# Knowledge Graph–Augmented DNA Representation Learning

**Fengyu Cai** [1 *] **Erik Kubaczka** [1 2 *] **Shaobo Cui** [3] **Heinz Koeppl** [1 2]

## Abstract

Understanding the language of the genome remains a key challenge in biology, with pre-trained models such as DNABERT-2 achieving substantial advancement. These models leverage massive nucleotide sequences through a self-supervised learning paradigm, yet they often overlook the rich, structured knowledge already curated by human experts. Inspired by the knowledge-enhanced foundation models in other biological molecules (e.g., proteins and drugs), we introduce Knowledge Graph-Augmented DNABERT (KGA-DNABERT), augmenting the objective of masked language modeling with knowledge graph (KG) modeling. Specifically, we construct KGs by extracting factual triplets from GenomicKB, a comprehensive human genome database. In addition to DNABERT-2's MLM, we incorporate six popular KG embedding methods to model the curated KG beyond sequence-level representations. We did *not* observe substantial benefits from incorporating KGs into DNA representation learning with the KGs tested here and attribute this to the insufficient coverage of the constructed KGs, as they represent only an excerpt of GenomicKB. This motivates us to explore further a better integration of KG for DNA representation learning.

## 1. Introduction

Foundation models (Devlin et al., 2019; Radford et al., 2019), with millions to billions of parameters, exhibit a strong ability to capture complex relationships and dependencies among biological molecules. Notably, DNA-specific foundation models (Ji et al., 2021; Zhou et al., 2024; Dalla-Torre et al., 2025) pre-trained on genomic sequences enable effective nucleotide representations for downstream tasks such as transcription factor and promoter detection.

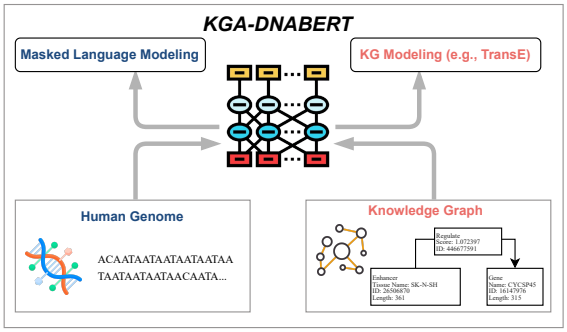


Figure 1: Illustration of KGA-DNABERT, which includes both classical mask language modeling and KG modeling as pertaining objectives.

While self-supervised paradigms in DNA foundation models effectively leverage large-scale nucleotide sequence datasets like the human genome (Mudge et al., 2025), they often overlook the structured human knowledge accumulated over decades through expert curation and research. In bioinformatics, one of the most prevalent forms of structured knowledge is the knowledge graph (KG). Integrating KGs has demonstrated notable improvements in molecular representation and downstream biological tasks. OntoProtein (Zhang et al., 2022) leverages Gene Ontology (GO) to enhance protein representation learning, improving the predictive performance on various protein properties and protein-protein interactions. Additionally, Hoang et al. (2024) construct a multimodal KG from diverse sources to refine drug and protein representations, thereby supporting more accurate drug–protein interaction prediction.

Inspired by their success, this work investigates the potential of bridging this gap in DNA representation learning by incorporating structured external knowledge. Focusing on the human genome, we extract structured knowledge from the Genomic Knowledgebase (GenomicKB, Feng et al. 2023), a large KG for the human genome. Specifically, we extract up to $95,416$ sequence-to-sequence relations to extend the pre-training. Building on DNABERT-2 (Zhou et al., 2024)

[*]Equal contribution [1]Department of Electrical Engineering and Information Technology, Technical University of Darmstadt, Darmstadt, Germany [2]Centre for Synthetic Biology, Technical University of Darmstadt, Darmstadt, Germany [3]Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland. Correspondence to: Fengyu Cai <fengyu.cai@tu-darmstadt.de>, Heinz Koeppl <heinz.koeppl@tu-darmstadt.de>.

*Proceedings of the ICML 2025 Workshop on Multi-modal Foundation Models and Large Language Models for Life Sciences*, Vancouver, Canada. 2025. Copyright 2025 by the author(s).

as the base model, we systematically investigate the integration of KG embedding methods—spanning translational, bi-linear, and geometric approaches—into DNA representation learning, jointly optimized with masked language modeling objectives (see Figure 1). Following DNABERT's setup, we further fine-tune the KG-augmented models on downstream tasks, including transcription factor binding prediction and promoter detection, selected from the Genome Understanding Evaluation Benchmark (Zhou et al., 2024). Although KG-augmented pre-training has not yielded substantial gains so far, we have discovered multiple potential directions for improving our approach.

## 2. Methods

### 2.1. KG Construction

A KG is a set of factual triples $(h, r, t)$, where $h, t \in \mathcal{E}$ are head and tail entities, and $r \in \mathcal{R}$ is a relation. The GenomicKB (Feng et al., 2023) combines data from diverse data sources such as GENCODE (Harrow et al., 2012), EnhancerAtlas (Gao & Qian, 2020), and RNACentral (RNAcentral Consortium, 2021), and by this provides rich annotation of the human genome as well as of the relationships between annotated elements. Besides sequence annotations like *coding*, *non-coding*, *promoter*, *protein* or *enhancer*, the GenomicKB integrates information on properties like tissue and cell lines or epigenomic features. We differentiate these by separating the entities $\mathcal{E}$ into those with a sequence representation ($\mathcal{E}_s$) and those without ($\mathcal{E}_p$). This induces a subdivision of the set of relations, where we focus on $\mathcal{R}_{s\to s}$, the set of sequence-to-sequence relations (i.e. $h, t \in \mathcal{E}_s$).

To evaluate the effect of integrating the knowledge graph into the pre-training of a DNA representation model, we derive two distinct datasets from $\mathcal{R}_{s\to s}$. These datasets are $\mathcal{D}_1$, which is the *Enhancer-Graph*, and $\mathcal{D}_2$, the *Diversified-Graph*. $\mathcal{D}_1$ includes all sequence-to-sequence relations with the head being of type *enhancer*. Furthermore, all sequences included have at least 50 and at most 500 nucleotides, which matches the usual sequence lengths of DNA representation learning and respects the length constraint of our base model. We chose ***enhancer*** sequences as they increase the expression of the associated gene upon binding of a gene-activatory protein, a so-called ***transcription factor*** (Alberts et al.). As such, they are related to all downstream tasks either directly or via their interaction with *promoters*. The *Enhancer-Graph* ($\mathcal{D}_1$) consists of $53,219$ unique sequences in $87,583$ relation triples with the unique relation ***regulate***.

To also provide a dataset with more diverse sequence and relation types, we create the *Diversified-Graph* ($\mathcal{D}_2$). After selecting 100 **seed entities** of type *enhancer*, for the reasons outlined above, the graph is created by traversing the GenomicKB (independent of the relation's direction) along all sequence-to-sequence relations starting at the seed entities. As before, the length constraint applies. $\mathcal{D}_2$, the *Diversified-Graph*, features $64,509$ unique sequences in $95,416$ relation triples with four relations: ***regulate***, ***transcribe_into***, ***correlated_with***, and ***eQTL_of***.

### 2.2. KG Modeling

Various KG embedding models aim to map entities and relations into a continuous vector space to preserve the structure of the KG. KG embedding (KGE) models aim to learn mapping functions that assign each entity $h, t \in \mathcal{E}$ and relation $r \in \mathcal{R}$ a vector representation ($\mathbf{h}, \mathbf{t}$ and $\mathbf{r}$), such that the plausibility of a triple $(h, r, t)$ can be measured by a scoring function $f(h, r, t)$. These models vary in how they define $f(\cdot)$ to capture structural patterns such as symmetry, inversion, and composition. Below, we summarize representative KGE models categorized by their scoring mechanisms.

**Translation-based models** represent relations as translations in a continuous vector space. Let $\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$ denote the embeddings of the head and tail entities, and $\mathbf{r} \in \mathbb{R}^d$ the embedding of the relation. For models employing relation-specific projection matrices, each relation $r$ has its own projection $M_r \in \mathbb{R}^{d\times d}$, transforming entity vectors $\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$ into the relation space $\mathbb{R}^d$ Representative translation-based models include:

- **TransE** (Bordes et al., 2013) models relations as translations between head and tail entities: $f(h, r, t) = -\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|$.
- **TransH** (Wang et al., 2014) projects entities onto a relation-specific hyperplane before translation: $f(h, r, t) = -\|\mathbf{h}_\perp + \mathbf{r} - \mathbf{t}_\perp\|$, where $\mathbf{e}_\perp = \mathbf{e} - \mathbf{w}_r^\top \mathbf{e} \cdot \mathbf{w}_r$ is the projection of entity $\mathbf{e}$ onto the hyperplane with normal vector $\mathbf{w}_r \in \mathbb{R}^d$.
- **TransR** (Lin et al., 2015) projects entities into relation-specific spaces using projection matrix $M_r$: $f(h, r, t) = -\|M_r\mathbf{h} + \mathbf{r} - M_r\mathbf{t}\|$.
- **TransD** (Ji et al., 2015) uses dynamic projection matrices dependent on entities and relations: $f(h, r, t) = -\|M_{r,h}\mathbf{h} + \mathbf{r} - M_{r,t}\mathbf{t}\|$.

**Bi-linear models** use multiplicative interactions for relation modeling. One of the representatives is **DistMult** (Yang et al., 2015), which uses element-wise multiplication of embeddings: $f(h, r, t) = \langle \mathbf{h}, \mathbf{r}, \mathbf{t} \rangle$, where $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^d$ are the embeddings of the head, relation, and tail.

**Geometric models** encode relations as spatial transformations; a prominent example is **RotatE** (Sun et al., 2019), which represents each entity and relation as a complex vector $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{C}^d$ (equivalently, $\mathbb{R}^{2d}$ when splitting into real

and imaginary parts) and treats each relation as an element-wise rotation (unit-modulus complex numbers), yielding the scoring function $f(h, r, t) = -\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|$, where $\circ$ denotes the Hadamard (element-wise) product.

**KG Modeling Loss** Given a batch of $N$ positive and negative triplet pairs $\left\{(h_i, r_i, t_i),\ (\tilde{h}_i, r_i, \tilde{t}_i)\right\}_{i=1}^{N}$, and a margin $\gamma > 0$, the margin-ranking loss $\mathcal{L}_K$ is defined as:

$$\mathcal{L}_K = \frac{1}{N}\sum_{i=1}^{N} \max\Big(0,\ \gamma\ -\ \big[f(h_i, r_i, t_i)\ -\ f(\tilde{h}_i, r_i, \tilde{t}_i)\big]\Big).$$

**KG-augmented Pre-training Loss** To include the KG modeling into the training procedure, the combined loss $\mathcal{L}$ is computed as

$$\mathcal{L} = \mathcal{L}_M + \lambda \mathcal{L}_K,$$

where $\lambda$ denotes the weight coefficient.

## 3. Experiments

### 3.1. Evaluation Benchmarks

To assess how well knowledge-augmented DNA language models generalise to fundamental regulatory tasks, we adopt the GENOME UNDERSTANDING EVALUATION (**GUE**, Zhou et al. 2024) suite introduced by DNABERT-2. The GUE benchmark includes three human genome regulatory tasks. **Promoter detection** identifies proximal promoter regions using 300 bp sequences spanning $-249$ to $+50$ bp around transcription start sites (TSS), with datasets split by TATA presence: `prom_300_tata`, `prom_300_notata`, and a combined set `prom_300_all`. **Core promoter detection** focuses on a narrower region closer to the TSS, using 70 bp sequences centered from $-34$ to $+35$ bp, making it more challenging. It includes `prom_core_tata`, `prom_core_notata`, and `prom_core_all`. **TF binding site prediction** aims to identify transcription factor binding sites using 101 bp sequences centered at ChIP-seq peaks from the ENCODE database (Consortium et al., 2012). From 690 candidate datasets, five datasets (`TF_0` to `TF_4`) were selected based on task difficulty to ensure a balanced benchmark. With sequence lengths of 70 to 300 nucleotides, these tasks span both short and medium-length contexts, providing diverse and discriminative challenges for DNA language models. The details are given in Table 1.

| Task | # Datasets | # Classes | Seq. length (bp) |
|---|---|---|---|
| Promoter detection | 3 | 2 | 300 |
| Core promoter detection | 3 | 2 | 70 |
| TF binding site prediction | 5 | 2 | 100 |

Table 1: Human genome language model evaluation benchmarks applied in this work.

### 3.2. Experimental Setups

The training pipeline consists of three stages: Pre-training with MLM, KG-augmented pre-training, and task-specific fine-tuning.

**Pre-training** We follow the pre-training protocol of DNABERT-2 (Zhou et al., 2024), using the same training corpus comprising 32.49 billion nucleotide bases over 6 epochs. We apply the pre-trained tokenizer[1], which is designed based on Byte Pair Encoding (Sennrich et al., 2015) instead of k-mer tokenization (Ji et al., 2021). We perform pre-training for around 1.5 million steps using four A100 40GB GPUs, which takes approximately six days. For more detailed configurations, please refer to Appendix A.1.

**KG-Augmented Pre-training** Based on the model checkpoint in the pre-training phase, we use the combined objective of MLM and KG modeling to continually pre-train the model for 10 epochs. For the scoring function $f(h, r, t)$ in Section 2.2, we use the output vector of DNABERT-2 corresponding to the `[CLS]` token as the representation of the head and tail entities, $h$ and $t$, respectively. Their dimension is 768. For *Enhancer-Graph* $\mathcal{D}_1$ and *Diversified-Graph* $\mathcal{D}_2$, there are in total 13,679 and 14,910 training steps, respectively. The scheduler starts with a 2,000-step warm-up phase, after which the learning rate decays linearly. We select the weighing coefficient $\lambda$ as 1.0. Other setups are identical to pre-training. For brevity, we denote the KG-augmented pre-trained models on *Enhancer-Graph* and *Diversified-Graph* as $\mathcal{M}_1$ and $\mathcal{M}_2$, respectively.

**Fine-tuning** We fine-tune different pre-trained models on task-specific data, adjusting the input length and training schedule accordingly. Specifically, training uses FP16 mixed precision with a global batch size of 32. We evaluate checkpoints every 200 - 400 steps and select the best-performing checkpoint on the dev set for final testing. More details are in Appendix A.2.

### 3.3. Experimental Results

**Our Constructed KGs Cannot Help DNA Modeling** Table 2 reports $\mathcal{M}_2$'s F1 scores of the downstream classification tasks selected from the GUE benchmark. Except `tf_0` and `prom_300_notata`, $\mathcal{M}_2$ fails to outperform the original DNABERT-2. For the task of transcription factor prediction, the performances of $\mathcal{M}_2$ are close to the vanilla model. However, in promoter detection, the perfor-

[1] https://huggingface.co/zhihan1996/DNABERT-2-117M

| Model | TF binding site prediction | | | | | Core promoter detection | | | Promoter detection | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | TF_0 | TF_1 | TF_2 | TF_3 | TF_4 | all | notata | tata | all | notata | tata |
| Vanilla | 78.5 | **82.6** | **77.2** | **72.1** | **83.1** | **78.0** | **80.1** | **81.7** | **89.5** | 92.6 | **78.4** |
| DistMult | **79.9** | 77.1 | 73.5 | 64.7 | 80.5 | 74.5 | 78.2 | 73.8 | 87.8 | **92.7** | 69.1 |
| RotatE | 78.9 | 77.6 | 73.8 | 33.3 | 80.4 | 74.8 | 76.9 | 71.8 | 87.2 | **92.7** | 55.8 |
| TransD | 76.9 | 80.1 | 74.6 | 63.0 | 77.3 | 75.4 | 78.0 | 73.5 | 88.3 | 92.3 | 71.4 |
| TransE | 77.6 | 80.4 | 72.7 | 62.7 | 80.2 | 76.6 | 79.2 | 73.7 | 87.7 | **92.7** | 61.8 |
| TransH | 75.0 | 80.6 | 72.5 | 57.8 | 79.2 | 75.0 | 77.6 | 72.4 | 88.7 | **92.7** | 73.2 |
| TransR | 79.5 | 79.9 | 72.4 | 64.6 | 78.0 | 76.1 | 77.3 | 75.4 | 88.6 | 91.0 | 73.5 |

Table 2: Performance (F1 score, %) of KGA-DNABERT $\mathcal{M}_2$ with different KGE methods on the GUE benchmark for the human genome. During KG-augmented pre-training the models $\mathcal{M}_2$ are additionally trained on $\mathcal{D}_2$ (*Diversified-Graph*).

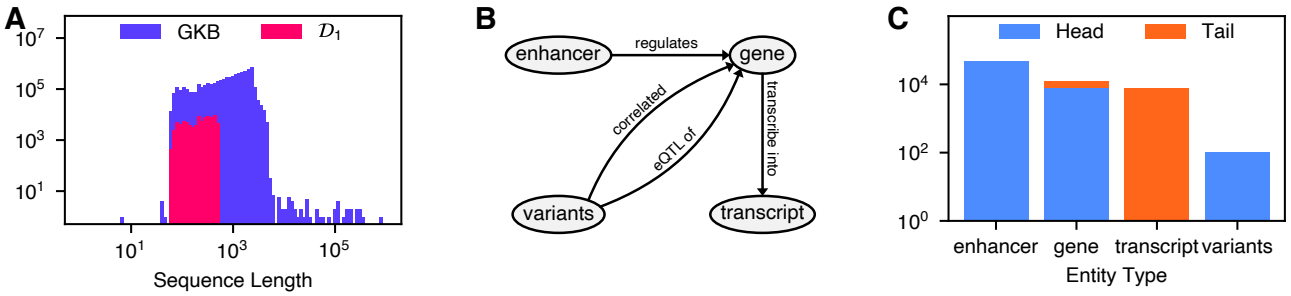


Figure 2: **Knowledge graph analysis. A:** Distribution of the sequence length of entities of type *enhancer* in GenomicKB (GKB, blue) and *Enhancer-Graph* ($\mathcal{D}_1$, red). The considered sequences span only a subrange of the *enhancer* sequences included in GenomicKB. **B:** High-level representation of the entity types and relations included in *Diversified-Graph* ($\mathcal{D}_2$). **C:** Overview on entity type representation as head (blue) or tail (orange) in *Diversified-Graph* ($\mathcal{D}_2$). While the relation heads are quite diverse with respect to the sequence entity, the types as well as the number of distinct tail entities are limited ($56,716$ vs. $12,023$ unique sequences).

mance gap between the vanilla model and $\mathcal{M}_2$ is significantly larger on the TATA split (`tata`) compared to the non-TATA split (`notata`), indicating the shift of model performance caused by KG-augmented pre-training. This pattern is also observed on $\mathcal{M}_1$, as shown in Table 3 in the Appendix.

## 4. Potential Improvements

Although KG modeling did not yield substantial improvement during DNA pre-training, we identify several possible reasons from both data and training perspectives.

### 4.1. Data-Related Factors

Our analysis suggests that the current selection of maximum sequence length of the KG may limit comprehensive genome-wide representation in GenomicKB (Feng et al., 2023), thereby impeding the effective integration of genomic knowledge into DNA sequence representations.

**Sequence Length Constraints Limit KG Expressiveness** Reconsidering the KG creation, the decision to extract only a subset of the GenomicKB with respect to sequence length and diversity of entity types is of particular importance. While the sequence length constraint aligns with the downstream tasks, many of the sequences included in the GenomicKB exceed this constraint and are therefore not considered (see Figure 2**A** and Figure 4 in the Appendix). Ultimately, the architecture of the deep learning model imposes a constraint on the maximum sequence length, which, however, can vary in dependence on the input encoding used. A strategy that combines the representations of sub-sequences or an alternative method is needed to account for long sequences that exceed this constraint.

**KG Construction is Essential for Complexity and Diversity** Although $\mathcal{D}_2$ is derived from connected subgraphs of GenomicKB, the final graph remains highly fragmented, mainly due to sequence length constraints, as discussed below. Greater connectivity may reduce representational

degrees of freedom per entity, encouraging convergence toward more representative embeddings.

The same applies to diversity in entity and relation types. Our focus on *enhancer* entities proofed beneficial in the cases of `tf_0`, `prom_core_tata`, and `prom_core_all`, while the other tasks likely reflect the low diversity of the relations and entity types present (see Figures 2**B** and **C**). Other downstream tasks may benefit from different entity or relation types or a more diverse representation. This is likely to have a positive effect on the number of distinct relation types included, which is again beneficial for DNA representation learning. Also, we only consider the relation type (e.g. regulate) for KG construction, instead of the specific property of the relation in GenomicKB (e.g., score).

As a preliminary analysis, we increase the maximum sequence length during $\mathcal{D}_2$ construction to $1,000$, $1,500$, and $2,000$, as shown in Figure 3. To assess graph connectivity, we compute the size of the largest connected component (LLC) and its ratio to the total number of nodes as a normalized metric. We observe that extending the length constraints to $2,000$ will significantly boost the ratio. In terms of diversity, increasing the cutoff to $1,000$ already captures most entity and relation types, with only marginal gains thereafter. These observations provide practical guidance for constructing a more comprehensive and diverse KG.

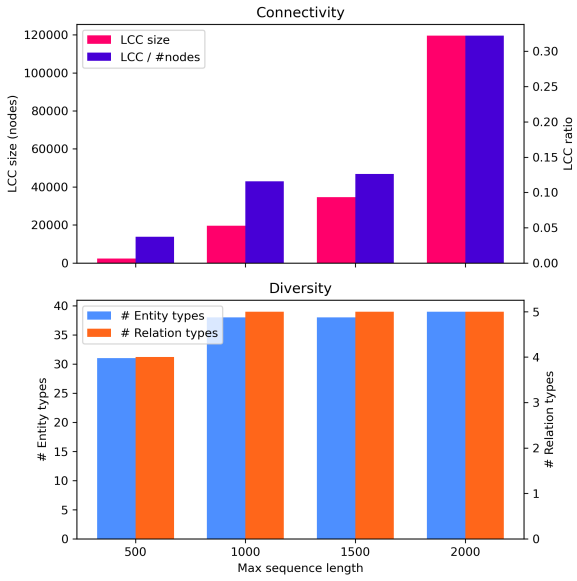


Figure 3: Effects of maximum sequence length ($500$, $1,000$, $1,500$, $2,000$) on connectivity and diversity. **Connectivity (Top)**: both largest component size and coverage increase sharply with longer sequences. **Diversity (Bottom)**: most entity and relation types are captured by $1,000$, with minimal gains beyond.

### 4.2. Training-Related Factors

We also anticipate that optimizing the training procedure could enable more effective fusion of the KG into DNA representations, although this lies beyond the scope of the current work due to computational constraints. One potential improvement is to jointly train KG modeling together with MLM, rather than modeling the KG only after MLM pre-training, as done in our current setup. While we continue MLM during KG-based continued pre-training, the integration of KG may disrupt the original representations learned via MLM, an issue commonly attributed to catastrophic forgetting (Wang et al., 2024). Another promising direction is to pre-train the KG component independently before joint training, as adopted in OntoProtein (Zhang et al., 2022). This allows the KG representations to be more robust before integration into the sequential model. However, the choice of modality for representing the KG remains an open challenge; for instance, OntoProtein encodes the KG using text-based sequential inputs.

## 5. Conclusion

In this work, we take a first step toward incorporating KG modeling into DNA representation learning. Specifically, we construct human genomic KGs by extracting structured knowledge from GenomicKB, a large-scale database of the human genome and its annotations. We integrate various KG modeling techniques into the pre-training process alongside classical masked language modeling and evaluate their impact on downstream genome understanding tasks such as promoter detection. While we have not yet observed substantial improvements, our analysis recognized several viable paths for future research.

## Acknowledgements

Fengyu Cai is supported by the Distr@l4a funding line of the State of Hesse (project number 493 24_0015_4A). Erik Kubaczka was supported by ERC-PoC grant PLATE (101082333). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding agencies.

## References

Alberts, Alberts, B., Bray, D., Hopkin, K., Johnson, A., Lewis, J., Raff, M., Roberts, K., and Walter, P. *Essential Cell Biology*. 3 edition.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pp. 2787–2795, 2013.

Consortium, E. P. et al. An integrated encyclopedia of dna elements in the human genome. *Nature*, 489(7414):57, 2012.

Dalla-Torre, H., Gonzalez, L., Mendoza-Revilla, J., Lopez Carranza, N., Grzywaczewski, A. H., Oteri, F., Dallago, C., Trop, E., de Almeida, B. P., Sirelkhatim, H., et al. Nucleotide transformer: building and evaluating robust foundation models for human genomics. *Nature Methods*, 22(2):287–297, 2025.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Feng, F., Tang, F., Gao, Y., Zhu, D., Li, T., Yang, S., Yao, Y., Huang, Y., and Liu, J. Genomickb: a knowledge graph for the human genome. *Nucleic Acids Research*, 51(D1): D950–D956, 2023.

Gao, T. and Qian, J. EnhancerAtlas 2.0: An updated resource with enhancer annotation in 586 tissue/cell types across nine species. *Nucleic Acids Research*, 48 (D1):D58–D64, January 2020. ISSN 0305-1048. doi: 10.1093/nar/gkz980. URL https://doi.org/10.1093/nar/gkz980.

Harrow, J., Frankish, A., Gonzalez, J. M., Tapanari, E., Diekhans, M., Kokocinski, F., Aken, B. L., Barrell, D., Zadissa, A., Searle, S., Barnes, I., Bignell, A., Boychenko, V., Hunt, T., Kay, M., Mukherjee, G., Rajan, J., Despacio-Reyes, G., Saunders, G., Steward, C., Harte, R., Lin, M., Howald, C., Tanzer, A., Derrien, T., Chrast, J., Walters, N., Balasubramanian, S., Pei, B., Tress, M., Rodriguez, J. M., Ezkurdia, I., van Baren, J., Brent, M., Haussler, D., Kellis, M., Valencia, A., Reymond, A., Gerstein, M., Guigó, R., and Hubbard, T. J. GENCODE: The reference human genome annotation for The ENCODE Project. *Genome Research*, 22(9): 1760–1774, January 2012. ISSN 1088-9051, 1549-5469. doi: 10.1101/gr.135350.111. URL http://genome.cshlp.org/content/22/9/1760.

Hoang, T. L., Sbodio, M. L., Galindo, M. M., Zayats, M., Fernandez-Diaz, R., Valls, V., Picco, G., Berrospi, C., and Lopez, V. Knowledge enhanced representation learning for drug discovery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 10544–10552, 2024.

Ji, G., He, S., Xu, L., Liu, K., and Zhao, J. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 687–696, 2015.

Ji, Y., Zhou, Z., Liu, H., and Davuluri, R. V. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*, 37(15):2112–2120, 2021.

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 2181–2187, 2015.

Mudge, J. M., Carbonell-Sala, S., Diekhans, M., Martinez, J. G., Hunt, T., Jungreis, I., Loveland, J. E., Arnan, C., Barnes, I., Bennett, R., Berry, A., Bignell, A., Cerdán-Vélez, D., Cochran, K., Cortés, L. T., Davidson, C., Donaldson, S., Dursun, C., Fatima, R., Hardy, M., Hebbar, P., Hollis, Z., James, B. T., Jiang, Y., Johnson, R., Kaur, G., Kay, M., Mangan, R. J., Maquedano, M., Gómez, L. M., Mathlouthi, N., Merritt, R., Ni, P., Palumbo, E., Perteghella, T., Pozo, F., Raj, S., Sisu, C., Steed, E., Sumathipala, D., Suner, M.-M., Uszczynska-Ratajczak, B., Wass, E., Yang, Y. T., Zhang, D., Finn, R. D., Gerstein, M., Guigó, R., Hubbard, T. J. P., Kellis, M., Kundaje, A., Paten, B., Tress, M. L., Birney, E., Martin, F. J., and Frankish, A. GENCODE 2025: Reference gene annotation for human and mouse. 53(D1):D966–D975, 2025. ISSN 1362-4962. doi: 10.1093/nar/gkae1078. URL https://doi.org/10.1093/nar/gkae1078.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

RNAcentral Consortium. RNAcentral 2021: Secondary structure integration, improved sequence search and new member databases. *Nucleic Acids Research*, 49(D1): D212–D220, January 2021. ISSN 0305-1048. doi: 10.1093/nar/gkaa921. URL https://doi.org/10.1093/nar/gkaa921.

Sennrich, R., Haddow, B., and Birch, A. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015.

Sun, Z., Deng, Z.-H., Nie, J.-Y., and Tang, J. Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations (ICLR)*, 2019.

Wang, L., Zhang, X., Su, H., and Zhu, J. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Wang, Z., Zhang, J., Feng, J., and Chen, Z. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1112–1119, 2014.

Yang, B., Yih, W.-t., He, X., Gao, J., and Deng, L. Embedding entities and relations for learning and inference in knowledge bases. In *International Conference on Learning Representations (ICLR)*, 2015.

Zhang, N., Bi, Z., Liang, X., Cheng, S., Hong, H., Deng, S., Zhang, Q., Lian, J., and Chen, H. Ontoprotein: Protein pretraining with gene ontology embedding. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=yfe1VMYAXa4.

Zhou, Z., Ji, Y., Li, W., Dutta, P., Davuluri, R. V., and Liu, H. DNABERT-2: Efficient foundation model and benchmark for multi-species genomes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=oMLQB4EZE1.

# A. Experimental Details

## A.1. Pre-training

We pre-train DNABERT-2 across four GPUs, using a per-device batch size of 16 with gradient accumulation over four steps to simulate a larger effective batch. The model is optimized with AdamW at a learning rate of $5e-5$, governed by a linear scheduler throughout training, and employs masked language modeling with 15% of tokens randomly masked each step.

## A.2. Fine-tuning

We fine-tune the pre-trained models on a range of tasks from the GUE benchmark using a unified and extensible training pipeline. Each dataset is formatted into three files: `train.csv`, `dev.csv`, and `test.csv`, each consisting of `sequence,label` pairs. Training is conducted with FP16 mixed precision and a global batch size of 32, achieved by setting `per_device_train_batch_size=8` and `gradient_accumulation_steps=1` across A100 GPUs clusters.

Task-specific configurations are tailored to reflect the sequence characteristics and difficulty. For instance, we set `model_max_length` to 20 for core promoter detection, 70 for proximal promoter task. The number of training epochs varies from 3 to 10, depending on the dataset. Model checkpoints are evaluated every 200 or 400 steps, and the checkpoint with the best development set performance is selected for final evaluation. Our training framework is fully modular, allowing seamless adaptation to new tasks with minimal configuration changes.

# B. Results

| **Model** | **TF binding site prediction** | | | | | **Core promoter detection** | | | **Promoter detection** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | TF_0 | TF_1 | TF_2 | TF_3 | TF_4 | all | notata | tata | all | notata | tata |
| Vanilla | 78.5 | **82.6** | **77.2** | **72.1** | **83.1** | 78.0 | 80.1 | **81.7** | **89.5** | **92.6** | **78.4** |
| DistMult | 78.0 | 78.7 | 73.6 | 64.8 | 80.0 | 87.6 | 91.5 | 71.2 | 75.1 | 78.5 | 74.1 |
| RotatE | **79.5** | 79.4 | 74.3 | 64.2 | 80.7 | **88.1** | **92.4** | 72.0 | 75.7 | 77.9 | 72.0 |
| TransD | 75.9 | 79.0 | 74.7 | 46.3 | 79.2 | 75.7 | 78.4 | 71.2 | 88.4 | 91.4 | 73.1 |
| TransE | 79.4 | 79.9 | 72.0 | 65.5 | 78.0 | 87.6 | 91.1 | 70.7 | 76.0 | 75.5 | 71.5 |
| TransH | 77.7 | 78.8 | 73.7 | 61.4 | 80.7 | 87.0 | 91.1 | 70.1 | 76.8 | 76.2 | 73.4 |
| TransR | 79.2 | 79.9 | 73.8 | 62.5 | 80.3 | 87.7 | 91.4 | 65.2 | 76.7 | 77.5 | 72.4 |

Table 3: Performance (F1 score, %) of various $M_1$ models on GUE benchmark tasks. The $M_1$ models are additionally trained on $\mathcal{D}_1$ (*Enhancer-Graph*) during pre-training.

# C. GenomicKB and Knowledge Graph Generation

The Genomic Knowledgebase (GenomicKB) (Feng et al., 2023) joins various data sources on the human genome and organizes them in a graph database. The entities in this graph database can refer to sequences and their annotations as well as ontology information, for example, structuring the relations between different tissues. The relations connecting the entities describe the type of relationship, like for example *regulates*, which expresses that the head entity regulates the tail entity.

In Figure 4, an overview on the sequence length distributions of a subset of entity types in the GenomicKB is given. As one observes, some entity types span only a very small range of possible sequence lengths (here *promoter* and *proximal_enhancer*), while others span multiple orders of magnitude. A particularity of GenomicKB is, that the actual DNA sequence of an entity is not stored with the entity itself but within the entities of type *chr_chain* and associated via the `locate_on_chain` relation to prevent redundancy from overlapping entities. In addition, the human genome is splitted up in parts of length 200 nucleotides. For the dataset generation, we retrieve the actual DNA sequence of an entity by extracting it from the corresponding *chr_chain* entities and joining the sequence chunks.

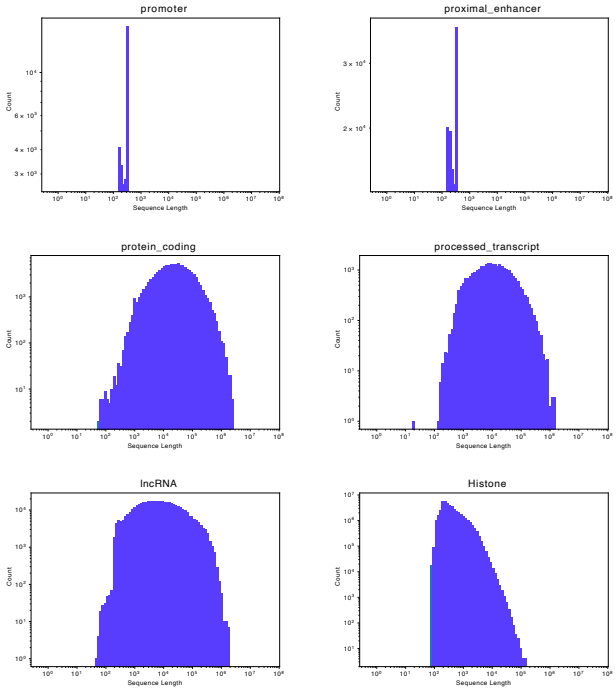


Figure 4: The distribution of sequence lengths for different entity types included in the GenomicKB.