An Element-wise Visual-enhanced BiLSTM-CRF Model for Location Name Recognition

Takuya Komada
Department of Computer Science
University of Tsukuba
komada@mibel.cs.tsukuba.ac.jp

Takashi Inui
Department of Computer Science
University of Tsukuba
inui@cs.tsukuba.ac.jp

Abstract

In recent years, previous studies have used visual information in named entity recognition (NER) for social media posts with attached images. However, these methods can only be applied to documents with attached images. In this paper, we propose a NER method that can use element-wise visual information for any documents by using image data corresponding to each word in the document. The proposed method obtains element-wise image data using an image retrieval engine, to be used as extra features in the neural NER model. Experimental results on the standard Japanese NER dataset show that the proposed method achieves a higher F1 value (89.67%) than a baseline method in location name recognition, demonstrating the effectiveness of using element-wise visual information.

1 Introduction

Since the 1990s, information extraction, in which computers are used to extract structured data from unstructured documents, has been extensively studied (Cowie and Lehnert, 1996; Grishman and Sundheim, 1996). Among the entities to be extracted, location information (where) is one of the essential components (5W1H) of event information to be extracted, and the process has evolved to include various tasks, such as location name disambiguation and mapping of location names to real-world geographic locations (Weissenbacher et al., 2019).

Location name recognition has been typically conducted as a named entity recognition (NER) task (Li et al., 2018). In this field, deep learning models using visual information have been actively studied in recent years, especially in the extraction of named entities (NEs) from posts in social networking services (SNSs) such as Twitter and SnapChat (Lu et al., 2018; Moon et al., 2019; Zhang et al., 2018). These methods use images attached to a post as multimodal features to disambiguate word meanings in the post. For example, the word *Washington* can be used to refer to Washington, D.C. (LOCATION) or the presidency of George Washington (PERSON). Looking at the attached image, *Washington* could be further disambiguated.

Figure 1: Images of urban cities: (left) Shenzhen, China, (right) Dubai, UAE.

Figure 2: Images of rural villages: (left) Manali, India, (right) Hakone, Japan.

As mentioned above, visual information is considered capable of explaining word meanings and provide useful information for location name recognition. For example, Figure 1 shows images of two different modernized cities, Shenzhen in China and Dubai in the UAE, and Figure 2 shows images of rural villages, Manali in India and Hakone in Japan. One can easily recognize common objects from these images: skyscrapers in Figure 1, and townscapes surrounded by mountains and rivers in Figure 2. Similarities like these would provide sufficient information to consider that words like *Shenzhen* and *Dubai* in documents have the same NE aspect.

Proceedings of the Third International Workshop on Spatial Language Understanding (SpLU 2020), pages 1–9

https://doi.org/10.18653/v1/P17

In this paper, we propose a method for location name recognition that utilizes images more effectively. Specifically, image data are obtained for each word in a document through an image retrieval engine, using the words in the document as a search query, and used as an extra multimodal feature in a neural NER model. The proposed model has two advantages. First, it is robust to unseen words that do not appear in the training data; standard NER models tend to be vulnerable to unseen words. Image data corresponding to each word in the document would provide additional information to clarify word meanings, as shown in the examples of Figure 1 and Figure 2. Second, our method can be applied to any documents to obtain element-wise image data corresponding to each word in the document; those of previous studies can only be applied to documents with images attached to them.

In addition, in the proposed method, we introduce a Gate mechanism to control the extent to which the visual features from images are input to the neural NER model. Polysemous words, abbreviations, and misspellings in a document could result in inappropriate instances in the image data obtained by the image retrieval engine. The gate's function is to remove the harmful effects derived from these instances by increasing or decreasing the degree of effect of a visual feature in the model when an image is appropriate or inappropriate for the document's context, respectively.

We evaluate the model's performance for location name recognition using a standard BiLSTM-CRF model as our baseline and then show the effectiveness of element-wise visual information and Gate mechanism, through our experimental results.

2 Related Work

2.1 Neural NER Model

In NER, machine learning models using conditional random fields (CRFs) have been widely used (Marcińczuk, 2015). Since the emergence of deep learning in recent years, it has become common to use various neural network-based NER models. Among them, bidirectional long short-term memory (LSTM) models that include a CRF layer, BiLSTM-CRF, are one of the most common models (Huang et al., 2015; Lample et al., 2016). Furthermore, a variation of BiLSTM-CRF with pre-trained language models for large unsupervised corpora such as Flair (Akbik et al., 2018) have been successful in achieving high performance.

2.2 Use of Visual Information

Visual information obtained from images (or pictures) has been used in neural NER models, especially when applied to SNS posts that include images related to them. Moon et al. (2019) proposed a neural NER model using images attached to a post as multimodal features. In the model, the image is transformed into a vector representation through a pre-trained CNN-based image recognition model and then combined with the input to the LSTM network for NER. Asgari-Chenaghlu et al. (2020) proposed a similar model to Moon et al. (2019) that could directly use object name class labels obtained by the image recognition model. Lu et al. (2018) and Zhang et al. (2018) proposed models that obtain one-to-one correspondences between a word in a document and an object in a picture attached to the document to obtain fine-grained visual features. These studies only use image data attached to the document, not element-wise image data corresponding to words in the document.

In Chinese NER, each part of a Chinese character in a document can be regarded as a visual feature and mixed into the NER model (Jia and Ma, 2019). Although this model handles element-wise visual information in the same manner as ours, that is, image data corresponding to each element (character or word) in the document are used in the NER model, it only focuses on the characters' patterns. Our model, described in detail in Section 4, focuses on images that express word meanings.

3 BiLSTM-CRF Model

This section describes the details of the BiLSTM-CRF model as a basis for our baseline model. As mentioned in the previous section, the BiLSTM-CRF model is one of the most common models for NER. The input is a word or character sequence in a document and the output is a sequence of labels representing NE information. In this study, we use a character-based model because the dataset used in the experiments is Japanese and errors caused by word segmentation can be ignored. Character-based models have been confirmed to outperform word-based models when Japanese documents are used (Misawa et al., 2017).

Let $\boldsymbol{C} = \{c_t\}_{t=1}^{M}$ be the input character sequence, $\boldsymbol{x} = \{\boldsymbol{x}_t\}_{t=1}^{M}$ be the input vector sequence corresponding to $\boldsymbol{C}$, and $\boldsymbol{y} = \{y_t\}_{t=1}^{M}$ be the output label sequence. Here, $\boldsymbol{x}$ is created by concatenating three types of vector (embedding) sequences $\boldsymbol{x}_c$, $\boldsymbol{x}_w$, and $\boldsymbol{x}_F$. The t-th element $\boldsymbol{x}_t$ of $\boldsymbol{x}$ is given by Equation (1).

$$\boldsymbol{x}_t = [\boldsymbol{x}_{c,t}; \boldsymbol{x}_{w,t}; \boldsymbol{x}_{F,t}] \quad (1)$$

The sequence $\boldsymbol{x}_c = \{\boldsymbol{x}_{c,t}\}_{t=1}^{M}$ is a sequence of character embeddings corresponding to $\boldsymbol{C}$. Each element of x_c, $x_{c,t}$ corresponds to a GloVe embedding (Pennington et al., 2014) for the corresponding character in $\boldsymbol{C}$.

In addition to $\boldsymbol{x}_c$, we also use $\boldsymbol{x}_w$ and $\boldsymbol{x}_F$, which are sequences of word embeddings to integrate the word meanings into the input. $\boldsymbol{x}_w$ is the character-based word sequence, which is a word sequence whose length is the same as that of the character sequence $\boldsymbol{C}$. Let $\boldsymbol{W} = \{w_t\}_{t=1}^{M}$ be a character-based word sequence, where M denotes the number of characters of a word sequence. Here, let $\boldsymbol{S} = \{s_i\}_{i=1}^{N}, (M \geq N)$ be a word sequence in the input document. $\boldsymbol{W}$ is a variation of $\boldsymbol{S}$, which is created by repeating each word $|s_i|$ times where $|s_i|$ denotes the number of characters in the word s_i. Note that w_t and w_{t+1} in $\boldsymbol{W}$ are the same value if they come from the same word s_i. The sequence $\boldsymbol{x}_w = \{\boldsymbol{x}_{w,t}\}_{t=1}^{M}$ is a sequence of word embeddings corresponding to $\boldsymbol{W}$. Each element of $\{\boldsymbol{x}_{w,t}\}$ is also trained by GloVe (Pennington et al., 2014). The sequence $\boldsymbol{x}_F$ is the alternative version of $\boldsymbol{x}_w$, using the Flair training scheme (Akbik et al., 2018) instead of GloVe.

The input $\boldsymbol{x}$ is given to the LSTM network layer. In this layer, each unit of the LSTM updates the state of the t-th element $\boldsymbol{x}_t$ on the basis of the previous LSTM ($\boldsymbol{c}_{t-1}$) and the hidden state ($\boldsymbol{h}_{t-1}$), and outputs the updated state as $\boldsymbol{h}_t$ and $\boldsymbol{c}_t$.

$$\boldsymbol{h}_t, \boldsymbol{c}_t = LSTM(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}, \boldsymbol{c}_{t-1}; \boldsymbol{\theta}) \quad (2)$$

In a BiLSTM network, the output $\overrightarrow{\boldsymbol{h}}$ of the forward LSTM and the output $\overleftarrow{\boldsymbol{h}}$ of the backward LSTM are combined to compute the total output $\overleftrightarrow{\boldsymbol{h}}$.

$$\overleftrightarrow{\boldsymbol{h}} = [\overleftarrow{\boldsymbol{h}}; \overrightarrow{\boldsymbol{h}}] \quad (3)$$

Next, the output of the LSTM network layer, $\overleftrightarrow{\boldsymbol{h}}$, is sent to the next CRF layer. In this layer, the labeling scheme that takes into account the transition probability between labels is carried out, and the output sequence $\boldsymbol{y}$ is calculated against $\boldsymbol{x}$. The output is selected for the optimal sequence on the basis of the Equation (4) where $\boldsymbol{\phi}$ is the feature function and $\boldsymbol{W}_{CRF}$ is a weight coefficient learned in this layer.

$$\boldsymbol{y}^* = \arg\max_{\boldsymbol{y}} \sum_t \boldsymbol{W}_{CRF} \cdot \boldsymbol{\phi}(\overleftrightarrow{\boldsymbol{h}}, y_t, y_{t-1}) \quad (4)$$

The element y_t of the output label sequence $\boldsymbol{y}$ represents the entity label for each character c_t. In general, named entity may be composed of multiple characters. So, we use the BIO scheme to represent the chunks of the named entity.

4 Proposed Method

The proposed method is a variation of the visual-enhanced BiLSTM-CRF models that enable to integrate visual features into the basic BiLSTM-CRF model described in the previous section. The proposed method utilizes element-wise visual features by obtaining image data for each word in the input document through an image retrieval engine where each word in the document is used as a search query. By retrieving images associated to words, the proposed method can be applied to any documents, while the previous visual-enhanced models mentioned in Section 2 can only be applied to documents with images attached to them.

Figure 3 shows an overview of the proposed method. The left-hand side shows the basic BiLSTM-CRF model. The right-hand side shows the proposed module to create element-wise visual features. In this section, we describe the proposed method step by step. First, we explain the procedure for constructing queries from the input document (Section 4.1). Next, we explain how to obtain visual embeddings (Section 4.2) and integrate the visual features to the original text features (Section 4.3). We then update the input vector sequence shown in Equation (1) to carry the visual features to the BiLSTM-CRF (Section 4.4).

4.1 Retrieving Image data

The given input document is transformed into a character-based word sequence $\boldsymbol{W} = \{w_t\}_{t=1}^{M}$ by using the same procedure described in the previous section. Then, we construct a query sequence

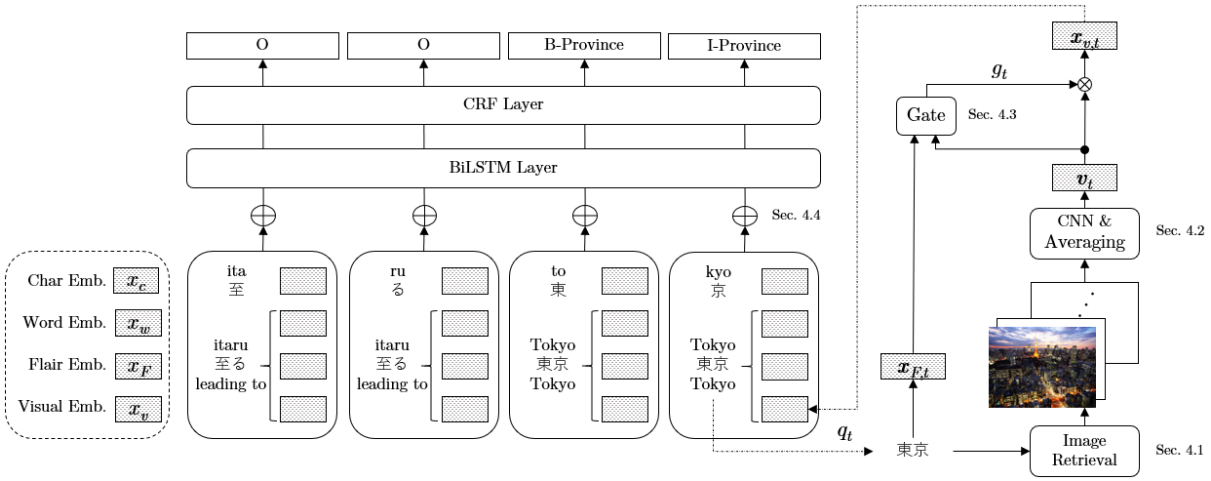

Figure 3: The overview of the proposed model. The left-hand side is the basic BiLSTM-CRF model. The right-hand side is the proposed module to create element-wise visual features.

$\boldsymbol{Q} = \{q_t\}_{t=1}^{M}$: if w_t is a noun, q_t is w_t itself, otherwise q_t is empty. As other word types would be irrelevant for image retrieval, we focus only on nouns. Nouns include not only proper nouns but also common nouns. The part-of-speech information is provided by the Japanese POS tagger MeCab, which is described below.

Each q_t in $\boldsymbol{Q}$ is used as the query for the image retrieval independently of each other; namely, we run the image retrieval M times. The top K retrieved images, referred to as $\boldsymbol{p}_t$, are saved for each run. If a query q_t is empty, no retrievals are performed and $\boldsymbol{p}_t$ is also set to empty. The $\boldsymbol{P} = \{\boldsymbol{p}_t\}_{t=1}^{M}$ is sent to the next step as element-wise visual information.

4.2 Obtaining Visual Embeddings

DenseNets (Huang et al., 2016) are one of the most powerful CNN-based deep neural network architectures, especially for image recognition. A pre-trained DenseNet model is applied to the retrieved images $\boldsymbol{p}_t$ to obtain visual embeddings. First, each image in $\boldsymbol{p}_t$ is sent to the DenseNet, and then the hidden representation of the final hidden layer of the DenseNet is saved. After K times running, the average of the K hidden representations is obtained as the visual embedding $\boldsymbol{v}_t$.

If $\boldsymbol{p}_t$ is empty, we define $\boldsymbol{v}_t$ as a *zero* vector where every element is 0.

4.3 Combining Visual Embeddings

The obtained visual embedding $\boldsymbol{v}_t$ are modified to adjust the balance of combinations between the original text features and our visual features. Here, we introduce the **Gate mechanism** to control how much of the visual features are input to the BiLSTM-CRF model. It works to decrease the degree of effect of the visual features when retrieved images from polysemous words, abbreviations, and misspellings are inappropriate. We also present another **simple procedure**, which we compared against the Gate mechanism..

Gate mechanism This procedure is formulated as follows.

$$\begin{aligned} g_t &= \sigma(\boldsymbol{W}_g \cdot [\boldsymbol{v}_t; \boldsymbol{x}_{F,t}] + b) \\ \boldsymbol{x}_{v,t} &= g_t \boldsymbol{v}_t \end{aligned} \tag{5}$$

The modified visual embedding $\boldsymbol{x}_{v,t}$ is obtained by $\boldsymbol{v}_t$ multiplied by g_t. The modification weight g_t is calculated on the basis of the visual feature $\boldsymbol{v}_t$ and the text feature $\boldsymbol{x}_{F,t}$. We use $\boldsymbol{x}_{F,t}$ because the feature relevance $\boldsymbol{v}_t$ and context information around w_t needs to be verified. Here, $\sigma()$ denotes the sigmoid function and $\boldsymbol{W}_g$ and b the weight coefficients to be trained. If a visual feature provides useful context, the g_t is close to 1, otherwise close to 0. Note that no visual features are considered when $g_t = 0$.

Simple This procedure is used as a comparison with the Gate mechanism where $\boldsymbol{x}_{v,t}$ is defined as follow.

$$\boldsymbol{x}_{v,t} = \boldsymbol{v}_t \tag{6}$$

Note that this procedure is equivalent to the gate function in which g_t is fixed at 1.

#words	5,149,521
#lexicons	75,024
#documents	10,158
#sentences	141,146

Table 1: Statistics of ENE corpus

class	#types	#mentions
City	2,936	12,687
Country	431	21,340
County	150	248
GPE_Other	95	1,203
Province	381	8,861
MIX	21	66

Table 2: Statistics of location name classes

4.4 Use of Visual Features

Finally, the input vector sequence shown in Equation (1) is updated to Equation (7) to input the visual features to the input layer of the BiLSTM-CRF.

$$\boldsymbol{x}_t = [\boldsymbol{x}_{c,t}; \boldsymbol{x}_{w,t}; \boldsymbol{x}_{F,t}; \boldsymbol{x}_{v,t}] \quad (7)$$

5 Experiments

5.1 Dataset

We used the Extended Named Entity corpus (ENE corpus) (Hashimoto et al., 2008), which uses the definition of Sekine's Extended Named Entity Hierarchy (Sekine et al., 2002) 7.1.0 including more than 200 types of named entities including a number of location name types. This corpus is one of the commonly-used datasets for evaluating Japanese NER methods. Each document in the corpus has no attached images. The statistics of ENE corpus are shown in Table 1.

We focused on six classes: Country, Province, County, City, GPE_Other, and MIX in the experiments. The first five classes are the original ones enclosed in Sekine's definition. We included MIX to indicate cases that have multiple NE classes. Hereafter, we ignore MIX for convenience because of rare cases. The statistics of each class are shown in Table 2[1].

Before training, we divided the dataset into three parts: training, develop and test in the ratio of 70:15:15. The statistics of dataset are shown in Table 3

class	#M(train)	#M(dev)	#M(test)
City	8,567	2,184	1,936
Country	14,534	3,593	3,213
County	172	43	33
GPE_Other	848	231	124
Province	5,888	1,525	1,448
MIX	54	9	3

#M(·) means number of mentions.

Table 3: Statistics of dataset

5.2 Settings

We constructed three models for location name recognition. The first is the baseline model and the others are the proposed models described in Section 4.

- **Baseline** is the BiLSTM-CRF model described in Section 3. No use of visual features.
- **Visual (Gate)** is the proposed visual-enhanced BiLSTM-CRF model that utilizes element-wise visual features with the Gate mechanism.
- **Visual (Simple)** is another proposed model. This model uses the Simple text/visual combination instead of the Gate mechanism.

For word embeddings $\boldsymbol{x}_w$ and character embeddings $\boldsymbol{x}_c$, we conducted the GloVe training with 300 dimensions with the BCCWJ corpus (Maekawa et al., 2014). We use MeCab (Kudo et al., 2004) with the unidic (Den et al., 2007) dictionary for word segmentation. The Flair embeddings (Akbik et al., 2018) were trained using BCCWJ and ten years of Mainichi newspaper data from 1991 to 2000 with 1024 dimensions.

We used Google Images[2] with photo options for the image retrieval used in the proposed models. The top 15 retrieved images for each query are saved. In the dataset, about 43% of words were nouns, enabling non-empty queries to be constructed. The visual embeddings were created from the final hidden layer representation of DenseNet, whose dimensions were 1024. We used the pre-trained DenseNet from PyTorch.

[1] In this study, we found a number of annotation errors in the ENE corpus. We carefully observed 328 mentions related to the location name and corrected them before conducting our experiments.

[2] https://images.google.com/

Model	Prec.	Recall	F1
Baseline	87.33	89.47	88.38
Visual (Simple)	**90.20**	87.78	88.97*
Visual (Gate)	89.33	**90.01**	**89.67****

We performed an approximate randomization test (Chinchor, 1992) on the F1 values. The mark "*" and "**" in the table show significant differences compared with the baseline at the 0.05 and 0.01 levels, respectively.

Table 4: Experimental Results

In the training of the models, we used Adam (Kingma and Ba, 2014) for optimization. The batch size was 20. We applied the dropout regularization (Srivastava et al., 2014) at $p = 0.5$ for each node of the input layer and each output node of the LSTM layer. We also used a gradient clipping (Pascanu et al., 2013) of 1.0 to reduce the effects of the gradient exploding.

We used the standard BIO schema (Tjong Kim Sang and Veenstra, 1999) for the chunk representation. The performance was measured by the Precision (Prec.), Recall, and F1 values. Only the exact matches were counted as the correct samples, while lenient matches were counted as incorrect.

5.3 Results and Discussion

Experimental results are shown in Table 4. Both models using element-wise visual features outperformed the baseline model. This result suggests that element-wise visual features are powerful features for location name recognition. Furthermore, the Visual (Gate) model achieved the best F1 value of 89.67%. From the results, the Gate mechanism is an essential part of integrating element-wise visual features into the baseline model.

The following example sentences are samples in the cases where the Visual (Gate) has a correct output while the Baseline has an incorrect output. Here, (1-J) and (2-J) are the original Japanese sentences, and (1-E) and (2-E) are the corresponding English translations.

- (ex.1-J) １９８２年１２月の**ジャマイカ**$_{Country}$における最終議定書及び条約署名会議において１１７か国及び２地域が署名し成立した
- (ex.1-E) Signed by 117 countries and two regions at the Final Protocol and Convention Signing Conference in **Jamaica**$_{Country}$ in December 1982.
- (ex.2-J) いよいよ、明日は、帰る日を除いて、フランス滞在の最終日です。**アヴィニヨン**$_{City}$に行く予定です。
- (ex.2-E) Finally, tomorrow is the last day of our stay in France, except for the day we leave. We're going to **Avignon**$_{City}$.

Figure 4: An example photo retrieved by the query ジャマイカ (Jamaica).

Figure 5: An example photo retrieved by the query アヴィニヨン (Avignon).

In the first example, ジャマイカ (Jamaica) is a country name to be recognized, and in the second example, アヴィニヨン (Avignon) is a city name in France. Moreover, the examples of retrieved image data corresponding to these location names are shown in Figure 4 and Figure 5, respectively. One can see that a typical scene or object is in each image; a beach in Figure 4 and a palace in Figure 5. It suggests that image data showing scenes or objects strongly relevant to locations provide helpful visual features.

Table 5 shows the fine-grained performances of the experimental results. It shows that the City class had the most significant improvement. In fact, we confirmed that the retrieved image data corresponding to city names showed many typical characteristics of the locations, such as buildings, landscapes, and skies. In contrast, for an example of other classes, image data corresponding to country names showed various weakly related objects to the countries. For example, we found some image data showing the president of its country.

class	Prec.	Recall	F1
Baseline			
City	84.18	77.90	80.92
Country	89.88	96.29	92.97
County	83.78	93.94	88.57
GPE_Other	85.90	75.28	80.24
Province	85.40	90.24	87.75
Visual(Simple)			
City	87.67	78.16	82.64
Country	93.07	93.68	93.37
County	82.35	84.85	83.58
GPE_Other	82.72	75.28	78.82
Province	87.37	88.10	87.73
Visual(Gate)			
City	89.38	78.67	83.68
Country	91.74	95.81	93.73
County	93.55	87.88	90.62
GPE_Other	84.42	73.03	78.31
Province	84.55	93.15	88.64

Table 5: Performance for each class

These seem to suggest not location names but person names.

Here we call words that appear in test data but not in training data as unseen words. In general, it is arduous to achieve accurate NER performance on unseen words because they do not appear in training data and thus have poor textual information. Here, we investigated whether our visual features provide supplemental information to unseen words. To realize the investigation, we conducted an analysis focusing on the City class. As shown in Table 2, the City class differs from other classes in that it has many types of mentions. It implies that there are many unseen mentions to be recognized to the City class. Therefore we compared the extraction performance between seen words and unseen words in the City class. Table 6 shows the details of the results. One can see that the unseen words achieved better performance improvements than the seen words. Furthermore, precision values improved most significantly (Seen(+5.08) → Unseen(+7.07)). This means that visual features improve the performance of not only true-positive samples but also true-negative samples. The example sentences are shown below. Each underline indicates the unseen true-negative word. And, the corresponding retrieved images are shown in Figure 6 and Figure 7. These samples were correctly classified by the proposed method while wrongly by the baseline method.

Figure 6: An example photo retrieved by the query ブンデスリーガ (Bundesliga).

Figure 7: An example photo retrieved by the query シスコ (Cisco).

- (ex.3-J) ドイツ・ブンデスリーガの強豪、バイエルン・ミュンヘンと「アドバイザー契約」を結んで４年目。
- (ex.3-E) This is the fourth year that he has signed an “advisory contract” with German Bundesliga powerhouse Bayern Munich
- (ex.4-J) このような……シスコの経営陣は……
- (ex.4-E) This kind of... Cisco’s management team...

Although, as discussed above, the element-wise visual features contributed to improve the performance of location name recognition, some types of errors remained. It observed that the proposed method tends to cause false-positive errors in compound words including location names. For example, 京都-議定書 (the Kyoto Protocol) in (5-J) and (5-E) was wrongly recognized as the location name. This type of error is caused by inadequate query construction. Because every single noun in the document is regarded as the query word independently in the proposed method, both 京都 (Kyoto) and 議定書 (Protocol) were used to the image retrievals. Then they led to the mistaken recognition of Kyoto.

- (ex.5-J) 温室効果ガス削減に向け、政府が決めた京都議定書の目標達成計画には様々な方策が並んでいる。

	Model	Precision	Recall	F1
Seen	Baseline	84.99	79.29	82.04
	Visual (Gate)	90.07 (+5.08)	80.05 (+0.76)	84.77 (+2.73)
Unseen	Baseline	68.54	54.95	61.0
	Visual (Gate)	75.61 (+7.07)	55.86 (+0.91)	64.25 (+3.25)

Table 6: Comparison between seen & unseen mentions for City class

Figure 8: An example photo retrieved by the query アンゴラ (Angola). This photo shows not “Angola” but “Angora rabbit”.

- (ex.5-E) The government’s plan to meet the Kyoto Protocol’s targets for reducing greenhouse gas emissions includes a number of measures.

It also observed that the proposed method tends to cause false-negative errors when inappropriate images are mixed to the retrieved images. For example, the proposed method missed recognizing アンゴラ (Angola) in (6-J) and (6-E) because the image data retrieved by the query アンゴラ (Angola) includes some inappropriate images of “Angora rabbit” like in Figure 8. アンゴラ (Angola) is not a polysemous word, but it found that the word means “Angora rabbit” in the specific domain [3].

- (ex.6-J) 内戦の続く**アンゴラ** $_{Country}$ ではコレラが流行。

- (ex.6-E) Cholera epidemic in **Angola**$_{Country}$ as civil war continues.

6 Conclusion

In this study, we proposed a NER model that uses images corresponding to all nouns in a document as features and a Gate mechanism that controls the extent to which visual features are provided as input to the neural NER model. We conducted experiments to confirm its performance in location name recognition. Experimental results show that the proposed method achieved a higher F1-value performance than the baseline model in the ENE corpus dataset, with a significant difference of $p < 0.01$.

In future research, we will investigate whether the proposed model is effective for cases other than location names. We also aim to improve our model to be more effective by conducting elaborate query investigations that are motivated by the error analysis. The hyper-parameter K, which means the number of images per word, would be critical for obtaining valuable visual embeddings. Therefore, we will also investigate whether the larger the K, the better the location name recognition performance. The experimental results showed that the proposed method has little contributions when query words are polysemous. We would like to attempt word sequence queries with nouns and adjectives/verbs instead of single noun queries.

[3] Note that the words “Angola” and “Angora” are transliterated into the same Japanese string “アンゴラ” although they are different in English.

Acknowledgments

We thank anonymous reviewers for their responsible attitude and helpful comments. This work was supported by JSPS KAKENHI Grant Number JP18K11982.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual String Embeddings for Sequence Labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649.

Meysam Asgari-Chenaghlu, M. Reza Feizi-Derakhshi, Leili Farzinvash, M. A. Balafar, and Cina Motamed. 2020. A multimodal deep learning approach for named entity recognition from social media. https://arxiv.org/abs/2001.06888.

Nancy Chinchor. 1992. The Statistical Significance of the MUC-4 Results. In *Proceedings of*

the Fourth Message Uunderstanding Conference (MUC-4), page 30–50.

Jim Cowie and Wendy Lehnert. 1996. Information extraction. *Communications of the ACM*, 39(1):80–91.

Yasuharu Den, Toshinobu Ogiso, Hideki Ogura, Atsushi Yamada, Nobuaki Minematsu, Kiyotaka Uchimoto, and Hanae Koiso. 2007. electronic dictionary, morphological analysis, database system, uniformity of units, identity of indexes. *Japanese Linguistics*, pages 101–123.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference-6. In *Proceedings of the 16th conference on Computational linguistics*, page 466–471.

Taiichi Hashimoto, Takashi Inui, and Koji Murakami. 2008. Constructing extended named entity annotated corpora. *IPSJ SIG Notes*, pages 113–120.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. 2016. Densely Connected Convolutional Networks. https://arxiv.org/abs/1608.06993.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging. https://arxiv.org/abs/1508.01991.

Yaozong Jia and Xiaopan Ma. 2019. Attention in Character-Based BiLSTM-CRF for Chinese Named Entity Recognition. In *ICMAI 2019 Proceedings of the 2019 4th International Conference on Mathematics and Artificial Intelligence*, page 1–4.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. https://arxiv.org/abs/1412.6980.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying Conditional Random Fields to Japanese Morphologiaical Analysis. *IPSJ SIG Notes*, 161:89–96.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2018. A Survey on Deep Learning for Named Entity Recognition. https://arxiv.org/abs/1812.09449.

Di Lu, Leonardo Neves, Vitor Carvalho, Ning Zhang, and Heng Ji. 2018. Visual Attention Model for Name Tagging in Multimodal Social Media. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1990–1999.

Kikuo Maekawa, Makoto Yamazaki, Toshinobu Ogiso, Takehiko Maruyama, Hideki Ogura, Wakako Kashino, Hanae Koiso, Masaya Yamaguchi, Makiro Tanaka, and Yasuharu Den. 2014. Balanced corpus of contemporary written Japanese. *Language Resources and Evaluation*, 48:345–371.

Michał Marcińczuk. 2015. Automatic construction of complex features in Conditional Random Fields for Named Entities Recognition. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 413–419.

Shotaro Misawa, Motoki Taniguchi, and Yasuhide Miura. 2017. Character-based Bidirectional LSTM-CRF with words and characters for Japanese Named Entity Recognition. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 97–102.

Seungwhan Moon, Leonardo Neves, and Vitor Carvalho. 2019. Multimodal Named Entity Recognition for Short Social Media Posts. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 852–860.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on International Conference on Machine Learning*, page III–1310–III–1318.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Satoshi Sekine, Kiyoshi Sudo, and Chikashi Nobata. 2002. Extended named entity hierarchy. In *Proceedings of the Third International Conference on Language Resources and Evaluation*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, page 1929–1958.

Erik F. Tjong Kim Sang and Jorn Veenstra. 1999. Representing Text Chunks. In *Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics*, page 173–179.

Davy Weissenbacher, Arjun Magge, Karen O'Connor, Matthew Scotch, and Graciela Gonzalez-Hernandez. 2019. SemEval-2019 Task 12: Toponym Resolution in Scientific Papers. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 907–916.

Qi Zhang, Jinlan Fu, Xiaoyu Liu, and Xuanjing Huang. 2018. Adaptive co-attention network for named entity recognition in tweets. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*, pages 5674–5681.