Efficient Peptide MRM Transition Prediction via Convolutional Hashing

Ramon Adàlia[1,2][0009−0004−9458−1922]✉, Gemma Sanjuan[1][0000−0002−1946−4345], Tomàs Margalef[1][0000−0001−6384−7389], and Ismael Zamora[2][0000−0002−7700−0354]

[1] Universitat Autònoma de Barcelona, Cerdanyola del Vallès, Spain
Ramon.Adalia@autonoma.cat✉,{Gemma.Sanjuan,Tomas.Margalef}@uab.cat
[2] Lead Molecular Design, S.L., Sant Cugat del Vallès, Spain
{ramon.adalia,ismael.zamora}@leadmolecular.com

Abstract. We present a novel method for predicting multiple reaction monitoring (MRM) transitions for peptides in targeted proteomics. Our approach employs a hash-based representation inspired by convolutional neural networks, efficiently encoding peptide fragments as sparse count vectors that capture local sequence context. Using gradient-boosted decision trees, our method achieves mean Hits@5 scores of 3.4318 (hash-based) and 3.5405 (hybrid model with target frequency), significantly outperforming baselines. Transpiling trained models into Zig enables exceptional computational efficiency, with low memory usage (1180 kB) and a throughput of 388-451 peptides/second even on mobile devices, enabling lightweight, high-speed processing for scalable peptide MRM transition prediction in high-throughput proteomics workflows.

Keywords: MRM transitions · Peptide quantification · Edge computing

1 Introduction

Multiple Reaction Monitoring (MRM) is a mass spectrometry technique enabling precise peptide quantification in applications ranging from biomarker discovery to pharmaceutical development. Mass spectrometry identifies compounds by measuring the mass-to-charge ratio (m/z) of ions, with MRM specifically employing a three-step process: (1) filtering ions with the desired m/z, (2) fragmenting these filtered ions via collision with inert gas, and (3) filtering fragments by m/z for highly-specific detection. Determining appropriate m/z values for the final filtering step requires analyzing pure samples with the second filter deactivated—a time-intensive and resource-demanding process [9], underscoring the value of computational prediction methods.

Peptides, amino acid chains linked by peptide bonds (typically <50 residues), fragment into characteristic **b ions** (N-terminal) and **y ions** (C-terminal) during

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_3

mass spectrometry. Amino acids contain an amino group, a carboxyl group, hydrogen, and a variable side chain determining their properties, all connected to a central alpha carbon atom. The 20 standard amino acids use single-letter codes (A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y), with the fragment notation indicating the amino acid count (e.g., in CYIQNCPLG, b_3=CYI and y_2=LG), as shown in Figure 1.

Existing approaches have significant limitations: small-molecule models [1] are unsuitable due to structural differences; simple representations such as amino acid composition [2] miss positional information despite wide use [3,4]; and neural networks like ESM [8] capture sequence features effectively but impose excessive computational demands.

Our approach addresses these challenges by replacing CNN [6] convolutions with hashing to efficiently encode peptide fragments as sparse vectors, capturing local sequence context with minimal overhead. Trained with LightGBM [5] and transpiled to Zig, our method achieves both high accuracy and computational efficiency for real-world proteomics applications.

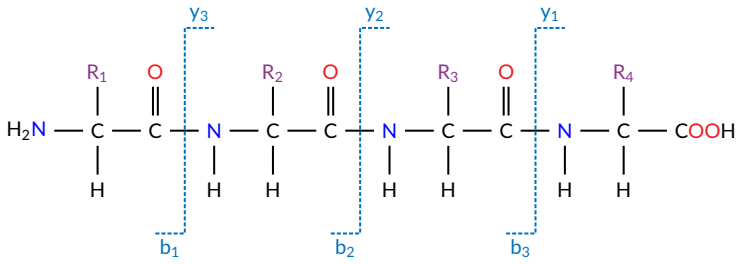

Fig. 1. Peptide fragmentation notation for a peptide with four amino acids.

2 Methodology

2.1 Data

The dataset used in this study was obtained from *Multiple Reaction Monitoring Assays for Large-Scale Quantitation* [7] and comprises 2,965 unique peptide sequences (ranging from 6-25 amino acids, median 11) with experimentally optimized MRM transitions. The majority of peptides (2,922) have five transitions, while the remainder have between 3-9, totaling 14,881 transitions represented as b/y ions with specific charge states (e.g., b9++). Charge distributions are

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_3

predominantly +1 (11,592), +2 (3,177), and +3 (111). Figure 2 illustrates key dataset characteristics, highlighting the prevalence of smaller ions.

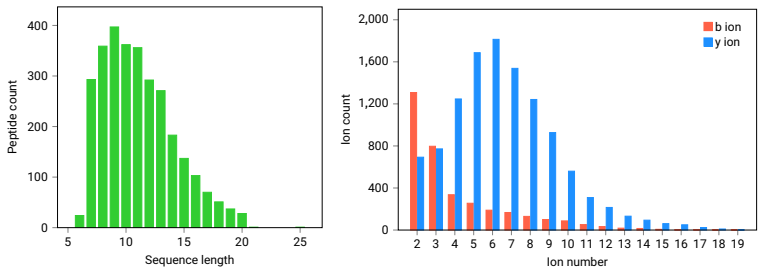

Fig. 2. Dataset composition showing sequence length distribution and b/y ion counts.

2.2 Peptide Fragment Representation

Algorithm 1 Peptide fragment sequence representation with neighborhood hashing

Require: Input sequence s of length $n \geq 3$, radius $R \geq 0$

Initialize empty array H for hashes
for $j \leftarrow 1$ to n **do**
  $v_j^0 \leftarrow [0] \parallel \mathit{description}(s_j)$ ▷ Initial encoding
  $h_j^0 \leftarrow \mathit{hash}(v_j^0)$; Append h_j^0 to H
end for
for $i \leftarrow 1$ to R **do**
  $h_1^i \leftarrow \mathit{hash}([i, h_1^{i-1}, h_2^{i-1}])$; Append to H ▷ First AA
  for $j \leftarrow 2$ to $n-1$ **do**
    $h_j^i \leftarrow \mathit{hash}([i, h_{j-1}^{i-1}, h_j^{i-1}, h_{j+1}^{i-1}])$; Append to H ▷ Middle AAs
  end for
  $h_n^i \leftarrow \mathit{hash}([i, h_{n-1}^{i-1}, h_n^{i-1}])$; Append to H ▷ Last AA
end for
return H

Our peptide fragment representation (Algorithm 1) draws inspiration from 1D CNNs. Each amino acid is mapped to a two-element integer vector encoding: an ordinal position (e.g., A=1, C=2, Q=3, etc.) and a binary inclusion indicator (1 if part of the fragment, 0 otherwise). For instance, in the b_2 fragment of peptide ACQA, the first amino acid 'A' encodes as [1,1] (included) whereas the final 'A' encodes as [1,0] (excluded).

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_3

Instead of standard convolutions, we leverage hashing to capture neighborhood context, aggregating local information through a hash function that processes neighboring amino acids. For a vector $[c_1, c_2, \ldots, c_n]$, we compute its hash as:

$$X_{i+1} = (a \cdot X_i + c_i) \mod m, \quad \text{for } i \in [1, n-1]$$

with parameters $m = 2^{32}$, $X_1 = 1013904223$, and $a = 1664525$.

Hashes are computed iteratively for radius R, transforming the resulting list H into a sparse count vector that comprehensively represents the fragment. Finally, charge states are incorporated as an additional integer feature.

2.3 Model Training

We employ Gradient Boosting Decision Trees via LightGBM, selected for their numerous advantages with high-dimensional sparse data, invariance to monotonic transformations, robustness to correlated features, and straightforward inference paths. Models are trained with default hyperparameters (100 trees, maximum 31 leaves) using LambdaMART with NDCG objective and a radius parameter of $R = 2$.

To enhance computational efficiency, we preprocess the sparse matrix by eliminating constant columns, reducing the feature count based on the selected R. For the simplest case where $R = 0$, only 41 features remain: one for the charge state and 40 for the amino acid presence/absence patterns. This dimensional reduction necessitates consistent feature selection between the training and inference phases.

2.4 Model Evaluation and Inference

We assess performance through 5-fold cross-validation on the dataset (593 peptides per fold), ranking candidate transitions for held-out peptides. Performance is quantified using the Hits@5 metric, which counts correct transitions appearing in the top five predictions.

For efficient inference, we transpile LightGBM's model output into optimized Zig functions that take hashmap input (column indices with corresponding counts) and produce prediction scores. This approach creates compact standalone binaries with Zig's advanced compiler optimizations, eliminating LightGBM dependencies while ensuring cross-platform compatibility and processing efficiency.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_3

3 Results and Discussion

3.1 Baseline Methods

For a random baseline model, we compute the exact Hits@5 distribution from empirical data rather than relying on simulations. For a peptide of length n, total number of possible transitions is $6(n-1)$ (derived from $n-1$ possible b/y-ions and 3 charge states). When randomly sampling 5 transitions, the number of correct predictions follows a hypergeometric distribution with the probability mass function:

$$P(X = k) = \frac{\binom{T}{k}\binom{6(n-1)-T}{5-k}}{\binom{6(n-1)}{5}},$$

where T represents experimentally identified transitions (typically 5). Averaging probabilities across all peptides yields a probability model with an expected Hits@5 value of 0.3396 for the random model.

A more effective baseline is the *target model*, which ranks transitions by their frequency in the dataset, always predicting [y6+, y5+, b2+, y7+, y4+] in the top 5. This straightforward approach achieves an average Hits@5 score of 2.2405.

3.2 Model Performance

We evaluated random, target, hash-based, and *hash+target* models using cross-validation. The hybrid model integrates target encoding with convolutional hashing, incorporating ion frequency within training folds to mitigate overfitting. This is implemented as an additional integer feature (e.g., for ion y_8^+, the feature represents how frequently y_8^+ appears among the top-5 fragment ions in the training set). Figure 3 illustrates performance distributions via Hits@5.

The results demonstrate marked improvement from the baseline models to hash-based models. The hash-based model achieves a mean Hits@5 of 3.4318 compared with 2.2405 for the target model (representing a 58% improvement, $p < 10^{-5}$, assessed with a one-sided paired permutation test). The *hash+target* model further enhances the performance to 3.5405 ($p = 0.00381$).

While all the models achieve comparable rates of at least one correct prediction, the hash-based approaches demonstrate substantially higher frequencies of multiple correct predictions, with the hybrid model producing 19.28× more perfect predictions than the target model does. This highlights how our approach excels particularly when maximizing correctly predicted transitions is critical for downstream applications.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_3

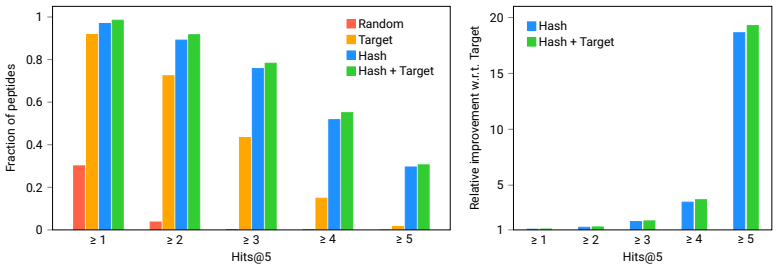

Fig. 3. Distribution of Hits@5 scores across evaluated models, showing hash-based models achieve higher frequencies of multiple correct predictions.

4 Ablation Study and Radius Effects

We conducted a detailed investigation into the impact of the radius parameter R on model performance, with $R = 0$ representing the complete removal of the convolutional effects. Figure 4 reveals substantial performance improvement when R increases from 0 to 1, with gains plateauing at higher values, suggesting diminishing returns from incorporating larger neighborhood contexts.

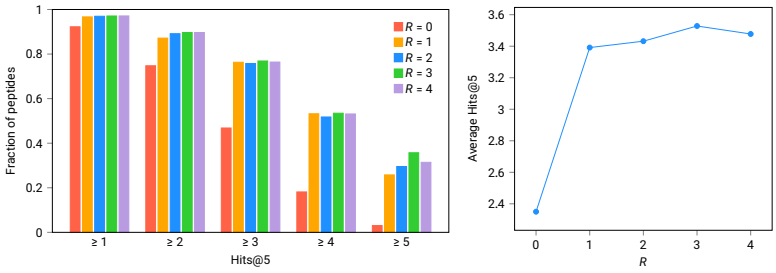

Fig. 4. Effect of radius R on Hits@5 scores, showing significant improvement from $R = 0$ to $R = 1$ and diminishing returns at higher values.

Table 1 presents computational metrics across different R values (Intel i7-8750H for training, Android Snapdragon 695 for inference). As R increases, the feature count grows exponentially (39 to 106,735), with the training time increasing from 18s to over 14 minutes. Notably, memory usage at inference remains constant (1180 kB) across all configurations, whereas inference throughput decreases marginally (14% from $R = 0$ to $R = 4$), demonstrating excellent scalability.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_3

Table 1. Training and inference performance for different radius values R.

			Inference		
R	**Features**	**Training Time**	**Mem. (kB)**	**Time (s)**	**Peptides/sec**
0	39	17.68 s	1180	6.564 ± 0.074	451.71
1	12,750	1:23.89 min	1180	6.789 ± 0.012	436.74
2	50,147	6:45.87 min	1180	6.989 ± 0.016	424.24
3	82,040	11:05.74 min	1180	7.205 ± 0.034	411.52
4	106,735	14:32.50 min	1180	7.632 ± 0.078	388.50

5 Conclusion

We have introduced a novel approach for predicting MRM transitions using a hash-based representation inspired by convolutional neural networks. By encoding peptide fragments as sparse count vectors with gradient boosting trees, our method achieves significant improvements over established baselines. The hybrid model integrating target frequencies with convolutional hashing achieves a mean Hits@5 of 3.5405, a 58% improvement over the baseline approach.

Our ablation study reveals the importance of the local sequence context, with peak performance gains occurring between $R = 0$ and $R = 1$. This demonstrates the value of neighborhood information while suggesting diminishing returns from larger contexts. The approach shows exceptional efficiency through Zig transpilation, maintaining consistent memory usage (1180 kB) across configurations while processing 388-451 peptides/second on mobile devices.

This combination of accuracy and efficiency makes our method practical for resource-constrained environments. The compact representation enables training on large datasets with minimal overhead. Future work could extend this methodology to other biomolecule types or incorporate domain knowledge to enhance prediction accuracy, streamlining integration into proteomics workflows.

Acknowledgments. This study was supported by Pla de Doctorats Industrials del Departament de Recerca i Universitats de la Generalitat de Catalunya (grant 2023-DI-00006).

Disclosure of Interests. The authors have no competing interests relevant to this article.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_3

References

1. Adàlia, R., Patel, S., Paiva, A., Kaufman, T., Zamora, I., Cai, X., Sanjuan, G., Shou, W.Z.: Development of a predictive multiple reaction monitoring (MRM) model for high-throughput ADME analyses using learning-to-rank (LTR) techniques. Journal of the American Society for Mass Spectrometry **35**(1), 131–139 (nov 2023). https://doi.org/10.1021/jasms.3c00363, https://doi.org/10.1021/jasms.3c00363
2. Chou, K.C.: A novel approach to predicting protein structural classes in a (20-1)-d amino acid composition space. Proteins **21**(4), 319–344 (Apr 1995). https://doi.org/10.1002/prot.340210406
3. Du, Q.S., Jiang, Z.Q., He, W.Z., Li, D.P., Chou, K.C.: Amino acid principal component analysis (aapca) and its applications in protein structural class prediction. Journal of Biomolecular Structure and Dynamics **23**(6), 635–640 (2006). https://doi.org/10.1080/07391102.2006.10507088, https://doi.org/10.1080/07391102.2006.10507088, pMID: 16615809
4. Jahandideh, S., Abdolmaleki, P., Jahandideh, M., Asadabadi, E.B.: Novel two-stage hybrid neural discriminant model for predicting proteins structural classes. Biophysical Chemistry **128**(1), 87–93 (2007). https://doi.org/https://doi.org/10.1016/j.bpc.2007.03.006, https://www.sciencedirect.com/science/article/pii/S0301462207000749
5. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y.: Lightgbm: A highly efficient gradient boosting decision tree. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017), https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf
6. Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., Inman, D.J.: 1d convolutional neural networks and applications: A survey. Mechanical Systems and Signal Processing **151**, 107398 (2021). https://doi.org/https://doi.org/10.1016/j.ymssp.2020.107398, https://www.sciencedirect.com/science/article/pii/S0888327020307846
7. Michaud, S.A., Pětrošová, H., Sinclair, N.J., Kinnear, A.L., Jackson, A.M., McGuire, J.C., Hardie, D.B., Bhowmick, P., Ganguly, M., Flenniken, A.M., Nutter, L.M.J., McKerlie, C., Smith, D., Mohammed, Y., Schibli, D., Sickmann, A., Borchers, C.H.: Multiple reaction monitoring assays for large-scale quantitation of proteins from 20 mouse organs and tissues. Communications Biology **7**(1), 6 (2024). https://doi.org/10.1038/s42003-023-05687-0, https://doi.org/10.1038/s42003-023-05687-0
8. Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C.L., Ma, J., et al.: Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proceedings of the National Academy of Sciences **118**(15), e2016239118 (2021). https://doi.org/10.1073/pnas.2016239118, https://www.pnas.org/doi/full/10.1073/pnas.2016239118, bioRxiv 10.1101/622803
9. Shou, W.Z., Zhang, J.: Recent development in high-throughput bioanalytical support for in vitro admet profiling. Expert Opinion on Drug Metabolism & Toxicology **6**(3), 321–336 (2010). https://doi.org/10.1517/17425250903547829, https://doi.org/10.1517/17425250903547829, pMID: 20163321

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97635-3_3