Proceedings of the International Conference on
Electrical Engineering and Informatics
Institut Teknologi Bandung, Indonesia June 17-19, 2007

Extraction of Defensin Antimicrobial Peptide from SWISS-PROT database using Extended Boyer-Moore Algorithm

N.H Ahamed Hassain Malim and Z. Zainol

School of Computer Sciences, Universiti Sains Malaysia, 11800, Penang, Malaysia

Antimicrobial Peptide (AMP) is a subset of protein that plays an essential role in innate immunity system. Researches on AMP are actively conducted in Immunology field where synthetics antibiotics are being developed. There are several classifications of AMP's families with different mechanism in immobilizing pathogens. Thus, family classification could help speed up a search for specific family AMP. Efforts have been made by active bioinformaticians to build databases to gather AMP data. However, explicit family classification is not featured in these databases due to the method they used for extracting AMP data which is based on literature mining using keywords such as "anticancer", "antibacterial" and etc. This research aimed to find AMP family's characteristics and method that enable us to extract AMP data based on its motif. Based on sequence signature resulted from Multiple Sequence Alignment on known antimicrobial peptide sequences from other research work, we proposed an Extended Boyer-Moore Algorithm (EBMA) as a method of extraction for Defensin AMP data from SWISS-PROT database. We constructed a local data mart specifically to store data extracted by EBMA. The evaluation of this algorithm shows that it is accurate in extracting Defensin AMP, fast and reliable.

1. Introduction

Proteins are the basic foundation of an organism. Besides being the building blocks of enzymes and genes, proteins may also serve as the second line of body defense. These proteins are called antimicrobial proteins(2). Antimicrobial protein also known as antimicrobial peptide(AMP) are isolated from fungi, plants, invertebrates and vertebrates. They are heterogeneous in length, sequence and structure, but most of them are small, cationic and amphiphatic. There are three important groups of human AMP which are defensins, cathelicidins and histatins(5). As stated by Smet et al(5), AMP is an important component of the natural defenses of most living organism. It plays essential roles in non-specific host defenses by preventing or limiting infections via their ability to selectively recognize potential pathogens. It possesses not only antibacterial and antifungal properties but also anticancer and antiviral properties.

Increasing resistance of bacteria and fungi to the commonly used antibiotics lead to the growing interest in peptide antibiotics(5). Thus, synthetic peptide antibiotics have been widely used as the alternative, driven by the awareness of the potential therapeutic application of these peptides or their synthetic analogues(5). Many structures and sequences of antimicrobial peptides have been solved experimentally and stored in biological databases. However to extract it, one will need to know the ID or the specific sequence of the stored data.

Many researches conducted in bioinformatics are focused to either protein-protein, protein-family or protein-function identification, classification, prediction etc. It is very unlikely to come across research in this area that identify and classify proteins into antimicrobial compound. Therefore, in this research, we shift our focus into the field of immunology, in which synthetics antibiotics are being developed based on the findings of corresponding AMP. Yount et al(7) performed multiple sequence alignment (MSA) on known antimicrobial peptide sequences and suggested several characteristics that could be used in this field. With the current format of AMP databases, immunologist may face difficulties to search for particular family that may contribute to overcome particular illness without knowing the exact sequence. Thus, we intend to extract antimicrobial peptide data from biological databases using data mining techniques based on the characteristics outlined by (7) and store them into a relational data mart specifically for Defensin, a family of AMP.

The following section reviews related work employed in data mining techniques. Section 3 presents the overview of method used to extract AMP data followed by discussion of the implementation and results in section 4. Finally conclusions and future work are discussed in section 5.

2. Related work

Since large volume of experimental data related to protein is stored in the fast-growing PubMed literature, researchers are performing literature mining to extract information that are not available in biological databases. There are several established databases that were built based on data that is extracted using Literature Mining. Among them are Antimicrobial Peptide Database (6), ANTIMIC(1), Synthetic Antimicrobial Peptide Database(9) and Antimicrobial Sequences Database(10).

Literature Mining is a technique to find intended literature and extract objects from it. It is also known as Text Mining. It incorporates Natural Language Processing (NLP) to process the text extracted. Hu et al(8) use this technique to design a Rule-based Literature Mining System for Protein Phosphorylation (RLIMS-P). This system find text intended from full-length articles in PubMed using text matching and processed them into sentences and tagged them. The tagged sentences then undergo entity and term recognition process that includes acronym detection. The next steps are the phrase detection, semantic type classification, relation identification and data extraction. This system managed to extract proteins that involve in phosphorylation and their target site. In the case of data gathering for AMP database, the terms mined were "antifungal", "anticancer", "antibacterial" and "antiviral" (6). Literature mining enables

information extraction from wide range of sources regardless literature or data sources.

3. Methodology

Our general framework is divided into three phases as shown in figure 1. The first phase is the determination of Defensin antimicrobial peptides characteristics in which we performed a short study on researches within the area of immunology and antimicrobial agent. The second phase is the extraction of Defensin antimicrobial peptide using Pattern Matching Algorithm. The final phase is to store the output of the second phase which is a set of protein data that is classified as Defensin antimicrobial peptides into a data mart. Each phase will be discussed in detail in the following sections.

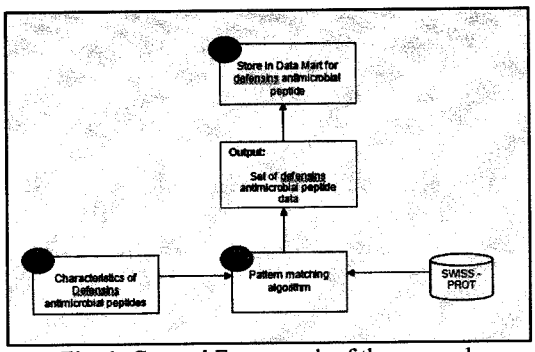

Fig. 1. General Framework of the research

3.1. First Phase: Determination of antimicrobial peptides characteristics

AMP are categorized by (7) into two classes. The first class contains AMP with disulfide bridges and the second class contains AMP without disulfide bridges. Note that, the research conducted by (7) was focused AMP with disulfide bridges. They performed a Multiple Sequence Alignment (MSA) on several known AMP from various sources such as human, animal and plants. As a result, they suggested a sequence signature called y-core motif as shown in figure 2. There are three characteristics that we derived from (7) for AMP as follows:

1. Length of AMP is in between the range of 10 -110 residues
2. AMPs are cationic
3. Presence of β-hairpin (a.k.a y-core (**γ-core**)) motif with sequence signature:
   NH_2---[X_{1-3}]- [GXC]-[X_{3-9}]-[C]---COOH

Thus, we will be using the sequence signature suggested by (7) (Figure 3) as our pattern to extract Defensin AMP from SWISS-PROT database.

$$NH_2\cdots[X_{1-3}]\text{-}[GXC]\text{-}[X_{3-9}]\text{-}[C]\cdots COOH$$

Fig. 2. Sequence signature suggested by (7).

3.2. Second Phase: Extraction of Defensin Antimicrobial Peptide using Pattern Matching Algorithm

In this phase, we used Extended Boyer-Moore Algorithm (EBMA) to perform pattern matching for the pattern shown in Figure 3. The original Boyer-Moore Algorithm (BMA) is an exact string matching algorithm that compares a pattern from right to left and generate a shift value regardless the occurrence of a match or mismatch which help to detect the occurrence of pattern easily and faster (3). Since the pattern we used actually denotes that any atom could be at the position 1 to 3 before G (X_{1-3}), and in between G and C (GXC), and position 3 to 9 after C (X_{3-9}) followed by another C, we made some modification to the BMA. The first modification is the insertion of the Pre-Pattern Matching step which aims to prepare the sequence and pattern to fit into the BMA procedure as at presence it has been working with a determined pattern. Where else, we are providing it with a half-determined pattern (X – could be any letter code for amino acids) to be matched. The second modification is inside the Pattern Matching step where we used Find Pattern Routine (FPR) with different sequence and pattern input. The output of this step is a set of Defensin AMPs' sequences that will be stored in Defensins AMP Data Mart. The procedure is divided into several sub-procedures as follows:

1. Data Downloading and Manual Filtering
2. Pre–Pattern Matching
3. Pattern Matching

3.2.1. Data Downloading and Manual Filtering

Data from SWISS-PROT database is downloaded into our local machine as a data file format. We divided it into a smaller file. Due to the first condition of Defensin AMP characteristics which state that the length of sequence is in range 10 - 110. Thus, we chose (filter) records which length are less than 200 and copied them into data file. This is being done manually due to the inconsistent length of each record. We managed to copy 600 records from the downloaded data into our data file.

3.2.2. Pre-Pattern Matching Steps

Our second step is to read one record at a time from input file to perform pattern matching. For each record, we will first scan for the "ID" notation and look at column 42 – 45 that is allocated for the sequence length. This information is read and matched with the first condition of AMP which states that the length of the peptide should be in the range of 10-110. If this condition is fulfilled then the next notation to be scan is the "(blank)" notation that is allocated for the peptide's sequence. This sequence is read and blank spaces that occur within the sequence are eliminated. This sub-procedure is called Pre–Pattern Matching step in which sequence and pattern are being prepared for insertion into the Find Pattern Routine (FPR) in Pattern Matching sub-procedure. Preparation of sequence is done by eliminating the blank spaces as mentioned before where else the preparation of pattern is done by dividing the pattern into three short patterns namely "G", "C" and "C". These short

patterns are the determined letter-code in the pattern. Figure 3 shows how the short patterns are produced.

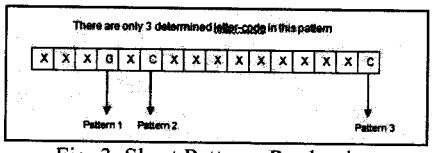

Fig. 3. Short Patterns Production

3.2.3. Pattern Matching Steps

Pattern 1 and the sequence are inputted into the first FPR. This routine actually uses BMA to find Pattern 1 (i.e "G") in the sequence. This routine will return an array of sequence starting from the index after it found "G" to the end of the sequence. This returned array is being used in the next pattern matching step. If FPR fails to find Pattern 1 in the sequence it will return a false value which explicitly indicates that the current record being read is not a Defensin AMP. Thus it will start reading the next record from the file.

The next step of this sub-procedure is to find Pattern 2 (i.e "C"). The input sequence is the array which was returned by the first FPR. Again, using BMA this routine finds Pattern 2 in the sequence. It returned and array starting from index after it found "C" to the end of the sequence. This index is later examined to determine whether index or location of Pattern 2 is actually the sum of index Pattern 1 plus 2. If the index of Pattern 2 does not satisfy the condition, the second FPR is repeated. In the case of failure to find Pattern 2, FPR returns the false value and the next record is read. Note that, the false value indicates that the record being read is not a Defensin AMP.

The final step is to find Pattern 3 (i.e "C"). The array returned by second FPR is being used as the input for this step. Note that, each time FPR returned an array and the length of each array is actually decreasing every time it is returned. The third FPR also uses the BMA to find Pattern 3. In the case of an array is returned, the index of Pattern 3 that is found is checked whether it is in the range of 3 to 9 letter-code away from Pattern 2. If it is, then the record is being extracted into our local data mart as it is proven to be a Defensin AMP. On the other hand, if it is not in the range the third FPR is repeated. In the case of failure, the false value is returned.

3.2.4. Find Pattern Routine (FPR)

The core procedure of the Pattern Matching steps is the FPR. This routine accepts two inputs. The first input is the pattern and the second input is the sequence. Upon receiving these inputs, this routine calls BMA to perform exact string matching on them. BMA works by matching the pattern with the sequence by comparing characters in a right-to-left manner. Since we are interested with the proceeding sequence after the match that will be used as the input for next FPR instead of the preceding one, we change slightly the BMA so that it will stop once a match is found and return the sequence along with the pattern index. FPR will accept the sequence and truncate the preceding part but maintain the proceeding part of the sequence that are not yet matched (Figure 4). It will set it as a new sequence input for the next FPR. This will further speed up the matching process as the remaining sequence is actually shorter than the original input.

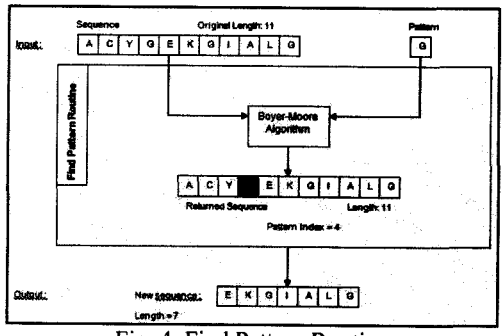

Fig. 4. Find Pattern Routine

3.3. Third Phase: Storing data in Defensin AMP Data Mart

We create a local data mart to store extracted Defensin AMP. The objective of this phase is to create a relational data mart specifically for Defensin antimicrobial peptide. The local data mart is called Defensin AMP Data Mart. Information will be passed into this data mart step by step. Once a sequence detected matched with the signature in phase two, its information will be stored into data mart.

4. Implementation

We implemented our algorithm in C using Microsoft Visual C++ compiler on our local machine The Defensin AMP Data Mart is also built on the same machine using Microsoft SQL Server 2000. The GUI is developed using Macromedia Dreamweaver MX and the input files for EBMA are read using UltraEdit-32 Professional Text/Hex Editor.

5. Results

We tested EBMA based on three criteria that are accuracy, timing and reliability.

5.1. Accuracy

Accuracy test of EBMA is done by comparing the data inside Defensin AMP Data Mart with Antimicrobial Peptide Database (APD) and ANTIMIC. APD and ANTIMIC store AMP data extracted from PubMed, GenBank and SWISS-PROT via Literature Mining method (1, 6). Data inside APD varies and are not formally classified into a specific family. ANTIMIC on the other hand, provide the family classification in the feature section. Thus, we tested for the existence of AMP data extracted regardless their family classification in these two databases and we also checked for their availability and classification as Defensin in ANTIMIC. The test is conducted by choosing 100 sequences from our local data mart and check for their availability in the two AMP databases.

The result shows that sequences tested exist in both databases and are classify as Defensin in ANTIMIC. We also compared our result to the result returned by PROSITE when this signature is used as a motif. We found out that our

result is slightly better than PROSITE where by 99% of our hits are of Defensins family compared to PROSITE's hits which is only 92.5%.

Thus, we concluded that the pattern (sequence signature) we used as input for EBMA is accurate and sufficient in order to classify a sequence as Defensin AMP. We also concluded that EBMA is accurate in detecting Defensin AMP pattern which exists in a sequence.

5.2. Timing

We tested our algorithm with different number of records from 100 – 1500. The result shows that EBMA's speed increased by 1 second for each 300 records being processed. This is because of the size of array processed is decreased each time FPR execute. Since the graph (Figure 5) produced is a linear graph, we concluded that the complexity of EBMA is in order N.

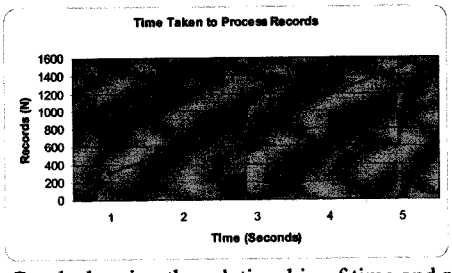

Fig. 5. Graph showing the relationship of time and records

5.3. Reliability

We collect a number of records which length is less than 110. This collection contains a mixture of Defensin AMP and other non-AMP sequence. We used these sequences to test our algorithm reliability. Table 1 shows the list of sequence tested and the output of our algorithm. This test is conducted to ensure that EBMA does not return any false-positive result that would jeopardize its reliability.

Table 1. List of sequence tested and the output of EBMA

1	PIAQIHILEGRSDEQKETLREYSEAISRSLDAPLT SVRVITEMAKGHFGIGGELASKVRR	Non-AMP	Pattern Not Found
2	GIFSSRKCKT VSKTFR[redacted]	Defensin	** Pattern Found ** G at index : 16 C at index : 18 C at index : 25
3	MAFLKKSLFL VLFLGLVPLF LCENEKREGE NERGENDDQG EEKRSLGSFM KGVGKGLATV GKIVADQFGK LLEAGQG	AMP but not Defensin	Pattern Not Found
4	PIAGLYLDG RSNEQRETLI REVSEAMSRS LDAPIERVRV IITEMPKNHF GIGGEPASKL NR	Non-AMP	Pattern Not Found
5	GLWKSLLKNV GVAAGKAALN AVTDMVNQ	AMP but not Defensin	Pattern Not Found
6	ALWFTMLKKL GTMALHAGKA ALGAAANTIS QGTQ	AMP but not Defensin	Pattern Not Found
7	ATCDILSFQS QWVTPNHAGC ALHCVIKGYK G[redacted]	Defensin	** Pattern Found ** G at index : 32 C at index : 34 C at index : 39
8	GLLASLGKVF GGYLAEKLKP K	Amphibian defense peptide	Pattern Not Found
9	ATCDALSFSS KWLTVNNHSAC AIHCLTKGYK G[redacted]	Defensin	** Pattern Found ** G at index : 32 C at index : 34 C at index : 39
10	VLCFAWLFLD VLFSTASMFH LCAISVDRYI AIKKPIQANQ YNSRATAFIK ITVVWLISIG	Non-AMP	Pattern Not Found

Based on Table 1, we concluded that EBMA is reliable to be used as an algorithm specifically to extract Defensin AMP data. It is able to distinguish between Defensin AMP sequences and others that are not.

6. Conclusion and Future Work

With the ability to decrease the array being matched each time a pattern matching procedure is done, EBMA successfully extracted Defensin AMP data from SWISS-PROT without returning any false-positive data.

EBMA can be further modified to extract other AMP families' data from any biological databases by changing the pattern input and several static conditions. Defensin AMP Data Mart can be expanded to store various family of AMP in a classified manner. It could also be added with facilities that would support 2D molecular structure data rather than sequence data as at present.

References

(1) Brahmacary, Krishnan, Koh, Khan, Seah, Tan, Brusic, Bajic. 2004. ANTIMIC: a database of antimicrobial peptide sequences. Nucleic Acid Research 32: pg 586-589

(2) Campbell, Reece and Mitchell. 1999. *Biology Fifth Edition*. Addison-Wesley.

(3) Cormen, Leiserson, Rivest. 1995. Introduction to Algorithms.McGraw-Hill.

(4) Frecer, Ho, Ding. 2004. De Novo Design of Potent Antimicrobial Peptides, Antimicrobial Agents and Chemotheraphy 48 (9): pg 3349 - 3357.

(5) Smet, Contreras. 2005. Human antimicrobial peptides: defensins, cathelicidins and histatins. Biotechnology Letters 27: pg 1337 - 1347.

(6) Wang, Wang. 2004. APD: the Antimicrobial Peptide Database, Nucleic Acids Research 32: pg 590 - 592.

(7) Yount, Yeaman. 2004. Multidimensional signatures in antimicrobial peptides. Proceedings of the natinal Academy of Sciences of the United States of America 101 (19) : pg 7363 - 7368.

(8) Hu, Narayanaswamy, Ravikumar, Vijay-Shanker, Wu. 2005. Literature Mining and database annotation of protein phosphorylation using a rule-based system. Bioinformatics 21(11): pg 2759-2765.

(9) Synthetic Antimicrobial Peptide Database, available online at URL http://oma.terkko.helsinki.fi:8080/~SAPD

(10) Antimicrobial Sequence Database, available online at URL http://www.bbcm.univ.trieste.it/~tossi/pag1.htm