A BASELINES DETAILS

In this section, we introduce the details of baselines mentioned in Section 4.1.

A.1 TEXT-ONLY METHODS

• **ARNN** (text only) [4], devises an Attention-RNN to reduce the noise in local contexts and proposes a novel method for sampling informative negative samples.

• **BERT** (text only) [3], is involved to learn the textual features and then learns the matching score between mentions and entity candidates through contrastive learning.

• **BLINK** (text only) [12], proposes a two-stage zero-shot linking algorithm, which first retrieves the entity candidates through two independent bi-encoder and then re-ranked the candidates through a cross-encoder.

• **GENRE** (text only) [2], is an autoregressive end-to-end framework that retrieves entities by generating their unique names, which helps to capture the fine-grained interactions between contexts and entities and also solves the memory storage problem for large entity sets.

• **GHMFC-onlytext** (text only) [10], utilizes the token-level and phrase-level feature extraction to capture the fine-grained textual representations. It also adopts the contrastive learning to minimize the distance between mentions and positive entities, while maximizing the distance between mentions and negative entities.

A.2 TEXT-VISION METHODS

• **JMEL** (text + vision) [1], employs fully connected layers to embed the visual and textual features into the same space and leverages triplet loss to measure the distance between positive and negative entities.

• **DZMNE-BERT** (text + vision) [7], designs a hybrid module with a modality attention to focus on addressing the entity disambiguation problem. For fairness, we also leverage pre-trained BERT and CLIP model to extract the textual and visual representations.

• **HieCoATT-Alter** (text + vision) [6], proposes an alternating co-attention mechanism to fuse the multimodal information and helps the model to capture the fine-grained joint representations.

• **GHMFC** (text + vision) [10], proposes a hierarchical multimodal co-attention module with a gated fusion to learn the fine-grained inter-modal correlations and a multimodal contrastive loss to reduce the noise.

• **LXMERT** (text + vision) [11], adopts two independent multimodal encoders to learn the representation of contexts and entity candidates. Then leverage the pre-trained LXMERT [9] to fuse the multiple modalities.
B IMPLEMENTATION DETAILS

In this section, we provide the implementation details of our method. Our MMEL framework is implemented with PyTorch on NVIDIA RTX A6000. We leverage the pre-trained base-uncased BERT model [3] as the textual encoder and CLIP model [8] as the visual encoder. We set the dimensions of textual and visual features, \( d_t \) and \( d_v \), to 512 and 768. The number of stacked modules \( q \) is 2 and the new size of visual features \( k \) is 4. The learning rate is selected as 5e-5 and the dropout rate is set 0.2 to avoid overfitting. We leverage the AdamW [5] to optimize the whole parameters with the batch size 32. Following [10], we employ the longest common subsequence algorithm, common prefix and normalized edit distance between contexts and entities to obtain \(|E| = 100\) candidate entities for each mention. The Top-k metrics are adopted to measure the performance of models and all the hyper-parameters are manually adjusted based on the top-5 result on the validation set.

REFERENCES


