information

Article

Deep Learning with Word Embedding Improves Kazakh Named-Entity Recognition

Gulizada Haisa [1,2,3] and Gulila Altenbek [1,2,3,*]

[1] College of Information Science and Engineering, Xinjiang University, Urumqi 830017, China; gulzada@stu.xju.edu.cn
[2] The Base of Kazakh and Kirghiz Language of National Language Resource Monitoring and Research Center on Minority Languages, Urumqi 830017, China
[3] Xinjiang Laboratory of Multi-Language Information Technology, Urumqi 830017, China
* Correspondence: gla@xju.edu.cn

Abstract: Named-entity recognition (NER) is a preliminary step for several text extraction tasks. In this work, we try to recognize Kazakh named entities by introducing a hybrid neural network model that leverages word semantics with multidimensional features and attention mechanisms. There are two major challenges: First, Kazakh is an agglutinative and morphologically rich language that presents a challenge for NER due to data sparsity. The other is that Kazakh named entities have unclear boundaries, polysemy, and nesting. A common strategy to handle data sparsity is to apply subword segmentation. Thus, we combined the semantics of words and stems by stemming from the Kazakh morphological analysis system. Additionally, we constructed a graph structure of entities, with words, entities, and entity categories as nodes and inclusion relations as edges, and updated nodes using a gated graph neural network (GGNN) with an attention mechanism. Finally, through the conditional random field (CRF), we extracted the final results. Experimental results show that our method consistently outperforms all previous methods by 88.04% in terms of F1 scores.

Keywords: stem embedding; gazetteer; attention; Kazakh; NER

Citation: Haisa, G.; Altenbek, G. Deep Learning with Word Embedding Improves Kazakh Named-Entity Recognition. *Information* **2022**, *13*, 180. https://doi.org/10.3390/info13040180

Academic Editor: Miltiadis D. Lytras

Received: 2 March 2022
Accepted: 28 March 2022
Published: 2 April 2022

1. Introduction

Kazakh is a typical agglutinative and morphologically rich language, and is one of the low-resource languages [1,2]. It is the official language of Kazakhstan, and is also widely spoken in China's Xinjiang and Gansu provinces. By adding different affixes, a root can have various word forms, and due to the highly inflected words in Kazakh, well-known issues of data sparsity arise.

Named-entity recognition (NER) is an essential primary task, and it has been widely used in various NLP (natural language processing) tasks, such as question-answering systems, dialogue systems, and sentiment analysis [3]. As a result, the performance of NER can affect the quality of a variety of downstream tasks. Kazakh is a low-resource language, unlike other languages—such as English and Chinese—which prefer to use rich annotated corpora and NLP toolkits. In addition, the first letter of the named-entity words in the text needs to be capitalized in English; this is significant to NER. However, there is no capitalization feature in Kazakh. Therefore, choosing appropriate linguistic features becomes very important for NER tasks in different languages.

The research on Kazakh NER started relatively late, and there are currently fewer open datasets and few public evaluation projects, restricting the application of Kazakh NER.

In this work, we derive a hybrid neural network for Kazakh NER, which integrates the morphological and gazetteer features. We take word-based and stem-based embedding for the model's input layer, and we believe that the combined word embedding exploits words' semantic meanings more efficiently than other approaches that only use the word (token) embedding in the original text. Furthermore, we also introduce gazetteer graph structures

to select the named entities in the sentence, and expect such structures to ignore the NER problem of unclear boundaries, polysemy, and nesting to some extent. Thirdly, we integrate the word features into the GGNN with an attention mechanism. Our contributions are as follows:

- We take the word-based and stem-based embedding pre-trained as an input, rather than only using word embedding. Additionally, we first consider the named-entity gazetteers with a graph structure for Kazakh NER.
- We design a novel WSGGA model for Kazakh NER, which integrates word–stem-based embeddings, gazetteer graph structures, GGNN, and an attention mechanism.
- We comprehensively analyze the structural characteristics of named entities in the Kazakh tourism domain, and effectively combine these specifics with neural networks. In addition, we evaluate our model on a benchmark dataset, with a considerable improvement over most state-of-the-art (SOTA) methods.

2. Related Works

NER identifies named entities to obtain structured data from unstructured text. Named entities include names of people, locations, organizations, times, and numeric expressions. To our knowledge, NER methods can be classified into dictionary-based, rule-based, machine learning, and deep learning methods. The dictionary-based and rule-based methods do not require large amounts of training data, and they are relatively easy to implement. However, they are time-consuming and labor-intensive, and require someone to write the linguistic rules manually. In addition, since the rules need to be manually formulated, they will inevitably produce subjective arbitrariness and cause uncertainty in identification. The traditional machine learning methods—such as SVM [4], HMM [5], and CRF [6]—need to use the method of feature engineering to manually define features and generate feature templates. The deep learning methods can significantly reduce the sparseness of the data by using word embedding, and can better extract the semantic features of the text. Most of the current methods include Word2vec [7], glove [8], fastText [9], Elmo [10], and BERT (bidirectional encoder representations from transformers) [11]. The word embedding is obtained using unlabeled large-scale text, expressing part of the semantic information and the contextual relationship. There are also several deep learning models, such as RNN+CRF [12], CNN+BiLSTM+CRF [13], BERT+BiLSTM+CRF [14], etc. Usually, the feature set is automatically constructed through the multilayer network structure, and the output vectors are fed to the CRF layer to jointly decode the best label sequence. LSTM-CRF models [15] leveraging word-level and character-level representations achieve state-of-the-art results in most languages. Additionally, most of the current literature can be categorized into attention model [16], pre-training model [17], graph neural network [18], and transfer learning [19] methods. In the past two years, researchers have proposed new network models combined with gazetteers [20], and their multiple features [21] have also achieved the SOTA in other language research. However, deep-learning-based methods still have several shortcomings. First, the deep learning methods require a large amount of training data to improve the accuracy of NER. Secondly, entities have unclear boundaries, multiple nesting, and other problems that are hard to analyze. Finally, the labelled corpus cannot cover all entities.

Researchers in Kazakh NER have been studying for several years, starting from the early rule-based [22] and statistics methods [23], and then using statistical methods for Kazakh phrase recognition [24,25]. Kazakhstan has made some progress in recent years, and has given Kazakh NER a systematic study by using conditional random fields [26]. The authors of [27] introduced three types of representation for morphologically complex languages, including word, root, and POS, and this model outperformed the other CRF and BiLSTM-CRF models. Recently, [28] used the SOTA model BERT+BiLSTM+CRF to research Kazakh NER. However, these works of literature are simple to use or direct stitching of off-the-shelf neural networks. In contrast, we consider both morphological features and named-entity gazetteers with a graph structure. We also designed a deep

neural network to pick out better quality sequence labeling. Furthermore, there are fewer research reports on NER in specific areas, such as tourism. Thus, this study has tremendous research significance.

3. Description of the Problem

3.1. The Complex Morphological Features of Kazakh

In model training, the extracted features directly affect the NER model. Kazakh is a derived language, and Kazakh text is composed of naturally separated words. Prefixes and suffixes can be added to a stem (or root) to construct hundreds or thousands of words. A stem is a basic vocabulary unit with practical meaning, and affixes provide semantic and grammatical functions. Therefore, Kazakh can be called morphologically complex and rich in vocabulary. After morphological analysis and processing of the Kazakh texts, we can retain meaningful and effective text features and reduce these features' complexity and dimensions.

According to the word frequency statistics on a large number of texts, we found that the most common words were "ساحاراسىندا‌عى" (on the grassland), "ساحاراسىنىان" (from the grassland), "ساحاراسى" (the grassland), and "ساحاراسىنىڭ" (their grassland); all of these words' stem (or root) is "ساحارا" (grassland), and "سى", "ىن", "دا", "عى", and "نىڭ" are all suffixes.

The study of Kazakh morphological analysis [29,30] has also produced a lot of work and applied it in practical research. The above research results of Kazakh morphology make it possible to use morphological analysis in Kazakh NER. Therefore, data sparseness can be effectively solved through morphological analysis of Kazakh tourism text.

3.2. Specific Issues in Kazakh Tourism NER

Through this study, we found that the Kazakh named entities in the tourism domain has the following characteristics:

- Some scenic spots are named too long, and the naming conventions are arbitrary. The length of most scenic areas is between 1 and 13 words. For example: "بانفاڭگۋ اۋىلدىق وسى زامانعى اۋىلشارۋاشىلىق عىلىم تەحنيكاسىنان ۇلگى كورسەتۋ باقشاسى" (Banfanggou Township Modern Agricultural Technology Demonstration Park). According to statistics, 87.84% of the entities consist of more than two words, 55.12% of which are composed of two words, and in which the last word is the same to some extent. For example: "سايرام كولى" (Sayram Lake), "قاناس كولى" (Hanas Lake), "ۇلىڭگىر كولى" (Ulungur Lake), "حۋالين باقشاسى" (Hualin Park), "حوڭشان باقشاسى" (Hongshan Park), and "اققايىڭ باقشاسى" (White Birch Forest Park).
- Some names of peoples, places, and nationalities are often nested. For example, "ىلە قازاق اۆتونوميالى وبلىسى" (Ili Kazakh Autonomous Prefecture), "گەنارال تاۋى ورمان باقشاسى" (Jiangjunshan Forest Park), or "قۇربان تۇلىم ەسكەرتكىش سارايى" (Kurban Tulum Memorial Hall).
- Multiple scenic spots may have the same name, or one scenic spot may have several names. For example, "نارات" (Narat), "ىلە نارات" (Ili Narat), "نارات ساحاراسى" (Narat Prairie), "نارات جايلاۋى" (Narat grasslands), "نارات كورىنىس رايونى" (Narat scenic area), "نارات سايا‌حات وڭىرى" (Narat tourist spot), etc., are actually one entity.
- Several entities lack clear boundaries and nest, and those types of entities are diverse, as shown in Figure 1.

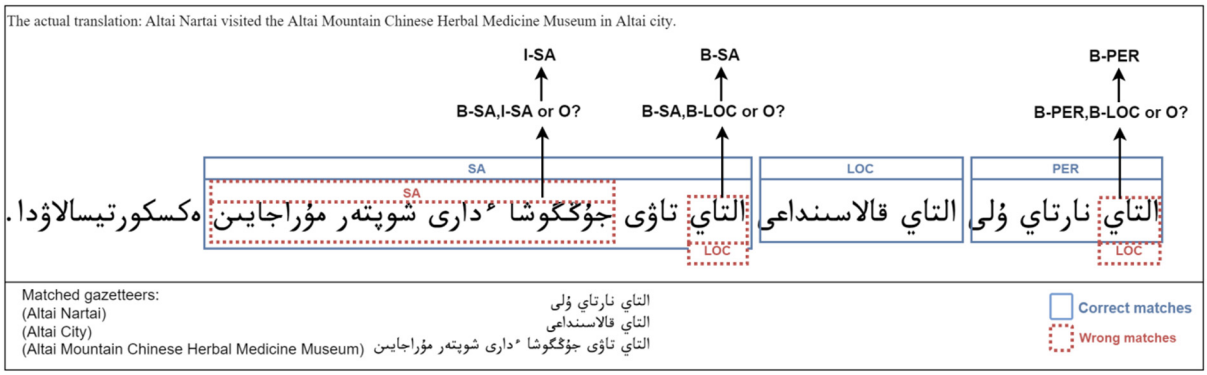

Figure 1. Example of entity matching (Kazakh can be written using both right-to-left (Arabic) and left-to-right (Latin or Cyrillic) scripts, and this work uses right-to-left).

As depicted in Figure 1, the first التايrepresents a person's name, the second التايrepresents a place name, and the third التايis a part of the scenic area. The boundaries of the words التاي نارتاي وُلى(Altai Nartai), التاي قالاسندىعى(Altai City), and التاي تاۋى جۇڭگوشا ءدارى شوپتەر مۇراجاين(Altai Mountain Chinese Herbal Medicine Museum) are the same. Therefore, making full use of the gazetteer's information would help to improve the Kazakh NER performance.

4. The Proposed Approach

4.1. Model Architecture

Our (WSGGA) models are based on Word2vec, GGNN, Attention, and CRF. The framework of the WSGGA is shown in Figure 2. The primary feature representation layer contains word–stem embedding and gazetteer graph structures. After the feature extraction layer, there is a GGNN layer and an attention layer.

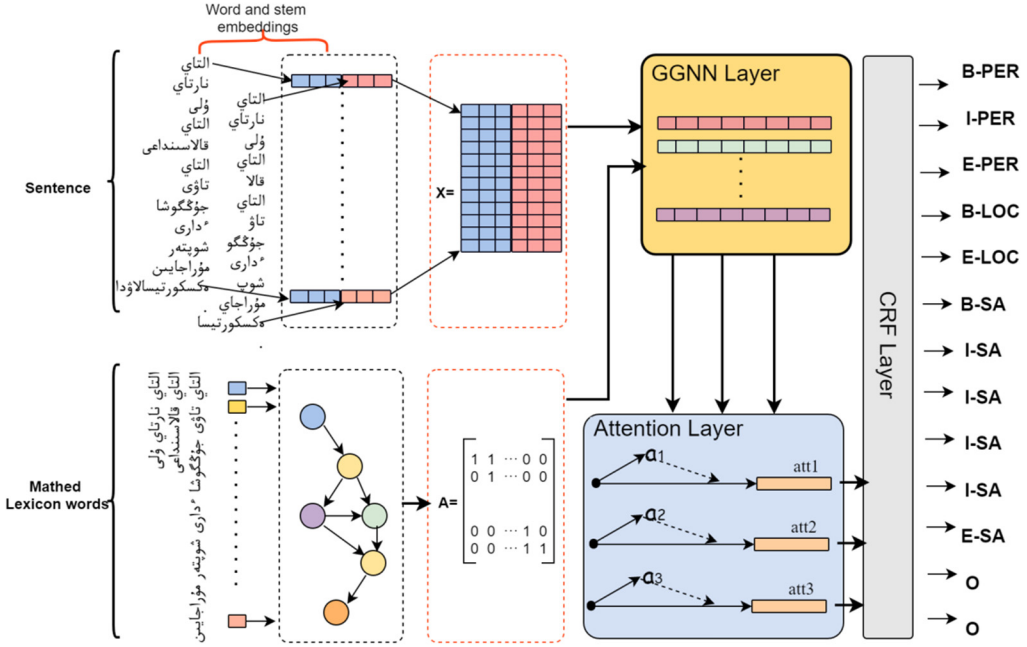

Figure 2. The framework of WSGGA for Kazakh NER.

4.2. Feature Representation Layer

There are two main tasks in the feature representation layer: First, word and stem embedding are trained on a large-scale unlabeled dataset. The word–stem embeddings are

initialized to combine embedding of the word's semantic features. The other task is to match the word in the sentence through gazetteers with a graph structure—the corresponding adjacency matrix represents the relationship between words in the original text.

4.2.1. Pre-Training Word–Stem Vectors

The Kazakh words are separated by spaces or punctuation marks; it is segmented at the morphological level, and extracts stem in order to retain meaningful and effective features. Here, we trained both the word-based and the stem-based embeddings using the Skip-Gram model of the Word2vec tool. The Skip-Gram model maximizes the following function for a given sentence $S = \{x_1, x_2, \ldots, x_M\}$:

$$F = \frac{1}{M} \sum_{m=1}^{M} \sum_{-n \le j \le n, j \ne 0} log\ p(x_{t+j}|x_t), \quad (1)$$

where n is the size of the training window, and contextually relevant words for the current word are obtained based on the window size. $w_i = [a_0, a_1, \ldots, a_d]$ represents word vectors, and d is the word vector dimension.

4.2.2. Training Word-Based and Stem-Based Embeddings

For word embedding, given a sentence of n words, $S = \{w_i, \ldots, w_n\}$, where w_i is the i-th word, with each word converted into a word-based vector $E^w(w_i)$, where E^w is the pre-trained word (token) vector table.

For stem embedding, given a sentence of n words, $S = \{c_i, \ldots, c_n\}$, where c_i is the stem of the i-th word, which is converted into a stem-based vector $E^c(c_i)$, where E^c is a pre-trained stem vector table.

Therefore, the final vector consists of two parts: the word-based vector $E^w(w_i)$ and the stem-based vector $E^c(c_i)$.

$$X_i = [E^w(w_i),\ E^c(c_i)], \quad (2)$$

4.2.3. Constructing a Graph Structure

Through gazetteer matching, we constructed the directed graph structure and obtained the corresponding adjacency matrix representing the relationship between the word and the entity in the sentence.

An adjacency matrix A can represent a graph with nodes. Given a sentence $S = \{w_i, \ldots, w_n\}$ containing n Kazakh words, each word in the sentence is taken as a node of the graph. The construction of the graph structure has two steps: First, a directed edge is added from right to left for each pair of adjacent Kazakh words. Secondly, a matching process takes places between the entities in the original sentence and the entities in the gazetteers. All comments successfully matched are added with an edge. If i and j are the start node and end node of the entity matched by the i-th word from the dictionary, respectively, then we can connect an edge between these two nodes—that is, let $A_{ij} = 1$. The multigraph in this paper is defined as $G = (V, E, L)$, where V is the node, E is the edge, and L is the set of labels. Node V is composed of word nodes—the entity's start and end word nodes. Each edge E is composed of links between two adjacent words in the sentence and the matching entity in the gazetteer. Label set L is the label of the adjacent words in the sentence and the text span that matches with the entity.

As illustrated in Figure 3, the given input sentence التاي نارتاي وُلى التاي قالاسنداعى التاي تاۋى جوُڭگوشا ءدارى شوپتەر مۇراجاين(Altai Nartai is in the Altai Mountain Chinese Herbal Medicine Museum in Altai City) consists of 11 Kazakh words, and the five gazetteers are SA1, SA2, PER, LOC1, and LOC2. We first used 11 nodes to represent a complete sentence, and each word corresponded to a node. We also operated another 10 nodes (start and end nodes for five entities) to represent the gazetteer. Next, we added edge information: First, an edge was added to each adjacent word from right to left, and then another edge was added between the words from the start to the end of each

entity matched by the gazetteer. Adjacency matrix A stored the structural information of the graph.

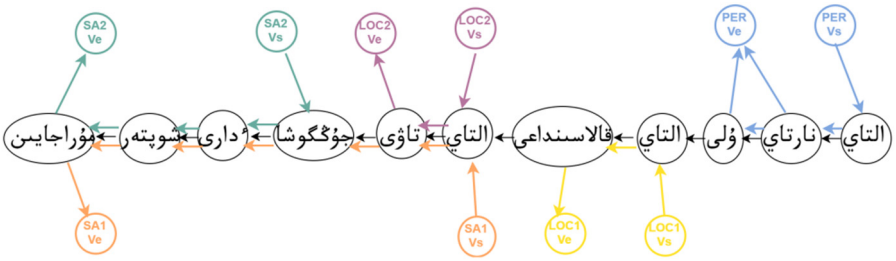

Figure 3. Named entity directed gazetteer graph.

4.3. Graph-Gated Neural Network

GGNN is a classical spatial domain message-passing model [31] based on gated recurrent units (GRUs) [32]. The authors of [33] incorporated the GRUs into a graph neural network, which unrolled the loop in a fixed number of steps and used backpropagation over time to calculate the gradient. In this work, the GGNN layer received the word–stem embeddings and adjacency matrices from the feature representation layers. Our goal was to send all of the feature representation information to the GGNN network, and then to obtain the embeddings of all nodes. The information transmission process is as follows:

$$h_i^{(0)} = [E^w(w_i)^T, E^c(c_i)^T]^T, \tag{3}$$

$$H = [h_1^{(t-1)^T}, \ldots, h_{|i|}^{(t-1)^T}], \tag{4}$$

$$a_i^{(t)} = A_{i:}^T H^T + b, \tag{5}$$

$$z_i^{(t)} = \sigma\left(W^z a_i^{(t)} + U^z h_i^{(t-1)}\right), \tag{6}$$

$$r_i^{(t)} = \sigma\left(W^r a_i^{(t)} + U^r h_i^{(t-1)}\right), \tag{7}$$

$$\hat{h}_i^{(t)} = tanh\left(W a_i^{(t)} + U\left(r_i^{(t)} \odot h_i^{(t-1)}\right)\right), \tag{8}$$

$$h_i^{(t)} = \left(1 - z_i^t\right) \odot h_i^{(t-1)} + z_i^t \odot \hat{h}_i^{(t)}, \tag{9}$$

where $h_i^{(0)}$ is obtained from the feature representation layer, $A_{i:}$ indicates the row vector corresponding to the node i is selected from the adjacency matrix, and W and U are parameters. Equation (3) creates the state matrix H at step $t-1$. Equations (6) and (7) show the general GRU update information process. Equation (8) combines the information of neighboring nodes with the current hidden state of the node, and calculates the new hidden state at step t. Finally, we get the final state $h_i^{(t)}$ of node i.

4.4. Attention Mechanism

The attention mechanism can effectively improve the model's ability to recognize keywords, and the transformer-based multi-head attention has been widely used in many NLP tasks. Multi-head attention is a combination of multiple self-attention structures. In order to highlight the importance of key information, we applied an attention mechanism to selectively assign higher weights to key content after the GGNN. After obtaining the output of the previous layer, we performed multi-head attention training on the current word, and calculated the similarity of Q (query), K (key), and V (value) to obtain the weight.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{10}$$

where d_k stands for the dimension of the K matrix, where the attention-scoring function is calculated using the dot product similarity. The original multi-head attention was defined as follows:

$$MultiHead(Q, K, V) = Concat(head_1, head_2, \ldots, head_n)W^O, \tag{11}$$

$$head_i = Attention\left(HW_i^Q,\ HW_i^K,\ HW_i^V\right), \tag{12}$$

where head_i denotes the self attention unit and note that n is the number of attention heads. H is the output of GGNN layer; W_i^Q, W_i^K, W_i^V and W^O are parameter matrix.

4.5. CRF Layer

Finally, the CRF layer obtains a global semantic representation and obtains a globally optimal sequence according to the adjacent labels. If the sentence sequence is $X = \{x_1, x_2, \ldots, x_n\}$, the predicted label sequence is $Y = \{y_1, y_2, \ldots, y_n\}$. We define the probability distribution $p(X, Y)$ as follows:

$$p(y/x) = \frac{\sum_{t=1}^{T} e^{f(y_{t-1}, y_t,\ x)}}{\sum_{y'}^{Y(x)} \sum_{t=1}^{T} e^{f(y'_{t-1}, y'_t, x)}}, \tag{13}$$

$$Y^* = argmax\ P(y/x), \tag{14}$$

where $Y(x)$ represents all possible annotation sequences. $f(y_{t-1}, y_t,\ x)$ calculate the probability of transition from y_{t-1} to y_t. y is the label with the maximum conditional probability.

5. Experiment

5.1. Dataset

The NER corpus and gazetteers were obtained from the Kazakh NER task by researchers who are in the base of the Kazakh and Kirghiz languages at national language resource monitoring and research Center on monitory languages. Moreover, researchers who are native Kazakh language speakers, including this article's author, constructed the datasets. There were 6000 sentences, including 15,006 named entities, and 7 kinds of named entities: person (PER), location (LOC), scenic area (SA), specialty (SC), organization (ORG), culture (CU), and nationality (NA, name of ethnic groups). The dataset was divided into the training set, validation set, and test set at an 8:1:1 ratio. Furthermore, we also used the tourism gazetteers—namely, SA, LOC, PER, SC, ORG, NA, and CU. Table 1 shows the specific distribution.

Table 1. Detailed statistics of the tourism dataset.

Data	Sentence	SA	LOC	ORG	PER	SC	NA	CU
Train	4800	4456	5064	510	728	914	241	629
Dev	600	360	475	73	64	70	35	101
Test	600	486	534	62	47	89	22	46
Gazetteers' size	–	2000	2350	1650	5000	2300	56	1630

In this task, one sentence may contain two or more entities. Thus, we used an annotation specification named BIOES (B-begin, I-inside, O-outside, E-end, S-single).

5.2. Parameter Settings

The experimental parameters of the WSGGA model are listed in Table 2.

Table 2. Experimental parameter setting.

Parameters	Parameter Size
Training batch size	10
Word vector dimension	200
Stem vector dimension	200
Word2vec window size	8
Word2vec min_count	3
GGNN_hidden_size	200
Dropout	0.5
Learning rate	0.001
Model optimizer	SGD
Attention heads	8
Epochs	50

5.3. Experimental Results and Analysis

5.3.1. Comparison of Word Embeddings

For word embeddings, we used the Python program to crawl five different Kazakh websites (China People's Daily Net (http://kazakh.people.com.cn (accessed on 2 March 2022)), Tianshan Net (http://kazakh.ts.cn (accessed on 2 March 2022)), Kunlun Net (http://kazak.xjkunlun.gov.cn (accessed on 2 March 2022)), Yili News Net (http://kazakh.ylxw.com.cn (accessed on 2 March 2022)), and Altay News Net (http://kazakh.altxw.com (accessed on 2 March 2022))) and obtain unlabeled texts with 1.89 million words. In this work, we used the Word2vec model to generate word embeddings and stem embeddings for Kazakh word sequences and stem sequences, respectively.

This paper made use of a stem segmentation system developed based on the research of [29] in order to extract stems from a Kazakh corpus. This method can convert them into stem sequences. After the stem segmentation, the stem-based vocabulary dropped sharply to 60% of the word vocabulary. Table 3 shows the proportions of words and stems in different sizes. Therefore, stem-based segmentation can reduce feature dimensions.

Table 3. Reduction in feature space dimension by stemming.

Text Size	Original Words	Words (Remove Duplicates)	Stems (Remove Duplicates)	Stem–Word Ratio
1.3 MB	108,589	20,137	12,627	62.71%
2.5 MB	217,815	34,615	22,764	65.76%
6.1 MB	617,396	57,112	36,694	64.25%
16.3 MB	1,737,125	106,885	71,787	67.16%

In training, Word2vec can generate each word vector according to its context, and we used the cosine distance to determine how similar the words were to another word or how similar the stems were to other stems. If the calculated cosine value is large, the semantics are closer; otherwise, the opposite is true. Tables 4 and 5 show the four words and four stems in Kazakh tourism vocabulary.

Tables 4 and 5 demonstrate the closest semantic vocabularies by calculating the cosine distance. The experimental results indicate that the trained similar vocabularies in the word vectors are generated by adding different affixes, and the semantics and word forms are relatively similar. In comparison, stem vectors can also obtain the same semantic vocabulary but different word forms, to some extent. Therefore, stem-based word vector training can effectively reduce the repetition rate and feature dimensions.

Table 4. Word vector semantic similarity.

Words	**1**	**2**	**3**	**4**	**5**
جۇڭگو	جۇڭگودا	جۇڭگوشا	جۇڭگونىڭ	جۇڭگوغا	جۇڭگودان
(China)	(In China)	(Chinese style)	(China's)	(To China)	(In China)
Cos distance	0.9706	0.9606	0.9549	0.9400	0.9391
قىستاق	قىستاقتا	قىستاققا	قىستاقتان	قىستاقتار	قىستاقى
(Village)	(In village)	(To the village)	(From village)	(Village)	(Their village)
Cos distance	09657	0.9616	0.9502	0.9433	0.9399
ساياھات	ساياھاتى	ساياھاتتا	ساياھاتشى	ساياھاتقا	ساياھاتتاي
(Tourism)	(Tourism)	(Travel)	(Traveler)	(On travel)	(Trip)
Cos distance	0.9022	0.8536	0.82.69	0.8161	0.8133
ساۋدا	ساۋدادا	ساۋدانى	ساۋداگەر	ساۋدانى	ساۋداغا
(Trade)	(In the trade)	(Trade on)	(Trader)	(Business)	(To business)
Cos distance	0.9701	0.9686	0.9499	0.9318	0.9221

Table 5. Stem vector semantic similarity.

Stems	**1**	**2**	**3**	**4**	**5**
جۇڭگو	جۇڭخۇا	امەريكا	جاپونيا	روسسيا	اۆسترليا
(China)	(In China)	(America)	(Japan)	(Russia)	(Australia)
Cos distance	0.9352	0.9115	0.9070	0.9066	0.9038
قىستاق	اۋىل	قىستاۋ	قالاشىق	اۋدان	ايماق
(Village)	(village)	(Rural)	(City)	(Town)	(Area)
Cos distance	0.9561	0.9283	0.8805	0.8632	0.8450
ساياھات	جولاۋشى	ساۋىق	سالتسانا	جول	كورىنىس
(Tourism)	(Traveler)	(Pleasure)	(Culture)	(Journey)	(Scenic)
Cos distance	0.8272	0.8265	0.8112	0.8049	0.7986
ساۋدا	ساتتىق	ايناليم	تاۋار	كاسىپ	باعا
(Trade)	(Purchase)	(Trade)	(Goods)	(Business)	(Price)
Cos distance	0.9666	0.9453	0.9224	0.9180	0.9076

In Table 6, we compare word embedding performance between word vector models. In contrast, fastText-generated embedding has a better performance than Glove. However, Word2vec-generated embeddings obtained more effective results for our task. This comparative experiment uses the WSGGA model as the base model, and uses Precision (P), Recall (R), and F1 score (F1) as evaluation indices.

Table 6. Comparison of three types of word embedding.

Word Embedding	**P (%)**	**R (%)**	**F1 (%)**
Glove	87.27	87.10	87.18
FastText	87.62	87.74	87.67
Word2vec	87.95	88.14	88.04

5.3.2. Experiment for Kazakh NER in Different Model

To verify the effectiveness of the WSGGA model for the Kazakh NER, we selected baselines as follows:

HMM [25] uses the HMM algorithm; it is the oldest method in the Kazakh sequence-labeling tasks.

CRF [26] uses the CRF algorithm; it is one of the most commonly used and well-known methods in Kazakh NER. In this experiment, we used a CRF model with word features.

BiLSTM+CRF [27] combines the traditional machine algorithm CRF and deep learning model BI-LSTM; it is also a classic model in NLP tasks. In this experiment, we used the BiLSTM+CRF architecture with word features.

BERT+BiLSTM+CRF [28] is the SOTA architecture in the NER system in many languages [11,14]. In this experiment, the "Bert-base-multilingual-cased" model that covers 104 languages was used for initialization.

In Table 7, we compare NER performance between challenging Kazakh NER models, ranked by F1 score.

Table 7. Comparative experimental results of different models.

Models	P (%)	R (%)	F1 (%)
HMM	64.01	60.99	62.46
CRF	76.88	62.23	68.78
BiLSTM+CRF	79.41	74.27	76.75
BERT+BiLSTM+CRF	84.91	83.19	84.04
WSGGA(ours)	87.95	88.14	88.04

As shown in Table 7, the WSGGA model achieves the best performance between whole baseline systems. In contrast, the CRF model achieves better improvement than HMM, because CRF solves partial label bias, while HMM does not adequately capture contextual semantic information. The performance of BiLSTM+CRF is significantly better than that of HMM and CRF, because the BiLSTM can extract long-distance semantic features. Because of the great success of BERT, the strong baseline BERT+BiLSTM+CRF obtained more effective results than previous baseline systems. However, in the tourism domain NER, the gazetteers are of obvious importance to improving NER abilities, so the WSGGA model is better than the BERT model. Furthermore, the WSGGA model fused with stem-embedding features captures the rich semantic features. Thus, our method consistently outperforms other baselines in Kazakh NER.

5.4. Ablation Experiment

To explore whether different features affect the WSGGA model, we designed a further comparative experiment. In this investigation, we removed some parts and performed NER.

Table 8 shows that the model achieves the best result when we use the complete model. However, when we remove the stem embedding, attention mechanism, and gazetteers, the model's F1 score decreases by 0.92%, 1.97%, and 3.92%, respectively. We can also see that the gazetteer's performance is better than that of the other features. Still, without these features, it is challenging for the model to identify entities, and it cannot fully express the named entities in the sentence. Thus, our proposed model gains more effective results with stem embedding, NER gazetteers, and an attention mechanism.

Table 8. The influence of different features.

Different Features	P (%)	R (%)	F1 (%)
WSGGA(complete model)	87.95	88.14	88.04
W/O (stem embedding)	86.70	87.55	87.12
W/O attention	85.26	86.89	86.07
W/O gazetteers	84.33	83.92	84.12

5.5. Case Study

To show the effect of NER in solving the actual texts, Table 9 gives the two case studies. In the first case, there is a scenic area entity "ئۈرمچى قالالىق علىمتەخنىكا سارايى" (Urumqi Science and Technology Museum) with nested "ئۈرمچى" (Urumqi), "ئۈرمچى قالالىق" (Urumqi

city), and "عىلمتەحنيكا سارايى" (Science and Technology Museum). Without gazetteer features, the model cannot integrate the self-matched named entity "Urumqi Science and Technology Museum" into the "قالالىق" and "عىلم". In the second case study, there is a scenic area entity "تاڭبالى ساحاراسىندااعى" (Tangbale Grassland). Without stem embedding, the model cannot integrate the nearest contextual entity word "تاڭبالى", and "ساحاراسىندااعى", so the predicted labels of the phrase "ساحاراسىندااعى" is "O". The reason for this is that the commonly used word forms of "ساحاراسىندااعى" is "ساحارا", and after adding the suffixes—such as "ن", "سى", "دا", and "عى"—the original word form changes, resulting in data sparseness problems.

Table 9. KzToNe recognition comparison example.

Language and Model	Text
Case 1: In Kazakh	ءۇرىمجى قالالىق عىلىم-تەحنيكا سارايى 8-ايدىڭ 10-كۇنى رسمي اشىلادى.
In English	Urumqi Science and Technology Museum officially opened on 10 August 2011
Common entity	ءۇرىمجى(LOC); ءۇرىمجى قالالىق (LOC); عىلىم-تەحنيكا سارايى(SA); عىلىم-تەحنيكا سارايى(SA)
Gold label	ءۇرىمجى(B-SA)قالالىق(I-SA) عىلىم(I-SA)-(I-SA)تەحنيكا(I-SA) سارايى(E-SA) 8(O)-(O)ايدىڭ(O) 10(O)-(O)كۇنى(O) رسمي(O) اشىلادى(O).(O)
w/o entity gazetteers	ءۇرىمجى(B-LOC) قالالىق(I-LOC) عىلىم(B-SA) -(I-SA) تەحنيكا(I-SA) سارايى(E-SA) 8(O)-(O)ايدىڭ(O) 10(O)-(O)كۇنى(O) رسمي(O) اشىلادى(O).(O)
with entity gazetteers	ءۇرىمجى(B-SA) قالالىق(I-SA) عىلىم(I-SA)-(I-SA)تەحنيكا(I-SA) سارايى(E-SA) 8(O)-(O)ايدىڭ(O) 10(O)-(O)كۇنى(O) رسمي(O) اشىلادى(O).(O)
Case 2: In Kazakh	نىلقى اۋداننىنداعى تاڭبالى ساحاراسىندااعى گۇلدەر شەشەك اتتى.
In English	Tangbale Grassland in Nelhe County is blooming with flowers.
Common entity	نىلقى اۋداننىنداعى(LOC) ; تاڭبالى ساحاراسىندااعى(SA)
Gold label	نىلقى(B-LOC) اۋداننىنداعى(E-LOC) تاڭبالى(B-SA) ساحاراسىندااعى(E-SA) گۇلدەر(O) شەشەك(O) اتتى(O).(O)
w/o stem embedding	نىلقى(B-LOC) اۋداننىنداعى(E-LOC) تاڭبالى(S-SA) ساحاراسىندااعى(O) گۇلدەر(O) شەشەك(O) اتتى(O).(O)
with stem embedding	نىلقى(B-LOC) اۋداننىنداعى(E-LOC) تاڭبالى(B-SA) ساحاراسىندااعى(E-SA) گۇلدەر(O) شەشەك(O) اتتى(O).(O)

6. Conclusions

In this paper, we introduced a hybrid network with multidimensional features for NER in Kazakh that obtains word embedding from original data through a parallel feature representation layer, with word–stem embeddings and a gazetteer graph structure. Then, we used the GGNN with an attention mechanism to learn semantic information thoroughly. Experimental results show that the WSGGA model significantly outperformed previous NER methods in the Kazakh language. Experimental results show that it can effectively solve NER problems, such as entity nesting, data sparsity, and polysemy. Although our model achieved good results, there are some limitations. For example, the current dataset was based on a limited field, and generalization needs improvement. In the future, we will use this method for other NLP tasks, such as information retrieval, text classification, and question answering. Furthermore, for the issue of insufficient data in other areas, we will use transfer learning methods to improve the model's generalization ability.

Author Contributions: G.H. conceived, designed, and performed the experiments; G.A. contributed materials; G.H. wrote the paper; G.A. revised the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 62062062).

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

References

1. Kuwanto, G.; Akyürek, A.F.; Tourni, I.C.; Li, S.; Jones, A.G.; Wijaya, D. Low-Resource Machine Translation for Low-Resource Languages: Leveraging Comparable Data, Code-Switching and Compute Resources. *Arxiv* **2021**, arXiv:2103.13272.
2. Li, X.; Li, Z.; Sheng, J.; Slamu, W. *Low-Resource Text Classification via Cross-Lingual Language Model Fine-Tuning*; Springer: Cham, Switzerland, 2020.
3. Kumar, A.; Irsoy, O.; Ondruska, P.; Iyyer, M.; Bradbury, J.; Gulrajani, I.; Zhong, V.; Paulus, R.; Socher, R. Ask me anything: Dynamic memory networks for natural language processing. In Proceedings of the 33rd International Conference on Machine Learning, PMLR, New York, NY, USA, 19–24 June 2016.
4. Ekbal, A.; Bandyopadhyay, S. Named entity recognition using support vector machine: A language independent approach. *Int. J. Electr. Comput. Syst. Eng.* **2010**, *4*, 155–170.
5. Saito, K.; Nagata, M. Multi-language named-entity recognition system based on HMM. In Proceedings of the ACL 2003 Workshop on Multilingual and Mixed-Language Named Entity Recognition, Sapporo, Japan, 12 July 2003; pp. 41–48.
6. Lafferty, J.; McCallum, A.; Pereira, F.C. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of the 18th International Conference on Machine Learning 2001 (ICML 2001), Oslo, Norway, 22–25 August 2001.
7. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *Arxiv* **2013**, arXiv:1301.3781.
8. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
9. Bojanowski, P.; Grave, E.; Joulin, A.; Mikolov, T. Enriching Word Vectors with Subword Information. *Trans. Assoc. Comput. Linguist.* **2017**, *5*, 135–146. [CrossRef]
10. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. *arXiv* **2018**, arXiv:1802.05365.
11. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
12. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural architectures for named entity recognition. *arXiv* **2016**, arXiv:1603.01360.
13. Chiu, J.P.C.; Nichols, E. Named entity recognition with bidirectional LSTM-CNNs. *Trans. Assoc. Comput. Linguist.* **2016**, *4*, 357–370. [CrossRef]
14. Straková, J.; Straka, M.; Hajič, J. Neural architectures for nested NER through linearization. *arXiv* **2019**, arXiv:1908.06926.
15. Liu, L.; Shang, J.; Ren, X.; Xu, F.; Gui, H.; Peng, J.; Han, J. Empower sequence labeling with task-aware neural language model. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
16. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, .; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762.
17. Lample, G.; Conneau, A. Cross-lingual language model pretraining. *arXiv* **2019**, arXiv:1901.07291.
18. Wang, S.; Chen, Z.; Ni, J.; Yu, X.; Li, Z.; Chen, H.; Yu, P.S. Adversarial defense framework for graph neural network. *arXiv* **2019**, arXiv:1905.03679.
19. Peng, D.L.; Wang, Y.R.; Liu, C.; Chen, Z. TL-NER: A transfer learning model for Chinese named entity recognition. *Inf. Syst. Front.* **2019**, *22*, 1291–1304. [CrossRef]
20. Ding, R.; Xie, P.; Zhang, X.; Lu, W.; Li, L.; Si, L. A neural multi-digraph model for Chinese NER with gazetteers. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 1462–1467.
21. Zhang, J.; Hao, K.; Tang, X.; Cai, X.; Xiao, Y.; Wang, T. A multi-feature fusion model for Chinese relation extraction with entity sense. *Knowl.-Based Syst.* **2020**, *206*, 106348. [CrossRef]
22. Altenbek, G.; Abilhayer, D.; Niyazbek, M. A Study of Word Tagging Corpus for the Modern Kazakh Language. *J. Xinjiang Univ. (Nat. Sci. Ed.)* **2009**, *4*. Available online: https://xueshu.baidu.com/usercenter/paper/show?paperid=a3871a5f9467444c61107b80ce2cf989&site=xueshu_se (accessed on 1 March 2022). (In Chinese)
23. Feng, J. *Research on Kazakh Entity Name Recognition Method Based on N-gram Model*; Xinjiang University: Ürümqi, China, 2010.
24. Altenbek, G.; Wang, X.; Haisha, G. Identification of basic phrases for kazakh language using maximum entropy model. In Proceedings of the COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, 23–29 August 2014; pp. 1007–1014.
25. Wu, H.; Altenbek, G. Improved Joint Kazakh POS Tagging and Chunking. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*; Springer: Cham, Switzerland, 2016; pp. 114–124.
26. Gulmira, T.; Alymzhan, T.; Zheng, X. Named Entity Recognition for Kazakh Using Conditional Random Fields. In Proceedings of the 4-th International Conference on Computer Processing of Turkic Languages "TurkLang 2016", Bishkek, Kyrgyzstan, 23–25 August 2016.
27. Tolegen, G.; Toleu, A.; Mamyrbayev, O.; Mussabayev, R. Neural named entity recognition for Kazakh. *arXiv* **2020**, arXiv:2007.13626.
28. Akhmed-Zaki, D.; Mansurova, M.; Barakhnin, V.; Kubis, M.; Chikibayeva, D.; Kyrgyzbayeva, M. Development of Kazakh Named Entity Recognition Models. In *Computational Collective Intelligence*; International Conference on Computational Collective Intelligence; Springer: Cham, Switzerland, 2020; pp. 697–708.

29. Abduhaier, D.; Altenbek, G. Research and implementation of Kazakh lexical analyzer. *Comput. Eng. Appl.* **2008**, *44*, 4.
30. Altenbek, G.; Wang, X.L. Kazakh segmentation system of inflectional affixes. In Proceedings of the CIPS-SIGHAN Joint Conference on Chinese Language Processing, Beijing, China, 28–29 August 2010.
31. Gilmer, J.; Schoenholz, S.S.; Riley, P.F.; Vinyals, O.; Dahl, G.E. Message passing neural networks. In *Machine Learning Meets Quantum Physics*; Springer: Cham, Switzerland, 2020; pp. 199–214.
32. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv* **2014**, arXiv:1409.1259.
33. Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. Gated Graph Sequence Neural Networks. *arXiv* **2015**, arXiv:1511.05493.