applied sciences

MDPI

Article

An Improved Nested Named-Entity Recognition Model for Subject Recognition Task under Knowledge Base Question Answering

Ziming Wang, Xirong Xu *, Xinzi Li, Haochen Li, Xiaopeng Wei and Degen Huang

School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, China
* Correspondence: xirongxu@dlut.edu.cn

Abstract: In the subject recognition (SR) task under Knowledge Base Question Answering (KBQA), a common method is by training and employing a general flat Named-Entity Recognition (NER) model. However, it is not effective and robust enough in the case that the recognized entity could not be strictly matched to any subjects in the Knowledge Base (KB). Compared to flat NER models, nested NER models show more flexibility and robustness in general NER tasks, whereas it is difficult to employ a nested NER model directly in an SR task. In this paper, we take advantage of features of a nested NER model and propose an Improved Nested NER Model (INNM) for the SR task under KBQA. In our model, each question token is labeled as either an entity token, a start token, or an end token by a modified nested NER model based on semantics. Then, entity candidates would be generated based on such labels, and an approximate matching strategy is employed to score all subjects in the KB based on string similarity to find the best-matched subject. Experimental results show that our model is effective and robust to both single-relation questions and complex questions, which outperforms the baseline flat NER model by a margin of 3.3% accuracy on the SimpleQuestions dataset and a margin of 11.0% accuracy on the WebQuestionsSP dataset.

Keywords: natural language processing; knowledge base question answering; subject recognition; named-entity recognition; deep learning; robustness

check for updates

Citation: Wang, Z.; Xu, X.; Li, X.; Li, H.; Wei, X.; Huang, D. An Improved Nested Named−Entity Recognition Model for Subject Recognition Task under Knowledge Base Question Answering. *Appl. Sci.* **2023**, *13*, 11249. https://doi.org/10.3390/app132011249

Academic Editor: Douglas O'Shaughnessy

Received: 19 July 2023
Revised: 28 September 2023
Accepted: 11 October 2023
Published: 13 October 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Knowledge Base Question Answering (KBQA) is a Natural Language Processing (NLP) task which is aimed at answering natural language questions automatically with facts in a Knowledge Base (KB). In a KBQA task, subject recognition (SR) is usually the prerequisite to obtain the answer to both single-relation questions and complex questions. In general, a general flat Named-Entity Recognition (NER) model is usually trained and employed in the SR task to recognize the entity in the question as the subject. However, sometimes, the golden entity recognized by the flat NER model is different from the golden subject in the subject recognition task of a similar question; examples are shown in Table 1.

In these examples, the golden subject in the KB may be different from the golden entity recognized from the question by a flat NER model because of spelling errors, missed or unexpected words, characters, or spaces, unknown subject aliases, etc. In these cases, the golden subject could not be strictly matched to all n-grams, which could be generated from the question [1], so it is difficult to obtain the golden subject even though the golden entity is recognized. Further, if the recognized entity is not the golden one, it would be more difficult to obtain the golden subject. As a result, a flat NER model is not effective in these cases.

Unfortunately, in practical applications, these cases would occur frequently: a question may be fed to a KBQA system after multiple processes of transmission, transformation, or translation, and the actual input question to a KBQA system may contain various errors caused by users or noise in these processes. In addition, it is impractical for a KB to contain

all possible aliases to an entity, and for a NER model to achieve an accuracy of 100%. As a result, it is necessary to propose a model which is effective and robust to these questions.

Table 1. Difference between golden entities and subjects.

Question	Golden Entity	Golden Subject
what is iqsdirectory.com?	iqsdirectory.com	iqs directory
in footbal, what position does Tserenjav Enkhjargal play?	Tserenjav Enkhjargal	Tserenjavyn Enkhjargal
what area is blackwireuk from?	blackwireuk	black wire
what country is Guanica, Puerto Rico in?	Guanica, Puerto Rico	Guánica
what label is chrisadamsstringdriventhing under?	chrisadamsstring-driventhing	string driven thing

Before describing our proposed model, we first describe the general flat NER model under SR tasks, which is shown in Figure 1a. To the example question "where is mission san buenaventura located" with the golden subject "mission san buenaventura" (bold font), several successive tokens (gray background) in the question is selected as the recognized entity in a flat NER model (e.g., BERT-CRF). Then, the recognized entity is supposed to be strictly matched to a certain subject in the KB and the subject is outputted as the recognized subject.

Obviously, if the recognized entity could not be strictly matched to any subject in the KB, this model would fail. In this case, an approximate matching strategy would be employed, as shown in Figure 1b. In this strategy, as "mission san" is the recognized entity, all possible tokens started with "mission" or ended with "san" in the question are approximately matched to all subjects in the KB. The best-matched one is outputted as the recognized subject.

Further, entity tokens, start tokens, and end tokens could be considered as different types of entities in a NER model. As a token could belong to multiple types of entities, a nested NER model would be employed. As shown in Figure 1c, besides the recognized entity "mission san" (gray background), "mission" is recognized as the start token (green background) and "located" is recognized as the end token (blue background). Then, tokens starting with "mission" or ending with "located" in the question are approximately matched to subjects in the KB to find the best-matched one.

In addition, as different NER models could recognize different entities to the same question, an integrated method could also be employed as shown in Figure 1d, which is our proposed model. In this model, a flat NER model (Model A) is employed to find the strictly matched subject first. If it fails, another nested NER model (Model B) will be employed to find the approximately matched subject.

In this paper, we propose an Improved Nested NER Model (INNM) for subject recognition tasks under KBQA. In our model, a nested NER model is employed to label each question token. Then, all possible entity candidates are generated based on these labeled tokens and then all subjects in the KB are scored based on the similarity. The best-matched subject would be found based on the scores.

The contributions of this paper are summarized as follows:

- We propose an Improved Nested NER model rather than a flat NER model for the SR task under KBQA. In the case that a general flat NER model fails to recognize the golden entity, our model could still have an opportunity to recognize the golden entity, which shows more effectiveness than baseline flat NER models.
- We employ an approximate matching strategy rather than a strict matching strategy in our model. This strategy shows better effectiveness and robustness especially to noisy

questions where the golden entity is different from the golden subject. Experimental results show that our model is more robust than baseline flat NER models.

- Our model is effective to both single−relation questions and complex questions. Experimental results show that our model outperforms the baseline flat NER model by a margin of 3.3% accuracy on the SimpleQuestions dataset (single−relation questions) and a margin of 11.0% accuracy on the WebQuestionsSP dataset (complex questions).

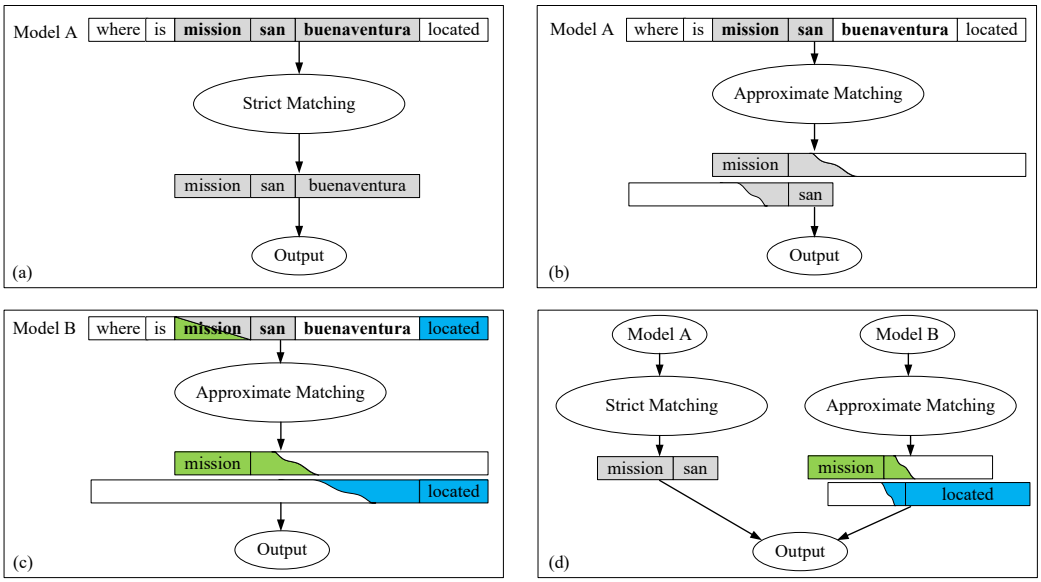


Figure 1. Different subject recognition models and their entity tokens (gray background), start tokens (green background), end tokens (blue background). (**a**) Flat NER model and strict matching strategy. (**b**) Flat NER model and approximate matching strategy. (**c**) Nested NER model and approximate matching strategy. (**d**) Our proposed model.

2. Related Work

The research of KBQA has evolved from earlier domain-specific question answering [2] to open-domain QA based on large-scale KBs such as Freebase [3]. The model of KBQA has also evolved from semantic parsing-based models [4], which parse questions into structured queries, to neural network-based models [5,6], which learn semantic representations of both the question and the knowledge from observed data. Some researchers [7–9] also attempt to combine multiple models to utilize information in natural language questions and KBs.

After pre-trained models such as BERT [10], ALBERT [11], XLNet [12], and ELECTRA [13] are proposed, they have been widely employed in various NLP tasks [14–17]. Many researchers employed NLP models based on pre-trained models and achieved good results. For example, Gangwar et al. [18] employed pre-trained models in the span extraction, classification, and relation extraction tasks focused on finding quantities, the attributes of these quantities, and additional information. Luo et al. [19] proposed a BERT-based approach for single-relation question answering (SR-QA), which consists of two models, entity linking, and relation detection. Zhu et al. [20] designed a comprehensive search space for BERT-based relation classification models and employed the neural architecture search method to automatically discover the design choices. However, in different situations, the best-performance model is also different. For example, ELECTRA achieves better performance in some tasks in GLUE [21], while ALBERT requires less training cost, and RoFormer is more effective for Chinese NLP tasks. However, it could be difficult to employ them in KBQA as the difference between the subject recognition task in KBQA and the general NER task is significant.

After the subject is recognized, the answer will be found using a relation extraction (RE) model for single-relation questions or other models for complex questions. For exam-

ple, a trainable subgraph retriever decoupled from the subsequent reasoning process is proposed to achieve better retrieval and QA performance [22], a dynamic program induction and a dynamic contextualized encoding are employed to address both the large search space and schema linking in a unified framework [23], and an effective and transparent model is proposed to support both label and text relations in a unified framework [24]. However, these models rely on the accurate recognition of subjects, and a wrong subject would lead to a wrong answer.

In practical applications, an NLP model is often designed to answer noisy and abnormal questions caused by various reasons (e.g., noise in the processes of transmission, transformation, or translation). Sometimes, the input to an NLP system is even transformed from a piece of voice, video, or image. If the raw voice, video, or image is available, we could feed it directly into special models such as VL-BERT [25], LXMERT [26], VideoBERT [27], ClipBERT [28], wav2vec [29], or SpeechBERT [30] to avoid errors caused by transformation. In addition, the structure of the original model could also be improved so that such noise could be handled automatically by the model. For example, Yang et al. [31] proposed a robust and structurally aware table-text encoding architecture, TableFormer, where tabular structural biases are incorporated completely through learnable attention biases. Su et al. [32] proposed a pre-trained Chinese BERT that is robust to various forms of adversarial attacks like word perturbation, synonyms, typos, etc. Liu et al. [33] proposed a robustly optimized bidirectional machine reading comprehension method by incorporating four improvements. Furthermore, a number of research works focus on finding and eliminating noisy labels in datasets so that the model could be trained without noise. For example, Zhu et al. [34] showed that for text classification tasks with modern NLP models like BERT, over a variety of noise types, existing noisehandling methods do not always improve its performance. Ye et al. [35] proposed a general framework named label noise-robust dialogue state tracking to train DST models robustly from noisy labels instead of improving the annotation quality further. Nguyen et al. [36] studied the impact of instance-dependent noise to the performance of product title classification by comparing data denoising algorithm and different noise-resistance training algorithms, which were designed to prevent a classifier model from over-fitting to noise. However, compared to an RE model, a NER model is much more sensitive to noise and an entity with a wrong character would be matched to a wrong subject. As a result, it is difficult to employ these methods directly for subject recognition in KBQA.

GlobalPointer (GP) [37] and its improvement Efficient GlobalPointer (EGP) [38] are two of the latest nested NER models, which have achieved satisfactory performance in both flat and nested NER tasks with less training time. Compared with other nested NER models, they have mainly two advantages: (1) they can work as both flat and nested NER models; (2) they are pre-trained models, so it is easy to modify and fine-tune them for different tasks, just like BERT. For these reasons, we choose these models as the baseline nested NER models in this paper. However, our experimental results show that it is not particularly effective on the SimpleQuestions dataset. As a result, in this paper, we modify them as components in our model rather than simply employ them, and our experimental results show that our model is effective and robust on both the SimpleQuestions dataset and WebQuestionsSP dataset.

3. Approach

3.1. Overview

A KB, such as Freebase, contains three components: a set of entities E, a set of relations R, and a set of facts $F = \{< s, r, o >\} \subseteq E \times R \times E$, where $< s, r, o >$ are subject–relation–object tuples. In subject recognition in KBQA, to an input question q, a flat NER model (e.g., BERT-CRF) is usually employed to recognize the best-matched subject $s' \in E$.

In this model, an input question q is split into n tokens $t_1, t_2, \ldots, t_n$ and each token is labeled as entity tokens or no-entity tokens. If $t_a, t_{a+1}, \ldots t_b$ are successive tokens, which are labeled as entity tokens, the best-matched entity as well as the best-matched subject would

be recognized as the combination of these entity tokens (as shown in Figure 1a). Obviously, the recognized subject would be a golden one only if:

- t_a is the start token of the golden subject;
- t_b is the end token of the golden subject;
- All tokens between t_a and t_b could be strictly matched to corresponding golden subject tokens.

In the case that t_a or t_b is wrongly labeled, or one or more entity tokens are different from their corresponding golden subject tokens, this model would fail.

To improve this model, in this paper, we propose IMNM, which mainly contains three components: token labeling, entity candidate generation, and approximate matching (as shown in Figure 2).

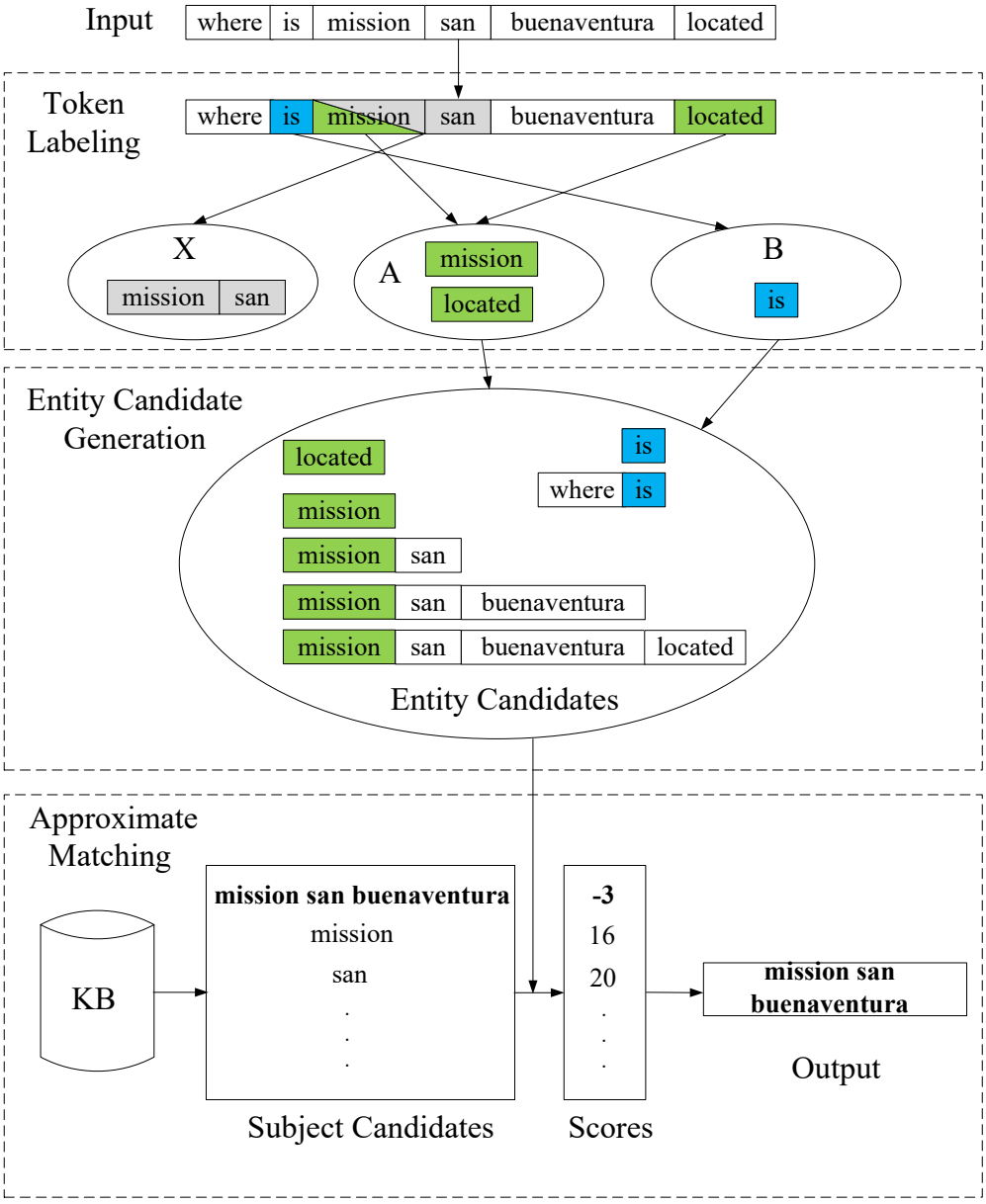


Figure 2. The overall structure of our model and its entity tokens (gray background), start tokens (green background), end tokens (blue background).

In token labeling, to an example question "where is mission san buenaventura located" with the golden subject "mission san buenaventura", a nested NER model is employed and three sets are generated: a set of entity tokens X, a set of start tokens A, and a set of

end tokens B. Unlike the flat NER model, each token that can be labeled multiply in a nested NER model. For example, token "mission" could be labeled as an entity token as well as a start token. Ideally, each of these sets should contain one and only one element: X = {mission san buenaventura}, A = {mission}, B = {buenaventura}. In this case, the element in X could be considered as the recognized subject.

However, sometimes, a set may contain no elements, a wrong element, or more than one element. In this case, all possible entity candidates are generated based on these elements in entity candidate generation. Then, all subject candidates in a KB is scored based on their similarity in approximate matching. The subject with the lowest score would be considered as the best-matched one. For example, the subject candidate "mission san buenaventura" in the KB has the highest similarity to one of the entity candidates ("mission san buenaventura") with a score of -3.0, so it is considered as the recognized subject. Our model will be described in detailed in the following subsections.

3.2. Token Labeling

To label each token in $q = \{t_1, t_2, \ldots, t_n\}$ (mainly based on semantics), a NER model would be employed. In a flat NER model, each token is labeled as one and only one entity type, and a vector $l_q = \{l_1, l_2, \ldots, l_n\}$ would be generated (each of $l_1, \ldots, l_n$ belongs to a certain entity type). In a nested NER model, as two recognized entities could have common tokens and one entity could belong to several entity types, we set

$$s_\alpha(i,j) = q_{i,\alpha}^{\mathrm{T}} k_{j,\alpha}^{\mathrm{T}} \tag{1}$$

as the score of the span $t_i, \ldots, t_j$ to be an entity of type α. In the equations

$$q_{i,\alpha} = H_{q,\alpha} e_i + b_{q,\alpha} \tag{2}$$

$$k_{i,\alpha} = H_{k,\alpha} e_i + b_{k,\alpha} \tag{3}$$

where H and b would be trained by the model, and $e_1, \ldots, e_n$ are the embedding for token $t_1, \ldots, t_n$ by the pre-trained model. Then, we would be able to determine whether the span $t_i, \ldots, t_j$ belongs to type α based on the score $s_\alpha(i,j)$ (further details of this model can be found in references [37,38]).

In this process, m matrices ($n \times n$) $W_1, W_2, \ldots, W_m$ would be generated for n tokens of m types. In each matrix, $w_{i,j} = 1$ represents token t_i. $t_{i+1}, \ldots t_j$ are labeled as the corresponding type. As the original GP or EGP is a pre-trained model for general NER tasks and could not be simply employed in SR tasks, we modify and fine-tune it as follows.

In our model, there are three types of tokens (entity tokens, start tokens, and end tokens) to be labeled, so we set $m = 3$ and the three matrices W_1, W_2, W_3 are generated. For example, for the question "what city was alex golfis born in" of seven tokens, three matrices of 7×7 are generated as shown in Figure 3.

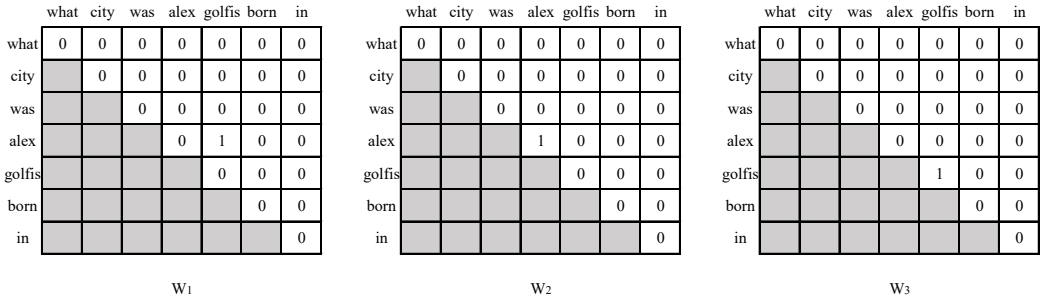


Figure 3. Matrices for the example question. (Gray background represents empty elements.)

In these matrices, $w_{4,5} = 1$ in W_1, $w_{4,4} = 1$ in W_2, and $w_{5,5} = 1$ in W_3. As a result, token "alex" is labeled as an entity token and a start token, and token "golfis" is labeled as

an entity token and an end token. Then, sets of these tokens would be generated: X = {alex golfis}, A = {alex}, B = {golfis}.

In particular, if we set $m = 1$ to label only entity tokens and generate only X, the model would be considered as a flat NER model. In addition, in a general flat NER model such as BERT-CRF, we could also obtain the start or end token of the entity simply based on the first token and the last token of the recognized entity token. However, experimental results show that these strategies result in a worse performance than our model. This model is then trained and evaluated in Section 4.

3.3. Entity Candidate Generation

If there is only one element in X and it could be strictly matched to a subject in the KB, it would be considered as the best-matched subject. Otherwise, entity candidates would be generated based on elements in A and B.

For each element in A and B, entity candidates could be generated as combinations of all possible successive tokens in the questions, which are started or ended with the element. In the aforementioned example "what city was alex golfis born in", there are four candidates starting with "alex" ("alex", "alex golfis", "alex golfis born", "alex golfis born in"), and five candidates ending with "golfis" ("golfis", "alex golfis" (repetition), "was alex golfis", "city was alex golfis", "what city was alex golfis").

Ideally, among these candidates, there should be one and only one best-matched entity, which could be strictly matched to a subject (the best-matched one) in the KB. However, sometimes, there are either none or a multiple of such candidates. As a result, following approximate matching is proposed to find the best-matched subject.

3.4. Approximate Matching

After entity candidates are generated mainly based on the semantics, it would be further pruned based on the string similarity in this section. In fact, words in entities and subjects (e.g., "golfis") are often beyond the vocabulary in BERT, so it is much more difficult to evaluate the semantic similarity than the string similarity between an entity candidate and a subject. In fact, alternative descriptions to a subject (with high semantic similarity) are supposed to be contained in the KB as subject aliases, so the string similarity between an entity candidate and a subject (or alias) could represent the relativity between them.

Levenshtein distance [39] is a common way to calculate the string similarity. To two strings $str1, str2$, Levenshtein distance describes the minimum number of editing operations (character added, deleted, or replaced) to convert $str1$ to $str2$. In general, a low Levenshtein distance indicates a high string similarity between two strings. However, if multiple entity candidates could be matched to different subjects, their Levenshtein distance would be equal. For example, the subjects "alex" and "alex golfis" in the KB have an equal Levenshtein distance of 0 to entity candidate "alex" and "alex golfis", respectively. Obviously, "alex golfis" is more likely to be the golden subject as it matches more successive characters in the question. As a result, spaces in both entity candidate c and subject s are ignored and scores can be calculated using the following expression:

$$Score(c, s) = L(c, s) - n(c) \tag{4}$$

In the expression, $L(c, s)$ is the Levenshtein distance (with spaces ignored) and $n(c)$ is the number of tokens in c.

To an input question q, the minimum score of the candidate in the set of all entity candidates C and a certain subject s would be considered as the similarity between the best-matched entity c' in the question and s:

$$Score(q, s) = Score(c', s) = \min_{c \in C} Score(c, s) \tag{5}$$

Among all subjects in the set E of the KB, the subject with the minimum $Score(q, s)$ would be considered as the best-matched subject s' to the question q:

$$s' = \arg\min_{s \in E} Score(q, s) \tag{6}$$

In this way, a best-matched subject to a question could be recognized in the case that the entity recognized by a general flat NER model could not be strictly matched to any subjects in the KB.

4. Experiments

4.1. Dataset

The SimpleQuestions (SQ) dataset [40] is a KBQA dataset of single-relation questions. It provides 108,442 single-relation questions with their answer facts, which are organized as subject–relation–object tuples in Freebase. The whole dataset is split into a training set, a validation set, and a test set, with 75,910, 10,845, and 21,687 samples, respectively. Among all samples in the test set, there are 1385 abnormal samples where the golden subject is not in FB5M (a subset of Freebase which provides E in our experiments) or different from the golden entity in the question. To these samples, even if golden entities could be recognized, they could not be strictly matched to any subject in the KB. In our experiments, the test set is divided into Dataset I, which contains 20,302 normal samples, and Dataset II, which contains 1385 abnormal samples.

The WebQuestionsSP (WQSP) dataset [41] is a KBQA dataset of complex questions (also based on Freebase), which contains 3098 samples in the training set and 1639 samples in the test set. In our experiments, the test set is also divided into Dataset III, which contains 1233 normal samples, and Dataset IV, which contains 406 abnormal samples in the similar way.

In addition, to further evaluate the robustness of our model, we generate Dataset V and Dataset VI by replacing 5% and 10% of characters randomly in all samples in WQSP. For example, the sample "who plays ken barlow in coronation street" in WQSP is converted into "who plays ktg barlpc in coronation street" in Dataset V and "who plais ken barloo in coronatign street" in Dataset VI.

4.2. Experiment Setting

Among various pre-trained models, BERT-base is a representative model as it has balanced cost and effectiveness. Compared to the original BERT model, Bert4keras (https://github.com/bojone/bert4keras, accessed on 10 June 2023) is an integrated framework that is more effective, flexible, and easier to employ and adjust. As a result, we choose BERT-base achieved by Bert4keras as the default pre-trained model in our experiments. Parameters are trained by an Adam optimizer [42] with a learning rate of 2×10^{-5} a loss function of multilabel categorical crossentropy, and a batch size of 16. Experiments are conducted with an AMD R9-5950X CPU and a GeForce RTX 3090. Both our model and the baseline models are trained by the training set in SQ and evaluated by the test set in SQ or WQSP. For comparability, we set the same batch size 16 and the same epoch 10 in each model. Their training time could be considered as the total training time of all the models as there are no trainable parameters in approximate matching.

Our model contains three components: Token Labeling (TL), Entity Candidate Generation (ECG), and Approximate Matching (AM). In TL, BERT-CRF, EGP, or other NER models could be employed, and then the best-matched subject would be generated via strict matching or our ECG + AM. In our experiments, the following methods are evaluated:

- BERT-CRF/GP/EGP (Figure 1a): BERT-CRF/GP/EGP, which is achieved by Bert4keras, is employed to generate X and the best-matched entity, which is strictly matched to the best-matched subject. In this case, GP and EGP work as flat NER models.
- BERT-CRF + AM (Figure 1b): BERT-CRF is employed to generate X, A, and B in TL. Then, the best-matched subject is generated by our ECG and AM.

- INNM-I (Figure 1c): EGP with $m = 3$ is employed to generate X, A, and B in TL. Then, the best-matched subject is generated by our ECG and AM.
- INNM-II/INNM-III (Figure 1d): BERT-CRF/EGP is employed to generate X and the best-matched entity. If the entity could be strictly matched to a subject in the KB, it would be considered as the best-matched subject. Otherwise, INNM-I/BERT-CRF + AM is employed to recognize the best-matched subject.

4.3. Experimental Results

Experiments are conducted on Datasets I–IV and the experimental results for the accuracies of subject recognition are shown in Table 2. For single-relation questions, BERT-CRF outperforms GP and EGP, and INNM further outperforms BERT-CRF. Among all of the INNM methods, INNM-II achieves the highest accuracy of 98.7% on Dataset I and 94.1% on the whole SQ (I + II), which outperforms BERT-CRF by a margin of 3.3%, while INNM-I achieves the highest accuracy of 27.4% on Dataset II. For complex questions, however, EGP outperforms BERT-CRF and GP and INNM-III achieves the highest accuracy of 92.2% on Dataset III, 6.4% on Dataset IV, and 71.0% on the whole WQSP (III + IV), which outperforms BERT-CRF by a margin of 11.0%.

Table 2. Experimental results for accuracies (%) of subject recognition.

Method	Dataset I	Dataset II	SQ	Dataset III	Dataset IV	WQSP
BERT-CRF	97.0	0	90.8	79.8	0	60.0
GP	95.5	0	89.4	80.3	0	60.4
EGP	95.6	0	89.5	81.8	0	61.5
BERT-CRF + AM	98.2	24.8	93.5	89.7	6.4	69.1
INNM-I	98.0	27.4	93.5	91.2	6.4	70.2
INNM-II	98.7	26.1	94.1	89.8	6.2	69.1
INNM-III	98.6	22.2	93.8	92.2	6.4	71.0

In addition, to abnormal questions in Dataset II, general flat NER models fail to recognize golden subjects because of the difference between the golden entity and subject, while our proposed models could recognize some of them correctly. To the normal questions in Dataset I, INNM-I fails to outperform BERT-CRF because EGP is not so effective in this case. As a result, it is a better choice to employ INNM-II where BERT-CRF (the more effective model) is employed first.

For complex questions, even normal questions in Dataset III could contain several interference words or entities, so it is much more difficult to recognize subjects correctly. In this case, EGP outperforms BERT-CRF and our INNM-I further outperforms EGP. Further outperformance would be achieved by employing INNM-III, where EGP (the more effective model) is employed first. Unfortunately, all methods fail to achieve satisfactory performance in Dataset IV. In fact, for many of the questions in Dataset IV, the golden entity is not similar to the golden subject. For example, the golden entity "jfk" (cannot even be encoded in BERT) in "who was vice president when jfk was president" is not similar to the golden subject "john f.kennedy". In this case, it is much easier to add such aliases to the KB than recognize them correctly by a model.

After the subject is recognized, to single-relation questions, we can simply employ a BERT-based RE model (also achieved by bert4keras) to predict the relation and retrieve the answer. To complex questions, relation predicting and subject recognition are usually considered as two individual tasks, so we only show experimental results for overall accuracies in SQ, which are shown in Table 3. BERT-CRF shows the best performance among traditional methods. Our models (INNM-I,II,III) further outperform it by a margin of 2.5%, 3.0%, and 2.7% respectively.

Table 3. Experimental results for overall accuracies (%) in SQ.

Method	Accuracy (%)
MemNN-Ensemble [40]	63.9
CFO [1]	75.7
BiLSTM-CRF + BiLSTM [43]	78.1
Structure Attention + MLTA [5]	82.3
BERT-CRF	82.6
GP [37]	81.3
EGP [38]	81.4
INNM-I	85.1
INNM-II	85.6
INNM-III	85.3

To further evaluate the robustness of our model, additional experiments are conducted on Datasets V and VI, and the experimental results for the accuracies of subject recognition are shown in Table 4. All INNM methods outperform BERT-CRF, and INNM-III shows the best performance among all methods listed, which outperforms BERT-CRF by an average margin of 10.4%. In addition, as EGP requires less training time than BERT-CRF in various NER tasks, our INNM-I also requires less training time (20% reduced) than BERT-CRF. To INNM-II and INNM-III, as both of BERT-CRF and EGP should be trained, more training time (80% increased) is required. As a result, INNM-I should be selected in the case that training time is more important, and INNM-II or INNM-III should be selected in the case that accuracy is more important or a trained NER model is available.

Table 4. Additional experimental results for accuracies (%) of subject recognition.

Method	WQSP	Dataset V	Dataset VI	Train Time (per Epoch)
BERT-CRF	60.0	30.2	16.6	469 s
INNM-I	70.2	37.9	21.8	376 s
INNM-II	69.1	41.1	25.1	469 s + 376 s
INNM-III	71.0	41.5	25.4	376 s + 469 s

5. Conclusions

In this paper, we propose a model for SR tasks under KBQA. In our model, a nested NER model EGP is modified and employed to label each token in an input question to generate entity tokens, start tokens, and end tokens. Then, entity candidates are generated based on these tokens. Approximate matching strategy is then employed to score all subject candidates in the KB based on their similarity, and the best-matched subject would be generated based on scores. Experimental results show that our model outperforms the baseline BERT-CRF by a margin of 3.3% accuracy on the SimpleQuestions dataset and a margin of 11.0% accuracy on the WebQuestionsSP dataset. As future work, studies will be conducted to extend our model to multilingual KBQA tasks.

Author Contributions: Conceptualization, Z.W. and X.X.; methodology, Z.W.; software, Z.W., X.L. and H.L.; validation, Z.W. and X.X.; data curation, Z.W.; writing—original draft preparation, Z.W.; writing—review and editing, Z.W., X.X., X.W. and D.H.; visualization, Z.W.; supervision, X.X., X.W. and D.H.; funding acquisition, X.W. and D.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of China under Grant No. U21A20491, No. U1936109, No. U1908214.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

KBQA	Knowledge Base Question Answering
SR	Subject Recognition
NER	Named−Entity Recognition
KB	Knowledge Base
INNM	Improved Nested NER Model
NLP	Natural Language Processing
RE	Relation Extraction
GP	GlobalPointer
EGP	Efficient GlobalPointer
SQ	SimpleQuestions
WQSP	WebQuestionsSP
TL	Token Labeling
ECG	Entity Candidate Generation
AM	Approximate Matching

References

1. Dai, Z.; Li, L.; Xu, W. CFO: Conditional Focused Neural Question Answering with Large-scale Knowledge Bases. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; pp. 800–810.
2. Liang, P.; Jordan, M.I.; Klein, D. Learning dependency-based compositional semantics. *Comput. Linguist.* **2013**, *39*, 389–446. [CrossRef]
3. Bollacker, K.; Evans, C.; Paritosh, P.; Sturge, T.; Taylor, J. Freebase: A collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, Vancouver, BC, Canada, 10–12 June 2008; pp. 1247–1250.
4. Yao, X.; Durme, B.V. Information extraction over structured data: Question answering with freebase. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 22–27 June 2014; pp. 956–966.
5. Wang, R.; Ling, Z.; Hu, Y. Knowledge base question answering with attentive pooling for question representation. *IEEE Access* **2019**, *7*, 46773–46784. [CrossRef]
6. Qu, Y.; Liu, J.; Kang, L.; Shi, Q.; Ye, D. Question answering over freebase via attentive rnn with similarity matrix based cnn. *arXiv* **2018**, arXiv:1804.03317.
7. Zhao, W.; Chung, T.; Goyal, A.; Metallinou, A. Simple question answering with subgraph ranking and joint-scoring. *arXiv* **2019**, arXiv:1904.04049.
8. Jin, H.; Luo, Y.; Gao, C.; Tang, X.; Yuan, P. Comqa: Question answering over knowledge base via semantic matching. *IEEE Access* **2019**, *7*, 75235–75246. [CrossRef]
9. Wei, M.; Zhang, Y. Natural answer generation with attention over instances. *IEEE Access* **2019**, *7*, 61008–61017. [CrossRef]
10. Devlin, J.; Chang, M.W.; Lee, K.; 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, Minneapolis, MN, USA, 2–7 June 2019; Association for Computational Linguistics: Minneapolis, MN, USA, 2019; pp. 4171–4186.
11. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *arXiv* **2019**, arXiv:1909.11942.
12. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In Proceedings of the 2019 Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 5754–5764.
13. Clark, K.; Luong, M.T.; Le, Q.V.; Manning, C.D. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
14. Martinc, M.; Škrlj, B.; Pollak, S. TNT-KID: Transformer-based neural tagger for keyword identification. *Nat. Lang. Eng.* **2022**, *28*, 409–448. [CrossRef]
15. Blšták, M.; Rozinajová, V. Automatic question generation based on sentence structure analysis using machine learning approach. *Nat. Lang. Eng.* **2022**, *28*, 487–517. [CrossRef]
16. Wysocki, O.; Zhou, Z.; O'Regan, P.; Ferreira, D.; Wysocka, M.; Landers, D.; Freitas, A. Transformers and the Representation of Biomedical Background Knowledge. *Comput. Linguist.* **2023**, *49*, 73–115. [CrossRef]
17. Laskar, M.T.R.; Hoque, E.; Huang, J.X. Domain Adaptation with pre−trained Transformers for Query-Focused Abstractive Text Summarization. *Comput. Linguist.* **2022**, *48*, 279–320. [CrossRef]

18. Gangwar, A.; Jain, S.; Sourav, S.; Modi, A. Counts@IITK at SemEval-2021 Task 8: SciBERT Based Entity And Semantic Relation Extraction For Scientific Data. In Proceedings of the 15th International Workshop on Semantic Evaluation, Bangkok, Thailand, 5–6 August 2021; pp. 1232–1238.
19. Luo, D.; Su, J.; Yu, S. A BERT−based Approach with Relation-aware Attention for Knowledge Base Question Answering. In Proceedings of the International Joint Conference on Neural Networks, Glasgow, UK, 19–24 July 2020; pp. 1–8.
20. Zhu, W. AutoRC: Improving BERT Based Relation Classification Models via Architecture Search. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop, Virtual Event, 1–6 August 2021; pp. 33–43.
21. Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S.R. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Proceedings of the 7th International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
22. Zhang, J.; Zhang, X.; Yu, J.; Tang, J.; Tang, J.; Li, C.; Chen, H. Subgraph Retrieval Enhanced Model for Multi-hop Knowledge Base Question Answering. *arXiv* **2022**, arXiv:2202.13296.
23. Gu, Y.; Su, Y. ArcaneQA: Dynamic Program Induction and Contextualized Encoding for Knowledge Base Question Answering. *arXiv* **2022**, arXiv:2204.08109.
24. Shi, J.; Cao, S.; Hou, L.; Li, J.; Zhang, H. TransferNet: An Effective and Transparent Framework for Multi-hop Question Answering over Relation Graph. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Virtual Event, 7–11 November 2021; pp. 4149–4158.
25. Su, W.; Zhu, X.; Cao, Y.; Li, B.; Lu, L.; Wei, F.; Dai, J. VL-BERT: Pre-training of Generic Visual-Linguistic Representations. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 30 April 2020.
26. Tan, H.; Bansal, M. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Hong Kong, China, 3–7 November 2019; pp. 5100–5111.
27. Sun, C.; Myers, A.; Vondrick, C.; Murphy, K.; Schmid, C. VideoBERT: A Joint Model for Video and Language Representation Learning. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7463–7472.
28. Lei, J.; Li, L.; Zhou, L.; Gan, Z.; Berg, T.L.; Bansal, M.; Liu, J. Less Is More: ClipBERT for Video-and-Language Learning via Sparse Sampling. In Proceedings of the 2021 IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 7331–7341.
29. Schneider, S.; Baevski, A.; Collobert, R.; Auli, M. wav2vec: Unsupervised Pre-Training for Speech Recognition. In Proceedings of the 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15–19 September 2019; pp. 3465–3469.
30. Chuang, Y.S.; Liu, C.L.; Lee, H.Y.; Lee, L.S. SpeechBERT: An Audio-and-text Jointly Learned Language Model for End-to-end Spoken Question Answering. *arXiv* **2019**, arXiv:1910.11559.
31. Yang, J.; Gupta, A.; Upadhyay, S.; He, L.; Goel, R.; Paul, S. TableFormer: Robust Transformer Modeling for Table-Text Encoding. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; pp. 528–537.
32. Su, H.; Shi, W.; Shen, X.; Xiao, Z.; Ji, T.; Fang, J.; Zhou, J. RoCBert: Robust Chinese Bert with Multimodal Contrastive Pretraining. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; pp. 921–931.
33. Liu, S.; Li, K.; Li, Z. A Robustly Optimized BMRC for Aspect Sentiment Triplet Extraction. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Seattle, WA, USA, 10–15 July 2022; pp. 272–278.
34. Zhu, D.; Hedderich, M.A.; Zhai, F.; Adelani, D.I.; Klakow, D. Is BERT Robust to Label Noise? A Study on Learning with Noisy Labels in Text Classification. In Proceedings of the Third Workshop on Insights from Negative Results in NLP, Dublin, Ireland, 26–27 May 2022; pp. 62–67.
35. Ye, F.; Feng, Y.; Yilmaz, E. ASSIST: Towards Label Noise-Robust Dialogue State Tracking. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, 22–27 May 2022; pp. 2719–2731.
36. Nguyen, H.; Khatwani, D. Robust Product Classification with Instance-Dependent Noise. In Proceedings of the Fifth Workshop on e-Commerce and NLP, Dublin, Ireland, 26 May 2022; pp. 171–180.
37. Su, J.; Murtadha, A.; Pan, S.; Hou, J.; Sun, J.; Huang, W.; Wen, B.; Liu, Y. Global Pointer: Novel Efficient Span-based Approach for Named Entity Recognition. *arXiv* **2022**, arXiv:2208.03054.
38. Su, J. Efficient GlobalPointer: Less Parameters, More Effectiveness. 2022. Available online: https://kexue.fm/archives/8877 (accessed on 10 June 2023).
39. Levenshtein, V. Binary codes capable of correcting deletions, insertions and reversals. *Dokl. Akad. Nauk SSSR* **1966**, *163*, 845–848.
40. Bordes, A.; Usunier, N.; Chopra, S.; Weston, J. Large-scale simple question answering with memory networks. *arXiv* **2015**, arXiv:1506.02075.

41. Yih, W.T.; Richardson, M.; Meek, C.; Chang, M.W.; Suh, J. The Value of Semantic Parse Labeling for Knowledge Base Question Answering. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; pp. 201–206.
42. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2015** arXiv:1412.6980.
43. Petrochuk, M.; Zettlemoyer, L. SimpleQuestions Nearly Solved: A New Upperbound and Baseline Approach. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 554–558.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.