EDNQ	KRH
RA-L03-5A	FWYH	MILV	CATSP	G	NQDERK
RA-L03-5B	CFYW	MLIV	G	PATS	NHQEDRK
RA-B09-5A	CLVH	IAS	FWGM	KNRT	DEPQY
RA-B09-5B	AM	WLY	CFIV	GHTS	DEKNQRP
RA-B09-5C	ACIY	MFLV	GHTN	SWDE	PRKQ
RA-B09-5D	CILMVFWY	A	GHST	EK	DNRPQ
RA-W99-5A	CMFI	LVWY	ATGS	NQDE	HPRK
RA-S15-5A	AFILMVWY	C	DE	GHNPQST	KR
RA-S15-5B	FWY	CILMV	DEGKNS	APQT	HR
RA-L02-5A	MFILV	ACW	YQHPGTSN	RK	DE
RA-L02-5B	IMVL	FWY	G	PCAST	NHQEDRK

Table 3.4: Six-letters reduced alphabets and their sources

RA-Name	**1.**	**2.**	**3.**	**4.**	**5.**	**6**
RA-E07-6A	MFILV	W	C	KRQE	DNASTPGH	VILFY
RA-M00-6A	LVIM	ASGT	PHC	FYW	EDNQ	KR
RA-L03-6A	CFYW	MLIV	G	P	ATS	NHQEDRK
RA-S15-6A	A	C	DE	FILMVWY	GHNPQST	KR
RA-S15-6B	FWY	CILMV	DE	GKNQS	APT	HR
RA-S15-6C	AGPST	C	DENQ	FWY	HKR	ILMV
RA-S15-6B	ADEGKNQRST	C	FILMVY	H	P	W
RA-S15-6D	AGPST	C	DEKNQR	FILMVY	H	W
RA-S15-6E	AST	CP	DEHKNQR	FWY	G	ILMV
RA-S15-6F	ACST	DEKNQR	FHWY	G	ILMV	P
RA-S15-6J	AEKQR	CHST	DN	FIV	GP	LMWY
RA-S15-6H	AEFHIKLMQRVWY	CT	DN	G	P	S
RA-S15-6I	ALM	CHT	DNS	EKQR	FIVWY	GP
RA-S15-6J	ACGST	DENQ	FWY	HKR	ILMV	P
RA-L02-6A	MFILV	A	C	WYQHPGTSN	RK	DE
RA-L02-6B	IMVL	FWY	G	P	CAST	NHQEDRK

Table 3.5: Seven, eight, nine, and ten -letters reduced alphabets and their sources

RA-Name	**1.**	**2.**	**3.**	**4.**	**5.**	**6**	**7**	**8**	**9**	**10**
RA-L03-7A	FWYH	MILV	CATS	P	G	NQDE	RK			

RA-L03-7B	CFYW	MLIV	G	P	ATS	NHQED	RK			
RA-S15-7A	FWY	CILLMV	DE	K	GNPQS	AT	HR			
RA-S15-7B	A	C	DE	FILMVWY	G	HNPQST	KR			
RA-L02-7A	MFILV	A	C	WYQHP	GTSN	RK	DE			
RA-L02-7B	IMVL	FWY	G	P	CAST	NHQED	RK			
RA-M00-8A	LVIMC	AG	ST	P	FYW	EDNQ	KR	H		
RA-L03-8A	FWYH	MILV	CA	NTS	P	G	DE	QRK		
RA-L03-8B	CFYW	MLIV	G	P	ATS	NH	QED	RK		
RA-S15-8A	A	C	DE	FILMV	G	HNPQST	KR	WY		
RA-S15-8B	AGILV	CM	DE	FWY	HKR	NQ	P	ST		
RA-S15-8C	FWY	ILMV	C	DE	K	GNPQS	AT	HR		
RA-L02-8A	MFILV	A	C	WYQHP	G	TSN	RK	DE		
RA-E07-9A	G	P	IV	FYW	ALM	EQRK	ND	HS	TC	
RA-L03-9A	FWYH	ML	IV	CA	NTS	P	G	DE	QRK	
RA-L03-9B	CFYW	ML	IV	G	P	ATS	NH	QED	RK	
RA-S15-9A	FWY	ILMV	C	DE	K	GNQS	PT	A	HR	
RA-S15-9B	A	C	DE	FILMV	G	HNQST	KR	P	WY	
RA-L02-9A	MF	ILV	A	C	WYQHP	G	TSN	RK	DE	
RA-L02-9B	IMV	L	FWY	G	P	C	AST	NHQED	RK	
RA-M00-10A	LVIM	C	A	G	ST	P	FYW	EDNQ	KR	H
RA-L03-10A	FWY	ML	IV	CA	TS	NH	P	G	DE	QRK
RA-L03-10B	C	FYW	ML	IV	G	P	ATS	NH	QED	RK
RA-S15-10A	WY	F	ILMV	C	DE	K	GNQS	PT	A	HR
RA-S15-10B	A	C	DE	FILMV	G	HNQ	KR	P	ST	WY
RA-S15-10C	ACGS	DE	FWY	HKR	ILV	M	N	P	Q	T
RA-L02-10A	MF	ILV	A	C	WYQHP	G	TSN	RK	D	E
RA-L02-10B	IMV	L	FWY	G	P	C	A	STNH	RKQE	D

Table 3.6: 11, 12, and 13- letters reduced alphabets and their sources.

RA-Name	**1.**	**2.**	**3.**	**4.**	**5.**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
RA-E07-11A	G	P	IV	FYW	A	LM	EQRK	ND	HS	T	C		
RA-L03-11A	C	FYW	ML	IV	G	P	A	TS	NH	QED	RK		
RA-L03-11B	FWY	ML	IV	CA	TS	NH	P	G	D	QE	RK		
RA-S15-11A	WY	F	ILMV	C	DE	K	G	NPQS	T	A	HR		
RA-L02-11A	MF	IL	V	A	C	WYQHP	G	TSN	RK	D	E		
RA-L02-11B	IMV	L	FWY	G	P	C	A	STNH	RKQ	E	D		
RA-P09-12A	A	D	KER	N	TSQ	YF	LIVM	C	W	H	G	P	
RA-M00-12A	LIVIM	C	A	G	ST	P	FY	W	EQ	DN	KR	H	
RA-L03-12A	FWY	ML	IV	C	A	TS	NH	P	G	D	QE	RK	
RA-L03-12B	C	FYW	ML	IV	G	P	A	TS	NH	QE	D	RK	
RA-S15-12A	WY	F	IL	MV	C	DE	K	G	NPQS	T	A	HR	
RA-L02-12A	MF	IL	V	A	C	WYQHP	G	TS	N	RK	D	E	
RA-L02-12B	IMV	L	FWY	G	P	C	A	ST	N	HRKQ	E	D	
RA-E07-13A	G	P	IV	FYW	A	L	M	E	QRK	ND	HS	T	C
RA-L03-13A	FWY	ML	IV	C	A	T	S	NH	P	G	D	QE	RK
RA-L03-13B	C	FYW	ML	IV	G	P	A	T	S	NH	QE	D	RK
RA-S15-13A	WY	F	IL	MV	C	DE	K	G	P	NQS	T	A	HR
RA-L02-13A	MF	IL	V	A	C	WYQHP	G	T	S	N	RK	D	E

RA-L02-13B	IMV	L	F	WY	G	P	C	A	ST	N	HRKQ	E	D

Table 3.7: 14, and 15 letters reduced alphabets and their sources

RA-Name	**1.**	**2.**	**3.**	**4.**	**5.**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
RA-L03-14A	C	FYW	ML	IV	G	P	A	T	S	N	H	QE	D	RK	
RA-L03-14B	FWY	ML	IV	C	A	T	S	NH	P	G	D	QE	R	K	
RA-S15-14A	W	Y	F	IL	VM	C	DE	K	G	P	NQS	T	A	HR	
RA-L02-14A	MF	I	L	V	A	C	WYQHP	G	T	S	N	RK	D	E	
RA-L02-14B	IMV	L	F	WY	G	P	C	A	S	T	N	HRKQ	E	D	
RA-M00-15A	LVIM	C	A	G	S	T	P	FY	W	E	D	N	Q	KR	H
RA-L03-15A	C	FYW	ML	IV	G	P	A	T	S	N	H	QE	D	R	K
RA-L03-15B	FWY	ML	IV	C	A	T	S	N	H	P	G	D	QE	R	K
RA-S15-15A	W	Y	F	IL	VM	C	DE	K	G	P	NQS	T	A	H	R
RA-L02-15A	MF	IL	V	A	C	WYQ	H	P	G	T	S	N	RK	D	E
RA-L02-15B	IMV	L	F	WY	G	P	C	A	S	T	N	H	RKQ	E	D

Table 3.8: 16, 17, and 18- letters reduced alphabets and their sources

RA-Name	**1.**	**2.**	**3.**	**4.**	**5.**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
RA-L03-16A	C	FY	W	ML	IV	G	P	A	T	S	N	H	QE	D	R	K		
RA-L03-16B	W	FY	ML	IV	C	A	T	S	N	H	P	G	D	QE	R	K		
RA-S15-16A	W	Y	F	IL	M	V	C	DE	K	G	P	NQS	T	A	H	R		
RA-L02-16A	MF	I	L	V	A	C	WYQ	H	P	G	T	S	N	RK	D	E		
RA-L02-16B	IMV	L	F	W	Y	G	P	C	A	S	T	N	H	RKQ	E	D		

RA-L03-16A	C	FY	W	ML	IV	G	P	A	T	S	N	H	QE	D	R	K		
RA-L03-16B	W	FY	ML	IV	C	A	T	S	N	H	P	G	D	QE	R	K		
RA-S15-16A	W	Y	F	IL	M	V	C	DE	K	G	P	NQS	T	A	H	R		
RA-L02-16A	MF	I	L	V	A	C	WYQ	H	P	G	T	S	N	RK	D	E		
RA-L02-16B	IMV	L	F	W	Y	G	P	C	A	S	T	N	H	RKQ	E	D		
RA-L03-16A	C	FY	W	ML	IV	G	P	A	T	S	N	H	QE	D	R	K		
RA-P09-17A	A	D	KE	R	N	T	S	Q	Y	F	LIV	M	C	W	H	G	P	
RA-L03-17A	C	FY	W	ML	IV	G	P	A	T	S	N	H	Q	E	D	R	K	
RA-L03-17B	W	FY	ML	IV	C	A	T	S	N	H	P	G	D	Q	E	R	K	
RA-S15-17A	W	Y	F	I	L	M	V	C	DE	K	G	P	NQS	T	A	H	R	
RA-M00-18A	LM	VI	C	A	G	S	T	P	F	Y	W	E	D	N	Q	K	R	H
RA-L03-18A	C	FY	W	M	L	IV	G	P	A	T	S	N	H	Q	E	D	R	K
RA-L03-18B	W	FY	M	L	IV	C	A	T	S	N	H	P	G	D	Q	E	R	K
RA-S15-18A	W	Y	F	I	L	M	V	C	DE	K	G	P	N	QS	T	A	H	R

Table 3.9: 19, and 20- letters reduced alphabets and their sources

RA-Name	**1.**	**2.**	**3.**	**4.**	**5.**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
RA-L03-19A	C	F	Y	W	M	L	IV	G	P	A	T	S	N	H	Q	E	D	R	K	

RA-L03-19B	W	F	Y	M	L	IV	C	A	T	S	N	H	P	G	D	Q	E	R	K	
RA-S15-19A	W	Y	F	I	L	M	V	C	D	E	K	G	P	N	QS	T	A	H	R	
RA-P09-20A	A	D	K	E	R	N	T	S	Q	Y	F	L	I	V	M	C	W	H	G	P
RA-M00-20A	L	V	I	M	C	A	G	S	T	P	F	Y	W	E	D	N	Q	K	R	H
RA-L03-20A	C	F	Y	W	M	L	I	V	G	P	A	T	S	N	H	Q	E	D	R	K
RA-L03-20B	W	F	Y	M	L	I	V	C	A	T	S	N	H	P	G	D	Q	E	R	K
RA-W99-20A	C	M	F	I	L	V	W	Y	A	T	H	G	P	D	E	S	N	Q	R	K
RA-S15-20A	W	Y	F	I	L	M	V	C	D	E	K	G	P	N	Q	S	T	A	H	R
RA-L02-16B	I	M	V	L	F	W	Y	G	P	C	A	S	T	N	H	R	K	Q	E	D

In order to apply each alphabet reduction option to each set of peptide sequences, multiple java programs needed to be written. Each program is designed to traverse each sequence of each peptide set and express each amino acid by which cluster it is located in. A representative amino acid indicates which group the reduced amino acid is in. After the program is run, in 3 letters reduced alphabets, there are only three distinct letters, as opposed to the original 20. While in 4 letters reduced alphabet, there are only four distinct letters opposed to the original 20, as shown in table 3.10 below. The residue clusters were denoted by the letters (B, J, U, X, Z, and O) in this study. If the number of reduced alphabets exceeds these distinct letters, we combined it with numbers, e.g., B1J2U3 and so on.

Table 3.10: The example of reduced alphabets. Each letter contains a cluster of amino acid residues. The residue clusters were denoted by the letters (B, J, U, X, Z, and O).

Reduced Alphabet	**(B)**	**(J)**	**(U)**	**(X)**	**(Z)**
ra2-11	IMVLFWY	GPCASTN HQEDRK			
ra3-29	IMVLFWY	GPCAST	NHQEDRK		
ra4-47	IMVLFWY	G	PCAST	NHQEDRK	
ra5-64	IMVL	FWY	G	PCAST	NHQEDRK

3.2 N-grams

This dissertation integrates N-gram analysis and ML, two different approaches, to develop a computational model that is able to accurately classify AMPs. These methods allowed to analyze how well N-gram frequencies can be used to train ML algorithms. N-grams are a commonly used technique in computational probability, linguistics, text categorization, and biology. It is a sequence of N quantity characters from a given text or string. Depending on the size of the N-gram, it also named a unigram (1 letter), bigram (2 letters), or a trigram (3 letters).

An N-gram has been denoted as a contiguous string of N amino acid residues in the protein sequence. The amino acid sequence, or the primary structure of a protein, determines the protein's three-dimensional structure. This implies that disorder, or lack of stable structure, can also be encoded in the sequence. A sequence can be disintegrated into a list of overlapping N-grams. N-gram patterns have been previously used to show evolutionary relationships between protein sequences and to predict protein secondary structure (Masso, 2011). A key advantage of using N-gram frequencies is that they are a computationally low-cost way of analyzing complex patterns in protein sequences.

N-gram algorithm is used to estimate the probability of relative frequency counts. It reads each sequence in the dataset and calculates the relative frequency of N-grams to show each sequence composition regarding feature vectors. These vectors representation generated by N-grams used by WEKA ML algorithms to make the classification model.

Different sizes of N-grams were used in this work considering the computational expense, which leads to a different number of distinct combinations of amino acid sequences. For example, using a three-letters alphabet with N-gram of size three or trigram leads to 27 distinct combinations of amino acid residues (3^3 distinct letters), while N-gram of size four leads to 81 distinct combinations of amino acid residues (3^4 distinct letters).

The N-gram program reads in each sequence from the input dataset and calculates the frequency of occurrence of each gram. Now each sequence composition is merely represented regarding the frequency of occurrence of each of the distinct combinations of amino acid sequences. So, different AMPs have different N-gram frequencies.

The frequency of each peptide sequence was calculated by determining how frequently this particular letter of amino acid sequence occurred in the entire peptide sequence, and to divide that by the total number of possible N-grams in the peptide sequence. The total number of possibilities is essentially two less than the length of the peptide, as every other amino acid should following it:

$$f_{ijk} = \frac{n_{\ ijk}}{n_{\ total}}$$

Where $\boldsymbol{f}$ represents N-gram frequencies, $n_{\ ijk}$ is how often this three-letter (ijk) amino acid sequence occurred throughout the entire peptide sequence, $n_{\ total}$ is the total number of possible N-gram in the peptide sequence.

n could not be made extremely large because the total number of possible N-grams would run the risk of surpassing the size of the dataset, which could cause model overfitting. N-gram frequencies were then normalized to prevent the frequency of a feature from skewing the decision process and allowing the comparison of different sized amino acid sequences. The normalized expression is called log-likelihood and for trigrams is given by:

$$q_{ijk} = log\left(\frac{f_{ijk}}{f_i f_j f_k}\right)$$

where f_{ijk} is the frequency of occurrence of a particular trigram and f_i, f_j, f_k is the frequency of occurrence of each amino acid in the trigram in the entire sequence. After using N-gram analysis to extract features from AMPs and Non-AMP and representing each sequence distinctively with a vector of N-gram combination frequencies, ML algorithms are then applied.

3.3 Machine Learning (ML) Approach

Machine Learning (ML) refers to computer algorithms and artificial intelligence applications dealing with the formation and evaluation of algorithms that assist pattern recognition, classification, and prediction, based on models originated from existing data (Tarca et al., 2007). ML uses the theory of statistics in the construction of mathematical models because the primary task is making the inference from a sample. When provided a series of features and observations as input, an algorithm tries to deduce rules or patterns which yield to an appropriate solution (Dietterich et al., n.d.).

The history of relations between bioinformatics and the field of ML is long and complicated. Several ML approaches are applied to discover new meaningful knowledge from the biological databases, to investigate and predict diseases, to cluster similar genetic elements, and to find associations or relationships in the biological database, especially when dealing with complex and high-dimensional data. (Tarca et al., 2007). Examples of ML techniques that have successfully been used in numerous bioinformatics applications:

1. ANN have predicted protein cleavage sites.
2. Genetic algorithms have been applied to determine gene expression levels and DNA promoter binding sites.
3. Evolutionary algorithms have been utilized to microarray classification.
4. Decision trees have been used for protein secondary structure prediction.
5. RF and NB use training data to generate classifiers that can assign labels to new data.

ML algorithms, such as RF and NB, can be trained using protein sequences' N-gram frequencies to decide whether a sequence is ordered or disordered. In the context of this dissertation, we use N-gram and ML methods to focus on classify and predict between positive (AMP) and negative (Non-AMP) observations. In other words, given an unknown peptide sequence, to classify and predict it as an AMP or Non-AMP appropriately.

3.4 Machine Learning Classifiers

In this study, seven machine learning algorithms Random Forest (RF), Support Vector Machine (SVM), Bagging, Decision tree (J48), Naïve Bayes (NB), Artificial Neural Network (ANN), and AdaBoost are consequently employed to learn from the N-gram frequencies of sequences to develop classification models.

3.4.1 Random Forest (RF)

A new regression and classification tool, RF, is a meta-learner, meaning consisting of many individual trees and was introduced by Breiman (Livingston, 2005). RF is initiated and investigated for predicting a compound's quantitative or categorical biological activity based on the quantitative description of the compound's molecular structure (Svetnik et al., 2003). The RF uses multiple random trees classifications to votes on an overall classification for the given data set. This algorithm was modified to achieve both unweighted and weighted voting. Then, the forest chooses the individual classification that holds the most votes. In general, this algorithm studies each attribute and shows the importance of the attribute in predicting the accurate classification of the RF machine learner. The user afterward could filter out unnecessary attributes that would save time during data collecting and experimental run time (Livingston, 2005).

RF was the primary classifier used in this study. One advantage of using it over the decision tree is that it corrects the overfitting of the decision tree model. Basically, RF operates by constructing a certain quantity of decision trees, and it outputs the class that is the most frequently stated by the trees (Liaw & Wiener, 2002).

3.4.2 Support vector machines (SVM)

SVM is a classification method introduced in 1992 by Boser, Guyon, and Vapnik (Ben-Hur & Weston, n.d.). SVM performs classification by finding a hyperplane which separates the N-dimensional data perfectly into its two categories or classes. However, since some of the data are regularly not linearly separable, SVM's introduce the concept of a "kernel induced feature space" that casts the data into high dimensional space where the data is separable (Boswell, 2002). Consequently, SVMs belong to the general category of kernel methods which is an algorithm that depends on the data only through dot-products. This SVM classifier is widely used in bioinformatics and other fields due to its high accuracy, ability to deal with high-dimensional data set such as gene expression, flexibility in modeling various sources of data, and employ sophisticated mathematical principles to avoid overfitting (Tarca et al., 2007).

3.4.3 Decision tree (J48)

A decision tree is a predictive model machine-learning approach that selects the target value of a new sample-based on several attribute values of the existing data. A J48 algorithm is a simple form of the C4.5 decision tree developed by J. Ross Quinlan (Salzberg, 1994). In the classification process in decision trees, it first needs to generate a decision tree based on the attribute values of the existing training data. So, whenever it encounters a set of the training set, it classifies the attribute values that discriminate the different instances most clearly (Daraei & Hamidi, 2017).

3.4.4 Naïve Bayes (NB)

Naïve Bayes is a simple probabilistic classifier algorithm based on using Bayes theorem with strong an independence "naive" assumptions to produce independent feature model (Hu, 2017). This NB classifier assumes that the presence of a specific feature of a class which is not related to the existence of any other feature. NB classifiers can be trained appropriately in a supervised learning setting, depending on the exact nature of the probability model. In several applications, NB models use the maximum likelihood method, which assumed to be Gaussian (Gao et al., 2016).

3.4.5 Multilayer Perceptron (MLP)

MLP is one type of artificial neural network (ANN) model. It consists of sets of an input layer that obtains data, computations are performed in the hidden layers, and sets of the output layer, which provides the classification result. The output sets represent hyper-plane in the space of the input patterns (Hoffman et al., 2012).

Neural networks are capable of developing meaning from imprecise or complicated data and can be used to detect trends and extract patterns that are too complex to be identified by either computer techniques or humans. A trained neural network can be assumed as an "expert" in the category of data it has been given to analyze. This expert can then be utilized to provide predictions given new conditions of interest and answer "what if" questions (Gao et al., 2016).

3.4.6 AdaBoost

AdaBoost is a well-known ensemble algorithm learning-based classification, which was first proposed by Freund and Schapire (1997). AdaBoost produces the final output by weighting the instances in the dataset by how difficult or easy they are to classify using the majority vote method (Hu, 2017).

3.4.7 Bagging

Bagging or Bootstrap Aggregating is an ensemble method that aims to sample data sets for an ensemble of classifiers. The Bagging ensemble commonly contains the procedures of aggregation and bootstrap sampling(Gao et al., 2016). This algorithm produces separate samples of the training dataset and produces a classifier for each sample. The outcomes of these multiple classifiers are then merged (such as majority voting or averaged) (Bashir etal., 2016).

3.5 Randomization Features

In this experiment, we did random shuffling of class labels, in order to further confirm and support the significance of results, control datasets with randomly shuffled (mislabeled) classes were generated using N-gram Classification application (mentioned below in chapter six). Theoretically, if the classification models are significantly accurate and able to detect differences between gene sequences of different classes, the randomly permuted class label dataset should achieve an accuracy about 50% or below.

3.6 WEKA

The Waikato Environment for Knowledge Analysis (WEKA) is a ML workbench that offers a general-purpose environment for automatic regression, classification, clustering, association rules, visualization, and feature selection-common data mining problems in the bioinformatics field. It covers an extensive collection of ML algorithms and data preprocessing approaches complemented by graphical user interfaces (GUI) for data exploration and comparison of different ML methods on the same problem. WEKA is able to process data given in the form of a single table. The primary objectives of WEKA are to:

1. Assist users to extract useful information from data.
2. Enable users to simply identify a suitable algorithm to generate an accurate predictive model (Frank et al., 2004).

WEKA is open source software under the GNU General Public License. It is developed at the University of Waikato and has a collection of ML algorithms for data mining tasks (Frank et al., 2004). These algorithms can either be used directly to a dataset or called from user java code. The consistency of the N-gram methodology is validated by first testing its ability to confirm more existing information, evaluating the accuracy of which sequences from different subtypes can be classified. In this study, the output from N-gram java programs (N-gram Classification application) was converted to ARFF format, compatible with WEKA software version 3.8. The Explorer GUI in WEKA was used to classify sequences based on N-gram frequencies.

3.7 Datasets Creation

This study's sequence-based approach of classification required peptide sequences to be obtained prior to any analysis of AMPs or any of its constituent subclasses.

- **Positive (AMP) dataset:**

  1- **Positive set:**7984 sequences, not necessarily of any specific class from databases APD, CAMP, Uniref-Uniprot, AVPdb, CancerPPD, BACTIBASE, PhytAMP, HIVdb, LAMP, BAGEL, DADP, EROP, YADAMP, Bagel-Joomla, and DBAASP.

  2- **APD set:**1794 Positive sequences from Antimicrobial Peptide Database (APD). These AMPs are from natural sources, demonstrate antimicrobial activities, and all amino acid sequences of the mature peptides have been elucidated (The Antimicrobial Peptide Database, n.d.).

- **Class-specific AMP sets:**

  Class-specific AMP sets (antibacterial, antiviral, and antifungal) were downloaded from CAMP, AVPdb, BACTIBASE, HIVdb, BAGEL, and DBAASP. All these sets were mutually exclusive, verified subsets of the raw positive set. The number of sequences in each raw dataset are shown in table 3.11.

- **Negative (Non-AMP) dataset:**

  1- **Neg set1:**7984 sequences with less than 50% similarity from Uniprot-Uniref with UniRef 50, meaning that each sequence in the Non-AMP set was less than 50% similar to the others.

  2- **Neg set2:**1600 Non-secretory proteins randomly searched from the UniProt database without annotation as ‘antimicrobial’.

Table 3.11: Number of AMP sequences in each class-specific AMP set.

Antimicrobial peptides (AMPs)	**Antibacterial Peptides (ABPs)**	**Antiviral peptides (AVPs)**	**Antifungal Peptides (AFPs)**	**APD positive set**	**Neg set 1+2 (Non-AMPs)**
7984	1914	1091	758	1794	7984 + 1600

- Redundant sequences and sequences containing 'X' were eliminated to obtain the final dataset.

- Control datasets: Random shuffling of class labels in order to further confirm and support the significance of results.

- Convert all sequence files into FASTA format.

- Non-duplicate sequences between 20 and 120 residues in length considered.

- Sequences below 20 residues in length were eliminated to ensure substantial N-gram and ML analysis.

- Sequences above 120 residues in length were not considered since most AMPs have fewer than 120 residues; even those AMP sequences containing more than 120 residues were eliminated since only a small part of the AMPs sequence may have the antimicrobial activity.

- Sequences were containing unknown, Non-natural amino acids removed.

- Transduction: to balance our positive and negative sets. So that the AMP and Non-AMP datasets had the same number of sequences.

- To achieve transduction, a simple random subset of the larger set was taken so that datasets in each trial had precisely the same number of sequences.

- This technique was implemented in order to disregard the number of sequences as a confounding variable and decrease the probability of overfitting the data to one of the sets.

In this dissertation, in each coming experiment, different datasets were used -from the above datasets- according to the experiment requirements with same exact steps of data arranging.

3.7.1 Datasets length distribution

- **Positive (AMP) dataset:**

1. Positive set:

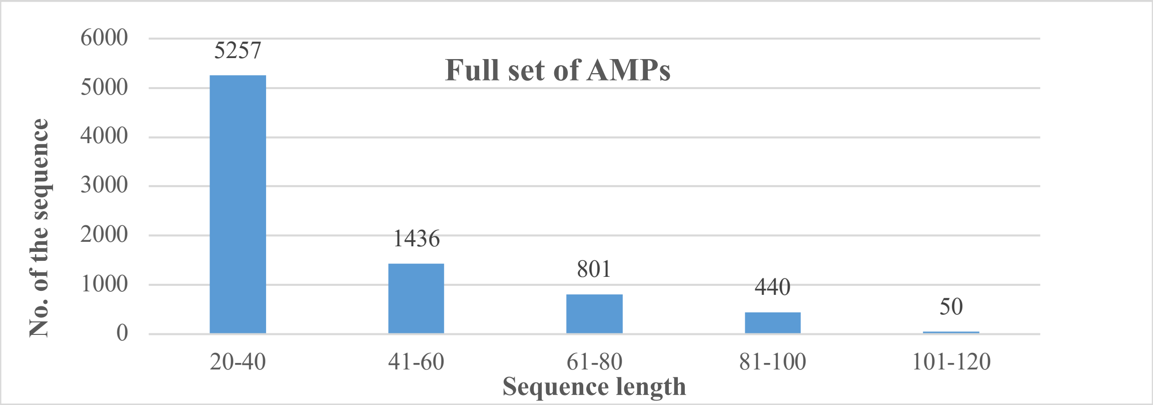


Figure 3.1: The length distribution of the sequences of full AMP dataset.

2. APD set:

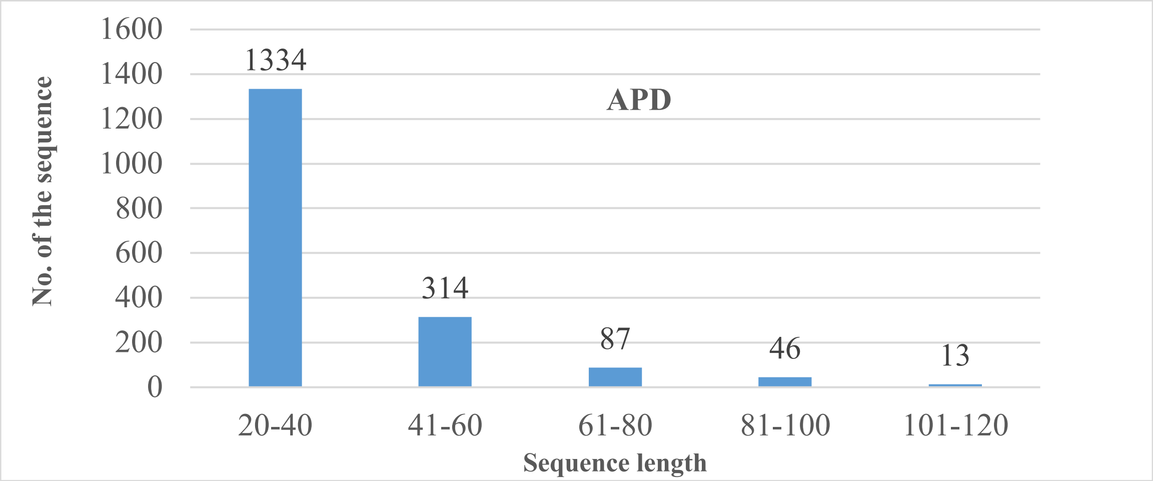


Figure 3.2: The length distribution of the sequences of APD dataset.

- **Negative (Non-AMP) dataset:**
- 1- Neg set1:

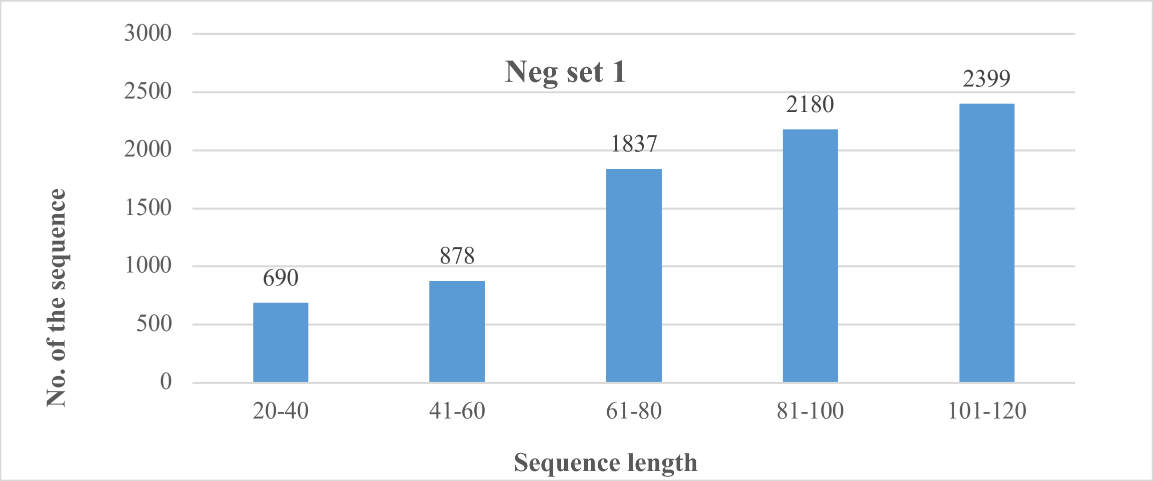


Figure 3.3: The length distribution of the sequences of Neg set1.

2- Neg set2:

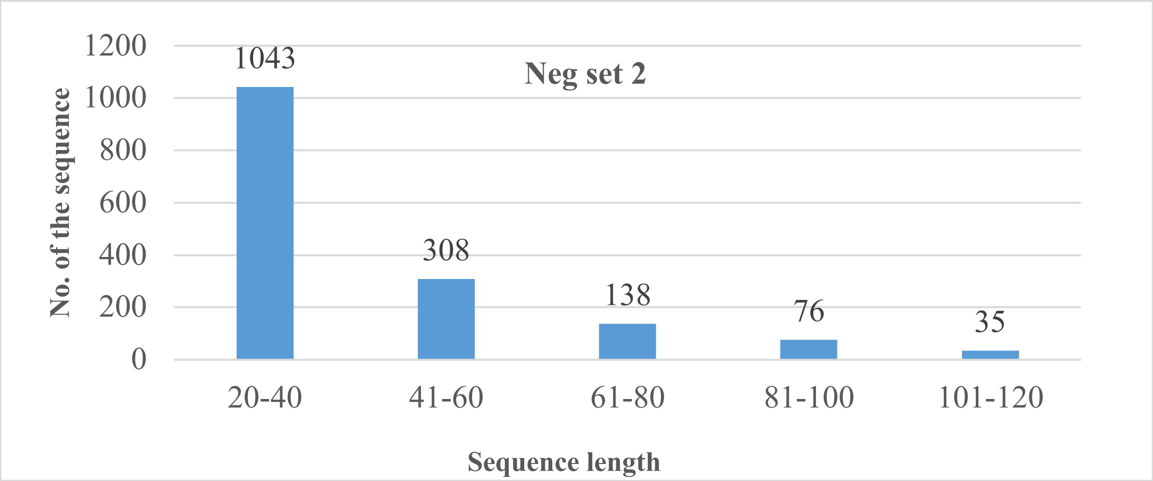


Figure 3.4: The length distribution of the sequences of Neg set2.

3.7.2 Sequence Composition of each Dataset and Subsets

In order to understand how each sequence worked with N-gram algorithm and the ML classification, sequence composition for each dataset and subset in this thesis was performed using three letters alphabet reduction (B, J and U) and unigram.

Table 3.12: Sequence composition of each dataset using 3 letters alphabet reduction and unigram.

Dataset name	No of Seq	B	J	U

Neg set1	7984	221,129	210,149	228,274
Neg set2	1600	20.727	22,502	22,797
Positive set	7984	92,353	120,385	101,763
APD	1794	18,341	27,623	18,996

Table 3.13: Sequence composition of each AMP dataset subclass using 3 letters alphabet reduction and unigram.

Dataset name	No of Seq	B	J	U
ABP	1914	23,509	33,084	23,626
Non-ABP	1914	52,915	50,050	54,495
AVP	1091	10,496	9,628	12,642
Non-AVP	1091	30,003	28,317	31,370
AFP	758	7,437	11,057	8,679
Non-AFP	758	20,839	19,534	21,671

Table 3.14: Sequence composition of each subset using 3 letters alphabet reduction and unigram.

Dataset	Seq No	B	J	U
Positive set (first half)	3992	41,846	57,271	47,769
Positive set (Second half)	3992	50,517	63,114	53,994
Positive set (Odd)	3992	45,858	59,772	50,982
Positive set (Even)	3992	45,884	59,891	50,848
Positive set (Odd)	1600	18,520	23,822	20,118
Positive set (Even)	1600	18,116	23,696	19,816
Positive set 100-120	53	776	1109	964
Positive set 60-80	848	8,727	11,118	9,334
Positive set 40-20 (1)	690	5,913	4,897	4,253
Positive set 40-20 (2)	690	2,953	3,495	3,006
Positive set	6190	74,012	92,762	82,767
APD	1600	15,702	24,064	15,985
Neg set1	1794	49,473	47,342	51,180
Neg set1 (first half)	3992	110,118	104,640	113,181
Neg set1(Second half)	3992	111,011	115,509	115,093
Neg set1(Odd)	3992	110,420	104,561	114,227
Neg set1(Even)	3992	110,709	105,498	114,047
Neg set1 100-120	53	998	1,053	1,069

Neg set1 60-80	848	10,819	10,251	11,018
Neg set1 40-20 (1, 2)	690	3,594	3,545	3,194
Neg set1	1600	26,703	25,941	27,480

Chapter 4: AMPs Sequences Characteristics

4.1 Selection of Alphabet Reduction

After arranging the sequences datasets, alphabet reduction needed to occur. Over 40 different alphabet reduction options of two, three and four letters were used from table 3.2 to figure out which of all these reduced alphabets will give a higher accuracy in order to use it in most of our experiments in this thesis.

4.1.1 Methods

In order to apply each alphabet reduction option to each set of peptide sequences, several reduction programs were written for AMP positive set and Neg set1of each classification, as there are over 40 different reduction options.

In this trial, N-gram of size three (trigram) is used, leading to 27 distinct combinations of amino acid sequences. The frequency of each sequence was calculated by determining how often this three-letter amino acid sequence occurred throughout the entire peptide sequence, and to divide that by the total number of possible N-grams in the peptide sequence.

The output files, each containing N-gram frequencies for all reduced alphabets, were then inputted into WEKA. Seven different types of ML classification models were used; RF, SVM, Bagging, J48, NB, ANN, and AdaBoost. The first three ML; RF, SVM, and Bagging, were used to run all reduced alphabet letters. RF produced the top three accurate results in each two, three, and four reduced letters. These nine, which provide the higher accuracies results, were used to run the other four ML algorithms. Ten-fold cross-validation was used.

4.1.2 Results

The output (table 4.2) below shows the accuracy of each reduced alphabet (2,3, and 4) using the seven ML algorithms. From the results, the three higher accuracies in each group of the reduced alphabets using RF were in table 4.1:

Table 4.1: The higher three accuracies in each reduced alphabet that were used RF algorithms.

	Counting No	No of RA	1	2	3	4	RF%
1	4	2	ACFGHILMVWY	DEKNPQRST			79.9
2	10	2	MFILVAW	CYQHPGTSNRKDE			81.8
3	11	2	IMVLFWY	GPCASTNHQEDRK			82.7
4	14	3	MHVYNDI	QLEKF	WPRGSATC		84.7
5	27	3	CFILMVWY	DEGKNQS	AHPRT		84.7
6	29	3	IMVLFWY	GPCAST	NHQEDRK		87
7	37	4	CLVIM	AFYHT	WGSNR	KDEPQ	85.8
8	46	4	MFILV	ACW	YQHPGTSNRK	DE	87.4
9	47	4	IMVLFWY	G	PCAST	NHQEDRK	87.7

From previous table we conclude the best reduced letter performance in each group:

1- In two reduced alphabet letters: ra2-11: IMVLFWY – GPCASTNHQEDRK

2- In three reduced alphabet letters: ra3-29: IMVLFWY – GPCAST – NHQEDRK

3- In four reduced alphabet letters: ra4-47: IMVLFWY – G – PCAST – NHQEDRK

4.1.3 Discussion

All the previous three reduced amino acid alphabets based on the residue pair counts for BLOSUM50 matrix according to study conducted by Liu and his group on simplified amino acid alphabets based on a deviation of conditional probability from random background. They have detected sequence homology in the SCOP database with the derived coarse-grained BLOSUM similarity matrices. This study verified that the reduced alphabets achieved well preserve information found in the original 20-letter amino acid alphabet (Liu et al., 2002). In this dissertation, according to the accuracy value, ra3-29 and ra4-47 will be optimal to use in most of the experiments.

Table 4.2: The accuracy of each reduced alphabets 2,3, and 4 letters that were used with seven ML algorithms

RA	1	2	3	4	RF	SVM	Bagging	J48	NB	AdBo	ANN
2	STQNGPAHRED	LIFVMYWCK			78.8	64.4	77.4				
2	LVIMCAGSTPFYW	EDNQKRH			80.6	63.5	79.1				
2	ACFGHILMVWY	DEKNPQRST			**79.9**	67.1	78.7	75.9	69.6	69.7	79.9
2	CMFILVWY	AGTSNQDEHRKP			81.7	68.2	80.5				
2	CLVIMAFYWGH	TSNRKDEPQ			79.9	67.1	78.7				
2	AMWLYCFIV	PGHTSDEKNQR			80.8	68.1	79.4				
2	LYMFIVCAWGHTS	DEKNPQR			80.2	63.7	79.0				
2	HCILMVFWYAGSTNR	DEKQP			79.8	61.2	78.4				
2	MFILVAW	CYQHPGTSNRKDE			**81.8**	67.0	80.9	78.7	72.6	73.6	81.7
2	IMVLFWY	GPCASTNHQEDRK			**82.7**	68.5	80.9	78.7	70.8	73.6	82.2
3	LASGVTIPMC	EKRDNQH	FYW		85.4	65.7	83.1				
3	MHVYNDI	QLEKF	WPRGSATC		**84.7**	70.7	83.2	80.1	77.8	78.3	80.5
3	ACFILMVWY	DEKNPQR	GHST		85.3	70.5	82.9				
3	CMFILVWY	AGTSP	NQDEHRK		86.1	71.1	83.6				
3	CLVIMAFYW	GHTS	NRKDEPQ		85.3	70.5	83.0				
3	AMWLYCFIV	PGHTS	DEKNQR		85.5	70.8	83.0				
3	LYMFIV	CAWGHTS	DEKNPQR		86.5	73.6	84.6				
3	CIMFLVWY	AGHTNSP	RDEKQ		85.8	71.0	83.7				
3	CILMVFWY	AGHSTP	DEKNQR		86.1	71.4	83.7				
3	CILMVFWY	AGHST	EKDNRPQ		85.8	71.3	83.2				
3	HCILMVFWY	AGSTNR	DEKQP		85.7	70.6	83.0				
3	CMFILVWY	ATHGPR	DESNQK		84.8	71.7	82.8				
3	ACGPSTWY	DEHKNQR	FILMV		86.5	72.8	84.6				
3	AFILMVWY	C	DEGHKNPQRST		85.6	74.3	84.0				
3	CFILMVWY	DEGKNQS	AHPRT		**84.8**	69.5	82.2	78.2	74.7	75.7	79.3
3	MFILVAW	CYQHPGTSNRK	DE		85.9	74.5	84.8				

4	IMVLFWY	GPCAST	NHQEDRK		**87.0**	74.0	84.8	81.5	78.2	79.4	82.5
4	AGPST	CILMV	DEHKNQR	FYW	86.9	73.1	84.9				
4	ADKERNTSQ	YFLIVMCWH	G	P	86.7	73.7	84.9				
4	LVIMC	AGSTP	FYW	EDNQKRH	86.6	72.6	84.3				
4	AFHTY	CILMV	DEKPQ	GNRSW	86.1	73.0	84.3				
4	CFYW	MLIV	GPATS	NHQEDRK	87.2	75.6	85.2				
4	CMFWY	ILV	AGTS	NQDEHRKP	87.3	75.2	85.1				
4	CLVIM	AFYHT	WGSNR	KDEPQ	**85.8**	73.0	84.2	79.3	79.3	78.6	56.0
4	CALM	VIFW	YGH	TSNRKDEPQ	86.7	74.5	84.7				
4	AMW	LYC	FIV	GHTSDEKNQRP	86.4	73.7	85.5				
4	CALY	MFIV	WGHT	SDEKNPQR	87.0	73.7	85.0				
4	ACIMHT	FLVWY	GNSR	PDEKQ	86.2	73.2	84.4				
4	CILMVFWY	AGHST	EKD	NRPQ	86.7	73.1	85.0				
4	FWY	CILMV	DEGKNQS	AHPRT	86.1	71.8	83.9				
4	AFILMPVW	CGNQSTY	DE	HKR	87.0	75.1	85.2				
4	AFILMVWY	C	DEKR	GHNPQST	87.3	75.6	85.1				
4	MFILV	ACW	YQHPGTSNRK	DE	**87.5**	77.5	85.5	81.8	79.0	80.8	79.4
4	IMVLFWY	G	PCAST	NHQEDRK	**87.7**	75.0	85.7	82.0	77.9	78.9	76.9

4.2 AMPs Classification

In this trial, we want to classify AMPs using a straightforward, sequence-based method that involved alphabet reduction, N-gram analysis with frequency, and ML. More sophisticated future goals of this study are classification between classes of AMPs and help the researcher to understand the AMPs' features in order to create of an artificial set of AMPs. Success rates in this experiment for some classification trials were comparable to that of previous studies by researchers conducting experiments with tangible AMPs in medical laboratories.

4.2.1 Methods

The 20-letter amino acid alphabet was reduced to fewer letters of an alphabet to simplify and quicken the machine learning process. From what we conclude from the previous study, ra3-29 and ra4-47 reduced alphabets were used. N-gram of size three or trigram is used, leading to 27 distinct combinations of amino acid sequences. After that, all the seven ML classification models were used. In order to apply each previous step N-gram Classification application were used.

The datasets applied in this experiment as shown in table 4.3:

1- 7984 of AMPs vs. 7984 Neg set1.

2- 1914 of antibacterial peptides (ABPs) vs. 1914 Non antibacterial peptides (Non-ABPs).

3- 1091 of antiviral peptides (AVPs) vs. 1091 Non antiviral peptides (Non-AVPs).

4- 758 of antifungal peptides (AFPs) vs 758 Non antifungal peptides (Non-AFPs).

5- 1091 of antibacterial peptides (ABPs) vs. 1091 of antiviral peptides (AVPs).

6- 758 of antibacterial peptides (ABPs) vs. 758 of antifungal peptides (AFPs).

7- 758 of antiviral peptides (AVPs) vs. 758 of antifungal peptides (AFPs).

8- Control dataset.

Table 4.3: Number of AMP sequences in each class-specific AMP set.

Antimicrobial peptides (AMPs)	**Neg set1**	**Antibacterial Peptides (ABPs)**	**Antiviral peptides (AVPs)**	**Antifungal Peptides (AFPs)**
7984	7948	1914	1091	758

- ra3-29 and ra4-47 reduced alphabets were used in all subclasses.
- ra3-29 were used in control.
- N-gram size: Trigram.
- Convert all sequence files into FASTA format.
- Non-duplicate sequences between 20 and 120 residues in length considered.
- Sequences were containing unknown, Non-natural amino acids removed.
- Transduction: to balance our positive and negative sets.
- All of these pervious steps done by N-gram Classification application.
- ML: RF, SVM, bagging, J48, NB, AdBo, and ANN.

4.2.2 Results

The classification of AMPs against Non-AMPs was successful. Models reached a maximum accuracy of 87.7% using frequency of N-gram analysis, alphabet reduction option 47, and the RF model with 10 trees cross-validation. The 10 trees cross-validation is what were used in all of the trials in this study. Label randomization was utilized as a control to ensure the fidelity of the dataset. The model of the control accuracy was 50.1%, 49.2% using RF and NB respectively. Implying that all models resulting from the non-randomized label experiments for this dataset yielded reliable results as we see in table 4.4.

Classification using more specific subclasses of AMPs was conducted next. First, classification of ABPs against Non-ABPs AMPs achieved a maximum accuracy of 86.8% using frequency N-gram analysis, alphabet reduction option 47, and RF model, while with bagging algorithm 84.3%. Second, classification of AVPs against Non-AVP AMPs achieved an accuracy of 92.7% and 92.3% using frequency N-gram analysis, alphabet reduction option 47 and 29 respectively, and with RF model.

This experiment also consisted of many other successful trials. A third successful trial classifying AFPs against Non-AFPs yielded a maximum accuracy of 89.4% using N-gram frequency analysis, alphabet reduction option 47, and the RF model with 10 trees and 10-fold cross-validation. A fourth successful trial classifying ABPs against AVP AMPs had a maximum accuracy of 84.4% using N-gram frequency analysis, alphabet reduction option 29, and the RF model. A fifth successful trial the classification between AVPs against AFPs and achieved a maximum accuracy of 82% using N-gram frequency analysis, alphabet reduction option 29, and the RF model as well.

Table 4.4: Performing classification tests, alphabet reductions, algorithms, and their respective accuracies. RF with 10-fold cross validation seems to be the best test for classifying AMPs.

N-gram	RA	Classification test	RF %	SVM %	BAGGING %	J48 %	NB %	ADBO %	ANN %
3	ra3-29	AMPs vs Non-AMPs	87	74	84.8	81.5	78.2	79.4	82.5
3	ra4-47	AMPs vs Non-AMPs	87.7	75	85.7	82	77.8	78.9	76.9
3	ra3-29	ABP (1914) vs Non-ABP	86.2	80.8	83.5	80	80	78.7	82.3
3	ra4-47	ABP vs Non-ABP	86.8	82.2	84.3	81.3	81.1	78.1	79.8
3	ra3-29	AVP (1091) vs Non-AVP	92.3	71.9	89.6	85.7	83.6	84.6	86.2
3	ra4-47	AVP vs Non-AVP	92.7	75.9	90.1	87	83.2	85.8	86.9
3	ra3-29	AFP (758) vs Non-AFP	88.7	79.2	85.9	82.7	80.9	80.9	82.2
3	ra4-47	AFP vs Non-AFP	89.4	80.3	87.5	82.4	81.9	81.9	84.4
3	ra3-29	ABP vs AVP	84.4	78	80.8	78.3	73.2	75.2	79.2
3	ra3-29	ABP vs AFP	51.8	63.2	59.1	62.9	64	64.5	55.6
3	ra3-29	AVP vs AFP	82	74.7	78.8	77	71.4	74.2	78.5
3	ra3-29	CONTROL	50.1	50.3	49.6	50.1	49.2	50.2	50.4

4.2.3 Discussion

However, this trial contained less successful results as well. ABPs against AFP AMPs trial obtaining accuracy below 70%. The explanation of this inconsistency likely lies in many inherent similarities between AFPs and ABPs. First, some AMPs in the dataset may have been both an AFP and an ABP, slightly misrepresenting the model and lowering the accuracy. Moreover, some fungi, such as yeast, are unicellular and can reproduce asexually, similar to the bacteria. Likewise, bacteria and fungi share the role of supporting multiple food webs by bolstering the nutritive properties of the soil in the ecosystem.

However, there were two trends present:

1- RF significantly outperforms each of the other six learning algorithms. This may have occurred because RF utilizes several unique decision trees, each with its own parameters. Due to the mechanism of the RF algorithm, a reduced chance of overfitting was present, authenticating the obtained accuracies. On the other hand, the next most accurate model was the Bagging model, as shown in figure 4.1.

2- Besides, alphabet reduction 47 most often yielded the highest classification accuracies. This finding implies that the 4-cluster alphabet is optimal for N-gram frequency analysis and ML. A 4-cluster alphabet reduces the alphabet so that amino acid sequences are simple enough for efficient ML but complicated enough to the extent that information losses in the original sequences are minor. Furthermore, alphabet reductions 29 yielded to high classification accuracy above 86% in three letters reduced options for the sequence-based method of analysis utilized in this study.

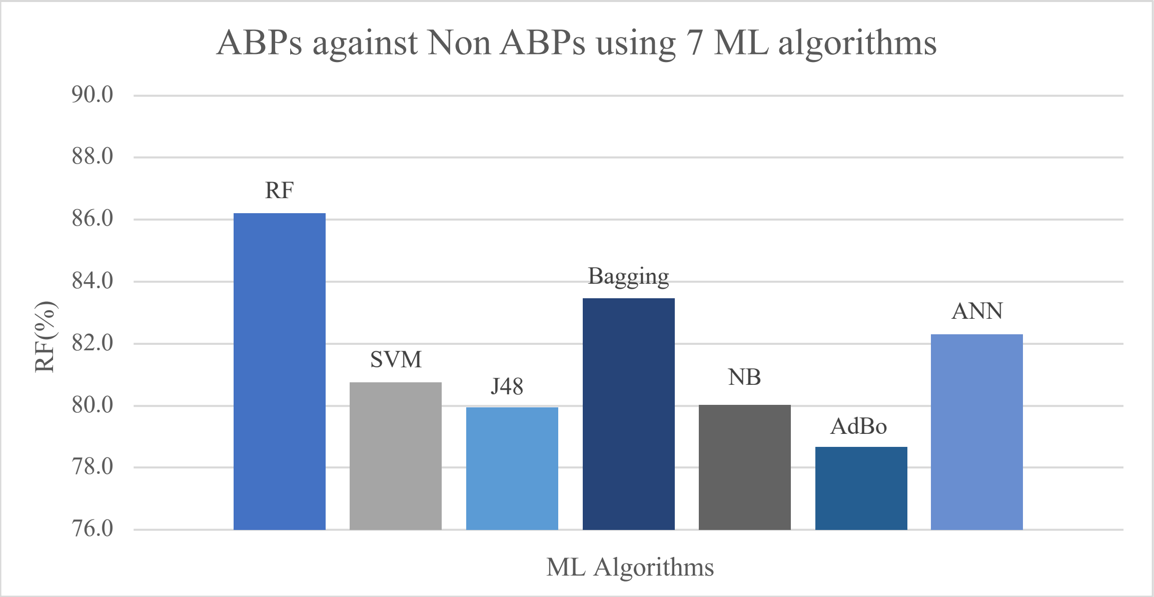


Figure 4.1: RF outperforms each of the other six learning algorithm in ABPs against Non ABPs using ra3-29.

4.3 Classification of AMPs with Different N-grams Size

To better understand how the size of N-grams could affect the accuracy of the ML, with several numbers of alphabet reduction in three several trials. In this experiment, different numbers of N-grams (1 to 4) and the different numbers of reduced alphabets (2, 3, and 4) on the same dataset were used.

4.3.1 Methods

The datasets applied in this experiment:

1- 1914 of ABPs vs 1914 Non-ABPs.

2- 1091 of AVPs vs 1091 Non-AVPs.

3- 758 of AFPs vs 758 Non-AFPs.

-Reduced alphabet letters:

1- ra2-11: IMVLFWY – GPCASTNHQEDRK

2- ra3-29: IMVLFWY - GPCAST - NHQEDRK

3- ra4-47: IMVLFWY – G – PCAST – NHQEDRK

-N-gram size:

1- N-gram of size one: $3^1 = 3$ possible combinations.

2- N-gram of size two: $3^2 = 9$ possible combinations.

3- N-gram of size three: $3^3 = 27$ possible combinations.

4- N-gram of size four: $3^4 = 81$ possible combinations.

-Three different ML classification models were used: RF, SVM, and J48.

-The next applied steps were mentioned in detail in the previous experiment.

4.3.2 Results

The classification of ABPs against Non-ABPs was successful. Models achieved a maximum accuracy of 86.8% and 87.2% with three and four reduced letters respectively, all using size 4 of N-gram analysis, and the RF model with ten trees cross-validation. According to the results from table 4.6 and 4.7, there is no significant change in the accuracies when using three or four sizes of N-gram. On the other hand, there is a significant drop when using size one, and two of N-grams with two letters reduced alphabets as shown in table 4.5.

Classification of AVPs against Non-AVP AMPs achieved an accuracy of 93.2% and 93% using N-gram size 4, alphabet reduction options 29 and 47, respectively, and with RF model while the model reaches 92.3% and 92.8% with N-gram size 3, and three and four reduced alphabets respectively.

The other successful trials were classifying AFPs against Non-AFPs yielded a maximum accuracy of 89.9% using size 4 of N-gram, alphabet reduction option 47, and the RF model with ten trees and 10-fold cross-validation, and 82.7% using J48. Moreover, when using size 1 of N-grams with two letters of alphabet reduction, the accuracy drops to 58.6% using the SVM model, as shown in table 4.5.

Trial 1: ra2-11

Table 4.5: Performing classification tests, alphabet reductions of ra2-11, N- grams size 1 to 4, 3 algorithms, and their respective accuracies.

N-grams	Subclasses	RF%	SVM%	J48%
1	ABP vs Non-ABP	69.6	58.6	60.8
2		75.3	60.0	65.5
3		79.3	72.9	76.4
4		82.4	72.9	78.7
1	AVP vs Non-AVP	76.7	50.6	53.4
2		82.9	63.8	76.3
3		86.1	67.2	82.3
4		89.6	68.4	85.7
1	AFP vs Non-AFP	72.4	61.9	61.8
2		78.9	61.8	73.1
3		82.9	75.7	79.2
4		86.3	77.0	84.0

Trial 2: ra3-29

Table 4.6: Performing classification tests, alphabet reductions of ra3-29, N- grams size 1 to 4, 3 algorithms, and their respective accuracies.

N-grams	Subclasses	RF%	SVM%	J48%
1	ABP vs Non-ABP	71.8	68.9	68.7
2		77.6	70.3	73.3
3		86.2	80.8	80.0
4		86.8	81.4	80.5
1	AVP vs Non-AVP	74.7	53.9	69.2
2		82.4	67.5	78.3
3		92.3	71.9	85.7
4		93.2	75.6	87.6
1	AFP vs Non-AFP	72.8	68.7	71.2
2		78.6	68.7	73.9
3		88.7	79.2	82.7
4		89.4	80.2	82.7

Trial 3: ra4-47

Table 4.7: Performing classification tests, alphabet reductions of ra4-47, N- grams size 1 to 4, 3 algorithms, and their respective accuracies.

N-grams	Subclasses	RF%	SVM%	J48%
1	ABP VS Non-ABP	75.5	72.7	75.3
2		82.8	74.6	77.7
3		86.8	82.2	81.4
4		87.2	82.2	79.8
1	AVP VS Non-AVP	80.1	56.4	76.9
2		89.8	71.4	85.2
3		92.8	75.9	87.0
4		93.0	78.1	88.0
1	AFP VS Non-AFP	79.6	72.6	76.7
2		85.4	73.8	79.5
3		89.5	80.3	82.5
4		89.9	80.5	82.7

4.3.3 Discussion

This study showed less successful results as well, with maximum accuracies ranging between 60% and 75% when using an N-gram size of 1 with two letters reduced alphabets. Like the previous experiment, RF significantly outperforms each of the other learning algorithms. On the other hand, the next most accurate model was the J48 model.

Throughout all classification tests, the unigram and bigram with two-cluster alphabet reductions never achieved the highest accuracies for a given classification. However, using unigram and bigram resulted in a severe loss of information stored in the original amino acid sequence when combined with two letters reduced alphabet. The two-cluster alphabets were grouped mainly by hydrophobicity, an important feature for AMPs. Thus, alphabet reductions with just two clusters were always outperformed by other alphabets.

Also, the N-gram size of three and four most often yielded to a higher classification accuracy without significant differences between them. This finding implies that trigram with either three or four letters of alphabet reduction and RF is optimal for AMPs classification.

N-gram size of four makes the amino acid sequences simple enough and efficient for ML without losing the original sequence information. Likewise, trigram yielded to high classification accuracy above 85% in three and four letters reduced alphabet see figure 4.2.

This result shows that even trigram are viable options for the sequence-based method of analysis utilized in this research.

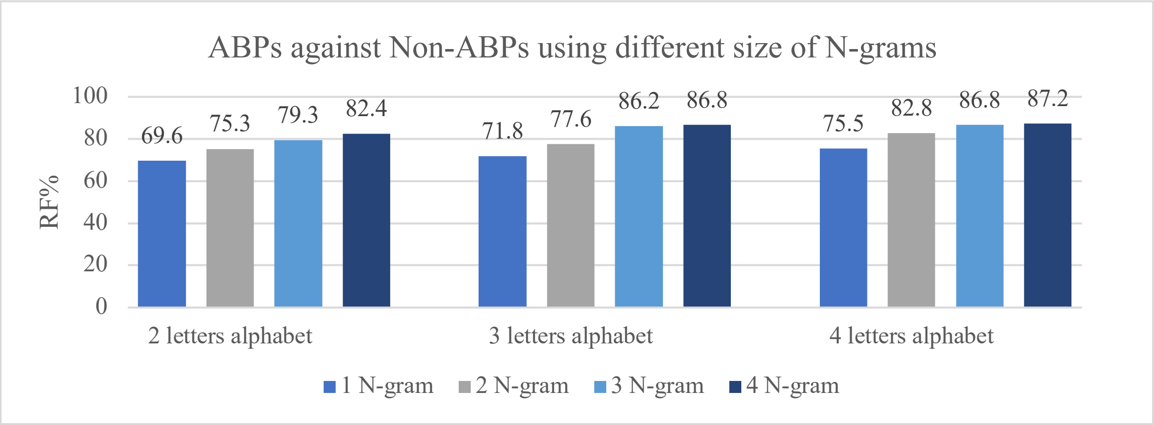


Figure 4.2: ABPs vs Non-ABPs using different size of N-grams (1 to 4) with 2, 3 and 4 reduced alphabet letters by RF ML.

4.4 Sums of N-grams

In this experiment, sums of a different number of N-grams (2-3-4) with three letters reduced alphabet in the ABPs dataset. In order to figure out if there any hidden information that could be extracted from different sizes of N-gram. Also, if the accuracy will increase when adding several sizes of N-grams together.

4.4.1 Methods

The datasets applied in this experiment:

-1914 of ABPs vs. 1914 Non-ABPs.

-3 alphabets reduce letters ra3-29

- N-gram size:

- N-gram of size one: $3^1 = 3$ possible combinations.
- N-gram of size two: $3^2 = 9$ possible combinations.
- N-gram of size three: $3^3 = 27$ possible combinations.

-The result of N2 will added to the result of N3.

(N3 + N2) = 27 + 9 = 36 possible combinations.

-Then, the result of N1 will added to the result of N3.

(N3 + N1) = 27 + 3 = 30 possible combinations.

-Finally, the result of N2 and N1 will added to the result of N3.

(N3 + N2 + N1) = 27 + 9 + 3 = 39 possible combinations.

-Three different ML classification models were used: RF, SVM, and J48.

-The next applied steps were mentioned in detail in 4.2 experiment.

4.4.2 Results

From table 4.8, we can realize that there is no significant change when sums the N-grams together. N-gram size of three still give us the highest accuracy around 85.8% using RF.

However, there was a slight increase in SVM and J48 when adding N-gram size of one to N-gram size of 3.

Table 4.8: Summation of N-grams of ABPs vs Non-ABPs using ra3-29 and 3 ML

N-grams	RF%	SVM%	J4%
N3	85.8	80.8	80.0
N3 + N1	85.5	81.2	80.2
N3 + N2	85.7	80.9	79.1
N3 + N2 + N1	85.5	81.3	79.8

4.4.3 Discussion

Typically, N-gram size of 3 outperforms the other sizes of N-grams with RF, which means that N3 can hold all the amino acid sequences information that already found in N1 and N2. On the other hand, using SVM with sums of N1, N2, and N3 gives higher accuracy than N3 alone but with no significant differences between both accuracies. Moreover, N2 + N3 with J48 gives higher accuracy than N3 without any substantial differences in the accuracy rates.

4.5 Different letters of alphabet reduction

Amino acid alphabet reduction assists in cluster-specific amino acids together based on features, diminishing the 20 distinct amino acids down to the number of clusters. This reduction impressively aids in the calculation of N-gram frequencies as demonstrated in all the previous experiments. In this study, we want to understand how the reduction of amino acids affect the accuracy of the model when diminishing the letters from 20 to 3 letters only.

4.5.1 Methods

The datasets applied in this experiment:

-1914 of ABPs vs. 1914 Non-ABPs.

-5 alphabets reduce were used:

1- ra3-29: IMVLFWY – GPCAST – NHQEDRK

2- ra6-81: IMVL – FWY – G – P – CAST – NHQEDRK

3- ra10-110: IMV – L – FWY – G – P – C – A –STNH – RKQE - D

4- ra15-149: IMV – L – F – WY – G - P – C – A – S – T – N – H – RKQ – E - D

5- ra20-177: I – M – V – L – F – W - Y – G - P – C – A – S – T – N – H – R – K - Q – E – D

-N-gram of size: Three

-Three different ML classification models were used: RF, SVM, and J48.

-The next applied steps were mentioned in detail in 4.2 experiment.

4.5.2 Results

The results in table 4.9 show, diminishing the 20 letters amino acid to 3 letters is still holding most of the sequence information and complexity with accuracy around 85% with 3 N-gram and RF. There are around 3% and 6% differences from decreasing the amino acid from 20 letters to 3 letters only with RF and J48 respectively.

Table 4.9: ABPs vs Non-ABPs using 5 different alphabets reduction letters with 3 N-gram and 3 ML.

RA	RF%	SVM%	J48%
ra3-29	85.9	80.7	80
ra6-81	87.6	83	81.5
ra10-114	87.7	85.3	80.5
ra15-149	87.8	84.9	81.5
ra20-177	88.5	79.9	86.8

4.5.3 Discussion

Throughout all five classification tests, the three-cluster alphabet reductions achieved a good percentage of accuracy when comparing to others. Alphabet reduction 20 most often yielded the highest classification accuracies in RF and J48 because of its hold all the sequence info. While with SVM, the accuracy of 20 letters almost equal to 3 letters, thus, maybe the SVM use generally “black-box” models for predictions, which depends on the data only through dot-products.

As stated previously, this finding proof that the 3-cluster alphabet is optimal for N-gram analysis and ML. A 3-cluster alphabet is simple enough for efficient ML and holds most of the data that encoded in the original sequences. Besides, this finding shows that even alphabet reductions with 6 or 10 clusters are viable options for the sequence-based method of analysis utilized in this study.

4.6 Feature Selection

Attribute or feature selection is a method that automatically searches for the best subset of attributes dataset to achieve the highest accuracy. The benefits of performing feature selection on this study to improves the model accuracy by reduce the misleading data, decrease overfitting by minimizing the opportunity to make decisions based on noise, and to lessen training time to train algorithms faster.

WEKA offers an attribute selection tool which is divided into two parts:

- Attribute Evaluator: Method to evaluate the attribute subsets.

- Search Method: Method to search for possible space for the subsets.

4.6.1 Methods

In this experiment, two attributes selection of WEKA (CorrelationAttributeEval and InfoGainAttributeEval) were used.

1- CorrelationAttributeEval: This is to evaluate the value of an attribute by assessing the correlation between it and the class.

2- InfoGainAttributeEval: This is to evaluate the value of an attribute by assessing the data gain with respect to the class.

These two attributes are the most suitable to use on AMPs classification models. In the next section, both of CorrelationAttributeEval and InfoGainAttributeEval can only be used with a Ranker Search Method, which assesses each attribute and lists the results in rank order.

The datasets applied in this experiment:

-7984 AMPs vs. 7984 Neg set1.

-4 alphabets reduce were used:

1- ra3-29: IMVLFWY - GPCAST – NHQEDRK

2- ra4-47: IMVLFWY – G – PCAST – NHQEDRK

3- ra10-110: IMV - L – FWY – G – P – C – A - STNH – RKQE – D

4- ra20-177: I – M - V – L – F – W - Y – G - P – C – A – S – T – N – H – R – K - Q – E – D

-3 letters reduced alphabet with N-gram of size three: $3^3 = 27$ possible combinations.

-4 letters reduced alphabet with N-gram of size three: 4^3 = 64 possible combinations.

-10 letters reduced alphabet with N-gram of size three: 10^3 = 1000 possible combinations.

-20 letters reduced alphabet with N-gram of size three: 20^3 = 8000 possible combinations.

-Attribute evaluator: CorrelationAttributeEval and InfoGainAttributeEval.

-Search method: Ranker.

-The top 2,3,5,10 and 20 of higher ranked attributes (N-gram value) were calculated.

-ML classification models were used: RF, SVM and J48.

-The next applied steps were mentioned in detail in 4.2 experiment.

4.6.2 Results

After the selecting the model, we run the attribute evaluator with the Ranker search method. The attribute selection output arranged the dataset according to higher N-grams frequency on the dataset as an example, the output from N-grams size of three with three letters reduced alphabet shown below in table 4.10, which shows the top 20 of higher N-gram value in sequences of the model.

Table 4.10: The top 20 higher N-grams in the all sequences using ra3-29.

	Ranked attributes					
	CorrelationAttributeEval			**InfoGainAttributeEval**		
1	0.28998	1	BBB	0.229	3	BBU
2	0.21416	5	BJJ	0.22	13	JJB
3	0.20904	7	BUB	0.219	19	UBB
4	0.18697	11	JBJ	0.216	7	BUB
5	0.18567	13	JJB	0.214	6	BJU
6	0.16016	27	UUU	0.213	8	BUJ
7	0.14661	21	UBU	0.208	22	UJB
8	0.1377	4	BJB	0.208	1	BBB
9	0.12874	24	UJU	0.205	11	JBJ
10	0.12744	15	JJU	0.199	16	JUB
11	0.12087	14	JJJ	0.196	23	UJJ
12	0.10709	6	BJU	0.194	26	UUJ
13	0.08658	23	UJJ	0.192	5	BJJ
14	0.07828	17	JUJ	0.191	15	JJU
15	0.06537	20	UBJ	0.183	18	JUU
16	0.06284	26	UUJ	0.182	12	JBU
17	0.05811	22	UJB	0.179	17	JUJ
18	0.04464	9	BUU	0.178	27	UUU
19	0.04091	19	UBB	0.171	2	BBJ
20	0.0378	16	JUB	0.169	20	UBJ

The final result showed, as all previous studies, RF outperforms all other ML and achieved accuracy around 86% when using the top 20 in both attribute selection with three alphabet reduction. SVM displayed the lowest accuracy results of around 78%. Generally, we can say the accuracy drop when we increase the number of reduced alphabet letters in both feature selection. On the other hand, the accuracy rises by increasing the rank number from 2 to 20. The is no significant change between rank 10 and 20. When using three and four alphabet reduction, the difference in the accuracy between rank 20 and using the full dataset is minor. While it is considerable when using ten and twenty alphabet reduction in all of ML, as indicated in table 4.11.

Table 4.11: The accuracy results of using two attribute evaluators: CorrelationAttributeEval and InfoGainAttributeEval, 3 different ML, 3,4 10, 20 reduced alphabet letters and size of 3 N-gram against AMPs and Neg set1 on top 2,3,5,10 and, 20 of N-gram frequency value.

RA	**ML**	**CorrelationAttributeEval**					**InfoGainAttributeEval**					**Full set**
		2	**3**	**5**	**10**	**20**	**2**	**3**	**5**	**10**	**20**	**27**
ra3-29	RF	79.7	81.5	83	85.2	86.3	79	79.6	82	85.2	86.3	87
ra3-29	J48	77.5	79.5	80.5	81.4	81.1	75.5	77.7	79.4	81	81.3	81.5
ra3-29	SVM	76.6	78.3	78.2	80.1	78.4	73.9	75.8	73.7	80.5	78.3	79.7
												64
ra4-47	RF	80	81.2	82.9	85.1	86.5	78.8	79.9	81.5	85.2	86.7	87.7
ra4-47	J48	78.3	79.9	80.5	82.4	82.3	77.8	79.4	80.5	81.4	81.2	82
ra4-47	SVM	67.7	67.5	71.8	72	73.1	50.7	59.4	59.4	68.8	72.1	75
												1000
ra10-114	RF	64.4	64.6	73	77.7	83	74.2	77.1	79.3	82.5	84.9	88.4
ra10-114	J48	64.5	64.7	73.2	78.5	82.7	73.8	76.9	79.7	81	80.8	84.1
ra10-114	SVM	63.3	60.6	64.6	67.3	70.9	52.7	53.1	54.5	58	61.3	84.3
												8000
ra20-177	RF	54	55.6	58	62.6	67.2	54.1	55.4	56.8	62.2	70.1	89.6
ra20-177	J48	54	55.6	58	62.8	67.4	54.1	55.4	56.9	62.4	70	85.2
ra20-177	SVM	52.6	51.2	53.1	59.3	66.6	50	51.8	53.7	58.5	59.3	81.3

4.6.3 Discussion

Discover feature selection to use it in AMPs and Non-AMPs classification as a suite of methods that can increase the model accuracy, decrease model training time, and reduce overfitting. From table 4.10 above, we can illustrate that each selection attribute has different N-grams rank with a different value according to how these attributes evaluate the model.

Taking only the top two (BBB and BJJ) on CorrelationAttributeEval or (BBU and JJB) on InfoGainAttributeEval causes severe loss of information that encoded within the peptide sequence. While using the top 20 of the highest-ranked N-grams of the model achieved better accuracy.

These results disclose that each of the N-gram in the sequence plays a role of holding some information of AMP characteristics that encoded within the sequences and discarded some of these N-grams could affect the model accuracy and cause the AMP to drop their antimicrobial activity.

4.7 Gaps Insertion Features

In this experiment, we introduce novel gap insertion features between amino acid sequences. This will help the researcher to go beyond basic composition and recognize direct correlations between neighboring and non-neighboring amino acid motifs within AMP sequences.

Many different trials of the gap insertion feature approach were made in this part. They were starting from zero gap to ten gaps inserted between amino acid with different sizes of N-gram and alphabet reduction.

4.7.1 Methods

Our method in this experiment is to understand the connections between neighboring and non-neighboring amino acids AMPs sequences and how these relationships will affect the accuracy of different ML algorithms.

Trial 1:

Introducing zero gap with N-gram size one using three and four alphabet reduced letters for AMPs and Non-AMPs datasets. Here no more gaps could add because there is only one N-gram.

Trial 2:

Introducing zero to five gaps with N-gram size two using three and four alphabets reduced letters for AMPs and Non-AMPs datasets as explained below. 0 means zero gap, 1 means one amino acid in the sequences in two N-gram, and X means gap.

- 0
- 1 X 1
- 1 XX 1
- 1 XXX 1
- 1 XXXX 1
- 1 XXXXX 1

Trial 3:

Introducing zero to five gaps with N-gram size three.

- 0
- 2 X 1
- 2 XX 1
- 2 XXX 1
- 2 XXXX 1
- 2 XXXXX 1

Trial 4:

Introducing zero to five gaps with N-gram size four.

- 0
- 2 X 2
- 2 XX 2
- 2 XXX 2
- 2 XXXX 2
- 2 XXXXX 2

Trial 5:

Introducing zero to 10 gaps with N-gram size three.

- 0
- 1 X 1 X 1
- 1 XX 1 XX 1
- 1 XXX 1 XXX1
- 1 XXXX 1 XXXX 1
- 1 XXXXX 1 XXXXX 1

Trial 6:

Introducing zero to 6 gaps with N-gram size four.

- 0
- 2 X 1 X 1
- 2 X 1 XX 1
- 2 X 1 XXX 1
- 2 XX 1 X 1
- 2 XX 1 XX 1
- 2 XXX 1 XXX 1
- 1 X 1 X 1 X 1
- 1 XX 1 X 1 X1

Trial 7:

Introducing zero and one gaps with N-gram size three using three and four alphabet reduced letters for AMPs, ABPs, AVPs, and AFPs datasets.

- 0
- 2 X 1
- 1 X 2

The datasets applied in this experiment:

-7984 AMPs vs. 7984 Neg set1.

-1914 of ABPs vs. 1914 Non-ABPs.

-1091 of AVPs vs. 1091 Non-AVPs.

-758 of AFPs vs. 758 Non-AFPs.

-Letters of alphabets reduce: ra3-29 and ra4-47

-N-gram size: 1, 2, 3 and 4

-Three different ML classification models were used: RF, SVM, and J48.

-The next applied steps were mentioned in detail in 4.2 experiment.

4.7.2 Results

Insertion of gaps within the sequence-based model based on amino acid alphabet reduction, N-gram frequency calculation, and ML classifiers is a novel experiment. The primary goal of this study was to successfully understand the connection between AMPs' amino acid sequences and how the structure of amino acid of the AMPs could also play roles of giving any peptide the antimicrobial characteristics.

The results from all seven trials were almost the same. Like all previous experiments, the N-gram size of one with no gap showed lower accuracy of 76.1% with RF on trial one as in table 4.12. While, the accuracy of RF increased to 85.2%, 87.7%, 88.1% when N-gram size of two, three, and four, respectively, were used. SVM and J48 displayed lower accuracy than RF in all numbers of N-gram.

The highest accuracy of AMPs resulted from four size N-gram and ra4-47 with RF around 88.3% by insertion of (1 XX 1 X 1 X1) gap on trial 6. In AMPs subclass, AVP obtained percent of classification rate of 93.7% by insertion of (1 X 2) gap. While 87.2% and 89.6 in ABPs and AFPs respectively, with a gap (2 X 1) on trial 7, see table 4.18.

The primary comparison in this study was between the three ML classifiers and how the gaps insertion affects the accuracy rates of the models. Increasing gaps between amino acids rise the accuracies, especially in the SVM algorithm. RF and J48 showed a slight increase in accuracy rates. The SVM accuracy rates increased from 67.1% to 84.5% in two N-gram, from 75% to 85.3% in three N-gram, and from 77.3% to 85.5% in four N-gram with ra4-47. On the other hand, the rates of the accuracy raised less than 5% in both RF, and J48 classifiers with an N-gram size of two see table 4.13. However, the accuracy rates drop for less than 1% in both RF and J48 ML when increasing the numbers of gaps between the amino acids, see tables 4.14, 4.15, 4.16, and 4.17.

Trial 1:

Table 4.12: The accuracy results of using zero gap with, 3,4 reduced alphabet letters, size of 1 N-gram and 3 different ML against AMPs and Neg set1.

N-gram	Class	Gaps	RF%		SVM%		J48%	
	7984 Seq		ra3-29	ra4-47	ra3-29	ra4-47	ra3-29	ra4-47
1	AMP	0	74.5	76.1	63.3	65.6	70	73.4

Trial 2:

Table 4.13: The accuracy results of using 1 to 5 gaps with, 3,4 reduced alphabet letters, size of 2 N-gram and 3 different ML against AMPs and Neg set1.

N-gram	Class	Gaps	RF%		SVM%		J48%	
	7984 Seq		ra3-29	ra4-47	ra3-29	ra4-47	ra3-29	ra4-47
2	AMP	0	80.4	85.2	67.1	67.1	74.4	79
2	AMP	1 X 1	82.3	85.8	77.3	77.3	78.5	80.8
2	AMP	1 XX 1	82.2	85.3	82.7	82.7	77.2	79.6
2	AMP	1 XXX 1	83.3	85.4	83.5	83.5	78.2	79.5
2	AMP	1 XXXX 1	83.4	85.5	84.1	84.1	78.8	80.2
2	AMP	1 XXXXX 1	84.3	85.7	84.5	84.5	79.6	80.6

Trial 3:

Table 4.14: The accuracy results of using 1 to 5 gaps with, 3,4 reduced alphabet letters, size of 3 N-gram and 3 different ML against AMPs and Neg set1.

N-gram	Class	Gaps	RF%		SVM%		J48%	
	7984 Seq		ra3-29	ra4-47	ra3-29	ra4-47	ra3-29	ra4-47
3	AMP	0	87	87.7	74	75	81.5	82
3	AMP	2 X 1	86.9	87.9	77.8	77.3	81.4	81.7
3	AMP	2 XX 1	86.7	87.5	83.4	83.1	80.8	81.4
3	AMP	2 XXX 1	86.9	87.7	84.2	84.2	80.9	81.2
3	AMP	2 XXXX 1	86.7	87.4	84.8	85	80.9	81.5
3	AMP	2 XXXXX 1	86.7	87.4	84.8	85.3	81.3	81.4

Trial 4:

Table 4.15: The accuracy results of using 1 to 5 gaps with, 3,4 reduced alphabet letters, size of 4 N-gram and 3 different ML against AMPs and Neg set1.

N-gram	Class	Gaps	RF%		SVM%		J48%	
	7984 Seq		ra3-29	ra4-47	ra3-29	ra4-47	ra3-29	ra4-47
4	AMP	0	87.9	88.1	74.8	77.3	81.8	82.8
4	AMP	2 X 2	88	88.1	78.4	79.2	81.5	82.4
4	AMP	2 XX 2	87.9	87.9	82.9	82.6	81.3	81.7
4	AMP	2 XXX 2	87.8	87.9	84.3	84.5	81.1	81.5
4	AMP	2 XXXX 2	87.6	88	85	85	81.4	81.8
4	AMP	2 XXXXX 2	87.6	88	84.8	85.5	81.2	81.8

Trial 5:

Table 4.16: The accuracy results of using 1 to 10 gaps with, 3,4 reduced alphabet letters, size of 3 N-gram and 3 different ML against AMPs and Neg set1.

N-gram	Class	Gaps	RF%		SVM%		J48%	
	7984 Seq		ra3-29	ra4-47	ra3-29	ra4-47	ra3-29	ra4-47
3	AMP	0	87	87.7	74	75	81.5	82
3	AMP	1 X 1 X 1	86.3	87.8	82.3	82.1	81.3	81.7
3	AMP	1 XX 1 XX 1	86.4	87.1	84.3	84.4	80.4	80.9
3	AMP	1 XXX 1 XXX1	86.7	87.4	84	84.7	80.7	81.4
3	AMP	1 XXXX 1 XXXX 1	86.7	87.6	84.5	85	81.8	81.6
3	AMP	1 XXXXX 1 XXXXX 1	86.8	87.5	84.8	85.2	81.6	81.5

Trial 6:

Table 4.17: The accuracy results of using 1 to 6 gaps with, 3,4 reduced alphabet letters, size of 4 N-gram and 3 different ML against AMPs and Neg set1.

N-gram	Class	Gaps	RF%		SVM%		J48%	
	7984 Seq		ra3-29	ra4-47	ra3-29	ra4-47	ra3-29	ra4-47
4	AMP	0	87.9	88.1	74.8	77.3	81.8	82.8
4	AMP	2 X 1 X 1	87.7	88.2	83	82.8	81.4	82.3
4	AMP	2 X 1 XX 1	87.9	88	84.7	84.5	80.4	82.1
4	AMP	2 X 1 XXX 1	88	88	85.1	85.5	81.4	82.3
4	AMP	2 XX 1 X 1	87.9	88.2	84.8	85	81.2	82.4
4	AMP	2 XX 1 XX 1	88	87.9	85.1	85.2	80.9	82.2
4	AMP	2 XXX 1 XXX 1	87.5	88	85	85.7	81.2	82
4	AMP	1 X 1 X 1 X 1	87.7	88	84.1	84.3	81.5	82.4
4	AMP	1 XX 1 X 1 X1	87.9	88.3	84.7	85	81.2	82.2

Trial 7:

Table 4.18: The accuracy results of using 1 gap with, 3,4 reduced alphabet letters, size of 3N-gram and 3 different ML on AMPs, ABPs, AVPs, and AFPs dataset.

N-gram	RA	Class	RF%			SVM%			J48%		
			0	2 X 1	1 X 2	0	2 X 1	1 X 2	0	2 X 1	1 X 2
3	ra3-29	AMP	87	86.9	86.9	74	77.8	77.5	81.5	81.4	81.5
3	ra4-47		87.7	87.9	87.9	75	77.3	77.6	82	81.7	81.2
3	ra3-29	ABP	86.2	86.2	85.1	80.8	80.9	80.7	80	79.1	79.3
3	ra4-47		86.9	87.2	86.7	82.2	82.1	82.5	81.4	80	79.4
3	ra3-29	AVP	92.3	92.5	92.4	72	77.7	76.8	85.7	85.8	86.3
3	ra4-47		92.8	93.1	93.7	75.9	78.2	76	87	86.7	86.8
3	ra3-29	AFP	88.7	88.4	88	79.2	80	80.2	82.7	81.7	81.5
3	ra4-47		89.5	89.6	89.2	80.3	81.4	81.5	82.5	80.4	81

4.7.3 Discussion

Insertion of gaps within reduced alphabet letters of amino acids with N-gram calculation and ML classifiers is a successful experiment that not only recognizes sequence motifs in AMPs but go beyond basic composition and capture related correlations between neighboring and non-neighboring amino acids motifs. This allows researchers to understand more about sequence features which can aid in the design of novel AMP sequences.

There would be no significant change in the antimicrobial activity of the peptide with gaps insertion between the amino acid sequences since the accuracy rate of the ML classifiers did not change remarkably. This means that the secondary structure of AMPs, either β-sheet, α-helix, extended, or loop peptides play an essential role in this regard. Also, SVM outperforms the RF and J48 ML algorithms. Thus, maybe because that the SVM generally uses "black-box" methods which is lacking transparency in how features are being used to make predictions.

Here, we can prove that the insertion of gaps within AMPs sequences could not play a significant role in the AMPs sequence-based model but may affect the secondary structure of the peptides and its activity. This result may uncover specific patterns within the primary structure of AMPs. Further studies about the linkage between the sequence and structure of a peptide to have its antimicrobial action need to be done.

From the results above, we conclude that the primary structure, or the amino acid sequence, of a protein, controls the protein's three-dimensional structure. This implies that disorder, or lack of stable structure, can also be encoded in the sequence.

More importantly, these new features of AMPs sequence-based models are more transparent compared to those of the previous studies by researchers conducting experiments. Likewise, provide an intuitive summary of what they are capturing. We hope this can also allow for more informed design choices for those designing novel AMPs sequences in silico.

4.8 30 Residues AMPs

Dividing the AMPs sequences to only 30 residue sequences may uncover unspecific sequence patterns. Since not the whole sequence of the AMPs showed the antimicrobial characteristics, by cut part of it may appeared as Non-AMP because the motif that gives the activity of the sequence was removed, especially in longer AMPs that contains more than 30 residues.

The primary goal of this study was to successfully understand the location of the motif in AMPs' long and short sequences. Moreover, to realize how these 30 residues will be implemented by using our novel, straightforward, sequence-based method. The more advanced goal of this study is to assist in creating an artificial set of AMPs.

4.8.1 Methods

To understand the performance of antimicrobial motifs in AMPs sequences, the AMPs and Non-AMPs were shopped to 30 residues. So, all sequences of the datasets were 30 residues or less. These 30 residues of AMPs and Non-AMPs were the test set of a model that has full sequences of AMPs and Non-AMPs. The ra3-29 and N-gram size of three were used. In this study, three trials had done in order to know the performance of these motifs in a shorter sequence.

Trial 1:

1- Full dataset (7984 Sequences) of AMPs and Neg set1were used to build the model, then each of them was shopped to 30 residues sequence as a test set.

2- The first half (3992 Sequences) of the previous datasets of AMPs and Neg set1 were used to build the model, then each half was shopped to 30 residues sequence as a test set.

3- Second half (3992 Sequences) of the full datasets of AMPs and Neg set1 were used to build the model, then each of the second halves was shopped to 30 residues sequence as a test set.

Trial 2:

Due to the variances of the accuracy rates that resulted from the previous halves as shown in table 4.18, new different halves of the same full dataset of AMPs were created:

1- First half: odd number of the full dataset (3992 Sequences) of AMPs and Neg set1were used to build the model, then each half of them were shopped to 30 residues sequence as a test set.

2- The second half: even number of the full dataset (3992 Sequences) of AMPs and Neg set1 were used to build the model, then each half were shopped to 30 residues sequence as a test set.

Trial 3:

To uncover more features of our previous tests, the Neg set2 of Non-AMPs was used. Here the two halves of AMPs dataset were created as follow:

1- The first half: (1600 sequences) of an odd dataset of AMPs and the Neg set2 to build the model, then each of them was shopped to 30 residues sequence as a test set.

2- The second half: (1600 sequences) even dataset of AMPs and the Neg set2 to build the model, then each of them was shopped to 30 residues sequence as a test set.

3- The model and the test set of the second half were used as a test set to the first half model.

4- The model and the test set of the first half were used as a test set to the second half model.

The datasets applied in this experiment:

Trial 1: 7984 AMPs vs. 7984 Neg set1.

Trial 1 and 2: Half dataset: 3992 AMPs vs. 3992 Neg set1.

Trial 3: Half dataset: 1600 AMPs vs. 1600 of the Neg set2.

-Divide all the halve sets to 30 residues.

-Letters of alphabets reduce: ra3-29

-N-gram size: Three.

-RF classifier was used.

-The next applied steps were mentioned in detail in 4.2 experiment.

4.8.2 Results

Trial 1:

The classification of AMPs against Neg set1 (Non-AMPs) was successful. The full dataset model achieved an accuracy of 87% using N-gram frequency analysis, alphabet reduction ra3-29, and the RF model with ten trees and 10-fold cross-validation. The model reached 89% with AMPs 30 residues test set. The other halves of the dataset showed a slight difference in the accuracy rates of the models. In the first half set, the accuracy was 88.6% and dropped to 85.9% in the second half. Similarly, the first half test set was 93.7% and decreased to 84.6%. the Neg set1 datasets showed very low accuracy percentage, see table 4.19 below.

Table 4.19: The accuracy results of using 30 residues test, 3 reduced alphabet letters, size of 3 N-gram and RF ML on AMPs dataset.

	Model	**AMPs test set (30)**	**Neg set1 test set (30)**
	All (AMPs + Neg set1)	all AMPs	all Neg set1
No of Seq	7984 + 7984	104591	429896
RF%	87	89.8	28.2
First half set	(AMPs + Neg set1)	AMPs	Neg set1
No of Seq	3992+3992	41079	213227
RF%	88.6	93.7	26.4
Second half set	(AMPs + Neg set1)	AMPs	Neg set1
No of Seq	3992+3992	63512	216669
RF%	85.9	84.6	35.5

In trial 2:

The odd and even halves dataset models achieved an accuracy of 86% and 86.2% respectively. And around 91.5 in both AMPs 30 residues test set. While the negative set showed very low accuracy as well around 24%, as shown below in table 4.20.

Table 4.20: The accuracy results of using 30 residues test, 3 reduced alphabet letters, size of 3 N-gram and RF ML on AMPs dataset.

	Model	**AMPs test set (30)**	**Neg set1 test set (30)**
Half set (odd)	All (AMPs + Neg set1)	AMPs	Neg set1
No of Seq	3992+3992	51643	214542
RF%	86	91.4	25.3

Half set (even)	(AMPs + Neg set1)	AMPs	Neg set1
No of Seq	3992+3992	52948	215354
RF%	86.2	91.5	24.2

In trial 3:

When we used the previous odd and even dataset with the Neg set2 (new negative set) the accuracy of the both 30 residue test sets significantly decreased to around 40%. However, the accuracy rate of this Neg set2 significantly increased to around 88%. The model accuracy of the full halves against the negative set dropped around 10% in comparison to the odd and even sets in the previous trial.

Classification using test set of AMPs was conducted next. First, classification of AMPs against Non-AMPs achieved a maximum accuracy of 87.3% in the odd set and used the even set as test set and 41.7% with the even 30 residues as test set as well. Moreover, the model achieved 86.7% with the even set and the odd set as test set. See table 4.21.

Table 4.21: The accuracy results of using 30 residues test with a Neg set2, 3 reduced alphabet letters, size of 3 N-gram and RF ML on AMPs dataset.

	Model	**AMPs test set (30)**	**Neg set2 test set (30)**
Half set (odd)	(AMPs 1 + Neg set2)	AMPs1	Neg set2
No of Seq	1600+1600	20721	22345

RF%	75.7	47.8	86.6

Half set (even)	(AMPs 2 + Neg set2)	AMPs2	Neg set2
No of Seq	1600+1600	19611	22345
RF%	76	40	88.4

Half set (odd)	AMPs 2 / 87.3	AMPs 2 (30) / 41.7	
Half set (even)	AMPs 1 / 86.7	AMPs 1 (30) / 41.9	

4.8.3 Discussion

In general, shopped the amino acid sequences to small chunks changed the accuracy rate of RF. This because the longer sequences of AMPs contain a motif that gives the antimicrobial activity features to the to a certain peptide; once chopped to smaller pieces, one piece only holds the antimicrobial activity motif while others not. Which means that one piece only is AMP, while the other pieces are not AMPs anymore, and the accuracy value of the models will change.

In the first trial, the full AMPs dataset arrange as most of the short sequences written at the beginning of the dataset, and most of the long sequences written at the end of the dataset. This arrangement explains the slight differences in the accuracy rates between the first and the second halves of the dataset. To overcome these variances, in the second trial, the first half were built by taking the odd number of the sequence from the full AMPs set, whereas the second half were created from the even number sequences. After this, the accuracy rates of both sets were almost equal. The figures (4.3, 4.4, 4.5, 4.6, and 4.7) below show the distribution of the sequence length through all of the five datasets; full set of AMPs, first half, second half, odd set, and even set.

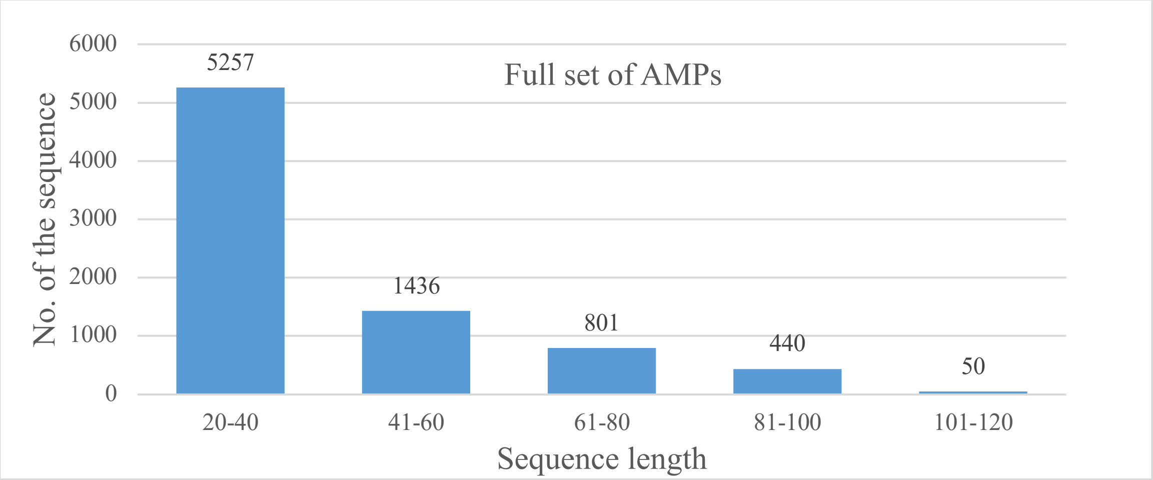


Figure 4.3: The length distribution in full dataset of AMPs.

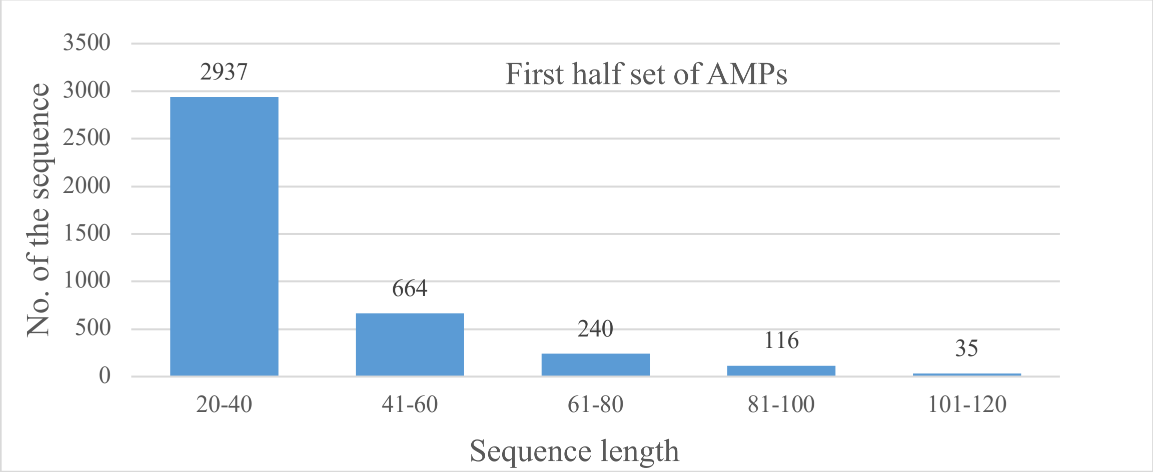


Figure 4.4: The length distribution of the sequences in first half of the full dataset of AMPs.

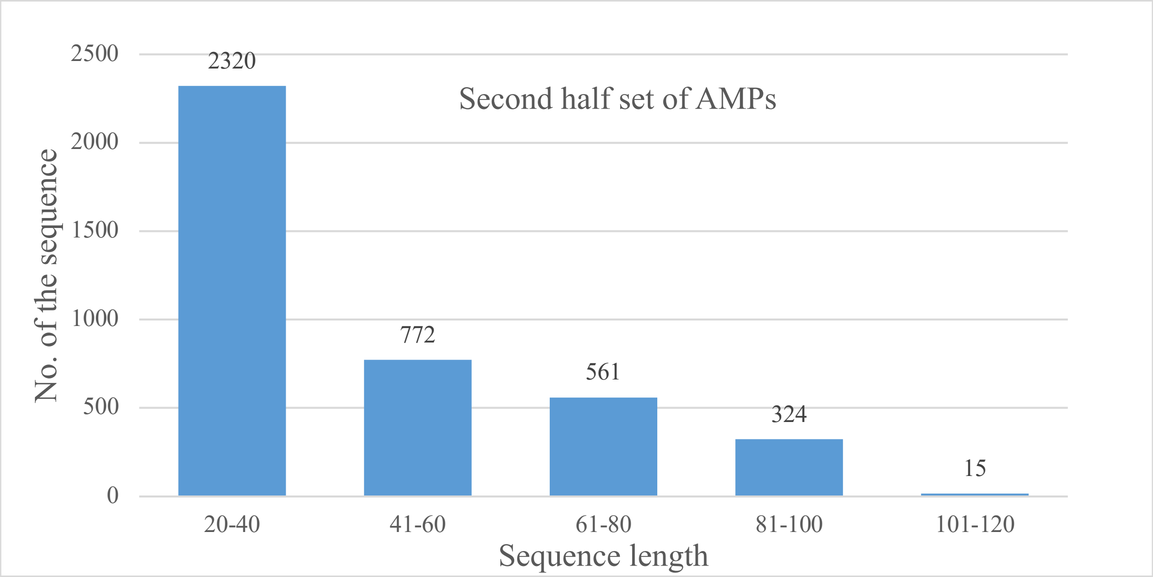


Figure 4.5: The length distribution of the sequences in second half of the full dataset of AMPs.

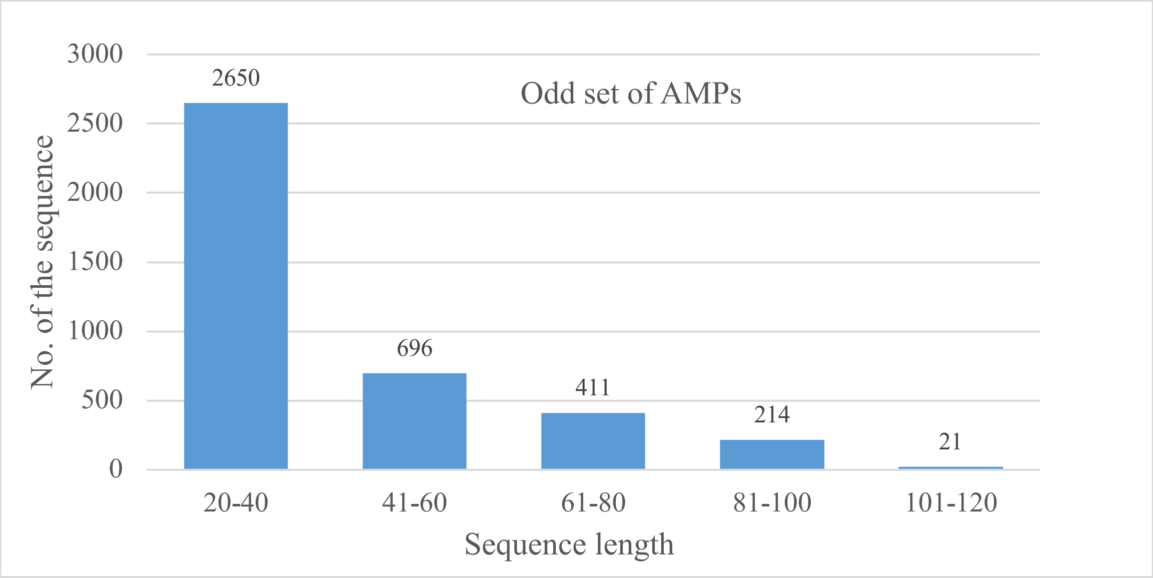


Figure 4.6: The length distribution of the sequences in the odd half of the full dataset of AMPs.

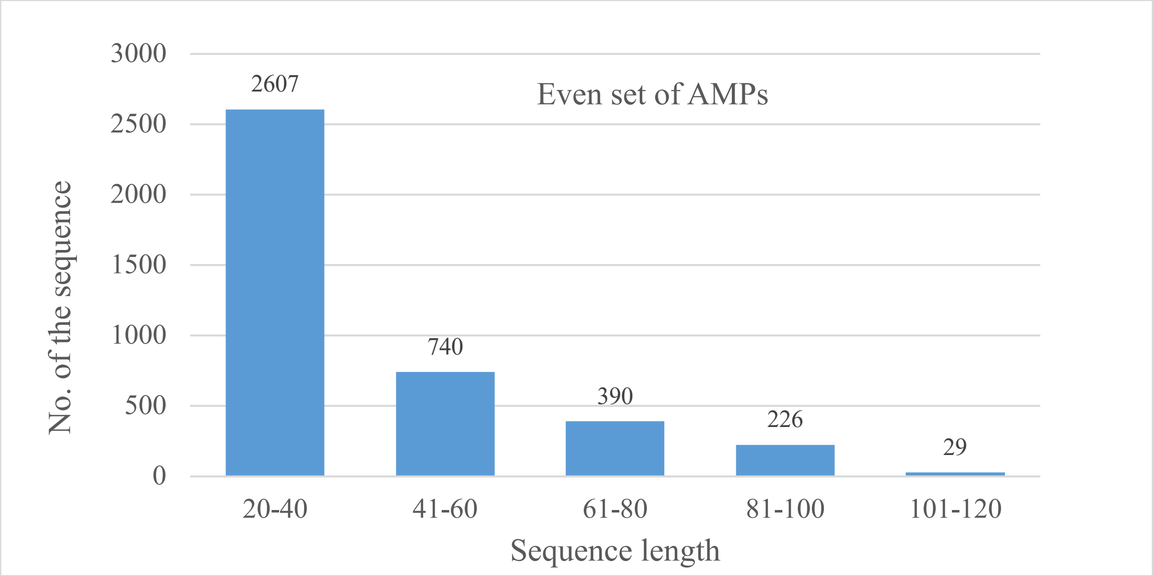


Figure 4.7: The length distribution of the sequences in the even half of the full dataset of AMPs.

4.9 The length feature of AMPs sequences

As we conclude from the previous experiment, the length of the AMP plays a vital role in the AMPs classification accuracy rates. By using a straightforward sequence-based classification of AMPs, we want to uncover more specific patterns of AMPs sequence length features. The outcomes from this study could be particularly interesting as this knowledge could applied toward synthesizing AMPs from a short amino acid sequences in the laboratory.

4.9.1 Methods

The method here was to classify the AMPs against Non-AMPs through using the sequence length. The datasets of the models were divided into three categories according to their length from:

1- 100 to 120 residues.

2- 60 to 80 residues.

3- 20 to 40 residues. The number of AMPs sequences in this range exceeds the numbers of Non-AMPs sequences. To resolve this issue, those AMPs sequences were divided into two parts and used the same Non-AMPs set in both parts.

Next, each of these sets was divided into 30 residue amino acid and test the models by using them as a test set. So, we ended up having four models, four positive AMPs of 30 residues test set, and four negative Non-AMPs of 30 residues test set.

The datasets applied in this experiment:

1- 100 to 120 residues: 53 AMPs vs. 53 Neg set1.

2- 60 to 80 residues: 848 AMPs vs. 848 Neg set1.

3- 20 to 40 residues: 690 AMPs vs. 690 Neg set1.

4- 20 to 40 residues: 690 AMPs vs. 690 Neg set1.

-Divide all the previous sets to 30 residues as a test set.

-Letters of alphabet reduce: ra3-29

-N-gram size: Three.

-RF classification model was used.

-The next applied steps were mentioned in detail in 4.2 experiment.

4.9.2 Results

The classification of AMPs against Non-AMPs was successful. Models achieved an average accuracy of 91.6 % using N-gram analysis, alphabet reduction ra3-29, and the RF model with ten trees and 10-fold cross-validation in the length of 20 to 40 residues sequences. The positive test set of this range achieved 100% accuracy, and the negative set 99.5%.

On the other hand, the accuracy decreased to 71.1% and 81.6 % in models 100-120 and 60-80 residues, respectively. As presented in table 4.22 below, the longer sequence displayed lower accuracy value as the AMP motifs will not be in each divided part of the original sequences, which complicate the ML automation process.

Table 4.22: The accuracy results of using different length of residues, 3 reduced alphabet letters, size of 3 N-gram and RF ML on AMPs dataset.

	Model	**Pos test set (30)**	**Neg set1 test set (30)**
100 - 120	All (AMPs + Neg set1)	AMPs	Non-AMPs
No of Seq	53+53	2095	2366
RF%	71.2	88	41.3
60 - 80	(AMPs + Neg set1)	AMPs	Non-AMPs
No of Seq	848+848	61888	19792
RF%	81.6	82.3	47.2
20 - 40	(AMPs + Neg set1)	AMPs	Non-AMPs
No of Seq	690+690	690	1285
RF%	90.3	100	99.5
20 - 40	(AMPs + Neg set1)	AMPs	Non-AMPs
No of Seq	690+690	3177	1285
RF%	93	96.5	88.1

4.9.3 Discussion

From this novel study, the result concludes that there is an inverse relationship between the length of the AMPs and the accuracy rate of a model. Each of AMP has its motif or motifs that give its antimicrobial activity characteristic of the particular sequence. Once this motif or these motifs removed from the sequence, the AMPs will lose their antimicrobial activity and become a natural peptide as stated earlier.

In the first part of 20-40 residues, the model achieved 100% accuracy with 30 residues test set this because all of the AMPs sequences in this range are less than 30 residues, and none of them were cut it to a shorter piece. Consequently, the ML count all of them as AMPs. While in the second part of 20-40 residues, some of the sequences were more than 30 residues and were chopped.

4.10 The Antimicrobial Peptide Database (APD)

In this study, we introduced a new positive dataset of AMPs from the Antimicrobial Peptide Database (APD). This dataset was used to build models as a test set and training sets. Here, we want to prove that our proposed straightforward method will work with any dataset models and provide reliable results, in order to use it soon, to classify any peptide using a simple and inexpensive way of alphabet reduction, N-gram analysis, and ML.

4.10.1 Methods

The method here was to classify the AMPs against Non-AMPs through using two different positive AMPs datasets and two different negative Non-AMPs datasets. All of those sets were used to build the models, test set, and training set. In this study, three trials were made to discover any concealed arrangements of this novel useful method. Furthermore, to realize how the length of the peptide affects the model's accuracy of ML.

Trial 1:

1- APD dataset (1600 Sequences) of AMPs and Neg set1 were used to build the model.

2- Each of them was shopped to 30, 40, 50, and 60 residues sequence as a test set.

Trial 2:

1- APD dataset of AMPs and Neg set2 to build the model.

2- Each of them was shopped to 30, 40, 50, and 60 residues sequence as a test set.

Trial 3:

1- APD dataset of AMPs and Neg set1 were used to build the model as trial 1.

2- Our original full AMPs dataset and the Neg set2 were used as the test set.

3- Each of the previous set was shopped to 30, 40, 50, and 60 residues sequence as a test set as well.

Trial 4:

To overcome the differences of the shopped 30 residues between the positive APD and the two negative sets, the negative sets were shorted to be equal to APD 30 residues.

1- APD dataset of AMPs and Neg set1 to build the model.

2- APD dataset of AMPs and Neg set2 to build the model.

3- Each of them was shopped to 30 residues sequence as a test set.

-The datasets applied in this experiment:

Trial 1: 1600 APD AMPs vs. 1600 Neg set1.

Trial 2: 1600 APD AMPs vs. 1600 Neg set2.

Trial 3: same models of trial 1and 2 with shorted 30 residues negative sets.

Trial 4: 1794 APD AMPs vs. 1794 Neg set1.

-Training set: 6190 AMPs and 1600 of Neg set2.

-Divide all the of these sets to 30, 40, 50, and 60 residues.

-Letters of alphabets reduce: ra3-29

-N-gram size: three.

-RF classification model was used.

-The next applied steps were mentioned in detail in 4.2 experiment.

4.10.2 Results

From trial one and trial two, the test set of the full dataset of the positive and negative AMPs achieved maximum accuracy of around 100% using alphabet reduction ra3-29 and N-gram size of three and RF ML classifiers. The model reached an accuracy of 89.3% on trial three see table 4.25. The APD test set results of trial one decreased with an increase in the length of the sequences. However, the accuracy of the Non-AMPs test set increased with an increase in the sequence's length, as in table 4.24.

On trial two, the accuracy of both positive and negative test sets improved with increasing the number of residues length to reach 96.7% on the Neg set2 Non-AMPs see figure 4.9. While on trial three, our original AMPs dataset displayed a decline in the RF%

by increasing the number of amino acids per sequence. On the other hand, the opposite effect occurred to the Neg set2 test set through using APD and Neg set1 datasets as a model, as displayed in table 4.25. The result from trial four showed that there were no significant differences in the accuracy rate when shorted 30 residues were used, see table 4.26. the accuracy of RF increased from 74.9% to 75.6% on shorted neg set1, and from 89.5% to 90.1% on shorted neg set2.

Trial 1 results:

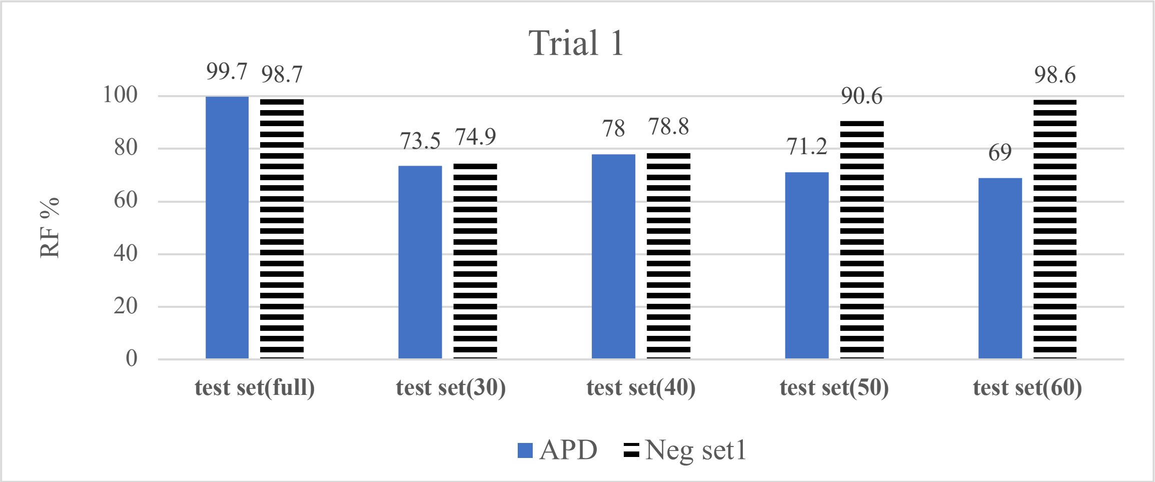


Figure 4.8: The accuracy of each test set full, 30, 40, 50, and 60 using RF classification model of APD dataset and Neg set1.

Table 4.23: The accuracy results of using different length of residues: full, 30, 40, 50, and 60 (test sets), 3 reduced alphabet letters, size of 3 N-gram and RF ML on APD dataset and Neg set1.

	Model	**APD**	**Neg set1**
	Full (APD + Neg set1)	test set (full)	test set (full)
No of Seq	1600+1600	1600	1600
RF%	84.9	99.7	98.7
		test set (30)	test set (30)

No of Seq	13020	35081
RF%	73.5	74.9
	test set (40)	test set (40)
No of Seq	6855	22556
RF%	78	78.8
	test set (50)	test set (50)
No of Seq	4227	12145
RF%	71.2	90.6
	test set (60)	test set (60)
No of Seq	3083	4520
RF%	69	98.6

Trial 2 results:

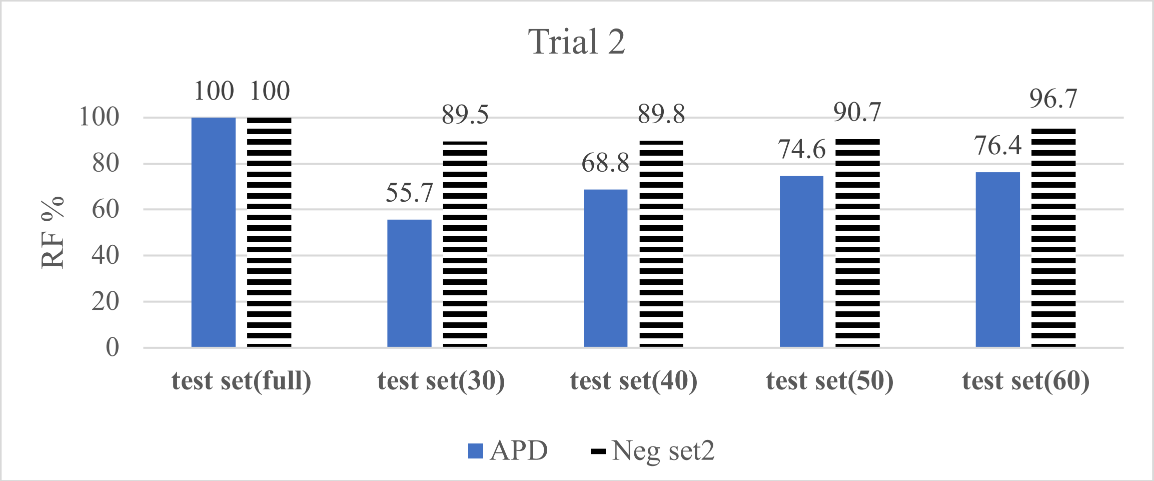


Figure 4.9: The accuracy of each test set full, 30, 40, 50, and 60 using RF classification model of APD dataset and Neg set2.

Table 4.24: The accuracy results of using different length of residues: full, 30, 40, 50, and 60 (test sets), 3 reduced alphabet letters, size of 3 N-gram and RF ML on APD dataset and Neg set2.

	Model	**APD**	**Neg set2**
	Full (APD + Neg set2)	test set (full)	test set (full)
No of Seq	1600+1600	1600	1600
RF%	82.8	100	100
		test set (30)	test set (30)
	No of Seq	13020	22345
	RF%	55.7	89.5
		test set (40)	test set (40)
	No of Seq	6855	14383
	RF%	68.8	89.8
		test set (50)	test set (50)
	No of Seq	4227	10215
	RF%	74.6	90.7
		test set (60)	test set (60)
	No of Seq	3083	7440
	RF%	76.4	96.7

Trial 3 results:

Table 4.25: The accuracy results of using different length of residues: full, 30, 40, 50, and 60 (test sets), 3 reduced alphabet letters, size of 3 N-gram and RF ML on APD dataset, Neg set1, AMPs, and Neg set2.

	Model	**AMPs**	**Neg set2**
	Full (APD + Neg set1)	test set (full)	test set (full)
No of Seq	1794 / 1794	6190	1600
RF%	89.3	72.4	48.6
		test set (30)	test set (30)
	No of Seq	87638	22345
	RF%	79.3	32.4
		test set (40)	test set (40)
	No of Seq	58214	14383
	RF%	67.5	53.2
		test set (50)	test set (50)

No of Seq	39342	10215
RF%	53.6	70.9
	test set (60)	test set (60)
No of Seq	26342	7440
RF%	45.1	79.8

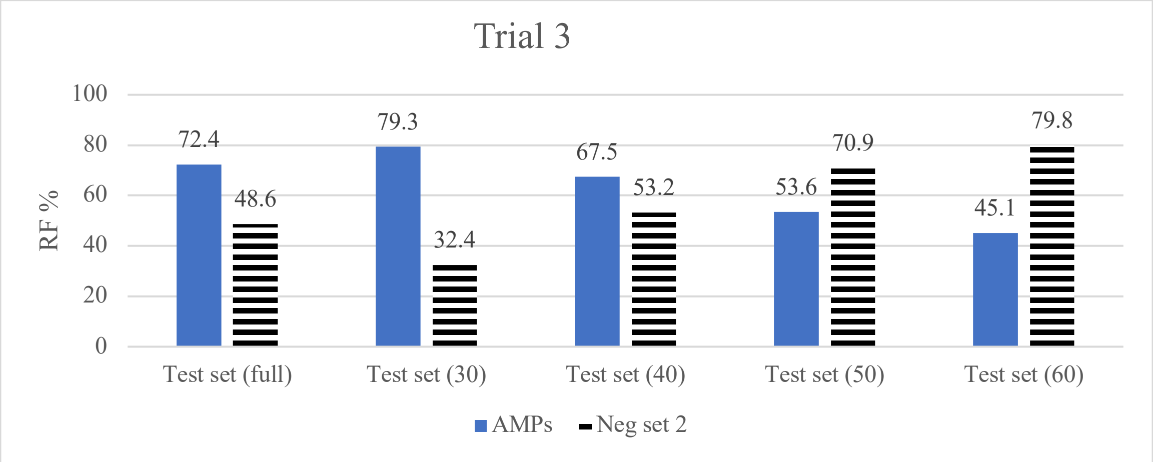


Figure 4.10: The accuracy of each test set full, 30, 40, 50, and 60 using RF classification model of APD dataset and Neg set1.

Trial 4 results:

Table 4.26: The accuracy results of using shorted 30 residues negative sets, 3 reduced alphabet letters, size of 3 N-gram and RF ML on APD dataset, Neg set1, and Neg set2.

	Model	**Pos test set (30)**	**Neg test set (30)**
	All (APD + Neg set1)	**APD seqs**	**Neg set1**
No of Seq	1600+1600	13020	13020
RF%	84.9	73.5	75.6
	All (APD + Neg set2)	**APD seqs**	**Neg set2**
No of Seq	1600+1600	13020	13020
RF%	82.8	55.7	90.1

4.10.3 Discussion

In the study, the straightforward, sequence-based classification of antimicrobial peptides was successful. In addition, we learned several general trends in the obtained accuracies throughout all the three classification trials.

From the figures above, illustrates this point on the model using peptide length as its only independent variable, AMP-activity can be seen as less probable as length increases. This is an important observation and corollary with known biological observations. AMPs tend to be short peptides compared to Non-AMPs.

One interesting result came from the attempted classification of smaller sized (specifically 30-60 amino acid) sequences between AMPs and Non-AMPs peptides. The antimicrobial region, in most cases, is commonly a short region of the peptide. Consequently, a smaller AMP would then have less "noise" coming from the region of the peptide that is not specifically antimicrobial. For this purpose, the original hypothesis was that this success rate would exceed that of the classification of unrestricted amino acid sequence size. While further studies need to be tested to account for potential experimental error, this test result could potentially provide the implication that the region of the peptide not specifically antimicrobial could actually be involved in antimicrobial response. Therefore, the model was able to classify longer peptides with higher accuracy than shorter peptides as in trial 2.

4.11 Effect of Dataset Contamination

After we performed all the previous experiments, we discover that our Neg set1 was accidentally contaminated. Around 5% of this dataset has some of the sequences are positive AMPs that were over 70% identity from the CAMP database.

4.11.1 Method

In order to overcome this error, A Blast on the neg set1 (7984 sequences) against the CAMP database was run, and all the sequences that above 65% identity were removed. As a result, the clean Neg set1 (C. Neg set1) contains 7513 sequences, and 471 sequences were disregarded.

To understand the effect of this contamination, this clean negative set was used to build a model with a positive AMPs dataset. Full dataset (7984 sequences) of AMPs was randomly reduced to 7513 sequences to balance between the positive and negative sets. Then these AMPs and the C. Neg set1 were shopped to 30 residues and used it as a test set. The ra3-29 and N-gram size of three were used.

The datasets applied in this experiment:

-7513 AMPs vs. 7513 C. Neg set1.

-Divide the sets to 30 residues.

-Letters of alphabets reduce: ra3-29

-N-gram size: Three.

-RF classifier was used.

-The next applied steps were mentioned in detail in 4.2 experiment.

4.11.2 Result

In this experiment, the result from using a clean negative set showed that the accuracy rate of the model increased about l% compared to the previous model that used the contaminated negative dataset (Neg set1). The RF of the model was 87%, as in table 4.19, and here elevated to 88%, as shown in table 4.27 below.

Table 4.27: The accuracy of the c. Neg set1 using size 3 N-grams, 3 letters alphabet reduction, and RF on AMPs dataset.

	Model	**Test set (30)**	**Test set (30)**
	(AMPs +C. Neg set1)	**AMPs**	**C. Neg set1**
No of Seq	7513 + 7513	101619	405530
RF%	88	90.2	27.3

4.11.3 Dissection

This experiment shows the effect of dataset contamination on the result. Moreover, this is very important because this contamination could happen any time during the testing phase, as in our case. However, it is possible that among those peptides in the negative set are not part of the CAMP database and may have some antimicrobial activity. Because these sequences were selected randomly from random protein sequences, and nobody tested it or discovered their antimicrobial activity, and they may have it, and we knew this from the beginning. Here we have an experiment that tells us that if a small number of sequences of the negative set have an antimicrobial activity, this does not change the accuracy of the result that much. This finding is exciting because it is addressing a critical issue of what is the effect of the contamination that cannot be discovered.

4.12 Real World Experiment

In order to test our sequence-based method, in this experiment, unknown 71 sequences were received. Three of our models were used. These models are all RF, size 3 of N-gram, 3-letter alphabet models based on different training sets. Model 1 uses all kinds of AMPs in a length range 20-40 residues (number of sequences=690), model 2 is all kinds of AMPs of length 20 (number of sequences =779), and model 3 is gram-negative AMPs only in a 15-45 residue range (number of sequences =280). The results of each model as shown below in tables 4.28, 4.29, and 4.30.

Table 4.28: The accuracy of unknown set on model 1, using size 3 N-grams, 3 letters alphabet reduction, and RF on AMPs dataset.

	Model 1	
20-40	**(AMPs + Non-AMPs)**	**AMPs**
No of Seq	690+690	71
RF%	93.3	90

Table 4.29: The accuracy of unknown set on model 2, using size 3 N-grams, 3 letters alphabet reduction, and RF on AMPs dataset.

	Model 2	
20	**(AMPs + Non-AMPs)**	**AMPs**
No of Seq	779+779	71
RF%	95.1	79.3

Table 4.30: The accuracy of unknown set on model 3, using size 3 N-grams, 3 letters alphabet reduction, and RF on AMPs dataset.

	Model 3 G-Neg AMP	
15-45	**(AMPs + Non-AMPs)**	**AMPs**
No of Seq	280+280	71
RF%	83.5	88.5

After we ran this unknown dataset on our models, we got the actual type of each sequence in the dataset. 24 of these peptides were AMPs, while the other 34 sequences are controls. These controls denote various randomizations, mutations, or truncations of the respective peptides.

As we can see from the results, models 2 and 3 predicted correctly all or almost all of the AMPs sequences as AMPs, and most randomized sequences as Non-AMPs. Model 1 achieved 90% accuracy, which has a higher rate of false positives. In comparison, almost all randomizations of the control sequences were predicted Non-AMP by model 3 that achieved 88.5%, which seems to be the best model from the current stable. Since most of these positive AMPs sequences are anti-gram-negative bacterial peptides.

4.13 Conclusion and Chapter Summary

This chapter has demonstrated that feature interactions of AMP sequences are essential and can help to improve AMP classification. The results of all experiments are indicative that the sequence-based classification of AMPs is a viable classification alternative with vast areas for additional research. This method allows for further differentiation into the subclasses of AMPs, specifically antibacterial, antiviral, and antifungal peptides. By continually improving the classification methods, biomedical researchers would be able to further advance the potential replacement of antibiotics with AMPs. Besides, by increasing the specificity of the model, there is a higher possibility that peptides can be found to combat specific classes of microbes.

The percent accuracies of most of the models were very high, around 80%, which indicates true positive rates as opposed to false positive rates. Alphabet reduction three or four letters in conjugation with trigram always demonstrates higher accuracy, and it is simple and enough for ML automation. RF significantly outperforms each of the other ML algorithms. The accuracy rate of SVM generally outperforms by RF and J48, thus maybe because, the SVM uses “black-box” method for prediction and these models are not clear as we mentioned previously.

Attribute selection of AMPs improved our model accuracy by reducing the ambiguous data, decrease overfitting and noise, and lessens training time to train ML algorithms faster.

Increasing gaps between amino acids rise accuracies, especially in the SVM algorithm. This means that the secondary structure of AMPs plays an essential role in this regard.

The length of the AMP plays a vital role in the AMPs classification accuracy rates. The probability of our model's activity decreases with length. So, we successfully understand the connection between AMPs' amino acid sequences and how the structure of the amino acid of the AMPs is encoded within the sequence. And how this connection would help in capturing related correlations between neighboring and non-neighboring amino acids motifs.

The contamination of the negative set could happen unintentionally and may occur at any time, and we should avoid it. Conversely, this contamination may not discover, and no one can identify it since not all the protein sequences were tested to check their antimicrobial activity. The effect of this contamination in our sequence-based method is minimal and insignificant. Basically, the need for a large dataset of confirmed Non-AMPs is appreciated.

In a real-world experiment, our sequence-based method achieved a high accuracy rate, which gives our model credibility and viability for the AMPs researcher community. Consequently, they can test their sequences using our model to discover or design new AMPs.

Direct comparisons between our method and commonly used methods are difficult due to the use of different datasets and the availability of these different classifiers. We are here just compared reported accuracy for each method that used to predict AMPs as shown in table 4.31 to our model's accuracies in table 4.32.

Table 4.31: Reported accuracy for some commonly used methods to predict AMPs.

Name	Method	Accuracy %	Weakness	Reference
Generic string kernel	ML	90	Only highly active AMPs	(Giguère et al., 2014)
SVM-LZ	SVM	87.6	Take longer time	(Ng et al., 2014)
AntiBP	SVM	85.2	For ABPs only	(Lata et al., 2010)
	ANN	77.3	For ABPs only	(Lata et al., 2007)
CAMP	Discriminant Analysis	87.5	Relies on similarity scores	(Thomas et al., 2009)
	SVM	91.5	Relies on similarity scores	(Thomas et al., 2009)
Seq alignment and feature selection	BLASTP + Nearest Neighbor Algorithm	80.2	Complexity and lower accuracy	(Wang et al., 2011)

Table 4.32: Our models accuracy for AMPs prediction.

Our Models	Accuracy %
Length feature of AMPs	93.3
AMPs with different N-grams size	93.0
30 Residues AMPs	91.5
Antimicrobial Peptide Database (APD)	89.3

However, further research would still greatly benefit this study in order to provide sufficient explanation for part of the interesting results, such as the varied success rates of the multiple alphabets for AMPs subclasses and how the insertion of many gabs would not affect the sequences feature. Finally, we conclude that the trigram with either three or four letters of alphabet reduction and RF is optimal for AMPs classification.

In the next chapter, we demonstrate different evaluation metrics to validate our results in this chapter. Some common performance metrics used for classification are briefly outlined next.

Chapter 5: Evaluation Metrics Considered for Model Performance

5.1 Introduction

Evaluating the model performance is one of the fundamental steps in the model development process. It helps to find out the best model that represents data, shows how successful the predictions of a dataset that have built in the training phase, and how well the chosen model will work in the future. Many metrics are offered for quantifying prediction and classification performance. Determining the suitable performance metric and acceptable level of type I error (when a null hypothesis is incorrectly rejected) and type II error (when a null hypothesis is incorrectly accepted) depends on the given problem. Predictions with correct classified observations are referred to as “true positives” (TP) or “true negatives” (TN), while erroneously classified observations are known as “false positives” (FP) or “false negatives” (FN). Several common performance metrics used for prediction and classification and employed throughout this chapter: learning curve, Mathew's Correlation Coefficient, Sensitivity and Specificity, Balanced Error Rate, Accuracy, Receiver Operating Characteristic, Precision, and Precision Recall Curves are briefly defined in the next part. Those metrics were performed on 17 models using dataset sizes ranging from 250 to 7984 (the size of the full positive set) and the control.

5.2 Evaluation Metrics

5.2.1 Learning Curve

To ensure that a small sample size of peptide sequences would not be indicative of overfitting, a learning curve was constructed for one classification test. In this study, the learning curve was constructed for the classification of AMPs against Non-AMPs. The curves had increments of 500 sequences. The learning curve was created using dataset sizes ranging from 250 to 7500 sequences.

5.2.2 Mathew's Correlation Coefficient (MCC)

MCC is fundamentally the correlation coefficient between the observed and the predicted in the binary classification performance. The value range will be from -1 to 1. The correlation coefficient of 1 signifies completely correct predictions in the recommend data while the value of -1 represents a completely opposite prediction. Larger values equate to better classification performance (Liu et al., 2015).

The formula of MCC defines as:

$$MCC = \frac{(\mathrm{TP} \times \mathrm{TN}) - (\mathrm{FN} \times \mathrm{FP})}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}}$$

5.2.3 Sensitivity and Specificity

Sensitivity and specificity analysis is used to evaluate the performance of a model. In the context of classification, sensitivity is the ability of an algorithm to correctly classify an amino acid as $'\mathrm{AMP}'$. It is equivalent to the true positive rate and can assesses type II error (Parikh et al., 2008).

The formula of sensitivity defines as:

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

Whereas specificity is the ability of an algorithm to correctly classify an amino acid as "Non-AMP". It is also called true negative rate and can assesses type I error (Parikh et al., 2008).

The formula of specificity defines as:

$$\text{Specificity} = \frac{TN}{TN + FP}$$

In both cases, values range from 0 to 1 and higher values associate to lower error rates.

5.2.4 Accuracy (ACC)

ACC represents the closeness of a measured value to a standard or known value when conditions remain constant. This refers to how well a method can predict sample classes. The values of ACC range from 0 to 100 percent. Larger values associate with better classification performance assumed that the number of positive and negative examples are similar in size (Taylor, 1997).

The formula of ACC defines as:

$$ACC = \frac{TP + TN}{TP + FP + TN + FN}$$

5.2.5 Balanced Error Rate (BER)

BER is the complement of the accuracy metric, that calculate the average of the errors on each class. It’s used to measure the effectiveness of our algorithms. So, a classification model that heavily favors one class will have a high BER even with a low error rate (Zhao et al., 2013).

The formula of BER defines as:

$$BER = 1 - ACC$$

Both ACC and BER are sensitive to the imbalanced data.

5.2.6 Receiver Operating Characteristic (ROC)

ROC curves compare true positive rate (Sensitivity) versus false positive rate (Specificity) across a range of values for the ability to predict a classifier performance. Each point on the ROC curve represents a pair corresponding to a particular decision threshold (Hajian-Tilaki, 2013). ROC curve provides a convenient graphical representation of classification and predictions accuracy. The closer the curve to the upper-left border of the ROC space, the more accurate classification. The ROC score is a summary statistic which refers to the area under the curve (auROC). If a classifier correctly predicts the class for all observations, then the auROC score reaches 1 (or 100 percent) whereas a random guess would have an auROC of 0.5 and would be represented by a straight diagonal line from the lower left to top right of the chart (Bewick et al., 2004).

5.2.7 Precision and Precision Recall Curves

Precision are commonly used for evaluating the classification performance. Precision is the ratio of correctly classify an amino acid as $'$AMP$'$ to the total predicted positive AMP. also known as positive predictive value (PPV).

The formula of precision defines as:

$$\text{Percision} = \frac{TP}{TP + FP}$$

The Precision Recall Curve (PRC) shows the relationship between Recall (sensitivity) and precision.

5.3 Model-Performance Results

The main goal of this study was to successfully classify AMPs using a straightforward, sequence-based method that involved N-gram analysis, and ML. More complex goals of this study were classification between subclasses of AMPs and creation of an artificial set of AMPs in silico. Success rates of the performance evaluation for the 17 models were comparable to that of previous studies by researchers conducting experiments with tangible AMPs in biochemistry laboratories.

The learning curves created showed that model accuracies varied minorly with respect to dataset size. The AMPs against Non-AMPs comparison yielded consistent accuracies above 1000 sequences. The curves had increments of 500 sequences, and ra3-29 with an N-gram size of three were used. The curve was generally flat between these points, as in the figure 5.1 below.

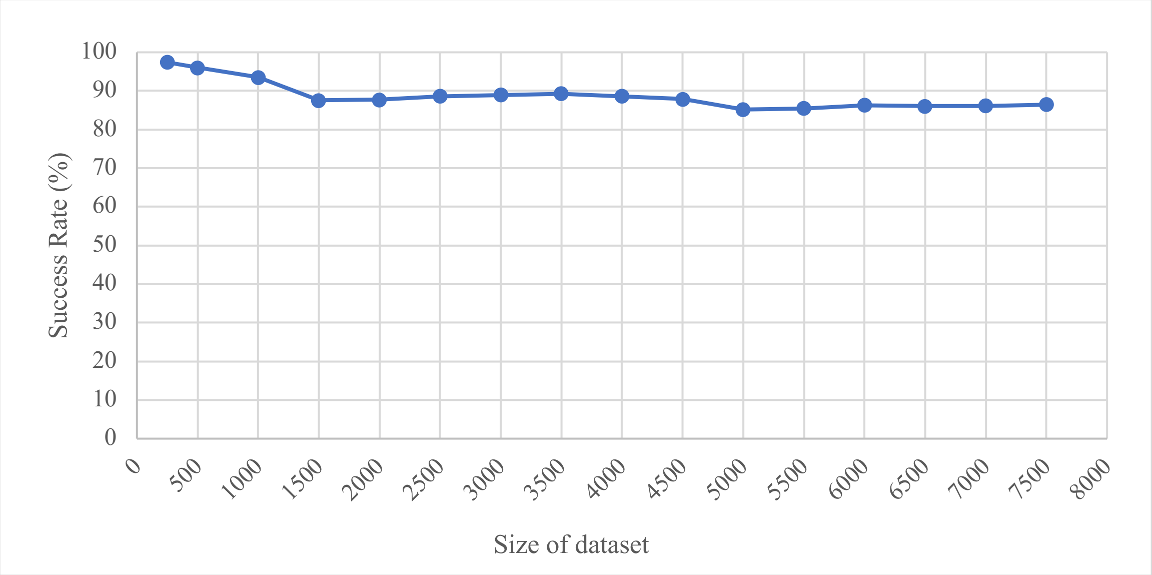


Figure 5.1: Model accuracies do not vary significantly with the size of the dataset, providing assurance that model accuracies will not be skewed as a result of small or large datasets.

Next, the 17 models were compared in the context of classification performance, comparing the models in terms of sensitivity, specificity, and MCC values. The sensitivity and specificity of all models achieved above 0.84, while MCC reached 0.74 in the full size of the AMPs dataset.

The effect on classification performance on all models can be seen in figure 5.2, which shows that all models have high sensitivity, specificity, and MCC value.

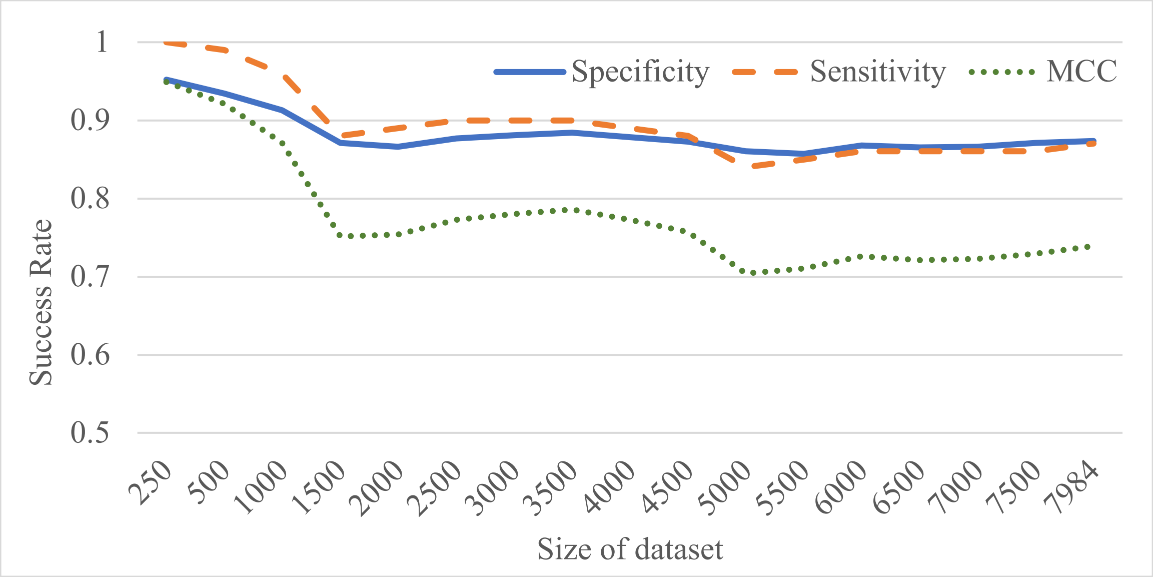


Figure 5.2: Classification performance of all 17 models on AMPs against Neg set1 datasets are evaluated in terms of sensitivity, specificity, and MCC.

Accuracies achieved a maximum at a dataset size of 250 sequences and a minimum at a dataset size of 5000 to 5500 sequences. ACC results showed good accuracy of classification between AMPs and Non-AMPs in all the models ranging between 97% to 85% using N-gram analysis, alphabet reduction option ra3-29, and the RF model with ten trees-fold cross-validation. BER value indicated the misclassification rate in each model. As we can see from figure 5.3, low BER value in all models with a maximum error rate 14.5% in 5500 sequence's models. The control trial showed low accuracy about 50% and high BER, implying that all models resulting from this method yielded reliable results.

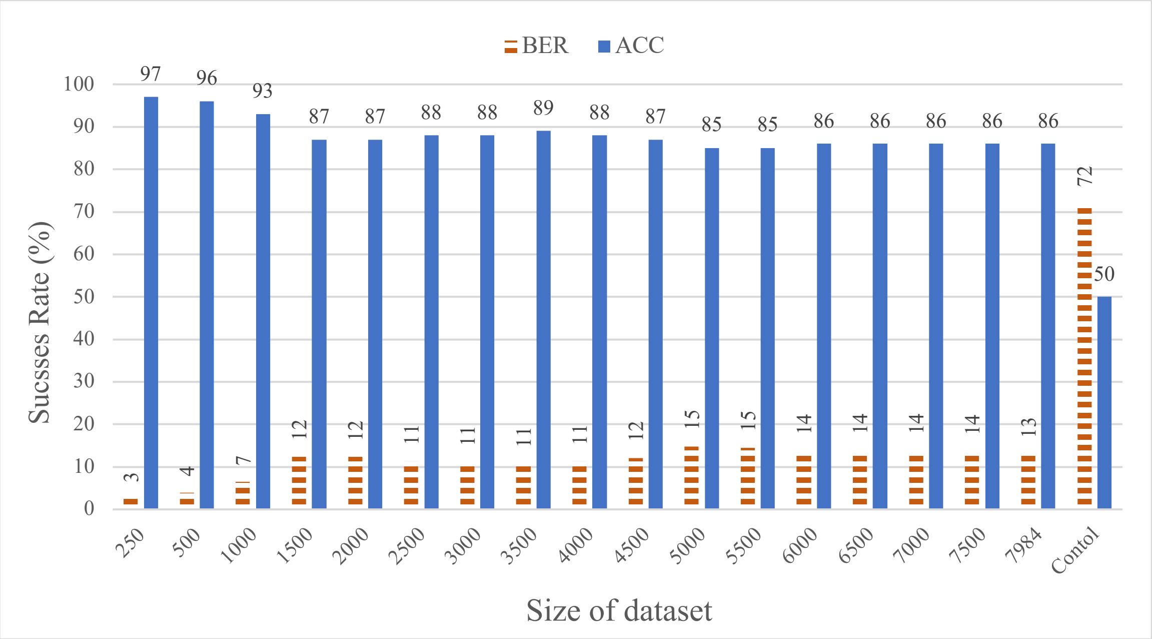


Figure 5.3: Classification performance of all 18 models on AMPs against Neg set1 dataset are evaluated in terms of ACC and BER.

A set of ROC (Receiver operating characteristic) area curves were created. ROC area is restricted to a real number between 0 and 1, with the area near 1 indicating few false positives in the data and with the area near 0.5 indicating truly random results. The area under the ROC curve (auROC) was 0.93 for a size of 7984 sequences (full dataset) to 5500 sequences, 0.92 for 5000 sequences, 0.94 for 4500 sequences, 0.95 for 4000 to 2500 sequences, 0.94 for 2000 to 1500 sequences, 0.99 for 500 to 250 sequences and 0.5 for the control trial with 7984 sequences. Figure 5.4 show the ROC curves for the full dataset and the control models. The ROC curves for the other 16 models available in the appendices, see figure A.1.

The 17 success rates of our models and the control were supported by the high accompanying auROC areas, indicative high true-positive rates compared to false positive rates, and lack of overfitting in the models. These success rates evince the classification power of alphabet reduction, N-gram, and ML models regarding differentiation between AMPs and Non-AMPs.

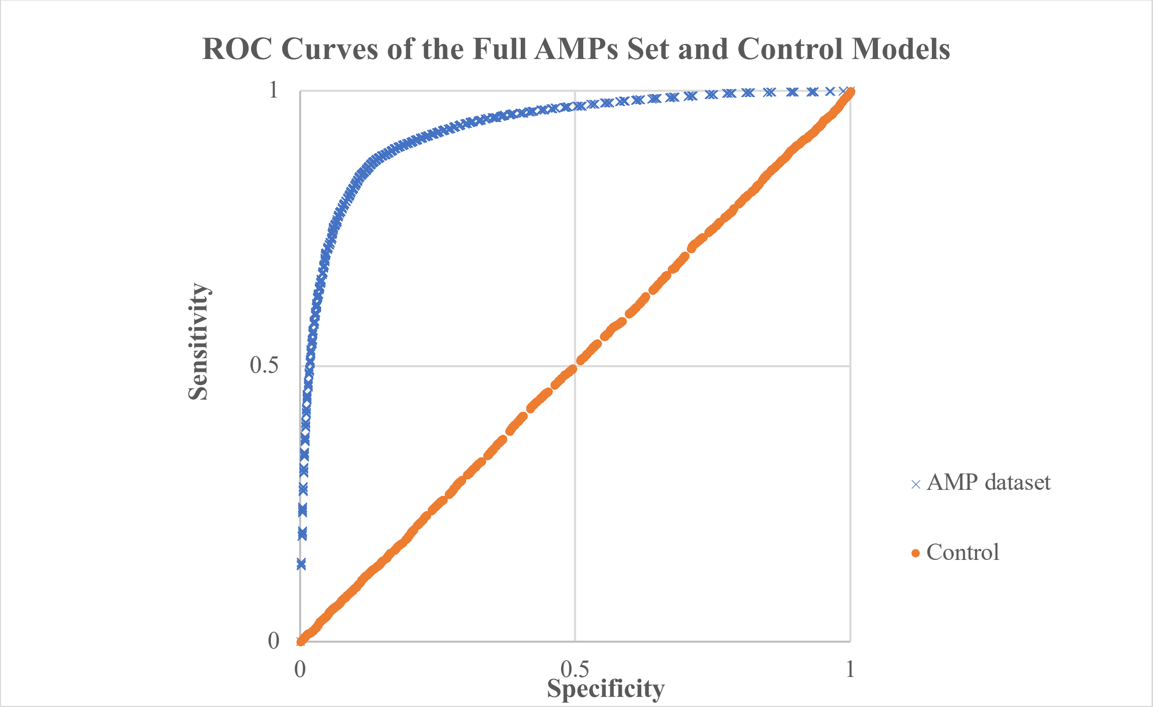


Figure 5.4: ROC Curves of the full-size dataset (7984 sequences) and the control models on AMPs against Non-AMPs datasets using RF 10-fold cross validation, N-gram size of 3, and reduced alphabet ra3-29

Precision and recall (sensitivity) are two vital model evaluation metrics. The result is a value between 0.0 for no precision / no recall and 1.0 for perfect precision and recall. All of our models achieved above 0.85 of precision. the minimum rate of recall was 0.84 as shown in figure 5.5 below, which indicates that our performance models have low false positives and high true positives rates of classification between AMPs and Non-AMPs.

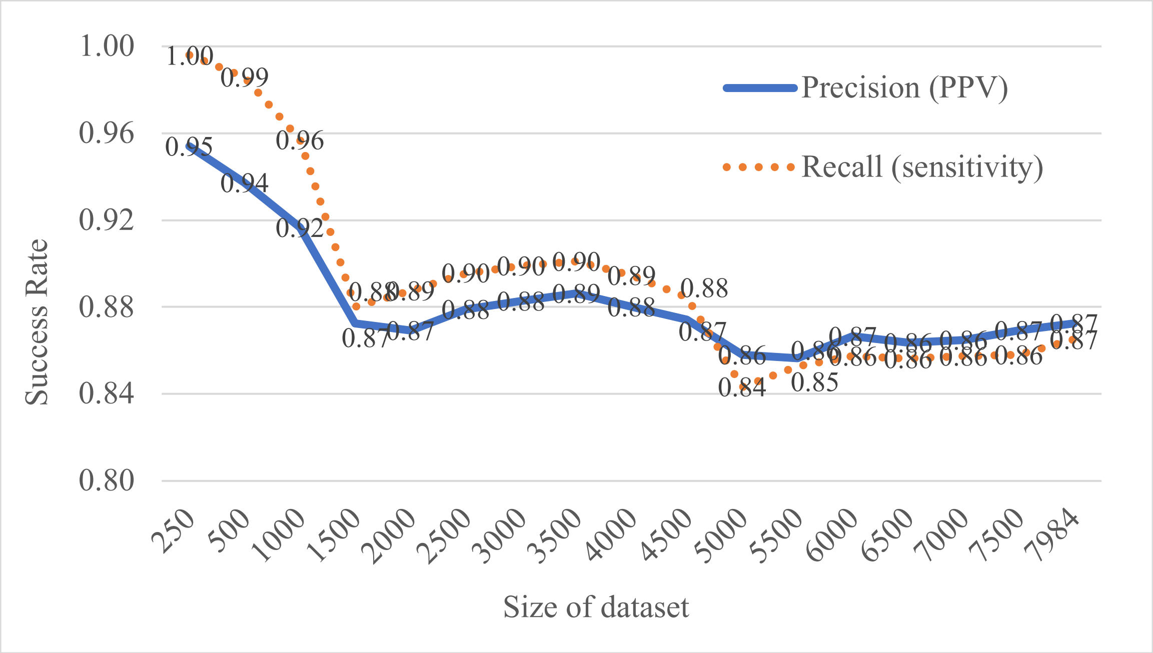


Figure 5.5: Classification performance of all 17 models on AMPs against Non-AMPs datasets are evaluated in terms of precision and recall.

5.4 Conclusion and Chapter Summary

The performance of the 17 models and the control among all of evaluation metrics prove that an N-gram based approach to discriminate between AMPs and Non-AMPs is an effective and efficient method. The quantitative findings from the learning curve indicated that even a small dataset size would not cause a substantial difference in the accuracy value of the models, which adds credibility to the accuracies mentioned in this dissertation.

The MCC, sensitivity, specify, precision, and recall displayed high success rate of over 0.8 in most of the models, which imply that our novel method has the power to correctly classify AMPs against Non-AMPs with high actual positive observations and low BER value, lack of false-positives and potential overfitting of the models to a dataset.

As mentioned before, 85.2% accuracy for this classification is comparable to the accuracies of previous studies utilizing more complex methods of classification. Furthermore, ROC area values are consistently high in all the models. Label randomization (control) verified the integrity of the dataset and implied that all models used in this comparison would yield reliable accuracies.

The results propose that the classifiers produced high predictive power and can be used in several medical and biological applications, potentially saving thousands of lives. AMPs are useful for modification of existing AMPs and for designing new synthetic AMPs. Again, consistently high accuracy, high auROC values, and other evaluation metrics corroborate for successful classification.

Chapter 6: N-gram Classification Application

6.1 Introduction

To make our proposed straightforward, sequence-based method that involved alphabet reduction, N-gram frequency, and ML reproducible and reachable to the greater AMPs research community, “N-gram Classification” has created, a free AMPs classification and prediction application available at http://www.binf.gmu.edu/mothman/N-gram-Classification-Application/. The Application is easy-to-use and simply requires the user to upload a FASTA or text file of sequences and reduced alphabet letters.

This application calculates the N-gram frequencies of each sequence that has been reduced to a reduced alphabet of 2 or more letters, selected by user. Also, the user can choose the way of reading the N-gram frequency, the range of the length of the sequences, remove redundancy, generate control of randomly labeled peptide, provide information about the sequence length and the number of sequences, and matching between the file of the original sequence and the chopped one. In this chapter, we will explain each of our application features in detail.

6.2 N-gram Application Methodology

As mentioned previously, the 20-letter amino acid alphabet was reduced to an alphabet of significantly fewer letters to simplify and quicken the ML process. N-gram algorithm is used to estimate the probability from relative frequency counts. It reads each sequence in the dataset and calculates the relative frequency of N-grams to show each sequence composition regarding feature vectors.

More details about how the alphabet reduction and N-gram frequency are implemented are available in Chapter 3. The performance results of ML classifiers that used in previous chapters had been reported using WEKA. A general workflow diagram for the application can be seen in figure 6.1.

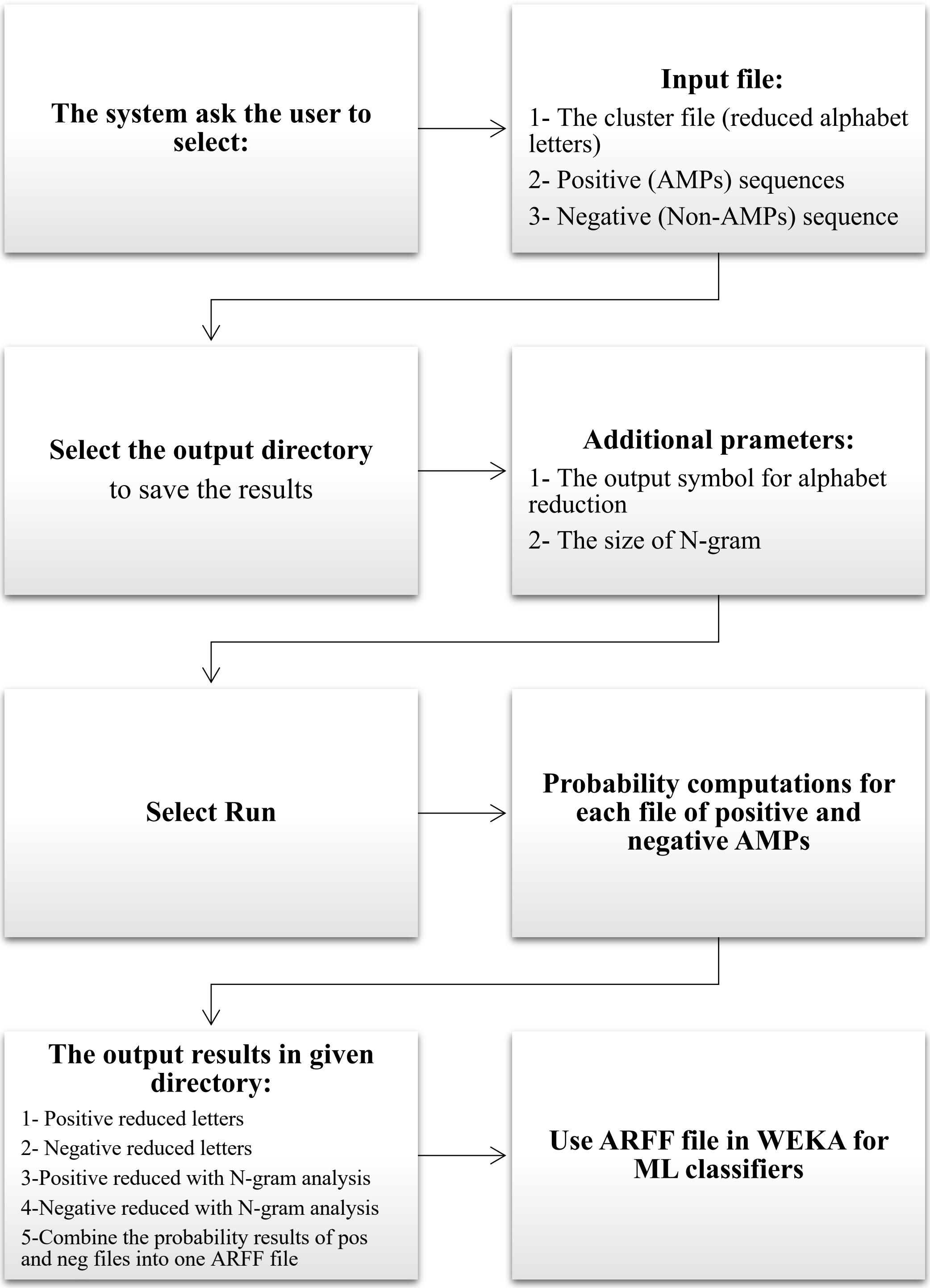
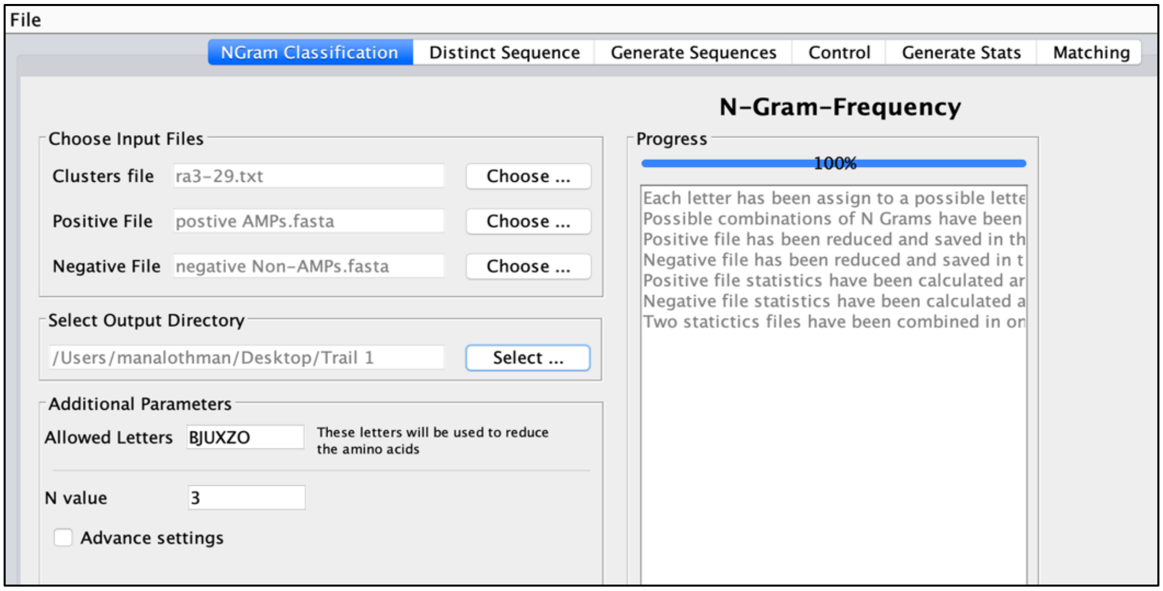

The system ask the user to select:
Input file:
1- The cluster file (reduced alphabet letters)
2- Positive (AMPs) sequences
3- Negative (Non-AMPs) sequence
Select the output directory
to save the results
Additional prameters:
1- The output symbol for alphabet reduction
2- The size of N-gram
Select Run
Probability computations for each file of positive and negative AMPs
The output results in given directory:
1- Positive reduced letters
2- Negative reduced letters
3-Positive reduced with N-gram analysis
4-Negative reduced with N-gram analysis
5-Combine the probability results of pos and neg files into one ARFF file
Use ARFF file in WEKA for ML classifiers

Figure 6.1: A workflow diagram detailing the structure of the N-gram classification application.

6.3 Application Features

6.3.1 N-gram Classification

The pipeline starts at the top left, as in figure 6.1 above when the user uploads their reduced letters text file, positive AMPs, and negative Non-AMPs sequences in FASTA file. After that, select where to save the output files. Then, select the output reduced letters and the size of N-gram. The program default size of N-gram is three and the reduced alphabet distinct letters are (BJUXZO)The user free use any letters or numbers that are not existing within the sequences. See figure 6.2.



Clusters file: ra3-29.txt

Positive File: postive AMPs.fasta

Negative File: negative Non-AMPs.fasta

/Users/manalothman/Desktop/Trail 1

Allowed Letters: BJUXZO — These letters will be used to reduce the amino acids

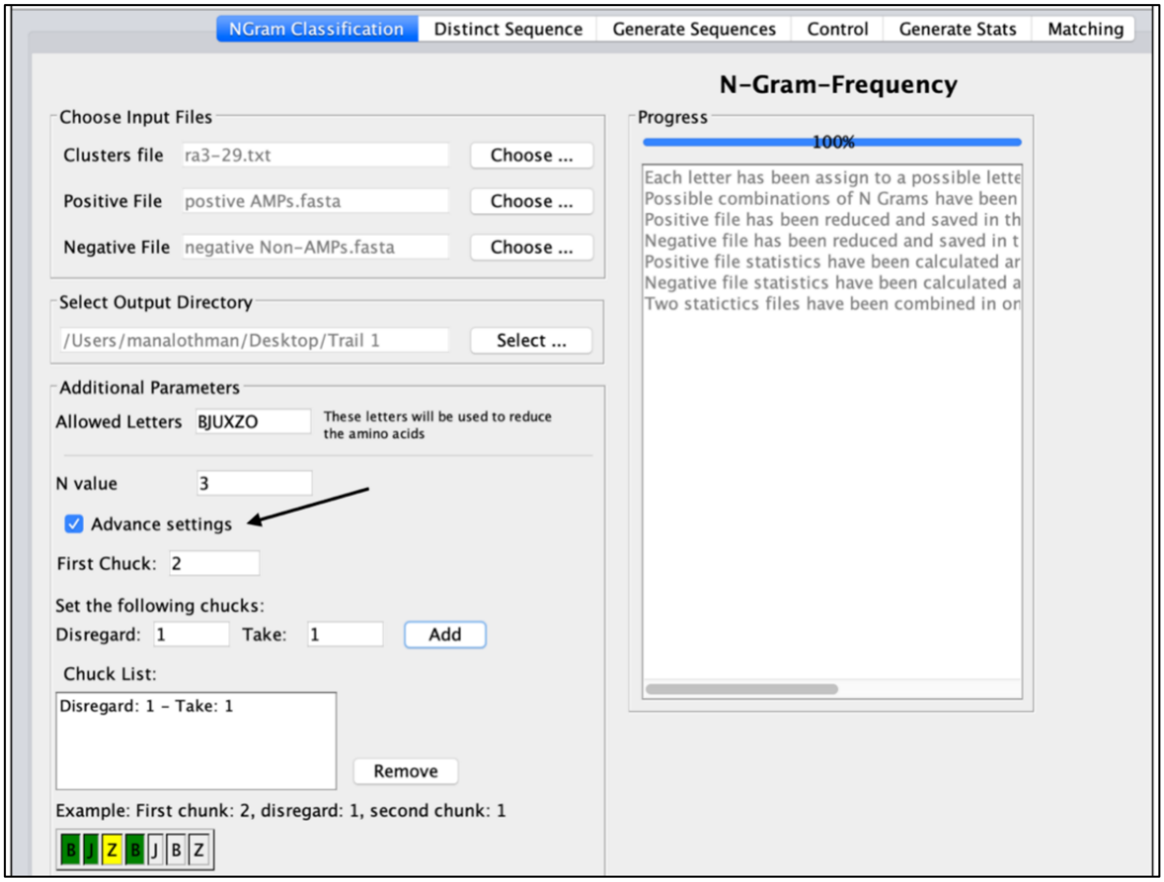
N value: 3

□ Advance settings

```
Each letter has been assign to a possible lette
Possible combinations of N Grams have been
Positive file has been reduced and saved in th
Negative file has been reduced and saved in t
Positive file statistics have been calculated ar
Negative file statistics have been calculated a
Two statictics files have been combined in or
```

Figure 6.2: The main graphical user interference of N-gram classification application.

One of the features of this application in advance setting button, which the user can specify the way of reading N-gram frequency from a given sequences. The user can add as many as he wants from disregard and take chunks. For example, take the first two letters, then disregard one letter, and then take another one letter, and disregard two letters, and at the end, take one letter and so on see figure 6.3 below. This setting was used previously in chapter 4 as in gaps insertion features experiment.



N-Gram-Frequency

Choose Input Files

Clusters file: ra3-29.txt

Positive File: postive AMPs.fasta

Negative File: negative Non-AMPs.fasta

Select Output Directory: /Users/manalothman/Desktop/Trail 1

Additional Parameters

Allowed Letters: BJUXZO — These letters will be used to reduce the amino acids

N value: 3

☑ Advance settings

First Chuck: 2

Set the following chucks:

Disregard: 1 Take: 1

Chuck List:

Disregard: 1 – Take: 1

Example: First chunk: 2, disregard: 1, second chunk: 1

🟩B 🟩J 🟨Z 🟩B J B Z

Progress

100%

```
Each letter has been assign to a possible lette
Possible combinations of N Grams have been
Positive file has been reduced and saved in th
Negative file has been reduced and saved in t
Positive file statistics have been calculated ar
Negative file statistics have been calculated a
Two statictics files have been combined in on
```

Figure 6.3: A screen shot of advance settings of N-gram classification application.

Next, the system will generate possible combinations of the amino acids of each positive and negative sequence using only a given letters and will calculate the N-gram algorithm by determining how often these three or any number of letters of amino acid sequence occurred throughout the entire peptide sequence. The outputs from the run button will be saved in the chosen directory. Five different files will be produced:

1- Positive reduced letters in txt file.

2- Negative reduced letters in txt file.

3- Positive reduced with N-gram frequency values in CVS file.

4- Negative reduced with N-gram frequency values in CVS file.

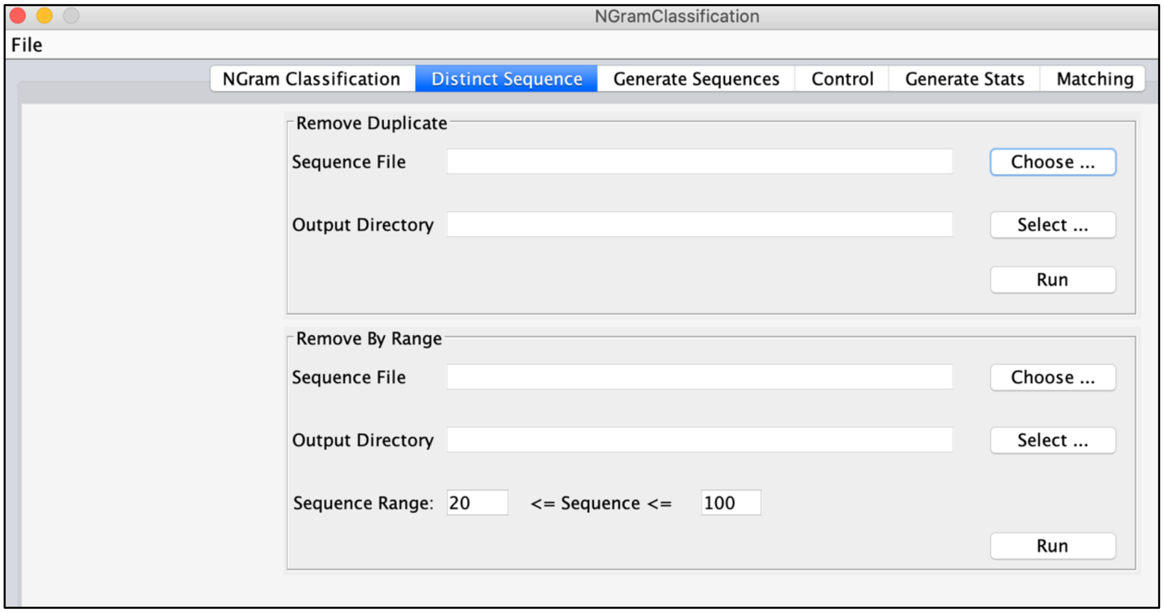
5- Combine the probability results of positive and negative files into one ARFF file. This model file well be ready for WEKA. WEKA will be used to classify sequences based on N-gram frequencies using different ML algorithms as RF, SVM, and decision tree.

6.3.2 Distinct Sequence

The second tab in the application is a distinct sequence, which has two options:

1- Remove duplicate: The user able to remove any duplication in the given file sequences to avoid redundancy in the dataset. The user will upload the fasta file of the sequence dataset from the selected directory, and then he will choose the location for the output file. This output file will contain distinct sequences without any duplication on the dataset.

2- Remove by range: Choose the range of the sequence's length. The output file will include all the sequences in a specific range of length or even one length. For example; as in this study, any sequence below 20 and above 120 was removed, or from 40 to 40 sequences only, which means only sequences that have 40 residues will be in the output file. The outputs from each option will be saved in the chosen directory see figure 6.4.



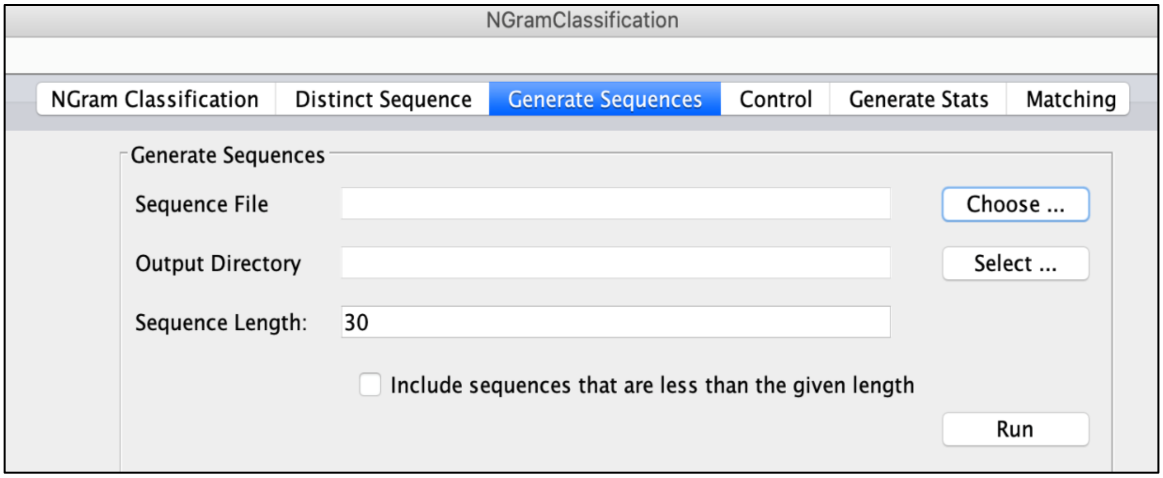
Sequence Range: 20 <= Sequence <= 100

Figure 6.4: A screen shot of distinct sequence tab of N-gram classification application.

6.3.3 Generate Sequence

The third tab of the application is to generate sequences from a given sequence file and a length. In other words, the program will divide each sequence to a given length, and the rest of the sequence will be a new sequence with the same length. The output file will be one size of length for all sequences and will saved in the chosen file as shown in the figure

6.5 below. The user has the option to discard or keep any sequences that are less than the required length. This option was used in 30 residues AMPs experiment in chapter 4.



Sequence File

Output Directory

Sequence Length: 30

□ Include sequences that are less than the given length

Figure 6.5: A screen shot of generate sequences tab of N-gram classification application.

6.3.4 Control

This feature allows the user to create a control dataset. There are two preferences available in this tab:

1- Make two controls set from two different files, positive and negative sets. The program will randomly shuffle the sequences in the two files in one list. Then, the program will generate a positive file and a negative file from the randomly shuffled sequences list. The size of each file will remain as the size of the original files.

2- One control set from one file to be randomly shuffled to change the order of the sequences in that file.

-The outputs from each option will be saved in the chosen directory, see figure 6.6.

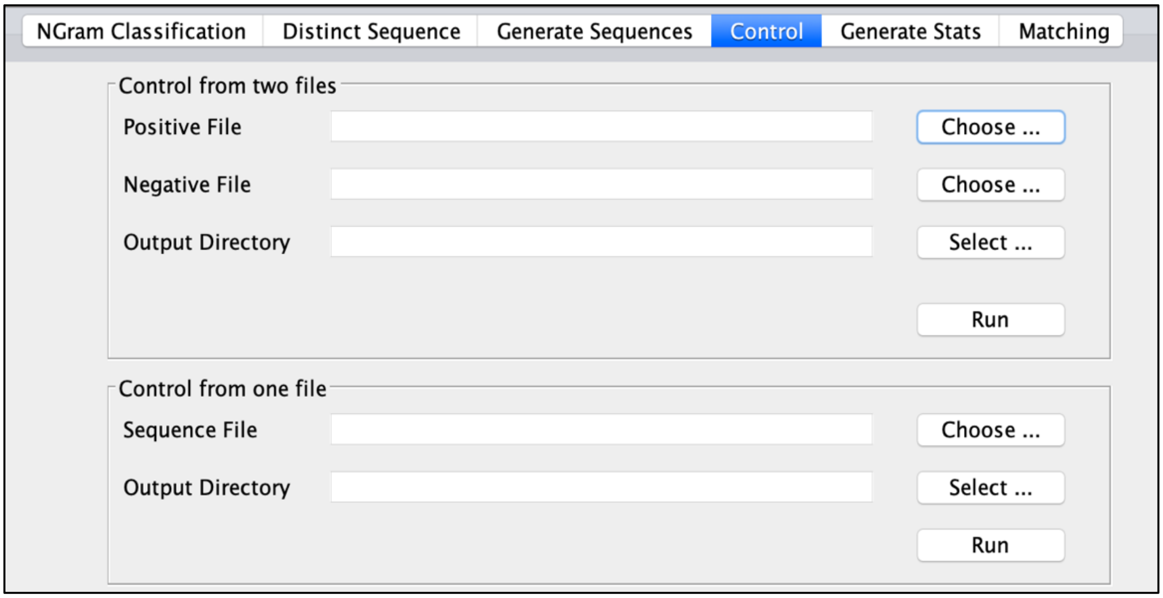


Figure 6.6: A screen shot of control tab of N-gram classification application.

6.3.5 Generate Status

This tab will generate info about the sequence file. The output of this feature will be two files:

1- The first file will show each sequence with its length.

2- The second one will count how many sequences in the file have the same length. See figures 6.7 and 6.8 below.

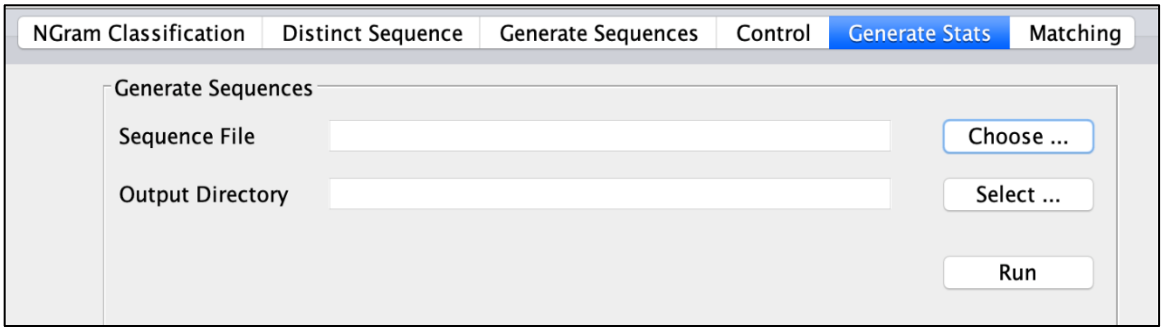
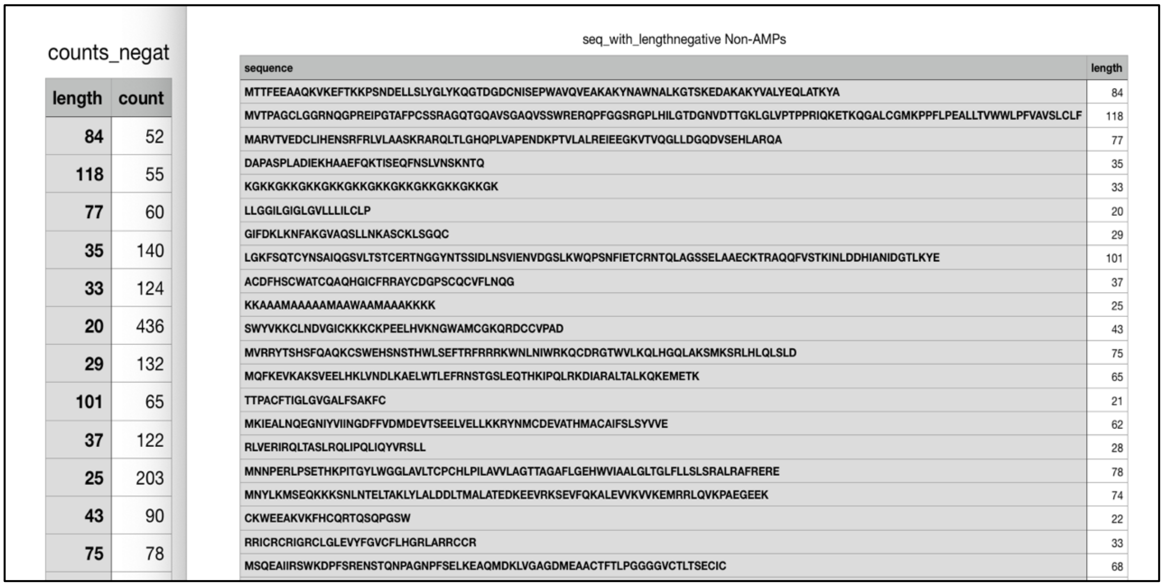


Figure 6.7: A screen shot of generate status tab of N-gram classification application.



counts_negat

length	count
84	52
118	55
77	60
35	140
33	124
20	436
29	132
101	65
37	122
25	203
43	90
75	78

seq_with_lengthnegative Non-AMPs

sequence	length
MTTFEEAAQKVKEFTKKPSNDELLSLYGLYKQGTDGDCNISEPWAVQVEAKAKYNAWNALKGTSKEDAKAKYVALYEQLATKYA	84
MVTPAGCLGGRNQGPREIPGTAFPCSSRAGQTGQAVSGAQVSSWRERQPFGGSRGPLHILGTDGNVDTTGKLGLVPTPPRIQKETKQGALCGMKPPFLPEALLTVWWLPFVAVSLCLF	118
MARVTVEDCLIHENSRFRLVLAASKRARQLTLGHQPLVAPENDKPTVLALREIEEGKVTVQGLLDGQDVSEHLARQA	77
DAPASPLADIEKHAAEFQKTISEQFNSLVNSKNTQ	35
KGKKGKKGKKGKKGKKGKKGKKGKKGKKGKKGK	33
LLGGILGIGLGVLLLILCLP	20
GIFDKLKNFAKGVAQSLLNKASCKLSGQC	29
LGKFSQTCYNSAIQGSVLTSTCERTNGGYNTSSIDLNSVIENVDGSLKWQPSNFIETCRNTQLAGSSELAAECKTRAQQFVSTKINLDDHIANIDGTLKYE	101
ACDFHSCWATCQAQHGICFRRAYCDGPSCQCVFLNQG	37
KKAAAMAAAAAMAAWAAMAAAKKKK	25
SWYVKKCLNDVGICKKKCKPEELHVKNGWAMCGKQRDCCVPAD	43
MVRRYTSHSFQAQKCSWEHSNSTHWLSEFTRFRRRKWNLNIWRKQCDRGTWVLKQLHGQLAKSMKSRLHLQLSLD	75
MQFKEVKAKSVEELHKLVNDLKAELWTLEFRNSTGSLEQTHKIPQLRKDIARALTALKQKEMETK	65
TTPACFTIGLGVGALFSAKFC	21
MKIEALNQEGNIYVIINGDFFVDMDEVTSEELVELLKKRYNMCDEVATHMACAIFSLSYVVE	62
RLVERIRQLTASLRQLIPQLIQYVRSLL	28
MNNPERLPSETHKPITGYLWGGLAVLTCPCHLPILAVVLAGTTAGAFLGEHWVIAALGLTGLFLLSLSRALRAFRERE	78
MNYLKMSEQKKKSNLNTELTAKLYLALDDLTMALATEDKEEVRKSEVFQKALEVVKVVKEMRRLQVKPAEGEEK	74
CKWEEAKVKFHCQRTQSQPGSW	22
RRICRCRIGRCLGLEVYFGVCFLHGRLARRCCR	33
MSQEAIIRSWKDPFSRENSTQNPAGNPFSELKEAQMDKLVGAGDMEAACTFTLPGGGGVCTLTSECIC	68

Figure 6.8: A screen shot of the output files from generate status tab of N-gram classification application.

6.3.6 Matching

This tab in the application will match back the produced prediction file from WEKA with sequences file from the generate sequences tab that divided into a given length with the original dataset file, see figure 6.9 below.

The user should upload 3 files from the selected directory:

1- Original sequence file: The main positive or negative AMP file before dividing it.

2-Divided sequence file: The output file from generate sequence tab that divide the sequences to a given length.

3- Predicted sequence file: This file is the output from WEKA after applying the ML algorithm.

This option allows the user to realize the original type of each sequence, either AMPs or Non-AMPs and what the ML classifiers predict it, each part of the divide sequences either changed or remained the same type.

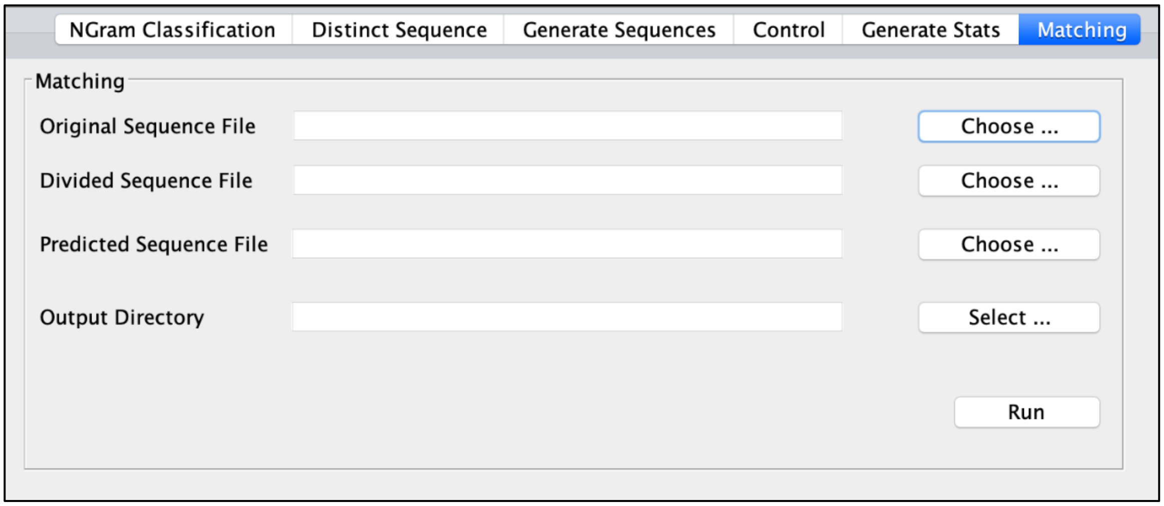
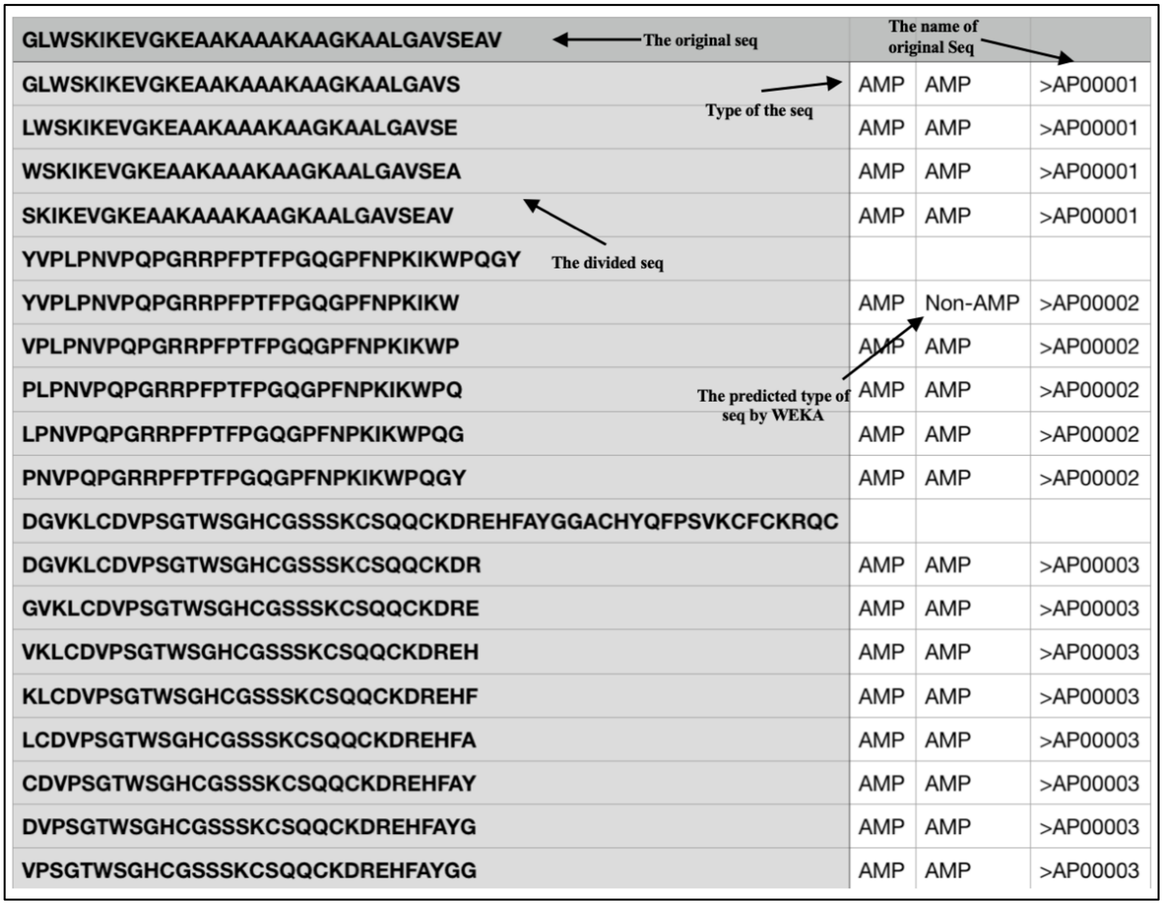


Figure 6.9: A screen shot of matching tab of N-gram classification application.

The output file will show in the first column the original sequence and all of its shopped sequences below the original one. The second column will be the original type of the sequence either, AMP or not. The third column will be the ML prediction type. The last column will be the original name of the sequence. See figure below 6.10.



GLWSKIKEVGKEAAKAAAKAAGKAALGAVSEAV			
GLWSKIKEVGKEAAKAAAKAAGKAALGAVS	AMP	AMP	>AP00001
LWSKIKEVGKEAAKAAAKAAGKAALGAVSE	AMP	AMP	>AP00001
WSKIKEVGKEAAKAAAKAAGKAALGAVSEA	AMP	AMP	>AP00001
SKIKEVGKEAAKAAAKAAGKAALGAVSEAV	AMP	AMP	>AP00001
YVPLPNVPQPGRRPFPTFPGQGPFNPKIKWPQGY			
YVPLPNVPQPGRRPFPTFPGQGPFNPKIKW	AMP	Non-AMP	>AP00002
VPLPNVPQPGRRPFPTFPGQGPFNPKIKWP	AMP	AMP	>AP00002
PLPNVPQPGRRPFPTFPGQGPFNPKIKWPQ	AMP	AMP	>AP00002
LPNVPQPGRRPFPTFPGQGPFNPKIKWPQG	AMP	AMP	>AP00002
PNVPQPGRRPFPTFPGQGPFNPKIKWPQGY	AMP	AMP	>AP00002
DGVKLCDVPSGTWSGHCGSSSKCSQQCKDREHFAYGGACHYQFPSVKCFCKRQC			
DGVKLCDVPSGTWSGHCGSSSKCSQQCKDR	AMP	AMP	>AP00003
GVKLCDVPSGTWSGHCGSSSKCSQQCKDRE	AMP	AMP	>AP00003
VKLCDVPSGTWSGHCGSSSKCSQQCKDREH	AMP	AMP	>AP00003
KLCDVPSGTWSGHCGSSSKCSQQCKDREHF	AMP	AMP	>AP00003
LCDVPSGTWSGHCGSSSKCSQQCKDREHFA	AMP	AMP	>AP00003
CDVPSGTWSGHCGSSSKCSQQCKDREHFAY	AMP	AMP	>AP00003
DVPSGTWSGHCGSSSKCSQQCKDREHFAYG	AMP	AMP	>AP00003
VPSGTWSGHCGSSSKCSQQCKDREHFAYGG	AMP	AMP	>AP00003

Figure 6.10: A screen shot of the output file from matching tab of N-gram classification application.

6.4 Conclusion and Chapter Summary

This chapter has presented N-gram classification, a useful tool for the AMPs research community for classifying sequences and proteomes for potential new synthesized AMPs sequences. Current predictors of AMPs use secondary structure analyses, multiple sequence alignments, distinctive residue compositions, or PSI-BLAST sequence profiles. These predictors require analyzing and comparing entire peptides sequences and take relatively longer time compared to N-grams, which decompose sequences into smaller chunks or parts, each of which can be readily analyzed quantitatively. To the best of our knowledge, it is the first prediction application to make predictions using a sequence-based model that involved alphabet reduction and N-gram analysis and produce files that are ready for ML classifier through WEKA. The application can handle user-requests with hundreds of thousands of AMPs and Non-AMPs, and report results in a reasonable time frame (typically less than 30 seconds).

The features of the application will assist the researcher in preparing the sequences file, remove redundancy, customize the length of the peptides, control the sequences length range, and get info of each peptide file. Of certain importance, is the need for a large dataset of confirmed Non-AMPs, so that the researcher of the medical field no longer has to rely on the negative datasets based on database keyword searches or homology as done in this dissertation.

Results from any experiments using this application, especially any incorrect predictions or classification, could help in making developments to our predictive models. In the meantime, the application will be continuously tested and improved, adding more features to be applied in the laboratory to synthesize AMPs targeting specific pathogens, directing us toward the more definite target when searching for alternatives to antibiotic treatments. We hope N-gram classification will be a beneficial tool for hypothesis generation in AMPs and antibiotics research. The application is currently available at http://www.binf.gmu.edu/mothman/N-gram-Classification-Application/. The application manual can be found at http://www.binf.gmu.edu/mothman/N-gram-Classification-Application/N-gram%20Classification%20manual.pdf

Chapter 7: Discussion and Future Directions

This dissertation has introduced a straightforward sequence method of AMP classification that would not only beat the success rates of earlier studies but also advance the sequence-based classification of AMP subclasses. In chapter 3, a detailed material about alphabet reduction and the N-gram analysis methods were provided, and those methods implemented in N-gram classification application.

In chapter 4, different experiments were made to uncover some of the AMPs sequential features. According to the results, the best-reduced alphabet letters to use for model classification was the ra3-29 that based on the residue pair counts for the BLOSUM50 matrix. Besides, the model performances of the ML algorithms indicate that an N-gram approach to differentiate between subclasses of AMPs, specifically antibacterial, antiviral, and antifungal peptides, is an efficient and effective method. RF significantly outperforms each of the other ML algorithms. This may have occurred because RF utilizes several unique decision trees, each with its own parameters. On other hands, some of ML algorithms like SVM and ANN have a significant issue with "black box" method, is a lack of transparency in how features are being applied to make predictions.

Furthermore, trigram with three-cluster alphabet reductions is simple enough for ML but sophisticated enough to the extent that loss of information in the original AMP sequences is minor. Feature selection of AMPs classification increased the model accuracy, decrease model training time, and reduce overfitting. Insertion of gabs between amino acids of AMPs captures some of the related correlations between neighboring and non-neighboring motifs.

The antimicrobial activity region of AMPs does not present in the entire sequence, and therefore longer peptide sequences have more extended regions missing AMP features. Nevertheless, the model for shorter length peptides had shown a higher percent accuracy than the unrestricted length of the peptides model. Consequently, AMPs tend to be short peptides compared to Non-AMPs.

In a real-world experiment, this sequence-based method achieved a high accuracy rate, which gives our model credibility and viability for the AMPs researcher community. However, more computational experiments are necessary to extend and corroborate the previous finding. More research would significantly benefit this study by providing explanations for a number of the exciting results, including the reasonably low model accuracy classifying antibacterial peptides against antifungal peptides. One reason for this discrepancy may be because that many fungi and bacteria have some similarities, such as unicellularity and acting as decomposers in an ecosystem, that impair the capability of ML models to distinguish between these subclasses.

Likewise, the transduction procedure used to reduce the possibility of overfitting our ML models by balancing datasets may not have been successful in small sample sizes as antiparasitic peptides. Because of this reason, the antiparasitic peptides were excluded from this study and require obtaining larger datasets when it is available in the future to meet the model criteria. Moreover, the need for a large dataset of confirmed Non-AMPs is appreciated. Because some of contamination of the negative may not discover, and no one can identify it, since not all the protein sequences were test their antimicrobial activity.

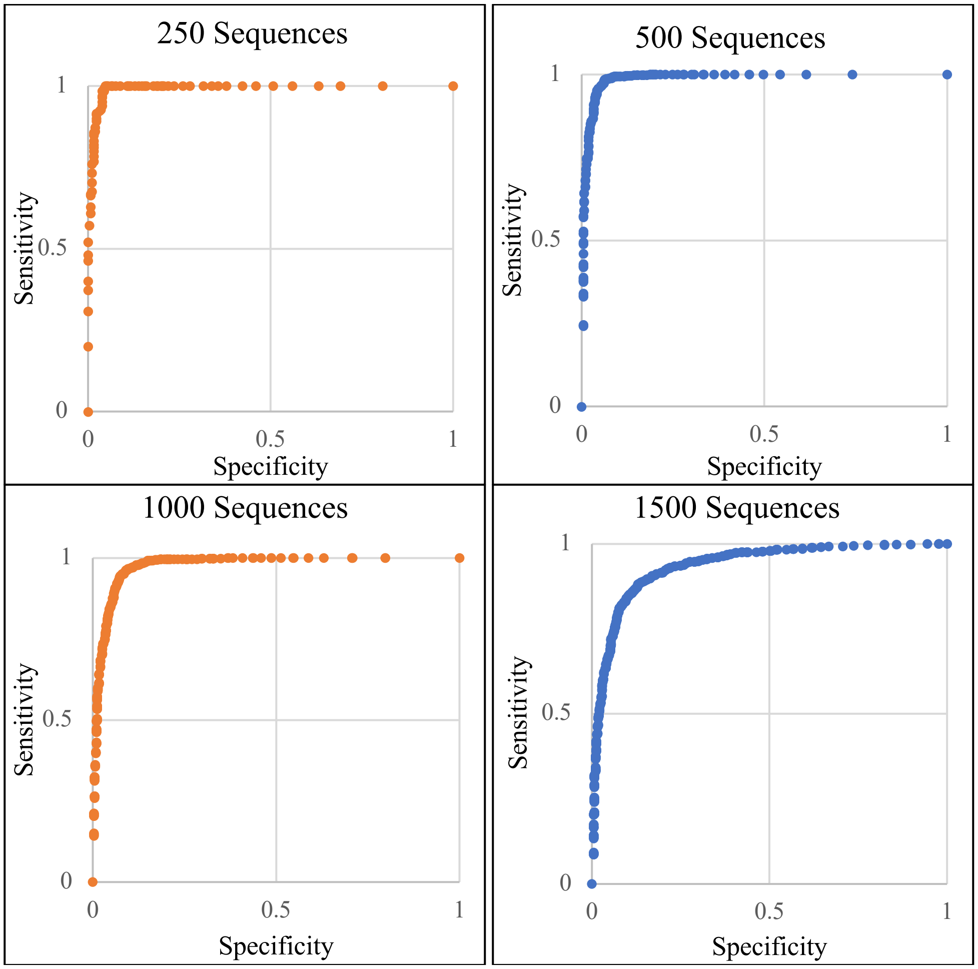
In this study, the models have much higher estimated accuracies than any other models that apply random guessing, and our maximum accuracies are comparable to those of previous studies by researchers running experiments with tangible AMPs in microbiological laboratories, as shown in chapter 5. These models consistently high in; auROC, MCC, sensitivity, specify, precision, and recall of over 80%, which validate the successful classification of the proposed method.

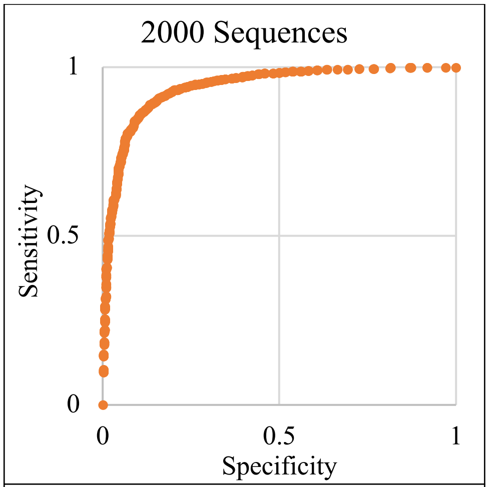
By continually improving the classification methods, biomedical researchers collaborating with other medical professionals would be able to advance the potential replacement of antibiotics with AMPs. In addition, by assessing the models with increased specificity, the innovation or synthesis of peptides to combat particular microbes becomes promising.

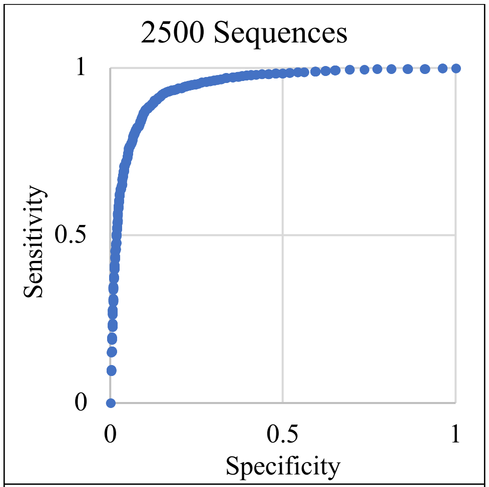
The outcomes of this research suggest that the classifiers produced a high predictive power and can be significantly used in numerous biological applications and saving thousands of lives. However, further computational experiments are still needed to provide support for the results.

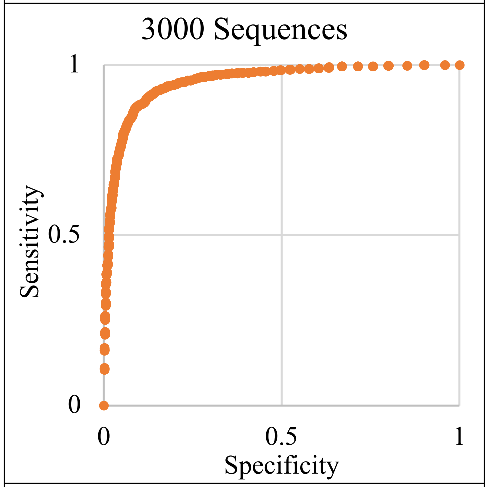
Finally, through this work, we make the above contributions available online through the application “N-gram Classification”. This application has different features that assist in building the models to be ready for ML classification, as detailed in chapter 6. Additional features can be added in the future, like finding class-specific motifs amongst different AMPs, and classification and prediction through AMPs secondary structure to combine it with the sequence classification. N-gram Classification is a free tool available at: http://www.binf.gmu.edu/mothman/N-gram-Cassification-Application/, which makes our novel method in this dissertation accessible and reproducible to all AMP researchers around the world.

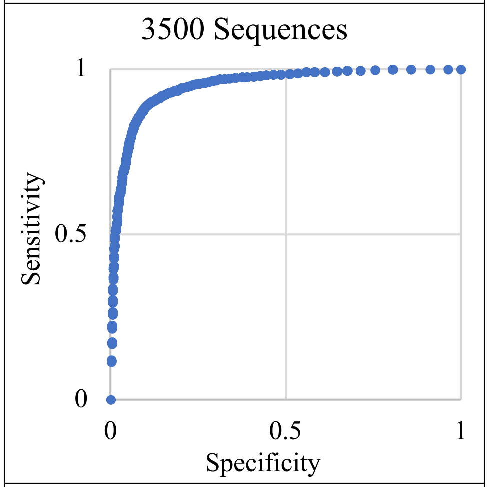
Appendix

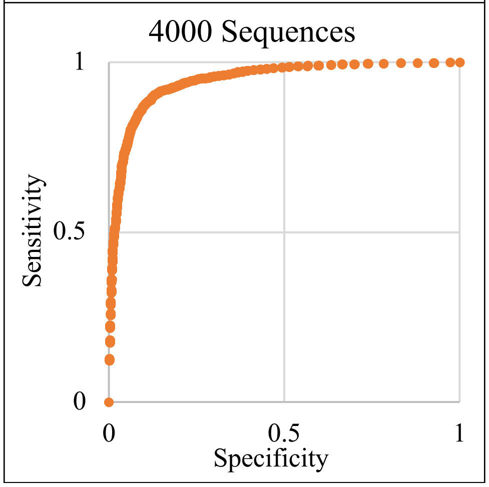

250 Sequences
Sensitivity
Specificity
1
0.5
0
0
0.5
1
500 Sequences
Sensitivity
Specificity
1
0.5
0
0
0.5
1
1000 Sequences
Sensitivity
Specificity
1
0.5
0
0
0.5
1
1500 Sequences
Sensitivity
Specificity
1
0.5
0
0
0.5
1

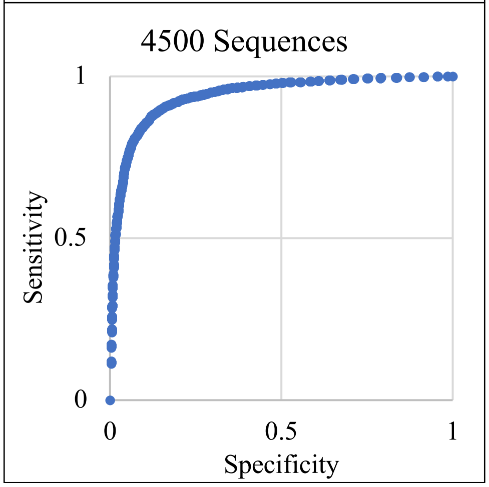

2000 Sequences
Sensitivity
Specificity
0
0.5
1
0
0.5
1


2500 Sequences
Sensitivity
Specificity
0
0.5
1
0
0.5
1


3000 Sequences
Sensitivity
Specificity
0
0.5
1
0
0.5
1


3500 Sequences
Sensitivity
Specificity
0
0.5
1
0
0.5
1


4000 Sequences
Sensitivity
Specificity
0
0.5
1
0
0.5
1


4500 Sequences
Sensitivity
Specificity
0
0.5
1
0
0.5
1

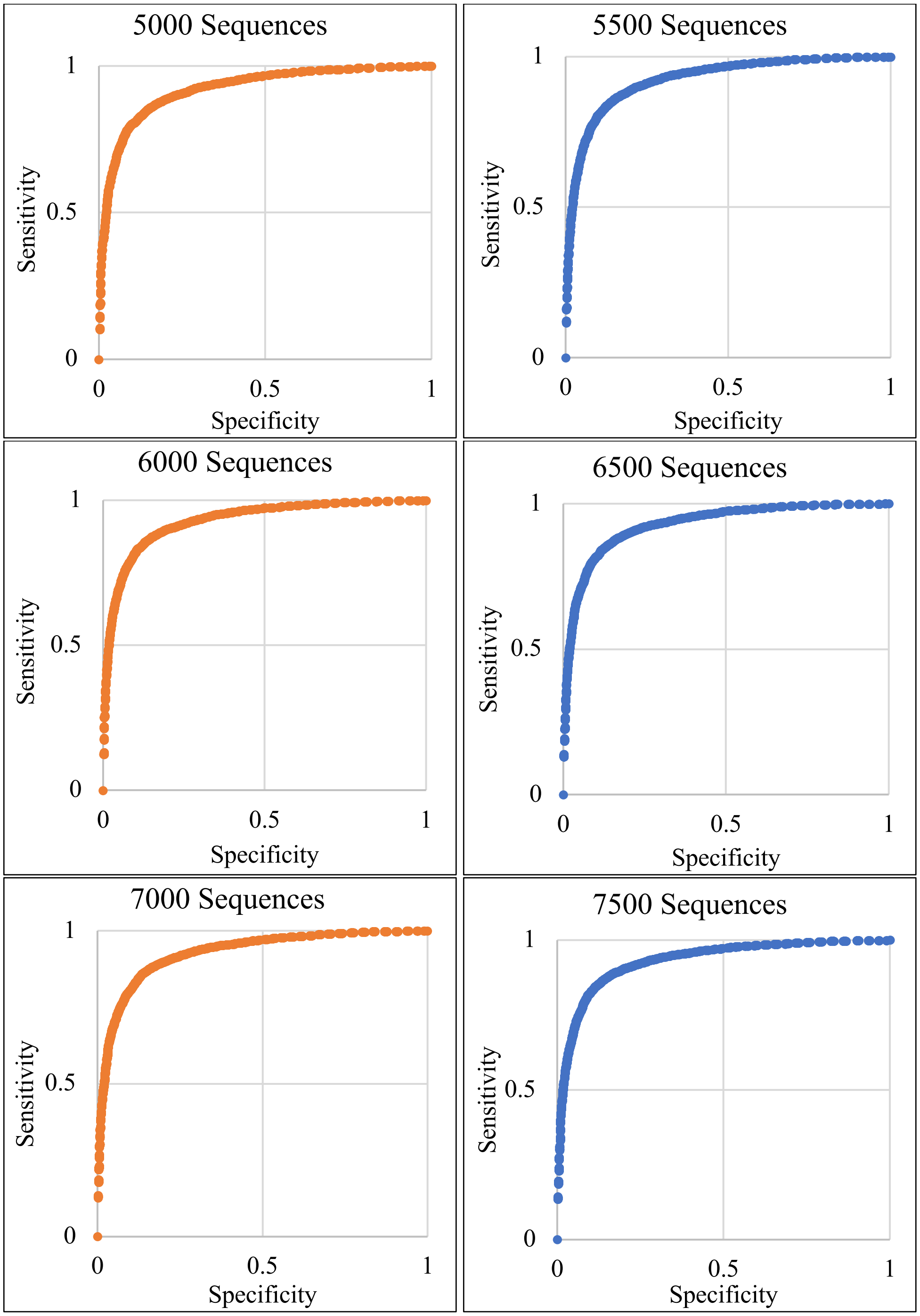


Figure A-1: ROC Curves of all 16 models on AMPs against Non-AMPs datasets using RF 10-fold cross validation, N-gram size of 3, and reduced alphabet ra3-29.

References

Amaral, A. C., et al. (2012). Predicting antimicrobial peptides from eukaryotic genomes: In silico strategies to develop antibiotics. *Peptides*, *37*(2), 301–308. https://doi.org/10.1016/j.peptides.2012.07.021

Aminov, R. I. (2010). A brief history of the antibiotic era: lessons learned and challenges for the future antibiotics and antibiotic resistance in the pre-antibiotic era. https://doi.org/10.3389/fmicb.2010.00134

Bacardit, J., Stout, M., Hirst, J. D., Sastry, K., Llorà, X., & Krasnogor, N. (2007). Automated alphabet reduction method with evolutionary algorithms for protein structure prediction. https://doi.org/http://doi.acm.org/10.1145/1276958.1277033

Bacardit, J., Stout, M., Hirst, J. D., Valencia, A., Smith, R. E., & Krasnogor, N. (2009). Automated alphabet reduction for protein datasets. *BMC Bioinformatics*, *10*(1), 6. https://doi.org/10.1186/1471-2105-10-6

Bahar, A. A., & Ren, D. (2013). Antimicrobial peptides. *Pharmaceuticals (Basel, Switzerland)*, *6*(12), 1543–1575. https://doi.org/10.3390/ph6121543

Bashir, S., Qamar, U., & Khan, F. H. (2016). IntelliHealth: A medical decision support application using a novel weighted multi-layer classifier ensemble framework. *Journal of Biomedical Informatics*, *59*, 185–200. https://doi.org/10.1016/j.jbi.2015.12.001

Cherkasov, A., & Artem. (2005). Inductive QSAR descriptors. distinguishing compounds with antibacterial activity by artificial neural networks. *International Journal of Molecular Sciences*, *6*(1), 63–86. https://doi.org/10.3390/i6010063

Daraei, A., & Hamidi, H. (2017). An efficient predictive model for myocardial infarction using cost-sensitive J48 model. *Iranian Journal of Public Health*, *46*(5), 682–692.

Alpaydin, E. (2004). Introduction to machine learning. The MIT Press

Fjell, C. D., Hiss, J. A., Hancock, R. E. W., & Schneider, G. (2011). Designing antimicrobial peptides: form follows function. *Nature Reviews Drug Discovery*, *11*(1), 37–51. https://doi.org/10.1038/nrd3591

Fjell, C. D., Jenssen, H., Hilpert, K., Cheung, W. A., Panté, N., Hancock, R. E. W., & Cherkasov, A. (2009). Identification of novel antibacterial peptides by chemoinformatics and machine learning. *Journal of Medicinal Chemistry*, *52*(7), 2006–2015. https://doi.org/10.1021/jm8015365

Frank, E., Hall, M., Trigg, L., Holmes, G., & Witten, I. H. (2004). Data mining in bioinformatics using Weka. *Bioinformatics*, *20*(15), 2479–2481. https://doi.org/10.1093/bioinformatics/bth261

Gao, V., Turek, F., & Vitaterna, M. (2016). Multiple classifier systems for automatic sleep scoring in mice. *Journal of Neuroscience Methods*, *264*, 33–39. https://doi.org/10.1016/j.jneumeth.2016.02.016

Gaynes R. (2017). The Discovery of Penicillin—New insights after more than 75 years of clinical use. *Emerging Infectious Disease*s, 23(5), 849–853. https://doi.org/10.3201/eid2305.161556

Giguère, S., Laviolette, F., Marchand, M., etal,. (2015). Machine learning assisted design of highly active peptides for drug discovery. *PLOS Computational Biology*, *11*(4), e1004074. https://doi.org/10.1371/journal.pcbi.1004074

Hajian-Tilaki, K. (2013). Receiver operating characteristic (ROC) curve analysis for medical diagnostic test evaluation. *Caspian Journal of Internal Medicine*, *4*(2), 627–635.

Hammami, R., & Fliss, I. (2010). Current trends in antimicrobial agent research: chemo- and bioinformatics approaches. *Drug Discovery Today*, *15*(13–14), 540–546. https://doi.org/10.1016/j.drudis.2010.05.002

Hoffman, M. R., Surender, K., Devine, E. E., & Jiang, J. J. (2012). Classification of glottic insufficiency and tension asymmetry using a multilayer perceptron. *The Laryngoscope*, *122*(12), 2773–2780. https://doi.org/10.1002/lary.23549

Hu, J. (2017). Automated detection of driver fatigue based on adaboost classifier with eeg signals. *Frontiers in Computational Neuroscience*, *11*, 72. https://doi.org/10.3389/fncom.2017.00072

Jenssen, H., Hamill, P., & Hancock, R. E. W. (2006). Peptide antimicrobial agents. *Clinical Microbiology Reviews*, *19*(3), 491–511. https://doi.org/10.1128/CMR.00056-05

Lata, S., Mishra, N. K., & Raghava, G. P. (2010). AntiBP2: improved version of antibacterial peptide prediction. *BMC Bioinformatics*, *11*. https://doi.org/10.1186/1471-2105-11-S1-S19

Li, T., Fan, K., Wang, J., & Wang, W. (2003). Reduction of protein sequence complexity by residue grouping. *Protein Engineering*, *16*(5), 323–330. https://doi.org/10.1093/protein/gzg044

Liaw, A., & Wiener, M. (2002). Classification and regression by random forest, R News 2, 18–22

Liu, X., Liu, D., Qi, J., & Zheng, W. M. (2002). Simplified amino acid alphabets based on deviation of conditional probability from random background. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, *66*(2), 4. https://doi.org/10.1103/PhysRevE.66.021906

Livingston, F. (2005). Implementation of Breiman's Random Forest Machine Learning Algorithm. in ECE591Q Machine Learning conference.

Maccari, G., Nifosì, R., & Luca, M. Di. (2013). Rational development of antimicrobial peptides for therapeutic use : design and production of highly active compounds. *Microbial Pathogens and Strategies for Combating Them: Science, Technology and Education*, 1265–1277. https://doi.org/10.13140/2.1.4408.6726

Magana, M., Pushpanathan, M., Santos, A. L. (2020). The value of antimicrobial peptides in the age of resistance. *The Lancet Infectious Diseases*. https://doi.org/10.1016/S1473-3099(20)30327-3

Masso, M. (2011). Sequence-based prediction of HIV-1 coreceptor usage. In *Proceedings of the 2nd ACM Conference on Bioinformatics, Computational Biology and Biomedicine.* https://doi.org/10.1145/2147805.2147841

Masso, M., & Vaisman, I. (2013). Sequence and structure based models of HIV-1 protease and reverse transcriptase drug resistance. *BMC Genomics*, *14 Suppl 4*(Suppl 4), S3. https://doi.org/10.1186/1471-2164-14-S4-S3

Melo, F., & Marti-Renom, M. A. (2006). Accuracy of sequence alignment and fold assessment using reduced amino acid alphabets. *Proteins: Structure, Function, and Bioinformatics*, *63*(4), 986–995. https://doi.org/10.1002/prot.20881

Murphy, L. R., Wallqvist, A., & Levy, R. M. (2000). Simplified amino acid alphabets for protein fold recognition and implications for folding. *Protein Engineering*, *13*(3), 149–152. https://doi.org/10.1093/protein/13.3.149

Ng, X. Y., Rosdi, B. A., & Shahrudin, S. (2015). Prediction of antimicrobial peptides based on sequence alignment and support vector machine-pairwise algorithm utilizing LZ-complexity. *BioMed research international,* 2015, 212715. https://doi.org/10.1155/2015/212715

Parikh, R., Mathai, A., Parikh, S., Chandra Sekhar, G., & Thomas, R. (2008). Understanding and using sensitivity, specificity and predictive values. *Indian Journal of Ophthalmology*, *56*(1), 45–50. https://doi.org/10.4103/0301-4738.37595

Peters, B. M., Shirtliff, M. E., & Jabra-Rizk, M. A. (2010). Antimicrobial Peptides: Primeval Molecules or Future Drugs? *PLoS Pathogens*, *6*(10), e1001067. https://doi.org/10.1371/journal.ppat.1001067

Peterson, E. L., Kondev, J., Theriot, J. A., & Phillips, R. (2009). Reduced amino acid alphabets exhibit an improved sensitivity and selectivity in fold assignment. *Bioinformatics*, *25*(11), 1356–1362. https://doi.org/10.1093/bioinformatics/btp164

Phoenix, D. A., Dennison, S. R., & Harris, F. (2013). Antimicrobial peptides. Weinheim, Germany: Wiley-VCH Verlag GmbH & Co. KGaA. https://doi.org/10.1002/9783527652853

Solis, A. D. (2015). Amino acid alphabet reduction preserves fold information contained in contact interactions in proteins. *Proteins: Structure, Function, and Bioinformatics*, *83*(12), 2198–2216. https://doi.org/10.1002/prot.24936

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *Journal of Chemical Information and Computer Sciences*, *43*(6), 1947–1958. https://doi.org/10.1021/ci034160g

Taboureau, O., Olsen, O. H., Nielsen, J. D., Raventos, D., Mygind, P. H., & Kristensen, H.-H. (2006). Design of Novispirin Antimicrobial Peptides by Quantitative Structure–Activity Relationship. *Chemical Biology Drug Design*, *68*(1), 48–57. https://doi.org/10.1111/j.1747-0285.2006.00405.x

Tarca, A. L., Carey, V. J., Chen, X., Romero, R., & Drăghici, S. (2007). Machine learning and its applications to biology. *PLoS Computational Biology*, *3*(6), e116. https://doi.org/10.1371/journal.pcbi.0030116

Taylor, J. R. John R. (1997). An introduction to error analysis : the study of uncertainties in physical measuremen*ts*. *Physics Today.* 51,1,57. https://doi.org/10.1063/1.882103

Veltri, D., Kamath, U., & Shehu, A. (2015). Improving recognition of antimicrobial peptides and target selectivity through machine learning and genetic programming. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1–1. https://doi.org/10.1109/TCBB.2015.2462364

Ventola, C. L. (2015). The antibiotic resistance crisis: part 1: causes and threats. *P & T : A Peer-Reviewed Journal for Formulary Management*, *40*(4), 277–283. PMCID: PMC4378521

Wade, D., & Englund, J. (2002). Synthetic antibiotic peptides database. *Protein and Peptide Letters*, *9*(1), 53–57. https://doi.org/10.2174/0929866023408986

Wang, G. (2010). *Antimicrobial peptides : Discovery, design and novel therapeutic strategies*. *Advances in molecular and cellular microbiology*. CABI: Oxfordshire, UK

Wang, G. (2016). Structural analysis of amphibian, insect, and plant host defense peptides inspires the design of novel therapeutic molecules. In *Host Defense Peptides and Their Potential as Therapeutic Agents* (pp. 229–252). Cham: springer international publishing. https://doi.org/10.1007/978-3-319-32949-9_9

Wang, J., & Wang, W. (1999). A computational approach to simplifying the protein folding alphabet. *Nature Structural Biology*, *6*(11), 1033–1038. https://doi.org/10.1038/14918

Wang, P., Hu, L., Liu. et al. (2011). Prediction of antimicrobial peptides based on sequence alignment and feature selection methods. *PLoS ONE*, *6*(4), e18476. https://doi.org/10.1371/journal.pone.0018476

Zhao, M.-J., Edakunni, N., Pocock, A., & Brown, G. (2013). Beyond fano's inequality: bounds on the optimal F-score, BER, and cost-sensitive risk and their implications. *Journal of Machine Learning Research*, 14:1033–1090, 2013.

Biography

Manal Othman earned her bachelor's degree in Medical Laboratory Technology from King Abdulaziz University, Jeddah Saudi Arabia, in 2010. After that, Othman earned her master's degree in Biomedical Informatics from the College of Osteopathic Medicine from Nova Southeastern University. Florida, USA., with honors. Later, she worked as Medical Bioinformatics Lecturer at the College of Medicine in Princess Nourah bint Abdulrahman University, Riyadh, KSA. Othman received a scholarship from the Ministry of Education in Saudi Arabia to study at the Bioinformatics and Computational Biology Ph.D. program at George Mason University in 2015.

In 2016, she received PHI BETA DELTA Honor Society for International Scholar for outstanding academic achievement. George Mason University. Virginia, USA. Then in Feb 2019, she become a member of Golden Key International Honor Society.

Othman published two abstracts in ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics. The first abstract in 2017 about Classification and Prediction of Antimicrobial Peptides Using N-gram Representation and Machine Learning. Next, in 2018 about Machine Learning Classification of Antimicrobial Peptides Using Reduced Alphabets.

Othman received the student research day Spring 2020 award for outstanding poster presentation.