Project Acronym:	**IDPfun**
Project Full Title:	**Driving functional characterization of intrinsically disordered proteins**
Grant Agreement:	**778247**
Project Duration:	**48 months (01/03/2018 - 28/02/2022)**

Deliverable D1.5

New version of the Mobi software

Work Package:	**WP1**
Lead Beneficiary:	**UNIPD**
Due Date:	**29 February 2020 (M24)**
Submission Date:	**28 February 2020**
Deliverable Type:	**D**
Dissemination Level:	**P**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 778247

Table of Content

Executive summary

This document describes the *"D1.5 New version of the Mobi software"* for the IDPfun project. The deliverable is an update on the software that automatically extracts annotations for intrinsic disorder and LIPs from experimental data.

List of Acronyms

Acronym	Definition
LIP	Linear Interactive Peptide
PDB	Protein Data Bank

Project overview

Linear Interacting Peptides, abbreviated as LIP, is a generic name used to refer to those peptides that makes interactions in a linear conformation. They are called peptides cause they are often found in chains shorter than 50 residues and even shorter when considering only the interaction fragment. The interacting property refers to the ability of the peptide to form chemical interactions, which in general are not that stable as in globular complexes, with others chains. The property of making temporary weak interactions is very useful in regulation or signaling. The example in Figure 1 (pdb id 1jsu) shows a LIP in red (chain C) and the interacting partners in green (chain A) and blue (chain B). In this example the LIP functions as inhibitor.

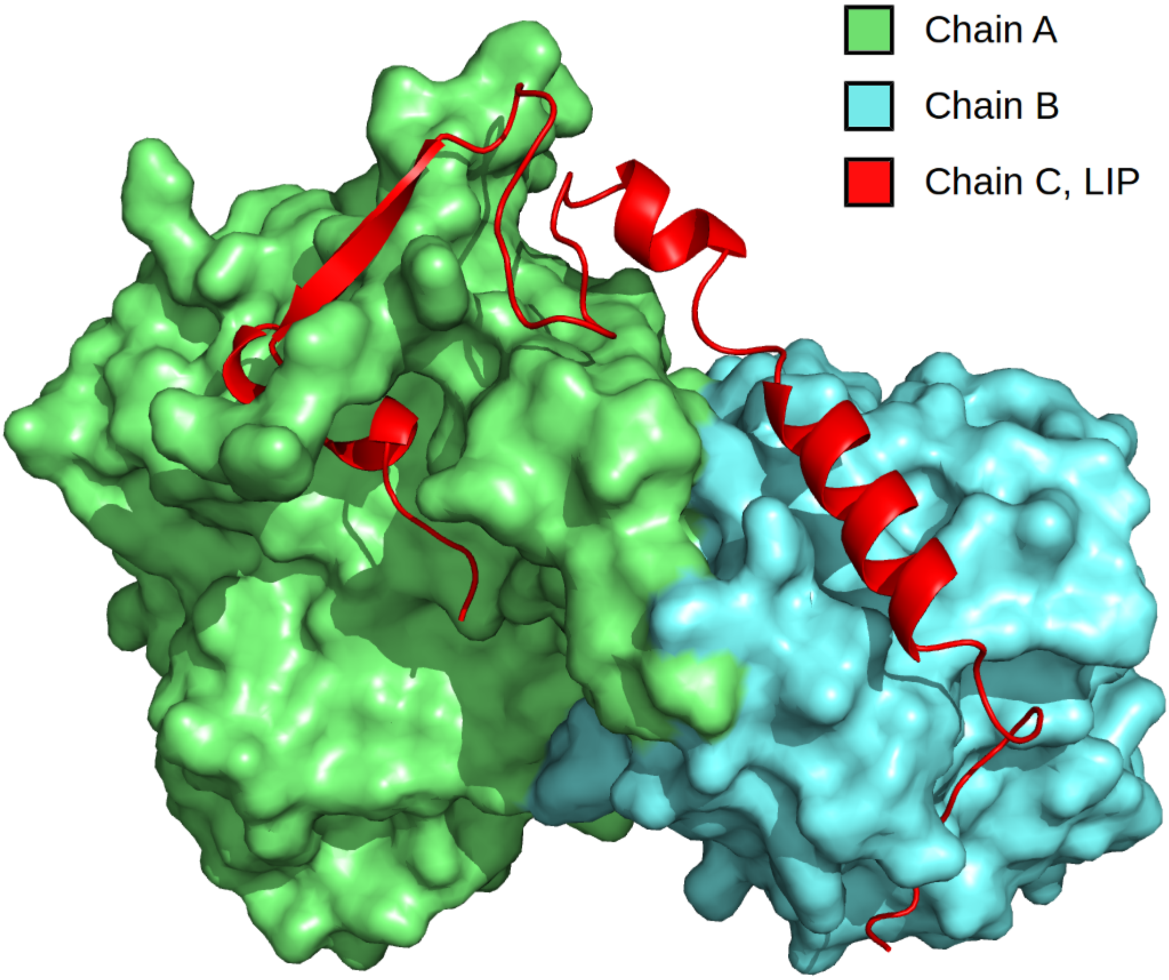

Figure 1: PDB id 1jsu, Chain C, LIP. Example of a long Linear Interacting Peptide

Identification of Linear Interacting Peptides is not an easy task even for a human expert. Mobi 2.0 was implemented to identify LIPS based on contacts measurements computed using the RING software. The problem of Mobi 2.0 is the high number of false positive predictions. FLIPPER is a new machine learning method that recognizes LIPs from 3D structure which replace Mobi 2.0 and perform significantly better.

Datasets

Datasets used to train ANCHOR, and PixelDB were used to train FLIPPER. ANCHOR is trained on a set of peptides that transit from disordered to ordered state upon binding with another protein and they are considered LIPs. Manual check on the dataset brought to conclude that all training examples are good candidates, exception made for PDB ID 1ymh. This particular example is strongly structured and does not fit our definition of LIP. Residues indexes that correspond to LIPs were selected by visually inspecting the structure. Negative examples to train the model come both from PDB-chains containing LIPs but where those LIPS are masked out and from PDB-chains that do not contain any LIPs. The result is a total of 70 non-redundant (sequence identity < 35%) PDB IDs forming the dataset LIP70.

Model

FLIPPER is a Random Forest Classifier with an ensemble of 20 Decision Trees, no fixed maximum depth and a minimum number of samples to build a new leaf that corresponds to 0.05% of examples in training set. Split quality is estimated with *Gini impurity*. A node is split if the split induces a decrease of the impurity greater than or equal to 0, where the decrease is weighted by the total number of examples.

Post Processing

Post processing refers to operations applied to predictions obtained from the classification model.

Blur

Blur is a filtering operation that smooths the prediction signal (probability to belong to positive - LIP - class). This operation creates a new signal using a sliding window of size w = 2∗c+1 centered on a target residue.

Gap Filling

Gap fill is the operation that aims at eliminating the discontinuities in the prediction signal considered in his binary form. A gap can be:

- Less or equal to *g* consecutive 1s surrounded left and right by 0 or beginning/end of the sequence. For example, considering *g=3* we can have: 111000.., ..001100.., or ..0011.
- Less or equal to *g* consecutive 0s surrounded left and right by 1. For example, considering *g=3* we can have: ..11011.., ..110011.., or ..1100011...

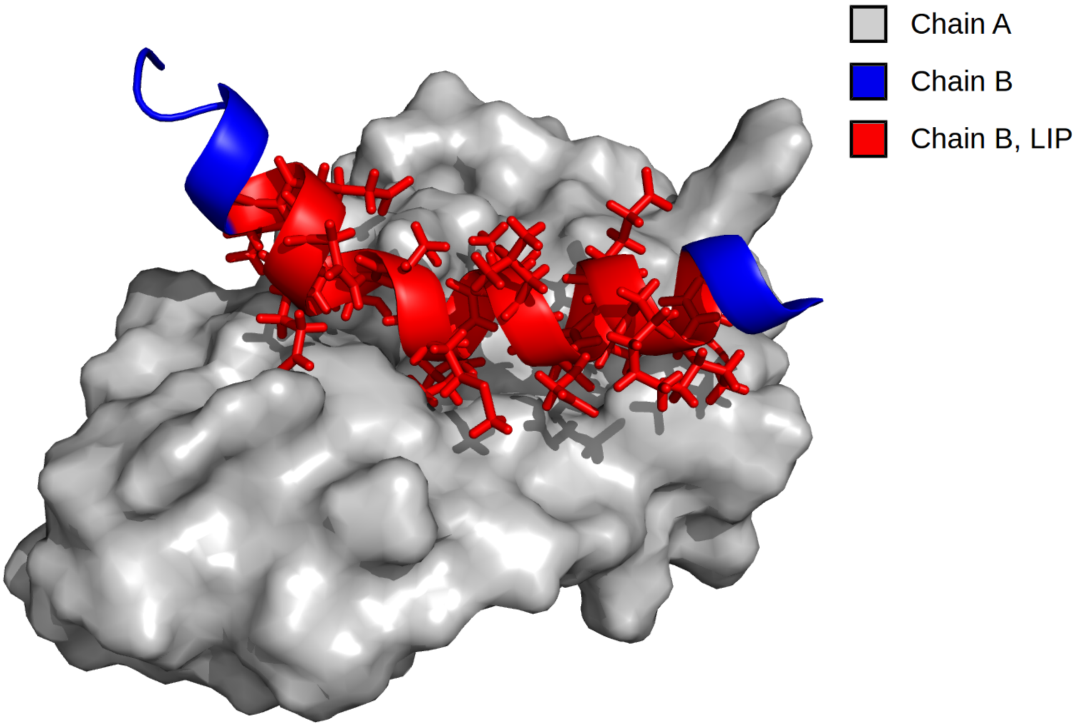

Figure 2: PDB 1sb0, Chain B, LIP Region. Explaining the algorithm used to fill prediction gaps during post processing

It is important to note that positives at the extremes of a sequence can be gaps while negatives cannot: this depends on how LIPs regions are defined. Considering the PDB ID 1sb0 chain B shown in Figure 2, we can see that the residues at the sides of the helix are not classified as LIP (blue). Replacing a negative classification in this case would be a mistake.

Features

FLIPPER gets in input the 3D structure of a protein to identify interacting flexible regions (LIPs). It follows the list of features.

- Inter Neighbors: Count of inter neighbors of a target residue. Inter neighbors are residues belonging to a different chain respect target residue in a range of 3.8 Å. Distance is measured as minimum distance between all atoms pairs.
- Intra Neighbors: Count of intra neighbors of a target residue. Intra neighbors are residues belonging to the same chain of target residue, in a range of 3.8 Å and separated by at least 7 residues in sequence. Distance is measured as minimum distance between all atoms pairs.
- Average Inter Neighbors: Inter chain neighbors counted for every residue in a window of 11 residues, normalized by window length.
- Average Intra Neighbors: Intra chain neighbors counted for every residue in a window of 11 residues, normalized by window length.
- Helix Content: Percentage of residues belonging to an helix in a 11 residues window.
- Beta-Strand Content: Percentage of residues belonging to a beta- strand in a 11 residues window.

- Coils Content: Percentage of residues not belonging to an helix nor a beta-strand in a 11 residues window.
- RSA: Relative Solvent Accessibility considering the chain in isolation, calculated as: RSA=ASA/M axASA. ASA is obtained using DSSP and M axASA from BioPython dictionary ”Wilke”.
- ∆ RSA: Difference between RSA calculated in isolation and in complex.
- Linearity: Distance between first and last residues in a sliding window of 41 residues, normalized by the window length. Distance is measured between α-carbons.
- Chain Length: Min between the number of residues in the chain and 100, divided by 100

Cross Validation

The prediction is made at the residue level, so a possibility would be to take random residues to train the model and test it on the rest. However, Considering that close residues in the chain will have similar features, because they are extracted using a sliding window, and the small size of the dataset, this approach is ineffective. To test FLIPPER, a 10-fold cross validation was adopted, dividing the examples at the PDB level. The data set used for this purpose is LIP70 (see Datasets). Cross validation has 7 steps in which the model is trained on 60 and tested on 10 pdb files. Results in Table 1 are the average of the 7 cross validation runs:

Precision-Negative:	0.9851
Precision-Positive:	0.9073
Recall-Specificity:	0.9880
Recall-Sensitivity	0.9141
Accuracy:	0.9762
Balanced-Accuracy:	0.9510
ROC-AUC:	0.9510
MCC:	0.8953

Table 1: FLIPPER Cross Validation Results on LIP70.

The scores indicate a good generalization on unknown targets. A detailed analysis of predictions highlighted that most of the errors are concentrated on flanking residues of LIPs: in other words, it is difficult to establish exactly start/end positions of a LIP.

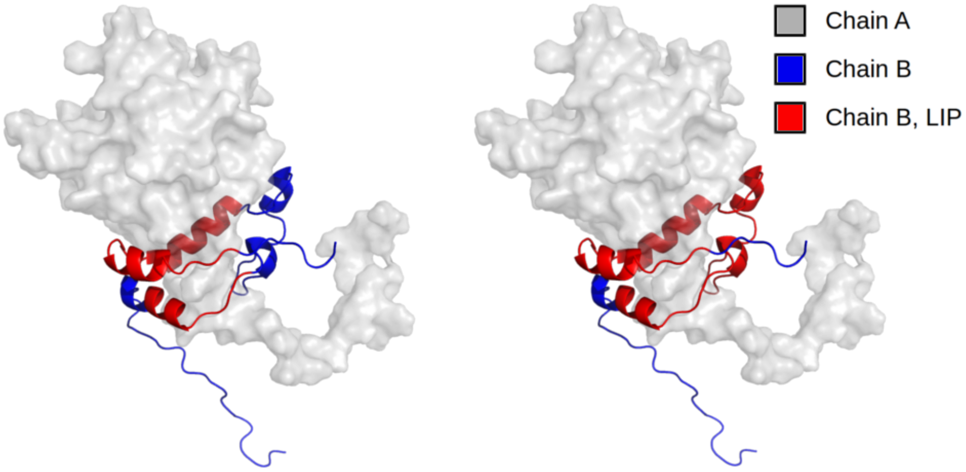

Figure 3: FLIPPER prediction on the left, ground truth on the right. Worst case in cross validation due to complexity and unique shape of the structure. PDB ID 1rf8.

The worst prediction is shown in Figure 3. This example, PDB ID 1rf8, comes from the ANCHOR data set and it has a very peculiar shape. FLIPPER is able to classify half of the residues correctly, and fails probably due to a high intra chain contacts rate in part of the LIP.

Validation On PixelDB

PixelDB is a database containing peptides bounded with one or more partners. Most PixelDB examples are short peptides bound with one or more “receptors”, and this means that they can be considered as LIPs. PixelDB contains 1.966 entries, divided in 486 different “receptor” clusters. Of the total 1.838 unique pdb identifiers, 11 that are used in the training (1dev, 1t08, 1ee5, 1ycq, 2fym, 1iwq, 1fv1, 2nl9, 1nx1, 1p4b, 2d1x) were leaved out from validation. FLIPPER was validated on this data set, with results shown in Table 2. The validation is at residue level, considering every residue in a peptide as positive and every residue in a “receptor” as negative.

Precision-Negative:	0.9993
Precision-Positive:	0.9118
Recall-Specificity:	0.9956
Recall-Sensitivity:	0.9849
Accuracy:	0.9951
Balanced-Accuracy:	0.9903
ROC-AUC:	0.9903
MCC:	0.9452

Table 2: FLIPPER Validation Results on PixelDB.

Run on PDB

FLIPPER has been tested against the entire PDB version of November 2019, containing a total of 156.300 files. To better understand the results, they are compared with Mobi 2.0. The latest publicly available results of Mobi 2.0 are relative to PDB version of October 2017. FLIPPER takes less than 18 hours on a 12 core CPU, while the same task takes more than one month for Mobi 2.0, running on a cluster of computers with much more cores.

Table 3 shows the results of running FLIPPER on the entire PDB. Results referring to UniProt are obtained with an ”at least one” assignment.

Total PDB LIPs:	79.249
Total PDB ids:	21.698
Total PDB Residues	1.393.619
Total UniProt LIPs:	11.022
Total UniProt ids:	8.509
Total UniProt Residues:	235.291
Short Regions (< 20):	6.951
Long Regions (>= 20):	4.372

Table 3: FLIPPER results on all files from PDB. Results are mapped to UniProt with an ”at-least-one” positive classification.

Stat	FLIPPER	FLIPPER80%	Mobi 2.0
Total UniProt LIPs:	8.264	10.609	2.133
Total UniProt ids:	5.916	5.59	13.596
Total UniProt Residues:	167.838	123.345	105.728
Short Regions (< 20):	5.187	8.621	21.603
Long Regions (>= 20):	3.077	1.988	530
Content:	2.23 %	1.63 %	1.40 %

Table 4: FLIPPER results on proteins containing LIPs from MobiDB 3.0 consensus. The first column refers to ”at-least-one” positive classification, while the second to 80 % consensus of FLIPPER predictions.

In Table 4 FLIPPER results are compared with results from Mobi 2.0 and the consensus generated at the protein level available in MobiDB 3.0. There are no statistics publicly available about direct output from PDB files, so they cannot be compared exactly. It is possible to compare instead results at the UniProt protein level, even if there are differences in the versions of PDB files used.

FLIPPER finds a total of 8.264 LIPs against the 22.133 of Mobi 2.0. While the number of LIPs is less, FLIPPER finds more residues than Mobi 2.0. A big difference is in the length (measured in residues) of LIPs. FLIPPER finds only 5.187 short regions (less than 20 amino acids), versus 21.603 found by Mobi 2.0. Furthermore, it finds 3.077 long regions (more than

20 amino acids) while Mobi 2.0 finds only 530 such regions. The consensus procedure in FLIPPER subdivides some of the long regions into shorter, but the gap between the results remains. The overall content of LIPs of the proteins under exam, is 2,23% for FLIPPER and 1,40% for Mobi 2.0, but with consensus at 80 % FLIPPER reaches 1,63%, that is quite similar to Mobi 2.0.

Conclusions

Considering the obtained results, FLIPPER is a good-performing tool that can bring innovation into the field of disordered and flexible proteins. The used features capture the desired definitions of Linearity and Interaction. The Cross Validation shows that the model generalizes well and does not easily overfit or underfit. A problem encountered is the limited size of the dataset that may not cover some very peculiar structures.

FLIPPER is reliable enough to perform the simple task of finding short peptides in- teracting with a partner (see Validation on PixelDB). This tells us that, with the right constraints, FLIPPER can be used to produce a data set of good quality peptides without the need of manual human intervention.

Compared to Mobi 2.0 FLIPPER really deserves to be called “Fast”. Faster execution time means that it is possible to test multiple versions of the program within reasonable time. For example it is possible to change the classification algorithm, retrain the model or change post processing parameters. That may be particularly useful when a new and better training set will be made available. FLIPPER finds 8.264 LIPs versus the 22.133 currently in MobiDB. There are for sure some totally ignored regions, cause there are approximately 7.500 UniProt proteins less. Since there are currently a lot of false positives in the database, this may be an indication of reduced errors. Considering the differences between short and long regions and the fact that FLIPPER finds more residues than Mobi 2.0, we can try to infer another conclusion: a lot of the regions found by Mobi2 are fragmentary, which is a consequence of the method that only takes into account residues contacts. Those fragmented regions are probably merged in FLIPPER results, giving longer LIPs. For the same reason we can explain why FLIPPER finds in total less regions than Mobi2.

Finally, FLIPPER is probably able to find regions not previously detected in MobiDB. The example in Figure 4 shows the PDB ID 2zi0 and the corresponding FLIPPER predictions. The chains come from RNA-silencing suppression protein of Tomato aspermy virus. It is a protein with a high degree of intrinsic disorder and it binds to double strand RNA. This makes a perfect example of LIP that Mobi 2.0 is unable to identify. The reason for it, is probably that Mobi 2.0 is biased against structures with high α-helical content and has problems computing protein-RNA contacts.

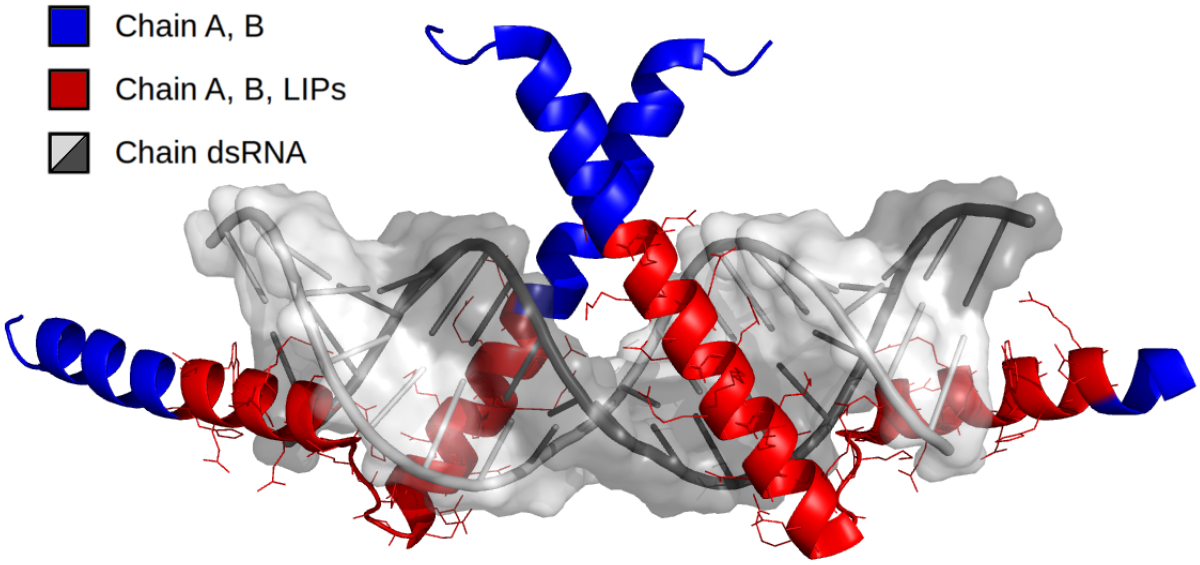

Figure 4: Chain A and B LIP in red. Example of a FLIPPER prediction which is missing in MobiDB 3.0. PDB ID 2zi0.

Availability

The project source code is available at:
https://gitlab.com/idpfun/flipper