# User Guide of DeepDigest

Version 1.7.0

Last revised February 13, 2023

Jinghan Yang

## **Contents:**

1 Introduction	2
2 Installation	2
2.1 Environment Requirements	2
2.2 Downloading DeepDigest	2
3 Running DeepDigest	3
3.1 Parameters	3
3.2 Running the command	4
4 Output file formats	4

#### **1** Introduction

DeepDigest is a Python-based command line tool, which integrates convolutional neural networks (CNNs) and long-short term memory (LSTM) networks to predict the proteotypic cleavage sites for eight commonly used proteases including trypsin, ArgC, chymotrypsin, GluC, LysC, AspN, LysN and LysargiNase. DeepDigest is freely available at <u>http://fugroup.amss.ac.cn/software/DeepDigest/DeepDigest.html</u>.

### **2** Installation

#### **2.1 Environment Requirements**

DeepDigest is a program based on Keras using TensorFlow backend. Here, Python (3.5), TensorFlow (1.10.0) and Keras (2.2.4) are required. Users should also make sure that all the following packages are installed in the Python environment: os, sys, re, getopt, numpy ( $\geq$ 1.14.5). For convenience, we strongly recommend users to install the <u>Anaconda 3</u> version (64-bit) or above in your local computer, and all the packages can be installed through pip.

#### 2.2 Downloading DeepDigest

DeepDigest can be freely downloaded from http://fugroup.amss.ac.cn/software/DeepDigest/DeepDigest.html. Users can download the release version of DeepDigest "DeepDigest.zip" (Figure 1.a), the help document "User Guide of DeepDigest.pdf" (Figure 1.b) and the test dataset "TestData.zip" (Figure 1.c) from this website.

	DeepDigest		
	INTRODUCTION DOWNLOAD PUBLICATION CONTACT		
	Introduction		
	DeepDigest is a deep learning-based tool to predict the proteotypic cleavage sites and peptide digestibility for eight commonly to proteases (trypsin, ArgC, chymotrypsin, GluC, LysC, AspN, LysN, and LysargiNase). It integrates convolutional neural networks long-short term memory networks to achieve high accuracy and robustness. Compared with traditional machine learning algorith (logistic regression, random forest, and support vector machine), DeepDigest has superior performance for all the eight proteases.		
	Input     Endedding     Consultant     Asse Rodding     Consultant       21 10 termin     21 10 termin     21 10 termin     21 10 termin       NR     31 certification     21 10 termin     21 10 termin       NR     31 certification     21 certification     21 certification       NR     31 certification     21 certification     21 certification		
	$\begin{array}{c} x_{0} \\ x_{0} \\$		
	Download		
	DeepDigest is written in Python. Please click the link below to download the DeepDigest 1.6.0 binary release:		
a	DeepDigest for Windows		
	v1.6.0 Release (2022.7)		
	Notes:		
	This is a command line tool for eight proteases (Trypsin, ArgC, Chymotrypsin, GluC, LysC, AspN, LysN, and LysargiNase). Please choose the consistent model for prediction.		
b	There is a user guide for this command line tool.		
c	Download the test dataset <u>here</u> .		
	Publication		
	Contact Address: No.55 Zbongguancun East Road, Haidian District, Beijing, China		
	Postcode: 100190		
	Any problem with the software or this website, please contact: Yan Fu's Research Group $\bigotimes$		
	© copyright (C) 2020 by Fu Group, Academy of Mathematics and Systems Science, CAS		

Figure 1. The screenshot of the DeepDigest website.

## **3 Running DeepDigest**

## **3.1 Parameters**

Users can set the custom parameters which are annotated in Table 1:

Parameter name	Meaning

input	The path of protein sequence file in FASTA format (.fasta)	
output	The path of output file in TXT format (.txt)	
regular	The regular expression used to extract the protein id	
	(default: ">(.*?)\s")	
protease	The digestion protease (default: Trypsin)	
missed_cleavages	The maximum number of missed cleavages allowed in each	
	theoretical peptide fragment (default: 2)	
min_len	The minimum length of the theoretical peptide fragments	
	(default: 7)	
max_len	The maximum length of the theoretical peptide fragments	
	(default: 47)	

#### 3.2 Running the command

Open the command interpreter "cmd.exe" and run DeepDigest in the path of the command line tool by the following format:

python the\_main.py --input=*the path of protein sequence file* --output=*the path of output file* --regular=">(.\*?)\s" --protease=Trypsin --missed\_cleavages=2 --min\_len=7 --max\_len=47

An example is as follows (Figure 2.):

E:\DeepDigest>python the\_main.py --input=E:\DeepDigest\nextprot-sparql-entry\_PE3.fasta --output=E:\DeepDigest\PredictResultsOfPE3\_Trypsin.txt --regular=">(.\*?)\s" --proteas e=Trypsin --missed\_cleavages=2 --min\_len =7 --max\_len=47\_

Figure 2. Illustration of running DeepDigest.

## 4 Output file formats

Once the calculation is done, DeepDigest generates a .txt file in the output directory. The detailed description of each column in this file is shown in Table 2.

 Table 2. Descriptions of headers in the result file.

Name	Description

Protein id	The identity of the protein from which the peptide
	is digested
Peptide sequence	The sequence of the theoretical digested peptide
Digestibility of the N-terminal	The predicted cleavage probability of the
site	cleavage site on the N-terminal of the peptide
Digestibility of the C-terminal	The predicted cleavage probability of the
site	cleavage site on the C-terminal of the peptide
Digestibility of the missed site(s)	The predicted cleavage probabilities of the missed
	cleavage sites in the peptide