CLAIMS

What is claimed is:

1. A system comprising a computing device, the computing device comprising a processer and a storage device storing computer executable code, wherein the computer executable code, when executed at the processor, is configured to:

provide a document;

embed a plurality of entities in the document into a plurality of embedding vectors; and

predict one of a plurality of relations between a first entity in the document and a second entity in the document based on a first embedding vector and a second embedding vector, the first embedding vector of the plurality of embedding vectors representing the first entity, and the second embedding vector of the plurality of embedding vectors representing the second entity,

wherein the computer executable code is configured to embed and predict using a language model stored in the computing device, each of the plurality of relations has an adaptive threshold, and the one of the plurality of relations is determined to exist when a logit of the relation is greater than a logit function of corresponding one of the adaptive thresholds of the relations.

2. The system of claim 1, wherein the computer executable code is configured to embed each of the plurality of entities by summarizing at least one hidden representation of at least one mention of the entity using LogSumExp (LSE).

3. The system of claim 1, wherein the computer executable code is configured to predict one of a plurality of relations by calculating a local context pooling for a pair of entities selected from the plurality of entities using:

$A^{(s,o)} = A_s^E \cdot A_o^E$,

$q^{(s,o)} = \sum_{i=1}^{H} {A_i}^{(s,o)}$,

$a^{(s,o)} = q^{(s,o)}/1^T q^{(s,o)}$, and

$$c^{(s,o)} = \boldsymbol{H}^T a^{(s,o)},$$

wherein the pair of entities comprises a subject entity and an object entity, A_s^E is a token-level attention heads of the subject entity, A_o^E is a token-level attention heads of the object entity, $A^{(s,o)}$ is a multiplication of A_s^E and A_o^E, H in $\sum_{i=1}^{H} A_i{}^{(s,o)}$ is a number of attention heads, $A_i{}^{(s,o)}$ is an i-th multiplication of H multiplications, $a^{(s,o)}$ is normalization of $q^{(s,o)}$ to sum 1, $\boldsymbol{H}$ in $\boldsymbol{H}^T a^{(s,o)}$ is last layer embedding of the language model that is pre-trained, and $c^{(s,o)}$ is the local context pooling for the pair of entities.

4. The system of claim 3, wherein hidden states of the subject entity and the object entity are determined by:

$$z_s^{(s,o)} = tanh\left(\boldsymbol{W}_s \boldsymbol{h}_{e_s} + \boldsymbol{W}_{C1} c^{(s,o)}\right), \text{ and}$$

$$z_o^{(s,o)} = tanh\left(\boldsymbol{W}_o \boldsymbol{h}_{e_o} + \boldsymbol{W}_{C2} c^{(s,o)}\right),$$

wherein $\boldsymbol{h}_{e_s}$ is the embedding of the subject entity, $z_s^{(s,o)}$ is hidden state of the subject entity, $\boldsymbol{h}_{e_o}$ is the embedding of the object entity, $z_o^{(s,o)}$ is hidden state of the object entity, and $\boldsymbol{W}_s$, $\boldsymbol{W}_o$, $\boldsymbol{W}_{C1}$, and $\boldsymbol{W}_{C2}$ are model parameters.

5. The system of claim 4, wherein the computer executable code is configured to predict relation between the subject entity and the object entity using:

$$logit_r = \sum_{i=1}^{k} z_s^{iT} \boldsymbol{W}_r^i z_o^i + b_r,$$

wherein $logit_r$ is logit function of the subject entity e_s and the object entity e_o in regard to the relation r, k is a positive integer, dimensions of the $z_s^{(s,o)}$ are divided by k to form a plurality of z_s^i, dimension of the $z_o^{(s,o)}$ are divided by k to form a plurality of z_o^i, and $\boldsymbol{W}_r^i$ and b_r are model parameters; and

wherein when the $logit_r$ is greater than a logit function of a learnable threshold TH of the relation r, the subject entity e_s and the object entity e_o have the relation r.

6. The system of claim 5, wherein the dimensions of the $z_s^{(s,o)}$ and the dimensions of the $z_o^{(s,o)}$ are 768, and *k* is 12.

7. The system of claim 1, wherein the language model comprises at least one of a bidirectional encoder representations from transformer (BERT), a robustly optimized BERT approach (roBERTa), SciBERT, a generative pre-training model (GPT), a GPT-2, and a reparameterized transformer-XL network (XLnet).

8. The system of claim 7, wherein loss function for training the language model is determined by:

$$logit_r = \sum_{i=1}^{k} z_s^{iT} \boldsymbol{W}_r^i z_o^i + b_r,$$

$$L_1 = -\sum_{r \in P_T} \log\left(\frac{exp(logit_r)}{\sum_{r' \in P_T \cup \{TH\}} exp(logit_{r'})}\right),$$

$$L_2 = -\log\left(\frac{exp(logit_{TH})}{\sum_{r' \in N_T \cup \{TH\}} exp(logit_{r'})}\right), \text{ and}$$

$$L = L_1 + L_2,$$

wherein $logit_r$ is logit function of the subject entity e_s and the object entity e_o in regard to the relation *r*, *k* is a positive integer, dimensions of the $z_s^{(s,o)}$ are divided by *k* to form a plurality of z_s^i, dimension of the $z_o^{(s,o)}$ are divided by *k* to form a plurality of z_o^i, $\boldsymbol{W}_r^i$ and b_r are model parameters, *TH* is a learnable threshold of the relation, P_T represents positive classes of relations, and *NT* represents negative classes of relations.

9. The system of claim 1, wherein the computer executable code is further configured to: use the first entity, the second entity, and the predicted one of the plurality of relations between the first entity and the second entity to construct a knowledge graph.

10. The system of claim 1, wherein the computer executable code is further configured to, when a question comprises the first entity and the second entity, and the document

is predetermined to comprise an answer to the question: use the predicted one of the plurality of relations to form the answer.

11. A method comprising:

providing, by a computing device, a document;

embedding, by a computing device, a plurality of entities in the document into a plurality of embedding vectors; and

predicting, by a computing device, one of a plurality of relations between a first entity in the document and a second entity in the document based on a first embedding vector and a second embedding vector, the first embedding vector of the plurality of embedding vectors representing the first entity, and the second embedding vector of the plurality of embedding vectors representing the second entity,

wherein the steps of embedding and predicting are performed by a language model stored in the computing device, each of the plurality of relations has an adaptive threshold, and the one of the plurality of relations is determined to exist when a logit of the relation is greater than a logit function of corresponding one of the adaptive thresholds of the relations.

12. The method of claim 11, wherein the steps of embedding of each of the plurality of entities is performed by summarizing at least one hidden representation of at least one mention of the entity using LogSumExp (LSE).

13. The method of claim 11, wherein the step of predicting comprises calculating a local context pooling for a pair of entities selected from the plurality of entities using:

$A^{(s,o)} = A_s^E \cdot A_o^E$,

$q^{(s,o)} = \sum_{i=1}^{H} A_i{}^{(s,o)}$,

$a^{(s,o)} = q^{(s,o)}/1^T q^{(s,o)}$, and

$c^{(s,o)} = \boldsymbol{H}^T a^{(s,o)}$,

wherein the pair of entities comprises a subject entity and an object entity, A_s^E is a token-level attention heads of the subject entity, A_o^E is a token-level attention

heads of the object entity, $A^{(s,o)}$ is a multiplication of A_s^E and A_o^E, H in $\sum_{i=1}^{H} A_i{}^{(s,o)}$ is a number of heads, $A_i{}^{(s,o)}$ is an i-th multiplication of H multiplications, $a^{(s,o)}$ is normalization of $q^{(s,o)}$ to sum 1, $\boldsymbol{H}$ in $\boldsymbol{H}^T a^{(s,o)}$ is last layer embedding of the language model that is pre-trained, and $c^{(s,o)}$ is the local context pooling for the pair of entities.

14. The method of claim 13, wherein hidden states of the subject entity and the object entity are determined by:

$$z_s^{(s,o)} = tanh\left(\boldsymbol{W}_s \boldsymbol{h}_{e_s} + \boldsymbol{W}_{C1} c^{(s,o)}\right), \text{ and}$$

$$z_o^{(s,o)} = tanh\left(\boldsymbol{W}_o \boldsymbol{h}_{e_o} + \boldsymbol{W}_{C2} c^{(s,o)}\right),$$

wherein $\boldsymbol{h}_{e_s}$ is the embedding of the subject entity, $z_s^{(s,o)}$ is hidden state of the subject entity, $\boldsymbol{h}_{e_o}$ is the embedding of the object entity, $z_o^{(s,o)}$ is hidden state of the object entity, and $\boldsymbol{W}_s$, $\boldsymbol{W}_o$, $\boldsymbol{W}_{C1}$, and $\boldsymbol{W}_{C2}$ are model parameters.

15. The method of claim 14, wherein the step of predicting relation between the subject entity and the object entity is performed using:

$$logit_r = \sum_{i=1}^{k} z_s^{iT} \boldsymbol{W}_r^i z_o^i + b_r,$$

wherein $logit_r$ is logit function of the subject entity e_s and the object entity e_o in regard to the relation r, k is a positive integer, dimensions of the $z_s^{(s,o)}$ are divided by k to form a plurality of z_s^i, dimension of the $z_o^{(s,o)}$ are divided by k to form a plurality of z_o^i, and $\boldsymbol{W}_r^i$ and b_r are model parameters; and

wherein when the $logit_r$ is greater than a logit function of a learnable threshold TH of the relation r, the subject entity e_s and the object entity e_o have the relation r.

16. The method of claim 15, wherein the language model comprises a bidirectional encoder representations from transformer (BERT) or SciBERT, and the loss function for training the language model is determined by:

$$logit_r = \sum_{i=1}^{k} z_s^{iT} \boldsymbol{W}_r^i z_o^i + b_r,$$

$$L_1 = -\sum_{r \in P_T} \log\left(\frac{exp(logit_r)}{\sum_{r' \in P_T \cup \{TH\}} exp(logit_{r'})}\right),$$

$$L_2 = -\log\left(\frac{exp(logit_{TH})}{\sum_{r' \in N_T \cup \{TH\}} exp(logit_{r'})}\right), \text{ and}$$

$$L = L_1 + L_2,$$

wherein $logit_r$ is logit function of the subject entity e_s and the object entity e_o in regard to the relation *r*, *k* is a positive integer, dimensions of the $z_s^{(s,o)}$ are divided by *k* to form a plurality of z_s^i, dimension of the $z_o^{(s,o)}$ are divided by *k* to form a plurality of z_o^i, $\boldsymbol{W}_r^i$ and b_r are model parameters,, *TH* is a learnable threshold of the relation, P_T represents positive classes of relations, and *NT* represents negative classes of relations.

17. The method of claim 11, further comprising:

using the first entity, the second entity, and the predicted one of the plurality of relations between the first entity and the second entity to construct a knowledge graph; or

when a question comprises the first entity and the second entity, and the document is predetermined to comprise an answer to the question: using the predicted one of the plurality of relations to form the answer.

18. A non-transitory computer readable medium storing computer executable code, wherein the computer executable code, when executed at a processor of an active computing device, is configured to:

provide a document;

embed a plurality of entities in the document into a plurality of embedding vectors; and

predict one of a plurality of relations between a first entity in the document and a second entity in the document based on a first embedding vector and a second embedding vector, the first embedding vector of the plurality of embedding vectors

representing the first entity, and the second embedding vector of the plurality of embedding vectors representing the second entity,

wherein the computer executable code is configured to embed and predict using a language model stored in the non-transitory computer readable medium, each of the plurality of relations has an adaptive threshold, and the one of the plurality of relations is determined to exist when a logit of the relation is greater than a logit function of corresponding one of the adaptive thresholds of the relations.

19. The non-transitory computer readable medium of claim 18,

wherein the computer executable code is configured to predict one of a plurality of relations by calculating a local context pooling for a pair of entities selected from the plurality of entities using:

$A^{(s,o)} = A_s^E \cdot A_o^E$,

$q^{(s,o)} = \sum_{i=1}^{H} A_i{}^{(s,o)}$,

$a^{(s,o)} = q^{(s,o)} / 1^T q^{(s,o)}$, and

$c^{(s,o)} = H^T a^{(s,o)}$, wherein the pair of entities comprises a subject entity and an object entity, A_s^E is a token-level attention heads of the subject entity, A_o^E is a token-level attention heads of the object entity, $A^{(s,o)}$ is a multiplication of A_s^E and A_o^E, H is a number of heads, $A_i{}^{(s,o)}$ is an i-th multiplication of H multiplications, $a^{(s,o)}$ is normalization of $q^{(s,o)}$ to sum 1, and $c^{(s,o)}$ is the local context pooling for the pair of entities;

wherein hidden states of the subject entity and the object entity are determined by:

$z_s^{(s,o)} = tanh\left(\boldsymbol{W}_s \boldsymbol{h}_{e_s} + \boldsymbol{W}_{C1} c^{(s,o)}\right)$, and

$z_o^{(s,o)} = tanh\left(\boldsymbol{W}_o \boldsymbol{h}_{e_o} + \boldsymbol{W}_{C2} c^{(s,o)}\right)$, wherein $\boldsymbol{h}_{e_s}$ is the embedding of the subject entity, $z_s^{(s,o)}$ is hidden state of the subject entity, $\boldsymbol{h}_{e_o}$ is the embedding of the object entity, $z_o^{(s,o)}$ is hidden state of the object entity, and $\boldsymbol{W}_s$, $\boldsymbol{W}_o$, $\boldsymbol{W}_{C1}$, and $\boldsymbol{W}_{C2}$ are model parameters; and

wherein the computer executable code is configured to predict relation

between the subject entity and the object entity using:

$logit_r = \sum_{i=1}^{k} z_s^{iT} \boldsymbol{W}_r^i z_o^i + b_r$, wherein $logit_r$ is logit function of the subject entity e_s and the object entity e_o in regard to the relation *r*, *k* is a positive integer, dimensions of the $z_s^{(s,o)}$ are divided by *k* to form a plurality of z_s^i, dimension of the $z_o^{(s,o)}$ are divided by *k* to form a plurality of z_o^i, $\boldsymbol{W}_r^i$ and b_r are model parameters, and when the $logit_r$ is greater than a logit function of a learnable threshold *TH* of the relation *r*, the subject entity e_s and the object entity e_o have the relation *r*.

20. The non-transitory computer readable medium of claim 18, wherein the language model comprises a bidirectional encoder representations from transformer (BERT), and the loss function for training the language model is determined by:

$$logit_r = \sum_{i=1}^{k} z_s^{iT} \boldsymbol{W}_r^i z_o^i + b_r,$$

$$L_1 = -\sum_{r \in P_T} \log\left(\frac{exp(logit_r)}{\sum_{r' \in P_T \cup \{TH\}} exp(logit_{r'})}\right),$$

$$L_2 = -\log\left(\frac{exp(logit_{TH})}{\sum_{r' \in N_T \cup \{TH\}} exp(logit_{r'})}\right), \text{ and}$$

$$L = L_1 + L_2,$$

wherein $logit_r$ is logit function of the subject entity e_s and the object entity e_o in regard to the relation *r*, *k* is a positive integer, dimensions of the $z_s^{(s,o)}$ are divided by *k* to form a plurality of z_s^i, dimension of the $z_o^{(s,o)}$ are divided by *k* to form a plurality of z_o^i, $\boldsymbol{W}_r^i$ and b_r are model parameters, *TH* is a learnable threshold of the relation, P_T represents positive classes of relations, and *NT* represents negative classes of relations.